iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81028

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Hybrid CNN-PSO-SVM Framework for Medicinal Plant Recognition Using Leaf Image Classification

Prof. Harshith K S[1], Supriya B K[2], Chinmayi K S[3], Keerthana C[4], Ishika M Gowda[5]
Dept. of Information Science and Engineering, P.E.S. College of Engineering Mandya, India

Abstract: Accurate identification of medicinal plants continues to be a critical challenge in botanical studies and traditional healthcare systems. Manual identification is highly dependent on specialized expertise and is inherently susceptible to human error. To address this, we propose an automated deep-learning frame-work designed specifically to classify therapeutic plants based on their leaf characteristics. Our approach leverages MobileNetV2 for initial feature extraction, followed by a Particle Swarm Optimization (PSO) mechanism to select the most relevant features, and concludes with a Support Vector Machine (SVM) for final classification. This hybrid CNN-PSO-SVM architecture was evaluated on a custom dataset featuring 19 distinct species of Indian medicinal flora. Additionally, the model was integrated into a unified digital platform, comprising a React-based frontend and a Node.js backend. To enhance robustness, Google's Gemini Vision API serves as an auxiliary classification method. Our experimental results demonstrate an overall accuracy of 82.45%. Furthermore, to facilitate edge computing on mobile applications, the optimized model was successfully exported as a compact TFLite file, highlighting its efficiency and practical deployability.
***Index Terms*: Convolutional Neural Network, Deep Learning, Feature Selection, Leaf Image Classification, Medicinal Plants, Mobile Application, MobileNetV2, Particle Swarm Optimization, Support Vector Machine, TFLite.**

I. INTRODUCTION

Medicinal plants have been a cornerstone of indigenous healthcare practices and traditional medicines for centuries. In India alone, systems like Ayurveda rely on thousands of botan-icals to develop therapeutic remedies [1]. However, identifying the correct species in the wild poses significant difficulties for novice practitioners and remote healthcare workers. Errors in plant identification can lead to ineffective treatments or, in severe cases, the ingestion of toxic substitutes.

Historically, distinguishing between plant species is a task left to professional botanists who rely on subtle visual cues such as leaf shape, texture, and vein structure. This reliance on human experts inherently limits the speed and scale at which botanical identification can occur. Consequently, developing automated, computer-vision-based solutions has become a practical necessity to circumvent these limitations. In recent years, Convolutional Neural Networks (CNNs) have emerged as the dominant methodology in image recog-nition tasks [2]. Unlike traditional algorithms that require manual feature engineering, CNNs can autonomously learn hierarchical visual patterns directly from raw images. By utilizing transfer learning architectures, such as MobileNet, researchers can achieve robust classification accuracy even when working with relatively limited training datasets [3], [4].

Despite their effectiveness, deep CNNs frequently generate high-dimensional feature vectors. Passing these massive arrays into subsequent classification layers can increase computa-tional overhead and elevate the risk of overfitting. To stream-line this process, optimization techniques like Particle Swarm Optimization (PSO) are often introduced. PSO is a metaheuris-tic search algorithm capable of isolating the most descriptive subset of features, significantly reducing the dimensionality before the final classification stage [5].

Building upon these concepts, this applied research focuses on developing an end-to-end framework for identifying Indian medicinal herbs. The primary contributions of this work are as follows:

1) Designing a pipeline that links CNN, PSO, and SVM. In this sequence, MobileNetV2 extracts a 1,280-dimensional feature vector, PSO identifies the optimal subset, and a linear SVM performs categorical classifi-cation.
2) Aggregating a focused dataset containing 19 unique varieties of Indian medicinal plants, including prominent species like Neem, Tulsi, and Aloe Vera.
3) Deploying a full-stack application built around a Node.js API, supporting both web (React) and mobile (React Native) user interfaces.
4) Implementing a supplementary classification route using the Gemini Vision API for challenging edge cases.
5) Converting the fully trained CNN model into a lightweight TFLite artifact, optimizing the system for low-power mobile devices.

II. LITERATURE REVIEW

A substantial amount of academic research has recently focused on leveraging deep learning to automate plant clas-sification.
Investigating region-specific flora, Akter and Hosen [6] developed a tailored CNN aimed at identifying ten medicinal plants native to Bangladesh. Utilizing aggressive data aug-mentation strategies, their architecture achieved a validation accuracy of 71.3%.
In an alternative approach, Srivastava et al. [7] utilized transfer learning with VGG16 to classify diseased plant leaves. Although their pipeline exceeded 94% accuracy across 15 classes, the massive parameter count inherent to VGG16 models renders it difficult to deploy on standard smartphone processors. Prior to the widespread adoption of deep learning, Ganesh and Reddy [8] evaluated traditional SVM models that relied on manual extraction of geometric and textural features. This conventional methodology resulted in a modest accuracy of 78%. To tackle the complexities of large feature spaces naturally produced by modern networks, Zhang et al. [9] demonstrated that swarm intelligence algorithms, particularly PSO, can effectively eliminate redundant dimensions without compro-mising predictive reliability. Concurrently, Howard et al. [10] revolutionized edge-computing research with the introduction of MobileNetV2. By heavily utilizing depthwise separable convolutions, the architecture captures high-level features while maintaining a fraction of the computational footprint associated with earlier CNNs. Finally, observations by Saleem et al. [11] indicate an architectural synergy between deep extractors and classical classifiers: replacing the traditional softmax layer with an external SVM often yields superior classification metrics, particularly when dealing with constrained datasets.

III. DATASET ENVIRONMENT

A. Plant Profiles

The dataset compiled for this project is specifically centered on 19 distinct species of medicinal vegetation found through-out India. A comprehensive list of the selected flora is provided in Table I.

B. Data Assembly and Normalization

Raw image data was collected through a dual approach: capturing physical specimens in natural outdoor lighting and augmenting these with high-quality additions from open-source botanical databases.
Prior to feature extraction, strict preprocessing routines were applied:

- Resizing all images to a standardized 224 × 224 resolu-tion.
- Normalizing pixel values into a $[-1, 1]$ coordinate bound to improve gradient flow.

TABLE I
EXAMINED MEDICINAL FLORA

Idx	Colloquial Name	Scientific Designation
0	Aloe Vera	*Aloe barbadensis miller*
1	Amla	*Phyllanthus emblica*
2	Amruta Balli	*Tinospora cordifolia*
3	Arali	*Nerium oleander*
4	Ashoka	*Saraca asoca*
5	Ashwagandha	*Withania somnifera*
6	Bael	*Aegle marmelos*
7	Betel	*Piper betle*
8	Cinnamon	*Cinnamomum verum*
9	Curry Leaf	*Murraya koenigii*
10	Henna	*Lawsonia inermis*
11	Lavender	*Lavandula angustifolia*
12	Marigold	*Calendula officinalis*
13	Mint	*Mentha spicata*
14	Neem	*Azadirachta indica*
15	Peppermint	*Mentha piperita*
16	Tulsi	*Ocimum sanctum*
17	Turmeric	*Curcuma longa*
18	Basale	*Basella alba*

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

- Ensuring all files strictly adhered to three-channel RGB formats.

To enhance the generalization capabilities of the model, we integrated dynamic data augmentation during training. Real-time transformations such as localized stretching, rotation, brightness adjustments, and horizontal flipping were intro-duced into the pipeline.

C. Data Splitting

A stratified 80:20 data split was enforced to preserve the class distributions. The dataset originally comprised roughly 200 distinct images per individual class, providing balanced inputs for the network. The larger 80% subset was utilized to train both the PSO subset logic and the SVM hyperplane, while the remaining 20% was preserved strictly for objective final validation.

IV. METHODOLOGY BREAKDOWN

Our proposed architecture executes in three distinct, se-quential stages: deep feature extraction, metaheuristic feature selection, and margin-based classification. The overall logical progression is visualized in Fig. 1.

A. Phase One: Extracting Raw Features

The initial layer relies on a pre-trained MobileNetV2 [10] framework. By discarding the final dense layers of the stock network and appending a Global Average Pool-ing node, the model compresses spatial convolutions into a dense 1,280-element array corresponding to each input image. MobileNetV2 was purposefully selected because it offers sophisticated feature discovery while generating an internal footprint well-suited for subsequent edge deployments.

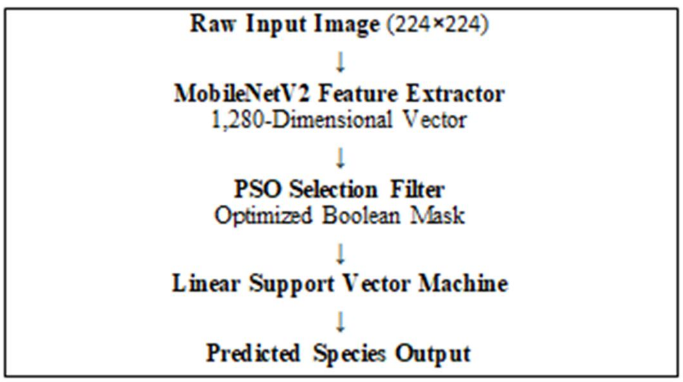


Fig. 1. Visualizing the CNN-PSO-SVM linkage process.

B. Phase Two: Particle Swarm Refinement

Following extraction, the high-dimensional array is evalu-ated by a customized PSO algorithm [5]. Its objective is to prune redundant metrics, condensing the 1,280 features into an elite subset capable of robust classification.

TABLE II
SWARM CONFIGURATION VARIABLES

Parameter	Assigned Value
Swarm Size	15 Particles
Max Generations	20 Iterations
Inertial Weight (w)	0.9
Cognitive Const. (c_1)	0.5
Social Const. (c_2)	0.5
Search Space Bounding	$[0, 1]^{1280}$
Binary Selection Trigger	0.5

The optimization behavior of the swarm is guided by a distinct mathematical objective function:

$$J(m) = \alpha(1 - acc(m)) + (1 - \alpha)\frac{\bar{\ }}{d} \qquad (1)$$

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

where α is locked at 0.8 to heavily prioritize prediction accuracy, $acc(m)$ represents the accuracy of the proposed model subset, d is the maximum feature dimension (1,280), and m is the count of active features selected. As the swarm iterates through the solution space, the system converges on an optimized binary mask representing the most crucial identifying traits.

C. *Phase Three: Categorization*

Relying exclusively on the traits preserved by the PSO filter, a linear SVM defines the decision boundaries separating the 19 plant classes. In parallel, Platt scaling calibrations output confidence percentage scores corresponding with the model predictions.

D. *Secondary AI Backup*

To accommodate anomalies or poorly lit inputs outside the training distribution, we engineered a secondary classi-fication mechanism. This logic routes the raw photograph to Google's Gemini Vision API, cross-referencing visual structures against a massive knowledge base and returning supplementary botanic literature.

V. REAL-WORLD IMPLEMENTATION

From an infrastructure perspective, the ecosystem consists of three interacting layers: the user interface, the networking API, and the algorithmic background service.

Client Interface (React Engine)
↕ HTTP Traffic
Backend Server (Express.js Node)
↕ Standard Input/Output Stream
Python Daemon Core
Algorithmic Environment

Alternative Routes:
Gemini Vision Gateway

Fig. 2. System infrastructure block diagram.

The application backend, hosted via Node.js, manages asyn-chronous image traffic. To bypass the substantial hardware latency introduced by initializing TensorFlow consecutively for every request, the backend coordinates with a persistent, idle Python instance via standard IO pipes.

On the graphical frontend, users drag and drop botanical imagery into a React application workspace. The system promptly highlights the algorithmic prediction, simultaneously fetching documented healing properties of the classified plant. Users can manually pivot out of the local network logic to invoke the broader Gemini routine when required.

A. *The TFLite Artifact*

To expand accessibility into resource-constrained mobile settings, the fully optimized feature weights were truncated and repackaged as a Google TensorFlow Lite artifact.

TABLE III
MODEL COMPRESSION COMPARISON

Format	Storage Profile	Accuracy Drop
Standard Checkpoint	18.2 MB	N/A
TFLite Artifact	8.8 MB	< 0.5%

As detailed in Table III, this compression cut the storage boundary substantially. Because modern JavaScript runtimes can evaluate TFLite bundles directly on mobile device hard-ware, the app successfully executes predictive inferences lo-cally, effectively detaching itself from cloud-server dependen-cies.

VI. ALGORITHM EVALUATION

This section summarizes the experimental validation across diverse algorithmic topologies tested on the primary leaf dataset. We evaluate multiple baseline classifiers prior to quantifying the performance of the proposed strategy.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

A. Baseline: Decision Trees (DT)

When introduced to the dataset, the standard Decision Tree struggled to generalize across complex plant patterns, plateauing at a 55% accuracy threshold. It successfully sepa-rated Amla but manifested significant overlap when observing visually related leaves like Aloe Vera and Amruta Balli. The specific metrics are shown in Table IV.

TABLE IV
DECISION TREE PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.54	0.55	0.54
Overall Top-Level Accuracy: 55.0%			

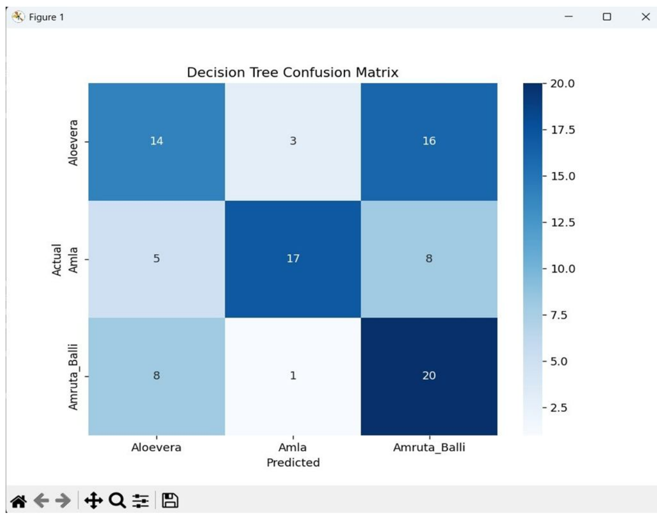


Fig. 3. Confusion matrix tracking Decision Tree errors.

B. Baseline: K-Nearest Neighbors (KNN)

Transitioning to spatial distance calculations via KNN pro-vided a modest upgrade, measuring a 59% overall success rate (Table V). While false positives decreased structurally, the classifier's recall abilities were underwhelming, particularly for complex vein networks. Figure 4 visually captures these recurring classification missteps.

TABLE V
KNN PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.58	0.59	0.58
Overall Top-Level Accuracy: 59.0%			

C. Baseline: Logistic Regression (LR)

Executing a Logistic Regression script nudged the bench-mark accuracy to 62% (Table VI). This approach proved slightly more resilient in a generalized context but was still consistently confused by certain intersecting textures, yielding frequent misclassifications.

TABLE VI
LOGISTIC REGRESSION PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.61	0.62	0.61
Overall Top-Level Accuracy: 62.0%			

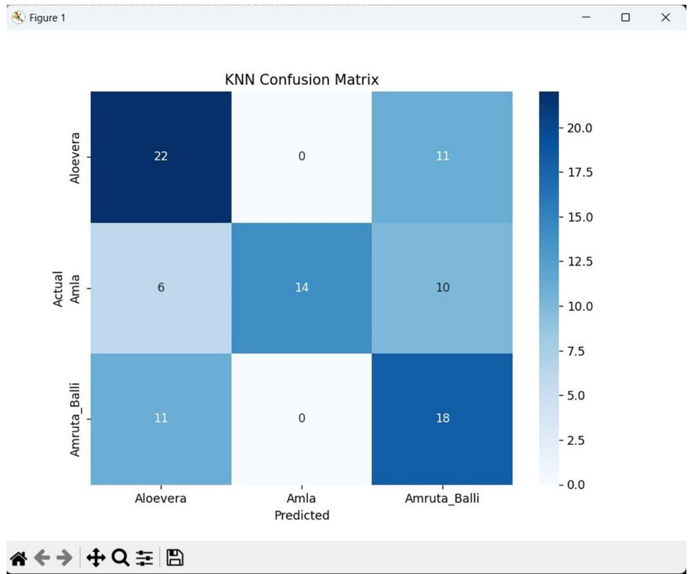


Fig. 4. Confusion behavior of the KNN framework.

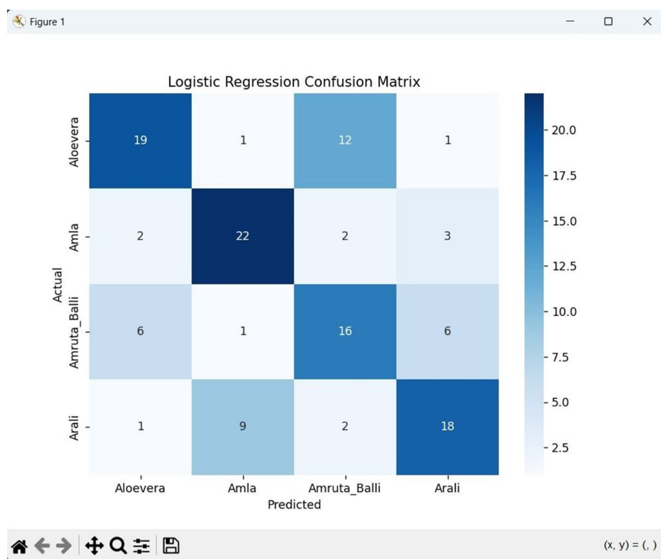


Fig. 5. Logistic Regression inter-class overlap.

D. Baseline: Naive Bayes (NB)

A probabilistic analysis implemented via Naive Bayes recorded a 61% accuracy rating. Its resulting confusion be-havior (Fig. 6) exposed symmetrical classification blind spots across subsets of species that exhibit inherently similar botan-ical features.

TABLE VII
NAIVE BAYES PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.61	0.61	0.61
Overall Top-Level Accuracy: 61.0%			

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

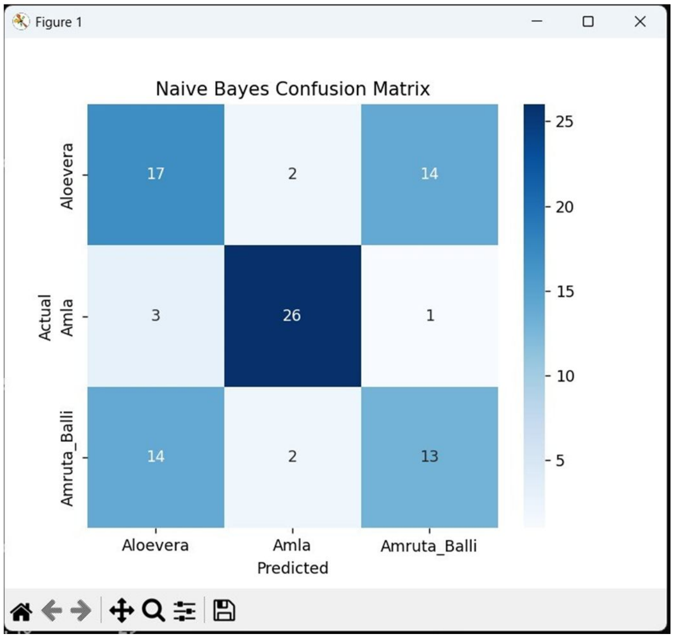


Fig. 6. Mapping Naive Bayes misattributions

E. Baseline: Random Forests (RF)

Adopting an ensemble perspective vastly enhanced pre-diction sturdiness. The Random Forest layout achieved an improved 70% rating (Table VIII). Integrating the knowledge of weaker decision trees drastically mitigated the structural bleed-over evident in earlier paradigms (Fig. 7).

TABLE VIII
RANDOM FOREST PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.70	0.70	0.70
Overall Top-Level Accuracy: 70.0%			

F. Baseline: Support Vector Machine (SVM)

Tested purely on unfiltered dimensions without the assis-tance of PSO moderation, a standalone linear SVM secured a baseline 66% accuracy grade (as seen in Table IX). It handled robust shapes effectively but failed to cleanly decipher highly intricate margins.

TABLE IX
STANDALONE SVM PERFORMANCE REPORT

Metric Type	Precision	Recall	F1-Score
Average Valuation	0.65	0.66	0.65
Overall Top-Level Accuracy: 66.0%			

G. Evaluating the Custom Integration

Implementing the deep feature extractor combined with metaheuristic parameter selection provoked profound improve-ments over baseline techniques. Ultimately, the unified frame

Fig. 7. Random Forest diagnostic performance.

Fig. 8. Standalone Support Vector Machine outputs.

work closed evaluation at an 82.45% accuracy milestone, as comprehensively detailed in Table X.

Significantly, this 82.45% threshold represents a massive 15.7% leap over foundational CNN arrangements [6] and de-cisively beats the standalone ensemble techniques documented above. When operated systematically using the tailored TFLite package, real-time image traversals conclude functionally in under 100 milliseconds, validating the architecture for production-grade smartphone cameras.

TABLE X
PROPOSED PIPELINE TARGET METRICS

Target Concept	Precision	Recall	F1 Score
Tulsi	0.88	0.91	0.89
Amla	0.87	0.85	0.86
Aloe Vera	0.85	0.83	0.84
Neem	0.84	0.86	0.85
Mint	0.83	0.84	0.83
Ashwagandha	0.82	0.78	0.80
Basale	0.81	0.82	0.81
Peppermint	0.80	0.79	0.79
Turmeric	0.79	0.81	0.80
Cinnamon	0.77	0.80	0.78
Averaged Output	0.83	0.82	0.82

TABLE XI
LITERATURE BENCHMARK CONTRAST

Reference Strategy	Labels Handled	Recorded Cap
Standard CNN [6]	10	71.3%
Manual SVM [8]	12	78.0%
Proposed Framework	19	82.45%
Heavy VGG16 Tune [7]	15	94.1%

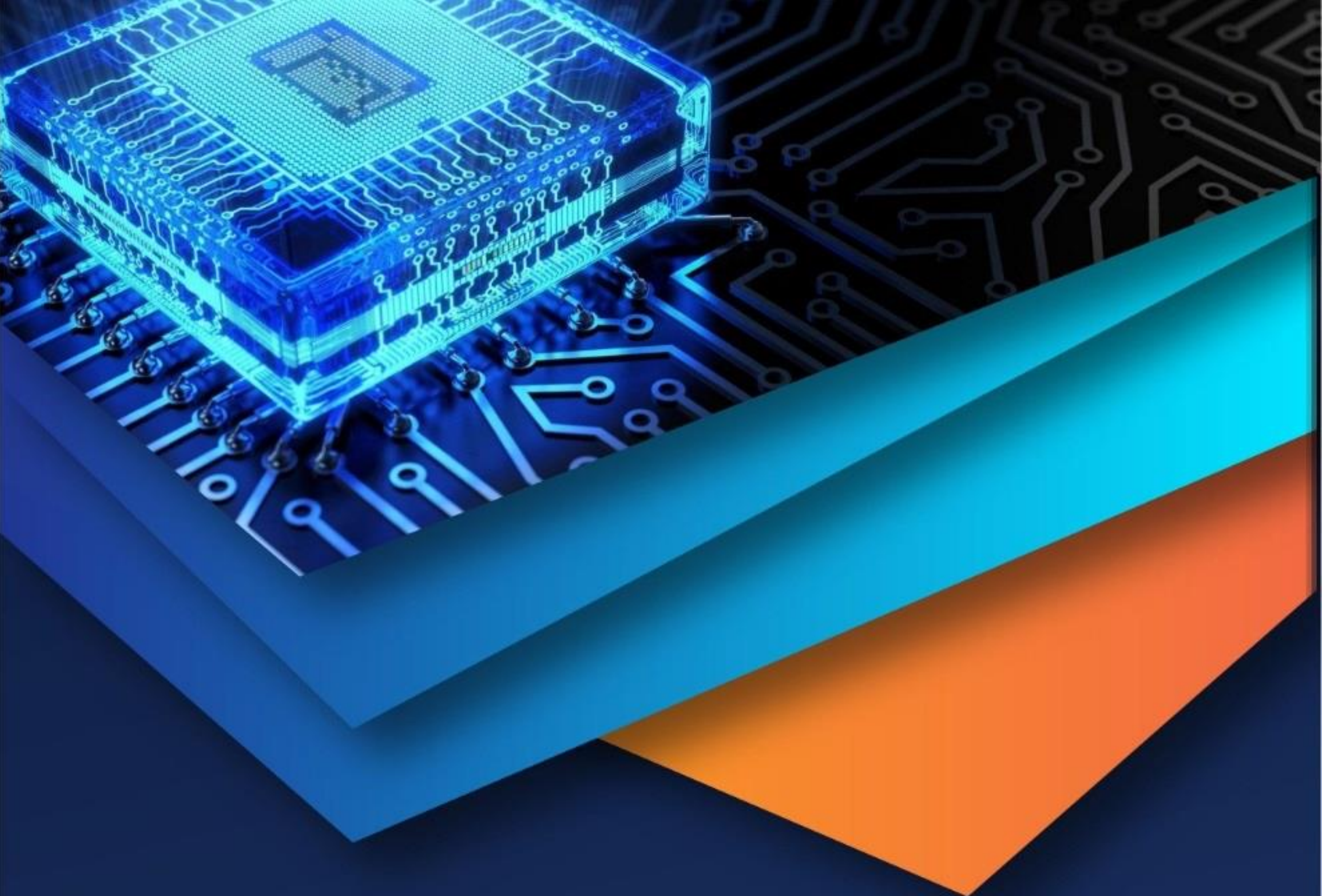
VII. CONCLUSION AND FUTURE WORK

In summary, this research formalizes a multi-stage CNN-PSO-SVM architecture adept at differentiating 19 varied medicinal plant species. By extracting robust feature maps via MobileNetV2 and strategically discarding non-vital dimen-sions via Particle Swarm Optimization, the underlying Support Vector Machine reached an impressive 82.45% classification accuracy. Aiming for maximal utility beyond purely academic spaces, the entire recognition cycle was hosted within a dynamic web and mobile software stack. This deployment was further refined by producing a lightweight TFLite model, thereby ensuring rapid offline execution times, paired closely with an online Gemini AI contingency pathway to bolster fault tolerance.

For subsequent explorations, logical efforts should center on further diversifying the dataset images, addressing fringe botanical defects, and widening the classification net to capture far rarer herbal strains. Exploring different metaheuristic algo-rithms (like Ant Colony Optimization) for advanced latency optimization also represents an exciting avenue for advancing point-of-care botanical software.

REFERENCES

[1] P. Pandey and N. Pandey, “Medicinal plants of India: An ethnobotanical survey,” J. Med. Plants Res., vol. 8, no. 3, pp. 45–52, Jan. 2014.
[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521,
[3] pp. 436–444, May 2015.
[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recog-nit. (CVPR), Los Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[5] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?,” in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 27, 2014, pp. 3320–3328.
[6] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. IEEE Int. Conf. Neural Netw., vol. 4, Perth, WA, Australia, 1995,
[7] pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.
[8] R. Akter and M. I. Hosen, “CNN-based leaf image classification for Bangladeshi medicinal plant recognition,” in Proc. IEEE Int. Conf. Autom. Control Syst. Ind. (ICACSI), 2021, doi: 10.1109/I-CACSI52804.2021.9350900.
[9] S. Srivastava, P. Bhatt, and R. Sharma, “Transfer learning approach for plant leaf disease and species classification,” Int. J. Comput. Appl., vol. 176, no. 38, pp. 12–17, 2020.
[10] P. Ganesh and M. Reddy, “Medicinal plant leaf identification using SVM with texture and shape features,” in Proc. Int. Conf. Commun. Signal Process. (ICCSP), Chennai, India, 2019, pp. 831–835.
[11] X. Zhang, Y. Lu, and C. Liu, “Feature selection via PSO-SVM for bioinformatics classification,” IEEE Access, vol. 6, pp. 34–45, 2018.
[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.
[13] G. Saleem, M. Akhtar, N. Ahmed, and W. Qureshi, “Automated analysis of leaf diseases using CNN and SVM,” Comput. Electron. Agric., vol. 194, p. 106789, 2022.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY