*Journal of Universal Computer Science, vol. 31, no. 6 (2025),623-647*
*submitted:15/6/2024, accepted:17/12/2024, appeared:28/5/2025* 

# Plant Leaf Recognition using OSSGabor filter and Vision Transformer

**Thuy Phuong Khuat**
(Computer Science Center, University of Science–Vietnam National University Ho Chi Minh City, and HUTECH University, Ho Chi Minh City, Vietnam
https://orcid.org/0009-0004-3349-3363, kt.phuong@csc.hcmus.edu.vn, ktphuong24nct@hutech.edu.vn)

**Trang Van**
(Faculty of Information Technology, Ho Chi Minh City University of Economics and Finance, Ho Chi Minh City, Vietnam
https://orcid.org/0000-0001-9254-5937, trangvtt@uef.edu.vn)

**Hoang Thien Van***
(Faculty of Information Technology, HUTECH University, Ho Chi Minh City, Vietnam
https://orcid.org/0009-0009-9359-1359, vt.hoang@hutech.edu.vn)

**Abstract:** Deep learning methods are increasingly used in automated plant species classification systems to support biodiversity conservation and ecological monitoring, particularly for medicinal plants. This study presents a novel approach to plant leaf recognition by integrating the Vision Transformer (ViT) model with the OSSGabor filter, termed the OGViT method. The OSSGabor filter is a leaf feature extraction technique that combines the responses of Gabor filters in 16 directions and optimizes their parameters using the Structural Similarity Index Measure (SSIM). These features capture intricate details such as leaf veins, texture, and frequency variations, which are essential for enabling ViT to fully leverage deep learning for leaf recognition. Experimental results on four public datasets—Swedish Leaf, Flavia, Folio, and UCI Leaf—demonstrate that the OGViT method outperforms state-of-the-art approaches, achieving accuracy scores of 100%, 100%, 100%, and 98.88%, respectively, with a 20% testing set and an 80% training set. This performance highlights the effectiveness of the proposed method for plant classification, offering a robust tool with potential applications in agriculture and biodiversity conservation.

**Keywords:** Plant classification, SSIM, Gabor Filter, Vision Transformer, Deep Learning
**Categories:** I.0, I.2, I.3, I.4, I.5, I.6, J.3
**DOI:** 10.3897/jucs.129624

## 1 Introduction

Plants play fundamental and pivotal roles in various crucial fields, including agriculture, industry, medicine, and environmental protection [Kaya et al., 2019, Lv and Zhang, 2023, Zhang et al., 2020]. Therefore, accurate plant species identification is vital across disciplines such as agronomy, conservation, and drug discovery. An automated system for plant species recognition is essential for botanical gardens, conservation efforts, and discovering new species [Zhang et al., 2020]. Such systems aim to assist non-experts in identifying plants rapidly and efficiently, saving time and resources [Kaya et al., 2019, Lv and Zhang, 2023]. Plant identification often relies on various parts like leaves, roots, branches, fruits, and veins. Among these, leaves offer a wealth of information, such as shape, texture, and color, making them a focus for developing plant identification methods [Kaya et al., 2019, Lv and Zhang, 2023, Yang, 2021, Zhang et al., 2020]. These methods typically fall into two categories: image processing-based techniques and deep learning-based techniques. Although traditional image processing techniques such as Gabor filters have made significant progress and proven effective in some specific cases [Chaki and Parekh, 2012, Chi et al., 2003], they still have certain limitations. The generalizability of these methods is often limited, especially when faced with diverse and complex datasets. In addition, they often require high-quality input data, intricate feature engineering, and careful parameter optimization. Deep learning, especially Convolutional Neural Networks (CNNs), has significantly transformed the field of plant classification [Dyrmann et al., 2016, Lee et al., 2015]. Although CNNs have shown the ability to capture intricate features from images for precise plant identification, they also require extensive and diverse datasets to minimize the risk of overfitting. To solve this challenge, scientists have proposed transfer learning, a practical solution that allows models to leverage pre-trained knowledge from large databases, thereby enhancing generalization and significantly reducing training duration. While CNNs and transfer learning have significantly advanced plant classification, they face limitations in capturing intricate spatial relationships present in images. Vision Transformers (ViTs) [Dosovitskiy et al., 2021] offer a novel approach to image processing, employing an attention mechanism to effectively capture detailed spatial relationships within images. ViTs have shown strong potential in image classification tasks, particularly in plant classification.

The objective of this study is to explore and demonstrate the enhanced performance of the Vision Transformer model when combined with images processed through a Gabor Filter to achieve high accuracy in plant classification. By integrating advanced technology with established knowledge, this methodology strives to significantly enhance the precision and efficiency of plant identification across various ecosystems. The contributions of this paper:

- Introducing the OGViT method, which integrates the ViT model with OSSGabor filters to enhance plant classification accuracy.
- Proposing the OSSGabor filter, which combines the responses of Gabor filters in 16 directions and optimizes their parameters using the Structural Similarity Index Measure (SSIM), capturing intricate leaf details such as veins, texture, and frequency variations for effective leaf recognition
- Extending ViT models to plant classification, demonstrating their adaptability and strong performance.
- Comprehensively evaluating the OGViT method alongside state-of-the-art approaches on four public datasets (Swedish Leaf, Flavia, Folio, and UCI Leaf) to demonstrate its superior classification accuracy and robust plant identification.

The remainder of the paper is structured as follows: Section 2 shows the related works. Section 3 presents the methodology, including feature extraction techniques and the proposed model. Section 4 covers the analysis and discussion of the results, and Section 5 concludes the paper.

## 2 Related Works

Plant classification has received increased attention and practice in recent years, with various methods and techniques focusing on image processing-based techniques and deep learning-based techniques.

The application of image processing methods for extracting plant features and classification has been extensively studied. Early approaches leveraged Gabor filter banks to analyze texture features, demonstrating their effectiveness in discriminating between species based on bark and leaf textures [Chi et al., 2003, Ishak et al., 2009]. Chi et al. [Chi et al., 2003] pioneered this approach by constructing Gabor filter banks and a classifier based on extracted texture features from plant bark. Ishak et al. [Ishak et al., 2009] further advanced this field by introducing an image analysis technique that integrated Gabor wavelets with gradient field distribution techniques to extract a unique set of feature vectors, leveraging directional texture properties for classifying weed species. Cope et al. [Cope et al., 2010] proposed a texture-based approach for plant classification, which involved calculating joint distributions at multiple scales of the Gabor filter. Chaki et al. [Chaki and Parekh, 2012] developed an automated plant species recognition system that applied Gabor Filter analysis to leaf images, with an emphasis on varying filter parameters to optimize accuracy.

Besides Gabor Filter, subsequent studies explored various approaches for plant leaf recognition. Zhang et al. [Zhang and Tao, 2015] achieved an accuracy of 91.19% on the Flavia dataset by using the wavelet fractal method in combination with the back propagation neural network. Chaki et al. [Chaki et al., 2015] achieved 97.6% accuracy on the Flavia leaf dataset by combining a Gabor filter and greyscale co-occurrence matrix for texture features and a multi-layered perceptron with a neuro-fuzzy controller classifier. Naresh et al. [Naresh and Nagendraswamy, 2016] employed an improved local binary pattern (MLBP) for leaf texture extraction, achieving accuracies of 97.55% and 96.83% on Flavia and Swedish datasets. Saleem et al. [Saleem et al., 2019] explored leaf visual features and various extraction techniques, achieving 97.6% accuracy on Flavia. Goyal et al. [Goyal et al., 2019] introduced a multi-class dual support vector machine method, reaching 98.11% accuracy on Flavia. Su et al. [Su et al., 2020] proposed plant classification method by extracting curvature, texture, and shape region features from leaf contours, achieving accuracies of 99.35% and 99.43% on the Swedish and Flavia plant leaf datasets, respectively.

Recent research has applied deep learning techniques, particularly CNNs, into the field of plant classification. Studies, such as Lee et al. [Lee et al., 2015] in 2015, utilized CNNs to achieve a remarkable 99.5% accuracy with unsupervised feature representations for 44 plant species. In 2016, Dyrmann et al. [Dyrmann et al., 2016] adopted CNNs for plant species identification in color images, achieving 86.2% accuracy across 22 species and aiding site-specific weed management. In 2017, Lee et al. [Lee et al., 2017] highlighted the capability of CNNs for plant classification by learning vein patterns directly from leaf images, highlighting the importance of hierarchical feature representation and contextual information. Studies have showed that CNNs are highly effective in plant classification, but they also have limitations, especially when working with limited datasets. Training a CNN from scratch requires a large and diverse dataset to avoid overfitting, as well as significant computational resources. When the dataset is not large or diverse enough, the model is prone to overfitting, resulting in poor performance on new data. In this context, transfer learning has emerged as an effective solution. By leveraging knowledge from models that have been pre-trained on large datasets, transfer learning allows fine-tuning the model for a specific plant classification task with less data. This not only reduces training time but also significantly improves the generalization ability of the model, allowing it to perform better on new and diverse data.

In 2017, Sun et al. [Sun et al., 2017] successfully utilized ResNet26 to classify plant species from image data. In 2019, Aydin Kaya et al. [Kaya et al., 2019] demonstrated that transfer learning, particularly with fine-tuning and deep feature extraction, significantly enhances deep learning model performance for automated plant identification. Meanwhile, Diazet al. [Mamani Diaz et al., 2019] proposed a deep learning system in the same year, with Xception outperforming the other models with 86.21% accuracy on the public dataset “Plant Seedlings Dataset”, and emphasized the impact of GPU hardware on classification model results. In 2021, Roopashree et al. [Roopashree and Anitha, 2021] introduced DeepHerb, a dataset of 40 Indian herbs. They used pre-trained deep learning models like VGG16, VGG19, InceptionV3, and Xception and achieved 97.5% accuracy with their proposed DeepHerb model, trained with Xception and ANN. In the same year, Venkatesh et al. [Venkatesh et al., 2021] proposed a fine-tuned MobileNet CNN model for fruit classification, achieving high accuracy (approximately 98.60%) with a low loss rate (around 0.38%) while maintaining low computational cost.

Although these studies have achieved impressive results, they also show that traditional deep learning methods can still struggle when dealing with limited or non-diverse datasets. Thus, it remains important to develop specialized datasets suitable for specific plants or tasks. In line with this, Abayomi-Alli et al. [Abayomi-Alli et al., 2024] presented FruitQ, a dataset for fruit quality assessment using five deep learning models (ShuffleNet, SqueezeNet, EfficientNet, ResNet18, and MobileNet-V2), highlighting the need for diverse and well-annotated datasets in the field of plant classification.

While powerful, CNNs and transfer learning struggle to fully grasp complex spatial relationships in images. In 2020, Dosovitskiy et al. [Dosovitskiy et al., 2021] demonstrated that pure transformer models, when directly applied to sequences of image patches, can achieve excellent classification performance without relying on CNN architectures. In 2021, Conde and Turgutlu [Conde and Turgutlu, 2021] introduced a multi-stage, multi-scale fine-grained visual classification framework based on ViT, utilizing a multi-head self-attention mechanism to capture distinctive features from diverse local regions. Additionally, Van Hieu et al. [Van Hieu et al., 2023] proposed PlantKViT, a hybrid approach that integrates ViT with the KNN algorithm, achieving a remarkable 93% accuracy in classifying plants from the Danang Forest Plant dataset, surpassing the ConvNeXt model (89%) and the Resnet-152 model (76%).

Recent studies have proposed a hybrid approach to improve plant classification. Specifically, Kayaalp & Kiyas [Kayaalp, 2024] proposed a hybrid deep learning method to classify medicinal plant leaves based on subtle visual cues, addressing the difficulties in distinguishing visually similar species. Furthermore, Qiu et al. [Qiu et al., 2023] demonstrated the potential of integrating hyperspectral imaging with deep learning for enhancing feature extraction and improving classification accuracy, particularly in distinguishing different vigor of Osmanthus fragrans seeds. Both of these studies highlight the promise of integrating multiple techniques to tackle the challenges of plant classification, motivating the exploration of hybrid approaches that combine spatial and texture-based feature extraction.

Building on these developments, we introduce the OGViT method, which integrates Gabor filters with the ViT model to take advantage of the strong capabilities of both to improve the efficiency of plant classification.

# 3 Our proposed method-OGViT

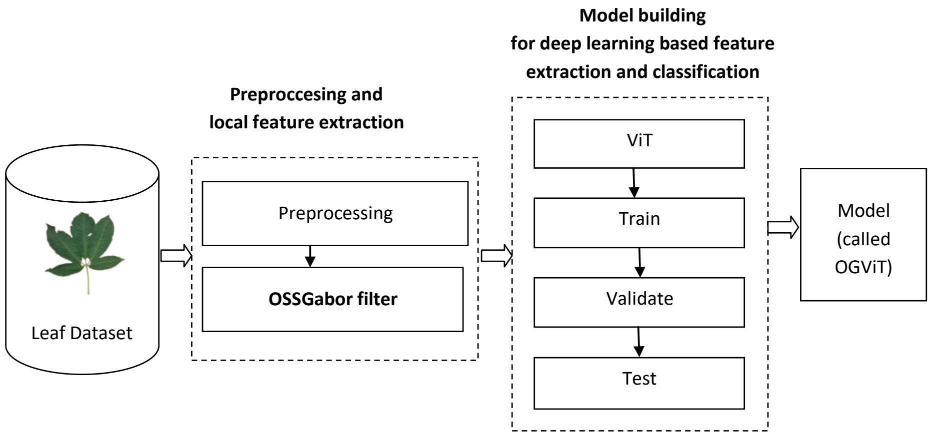



*Figure 1: The proposed framework for leaf recognition*

These related studies demonstrate the evolution of image processing techniques and the application of deep learning models in plant classification. ViTs offer several advantages over CNNs, particularly the ability to efficiently learn long-range dependencies in images through a self-attention mechanism. This allows ViTs to consider relationships between all patches in an image, regardless of their spatial distance, overcoming the limitations of CNNs, which struggle to capture long-range dependencies due to the limited receptive field of convolutional filters. Moreover, ViT has better scalability, especially when processing high-resolution images or large datasets. By using linear transformations and self-attention, ViT is highly parallelizable, enabling fast and efficient processing even with vast amounts of data, making it particularly suitable for modern vision tasks that require large datasets. However, ViT faces challenges in representing texture and frequency features, where Gabor filters excel, particularly in texture analysis, edge detection, and feature extraction. To address this, we propose a novel approach that combines Vision Transformer models with Gabor filters to enhance plant classification accuracy. Our proposed leaf feature extraction method, called the OSSGabor filter, integrates Gabor filter responses in 16 directions and optimizes the filter parameters using the Structural Similarity Index Measure (SSIM). This method aims to highlight leaf veins and capture subtle nuances in texture and frequency, which are crucial for boosting the ViT model's performance. A visual representation of our proposed framework for leaf recognition is illustrated in Figure 1. The following subsections will provide detailed descriptions of each processing step within our method.

### 3.1 Pre-processing

The pre-processing steps involve resizing the image and removing the background to reduce noise [Elhariri et al., 2014]. First, we convert the input image to grayscale and apply a threshold to separate the candidate object from the background regions. We then apply the GrabCut method [Goy et al., 2023, Joseph et al., 2020], with the iteration count set to 5, to compute the leaf mask and detect the leaf region on the original RGB image. Next, we convert the leaf region image to the CIELAB color space, apply a threshold, and perform morphological erosion and dilation using a 3x3 structural element on the binary mask to remove shadows and noise. Finally, we multiply the RGB image by the binary mask to extract the leaf from its background. Figure 2 shows a flowchart of the pre-processing phase steps.

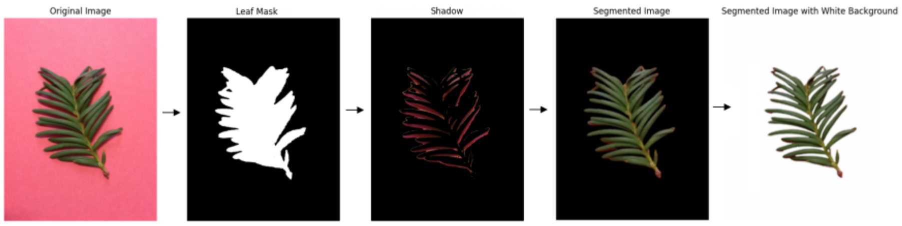



*Figure 2: Flowchart of the pre-processing phase steps*

### 3.2 Gabor filter

The Gabor filter is a type of convolution filter that combines a Gaussian function with a sinusoidal component, providing both spatial localization and directional sensitivity [Chaki and Parekh, 2012, Cope et al., 2010, Van et al., 2025]. The Gaussian function acts as a weighting factor, while the sinusoidal term introduces directional selectivity. The formulation of the Gabor filter is as follows: [Chaki and Parekh, 2012, Cope et al., 2010]:

$$g(x, y, \lambda, \theta, \psi, \sigma, \gamma) = exp\left\{-\frac{\acute{x}^2 + \gamma^2 \acute{y}^2}{2\sigma^2}\right\} exp\left\{i\left(2\pi\frac{\acute{x}}{\lambda} + \psi\right)\right\},$$
$$\acute{x} = xcos\theta + ysin\theta, \acute{y} = -xsin\theta + ycos\theta \quad (1)$$

here, $i = \sqrt{-1}$, $\lambda$ represents the wavelength of the sinusoidal component, $\theta$ defines the angle at which the filter is applied, $\psi$ represents the phase offset, which shifts the sinusoidal component of the filter, $\sigma$ controls the spread or width of the Gaussian function and $\gamma$ controls the ellipticity of the filter. Various digital filters (kernels) can be generated by adjusting these Gabor parameters (Figure 3). The selection of these parameters is critical, as they directly affect the filter's ability to capture important image features. The parameters $\lambda$, $\theta$, $\sigma$, and $\gamma$ (Table 1) were determined through experiments to optimize the detection of key leaf textures like vein patterns and structural details. Each parameter was varied within a specified range to evaluate its impact on feature clarity and completeness in controlled leaf images. In image processing, Gabor filters are widely used in edge detection, texture analysis, and feature extraction. Their ability to resonate with features of similar frequency and orientation makes them particularly adept at detecting edges and analyzing textures by capturing characteristics like orientation and scale. Furthermore, they excel at extracting salient image features such as lines, curves, and corners, which are vital for object recognition.

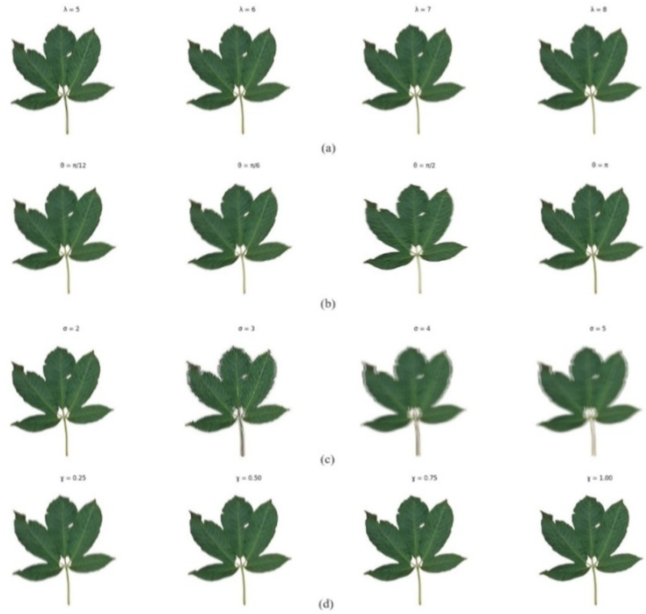



*Figure 3: Gabor Filter Response with Varying Parameters:(a) only change the lambda parameter from 5 to 8, with fixed parameters ($\theta = 0$, $\gamma = 0.25$, $\sigma = 2$, $\Psi = 0$), (b) only change the theta parameter to $\pi/12$, $\pi/6$, $\pi/2$ and $\pi$, with fixed parameters ($\lambda = 6$, $\gamma = 0.25$, $\sigma = 2$, $\Psi = 0$), (c) only change sigma parameter from 2 to 5, with fixed parameters ($\theta = 0$, $\lambda = 6$, $\gamma = 0.25$, $\Psi = 0$), and (d) only change gamma parameter to 0.25, 0.5, 0.75 and 1, with fixed parameters ($\theta = 0$, $\lambda = 6$, $\sigma = 2$, $\Psi = 0$)*

### 3.3 OSSGabor filter for optimal structure features

The Gabor filter captures local structural details like spatial frequency, location, and direction, but its effectiveness depends on carefully selecting the right parameters to enhance image structure while preserving fine details and minimizing sensitivity to lighting variations. To address this, we propose the OSSGabor filter, a leaf feature extraction approach that aggregates Gabor filter responses from 16 different orientations and optimizes the filter parameters using SSIM [Li et al., 2020]. SSIM, a full-reference metric for evaluating image similarity, measures brightness, contrast, and structural consistency, with values ranging from 0 (completely different) to 1 (identical), indicating the degree of image preservation. The formula for the OSSGabor filter is defined as follows:

$$f_{ossg} = \frac{1}{16}\sum_{\theta_i=\frac{\pi(i-1)}{16}, i=1,2,..,16} f * g(x, y, \lambda, \theta_i, \psi, \sigma, \gamma) \quad (2)$$

where $f_{ossg}$ is the filtered image, and the parameters (λ, ψ, σ, γ, and filter size) are optimized based on the training dataset through the following procedures:

1. Parameter Grid Initialization (Table 1): This step involves defining parameters like (λ, ψ, σ, γ, and filter size).
2. Generating a set of Gabor filters using the initialized parameter grid.
3. Apply formula (2) to each set of Gabor filters corresponding to a parameter set, and the resulting images are then converted to grayscale.
4. For each grayscale image, the SSIM value is calculated with its corresponding original image using the following formula [Li et al., 2020]:

$$SSIM(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \cdot \frac{2\delta_{xy} + C_2}{\delta_x^2 + \delta_x^2 + C_2} \quad (3)$$

where, $\delta_x^2$ and $\delta_y^2$ are the variances, μx and μy represent the mean values, C1 and C2 are constants to prevent division by zero, and δxy is the covariance between the original and filtered images.

5. If the current average SSIM exceeds the previously achieved best SSIM, the optimal parameter set is updated accordingly.
6. Steps 2 to 5 are repeated for each parameter combination in the grid..
7. After iterating through all parameter combinations, the best parameter set and its corresponding SSIM value are provided as output.

In this study, the parameter ranges in Table 1 are chosen based on the image size and resolution in the datasets. For leaf images with medium resolution, sizes such as (15,15) and (21,21) offer a good trade-off between preserving detail and ensuring computational efficiency. We varied λ from 5 to 8 to capture the appropriate texture features for leaf classification. Smaller lambda values are effective at capturing fine details but may also enhance noise and irrelevant textures, potentially complicating feature extraction. In contrast, higher λ values are more effective at reducing noise but may also blur important fine textures necessary for precise classification. The standard deviation σ was chosen within the 2 to 5 range, balancing detail preservation and noise reduction based on preliminary observations. Lastly, the range for γ (0.25, 0.5, 0.75) was selected to capture both elongated structures, like leaf veins, and more isotropic features, reflecting the varied morphological characteristics of leaf surfaces.

| **Parameter** | **# Values** |
|---|---|
| filter size | [(15,15), (21,21)] |
| σ | [2, 3, 4, 5] |
| λ | [5, 6, 7, 8] |
| ɣ | [0.25, 0.5, 0.75] |
| Ψ | [0] |

*Table 1: Parameter grid*

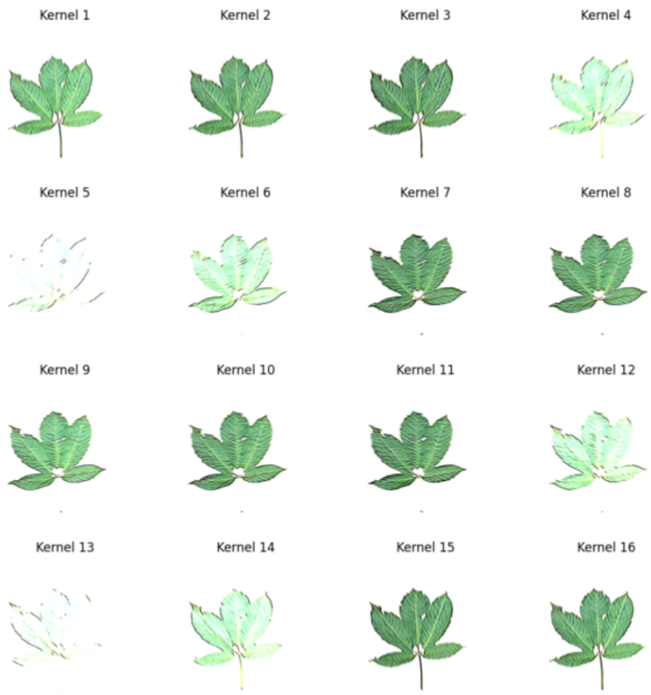



*Figure 4: 16 filtered images correspond to 16 different directions*

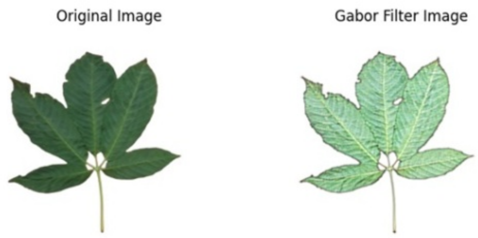



*Figure 5: Comparison between the original image and the image filtered by the OSSGabor filter*

Figure 4 shows the 16 filtered images correspond to 16 different directions for computing the optimal filtered image. Figure 5 illustrates a comparison between the original image and the filtered image using the OSSGabor filter, showcasing clearer texture while maintaining insensitivity to lighting variations. The filtered image retains detailed local structural information, including spatial frequency, location, and directionality, while also maintaining the original image's key structural features.

### 3.4 OGViT for leaf recognition

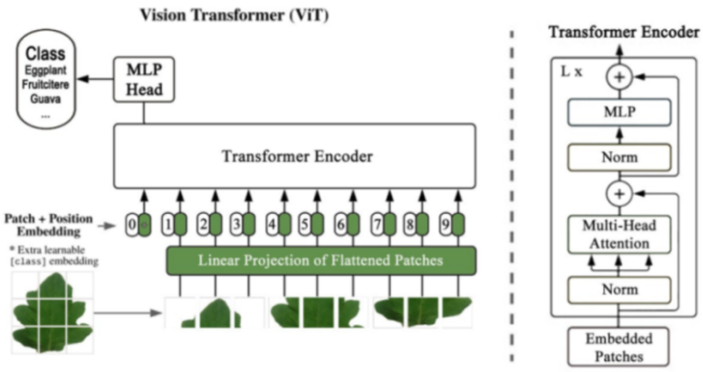
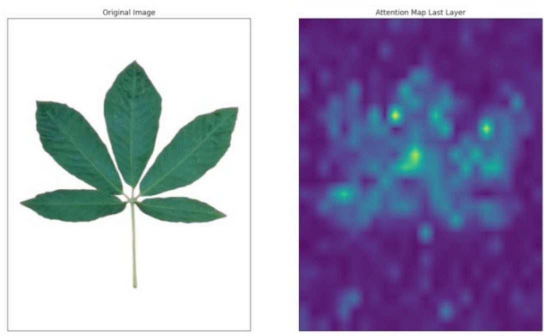



*Figure 6: ViT architecture*

The ViT [Dosovitskiy et al., 2021] marked a significant shift in computer vision by applying the transformer architecture, originally designed for natural language processing, to image analysis. Unlike convolutional neural networks (CNNs) that rely on localized spatial filtering, ViT captures both fine-grained and global contextual information through self-attention [Van et al., 2025]. The model processes images by first breaking them into a series of fixed-size patches, typically 16×16 for a 224×224 input image, resulting in 196 patches. Each patch is then flattened and projected into a high-dimensional vector using a learnable linear projection, effectively treating each patch as a token, analogous to word tokens in NLP. To maintain spatial awareness, ViT incorporates positional encodings that provide information about the relative positions of these patches, ensuring that the model captures spatial structure despite its non-convolutional design. This sequence of patch embeddings, combined with a special classification token (CLS token), is then fed into a stack of transformer encoder layers. Each encoder layer includes two key components: multi-headed self-attention (MSA), which captures long-range dependencies by allowing the model to focus on multiple image regions simultaneously, and a multi-layer perceptron (MLP) for non-linear feature transformation. The MLP consists of fully connected layers with Gaussian Error Linear Unit (GELU) activation, which helps capture complex feature interactions. To enhance training stability, each encoder layer also incorporates layer normalization (LN) and residual connections, ensuring that gradient flow is preserved throughout the network. After passing through these encoder layers, the final representation of the CLS token is extracted and processed through a classification head to generate the model's output. This combination of global attention and positional encoding enables ViT to outperform traditional CNNs in tasks requiring a more comprehensive understanding of image content.

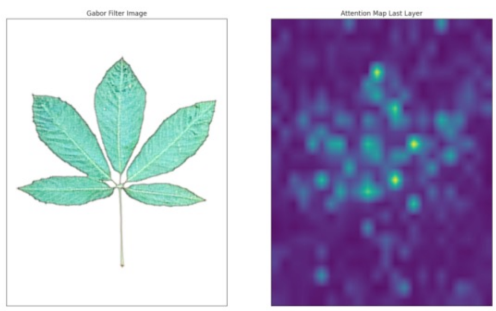
In this study, we propose the OGViT method, a combination of OSSGabor filters and ViT, adapted to better suit the task of classifying leaf images. In the OGViT method, the original images are first preprocessed to detect the leaf regions, after which the OSSGabor filter is applied. The filtered images are then used as inputs for training and testing the ViT model. The ViT model in our leaf recognition method consists of

12 transformer encoder layers, each containing 12 attention heads. Figure 7 illustrates an example of the original image and its corresponding 'attention map from the last layer.' Figure 8 shows the OSSGabor-filtered image and its corresponding 'attention map from the last layer.' The contrast between the attention maps in Figures 7 and Figure 8 demonstrates that ViT utilizes more regions (with higher values) from the filtered image for classification than from the original image, significantly improving classification effectiveness.




(a) (b)

*Figure 7: Key regions influencing the model's decision: (a) The original image and (b) the corresponding last-layer attention map.*




(a) (b)

*Figure 8: Key regions influencing the model's decision: (a)The OSSGabor-filtered image and (b) the corresponding last-layer attention map.*

# 4 Experimental Results

## 4.1 Datasets

In this study, we utilized four public plant datasets: Swedish Leaf [J. O. Söderkvist, 2001], Flavia[Wu et al., 2007], Folio [Pudaruth, 2015], and UCI Leafdataset [Silva, 2014], to assess the effectiveness of deep learning models in medicinal plant classification.Figure 9 shows sample images of the Swedish Leaf, Flavia, Folio, and UCI Leaf datasets.

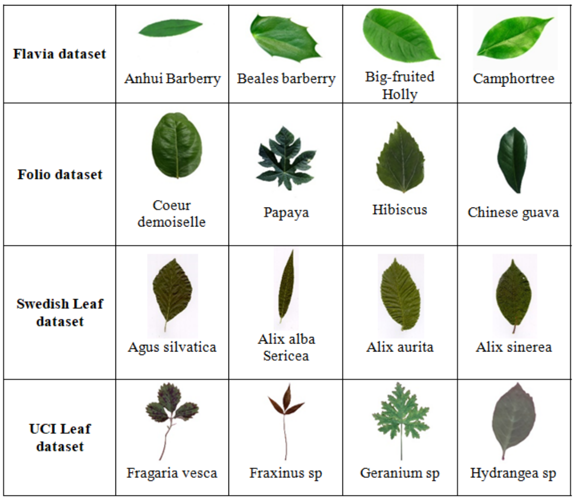



*Figure 9: Sample images from the Flavia, Folio, Swedish Leaf, and UCI Leaf datasets*

Table 2 shows the infomation for these datasets. Swedish Leafand Flavia consist of images with relatively high clarity and brightness, with each category containing approximately 35 to 75 images - a sample size sufficient for effectively training the models. Conversely, Folio and UCI Leaf datasets comprise low-quality images and a small number of images per class, which evaluate the model's performance with limited data per class. The selection of these diverse datasets facilitates a comprehensive exploration of the model's robustness across varying image qualities and dataset sizes.

| Dataset | Classes Number | Sample Total | Color | Average Images per Class |
|---|---|---|---|---|
| Swedish Leaf | 15 | 1125 | RGB | 75 |
| Flavia | 32 | 1097 | RGB | 35 |
| Folio | 32 | 637 | RGB | 20 |
| UCI Leaf | 40 | 443 | RGB | 11 |

*Table 2: Properties of the experimented datasets*

### 4.2 Evaluation Metrics

The performance of our proposed method is assessed through several evaluation metrics, including Accuracy (Acc), Precision (Pre), Recall (Re), and F1-score (F1). These metrics offer a thorough evaluation of the model's performance in different classification areas. The computation of these metrics for multi-class classification, utilizing macro-averaging, is outlined in Equations (4)–(7):

$$AvgAcc = \frac{1}{classes}\sum_{i}^{classes} Acc(i)\ ,\ Acc(i) = \frac{TP(i)+TN(i)}{TP(i)+TN(i)+FP(i)+FN(i)} \quad (4)$$

$$AvgRe = \frac{1}{classes}\sum_{i}^{classes} Re(i)\ ,\ Re(i) = \frac{TP(i)}{FN(i)+TP(i)} \quad (5)$$

$$AvgPre = \frac{1}{classes}\sum_{i}^{classes} Pre(i)\ ,\ Pre(i) = \frac{TP(i)}{TP(i)+FP(i)} \quad (6)$$

$$AvgF_1 = \frac{1}{classes}\sum_{i}^{classes} F_1(i)\ ,\ F_1(i) = 2 \times \frac{Pre(i)\times Re(i)}{Pre(i)+Re(i)} \quad (7)$$

where, for class *i*, TP (True Positive) denotes instances correctly identified as belonging to class *i*, TN (True Negative) refers to instances correctly identified as not belonging to class *i*, FP (False Positive) represents instances incorrectly labeled as class *i*, and FN (False Negative) represents instances incorrectly labeled as not belonging to class *i*.

### 4.3 Results

The test models were trained for a fixed duration of 50 epochs, with early stopping implemented. The early stopping criterion was configured to halt training if there was no improvement in performance for 5 consecutive epochs, with a threshold of 0.0 for change. The performance was evaluated using the 'accuracy' metric. The datasets were divided into two segments: 80% for training and 20% for testing. To enhance generalization and avoid overfitting, the K-Fold Cross-Validation method [Saud et al., 2020, Shiddiq et al., 2024] was employed in the experiments. With k set to 5, each subset was used as the validation set once, while the remaining k−1 subsets were used for training. The experiments were conducted on the Google Colab platform, utilizing T4 GPUs. We conducted two experiments to evaluate the effectiveness of the proposed model compared to state-of-the-art deep learning methods.

#### 4.3.1 Result of Experiment 1: Demonstrating that ViT outperforms VGG16, Xception, MobileNet, and DenseNet201 for plant classification

In the first experiment, we evaluated five different models, including VGG16, Xception, MobileNet, DenseNet201, and ViT, using the original dataset to determine the most effective architecture. The Vision Transformer (ViT) model demonstrated superior performance in terms of accuracy and F1 scores across all datasets, as shown in Table 3. Figure 10 shows that the ViT method consistently achieves the highest accuracy compared to the other methods across all datasets. The confusion matrices for the ViT method across all datasets are presented in Figures 11 and Figure 12. This outperformance is consistent and significant when compared to models like VGG16, Xception, MobileNet, and DenseNet201, indicating that the ViT model's architecture is highly effective and generalizes well for various leaf classification tasks. Given its high accuracy, the ViT model emerges as a particularly suitable choice for applications where optimal performance is crucial.

As illustrated in Table 4, the ViT model has a longer training time compared to the other models, but its testing time per sample is similar to that of the other methods. Notably, the ViT method is the smallest in size, Figure 13. These experiments demonstrate that the ViT method not only achieves high accuracy but also provides fast recognition with a compact model size. Therefore, we developed the proposed method based on ViT combined with the OSSGabor filter, which is evaluated in Experiment 2.

| **Dataset** | **VGG16** | **Xception** | **MobileNet** | **DenseNet201** | **ViT** |
|---|---|---|---|---|---|
| **Swedish Leaf** | | | | | |
| Accuracy (%) | 100 | 98.22 | 100 | 100 | 100 |
| F1 Score (%) | 100 | 98.00 | 100 | 100 | 100 |
| **Flavia** | | | | | |
| Accuracy (%) | 98.98 | 98.47 | 98.47 | 99.49 | 100 |
| F1 Score (%) | 99.00 | 98.49 | 98.49 | 99.49 | 100 |
| **Folio** | | | | | |
| Accuracy (%) | 92.97 | 93.75 | 94.53 | 96.09 | 99.22 |
| F1 Score (%) | 93.00 | 94.00 | 94.00 | 96.00 | 99.20 |
| **UCI Leaf** | | | | | |
| Accuracy (%) | 90.57 | 88.68 | 97.17 | 96.23 | 97.75 |
| F1 Score (%) | 90.00 | 88.00 | 96.00 | 96.00 | 97.33 |

*Table 3: Comparative performance of five models (VGG16, Xception, MobileNet, DenseNet201, ViT) across datasets on the test dataset*

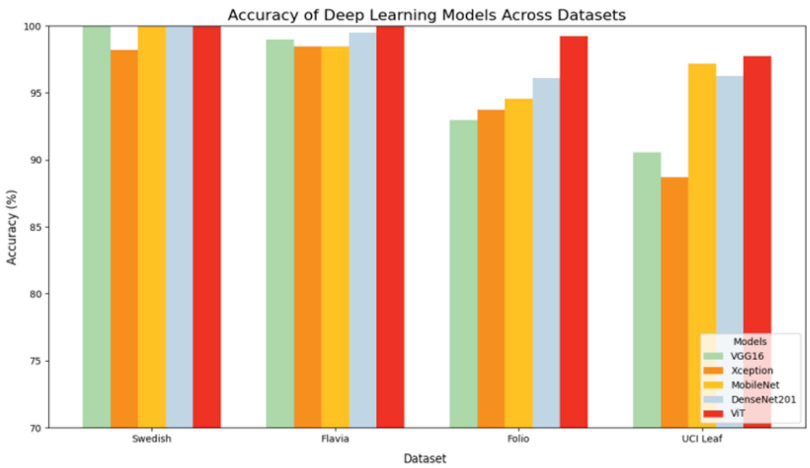
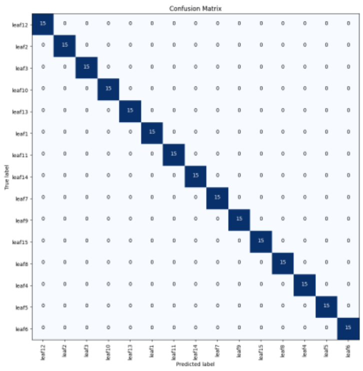
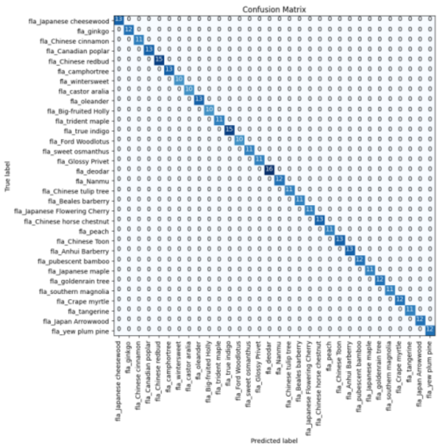



*Figure 10: Model accuracy comparison across datasets*

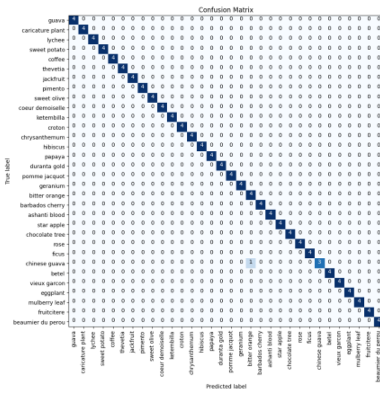



(a)

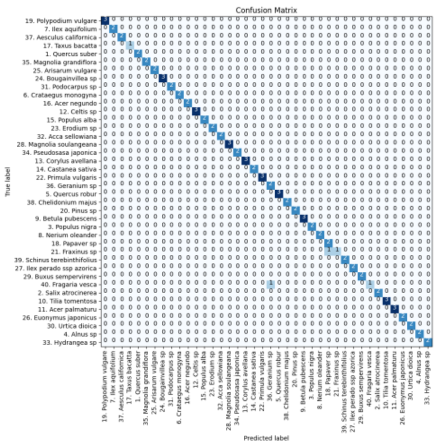



(b)

*Figure 11: Confusion matrix of the ViT model on (a) the Swedish Leaf dataset and (b) the Flavia dataset, respectively*




(a)




(b)

*Figure 12: Confusion matrix of the ViT model on (a) the Folio dataset and (b) the UCI Leaf dataset, respectively*

| Dataset | VGG16 | Xception | Mobile-Net | Dense-Net201 | ViT |
|---|---|---|---|---|---|
| **Swedish Leaf** | | | | | |
| Training time (second) | 836 | 552 | 489 | 803 | 1995 |
| Test time (second/sample) | 0.02 | 0.05 | 0.03 | 0.02 | 0.08 |
| **Flavia** | | | | | |
| Training time (second) | 2546 | 3071 | 2535 | 1630 | 2921 |
| Test time (second/sample) | 0.03 | 0.03 | 0.02 | 0.02 | 0.06 |
| **Folio** | | | | | |
| Training time (second) | 1755 | 1386 | 1135 | 1466 | 2430 |
| Test time (second/sample) | 0.08 | 0.09 | 0.08 | 0.09 | 0.13 |
| **UCI Leaf** | | | | | |
| Training time (second) | 279 | 463 | 397 | 583 | 992 |
| Test time (second/sample) | 0.01 | 0.03 | 0.01 | 0.02 | 0.04 |

*Table 4: Comparison of Processing Time for Models across Datasets*

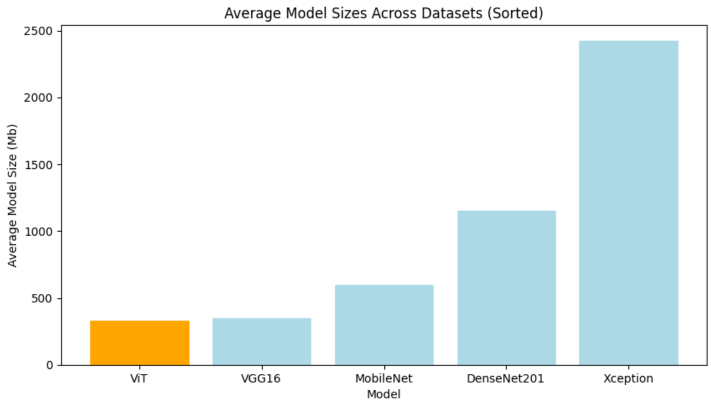



*Figure 13: Comparison of model sizes (Mb) across datasets*

### 4.3.2 Result of Experiment 2: Prove that OSSGabor filters can help improve model accuracy

In this experiment, we conducted an extended analysis using the ViT model on Gabor filter-processed data. We then compared the performance of two different model configurations—using the original dataset and the OSSGabor filter-processed dataset—to determine the most effective model. We used the parameter grid described in Table 1 (Section 3.3) to find the optimal parameters for the OSSGabor filter. Table 4 presents the optimal parameter set obtained by evaluating and comparing the SSIM values of various filters, covering all possible parameter combinations derived from Table 1.

Figure 14 shows the results of applying the OSSGabor filter with the optimal set of parameters to an image in the Swedish Leaf, Flavia, Folio, and UCI Leaf, respectively.

| **Dataset** | **Best Parameters** (number of filters, filter size, σ, λ, ψ, γ) | **# SSIM values** |
|---|---|---|
| Swedish Leaf | (16, (21, 21), 2, 5, 0.75, 0) | 0.731 |
| Flavia | (16, (15, 15), 2, 6, 0.5, 0) | 0.815 |
| Folio | (16, (15, 15), 3, 7, 0.25, 0) | 0.795 |
| UCI Leaf | (16, (15, 15), 3, 6, 0.75, 0) | 0.742 |

*Table 4: Best parameters of all datasets*

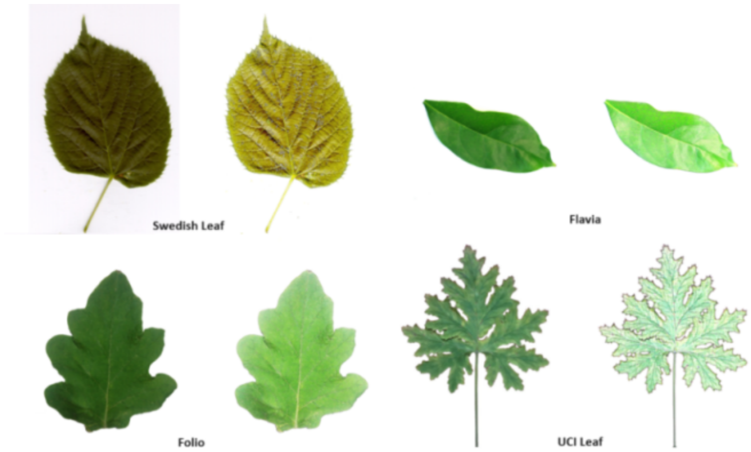



*Figure 14: Original image and corresponding Gabor filter responses across different datasets*

| **Name** | **Parameter Set** (number of filters, filter size, σ, λ, ψ, γ) | **Accuracy (%)** | **F1 score (%)** |
|---|---|---|---|
| Set 1 | (16, (15, 15), 3, 6, 0.5, 0) | 96.63 | 96.42 |
| Set 2 | (4, (15, 15), 3, 6, 0.75, 0) | 97.75 | 97.81 |
| Set 3 | (8, (15, 15), 3, 6, 0.75, 0) | 97.75 | 98.04 |
| Set 4 | (16, (15, 15), 3, 7, 0.75, 0) | 98.87 | 98.81 |
| Set 5 | (16, (21, 21), 3, 6, 0.5, 0) | 95.51 | 93.98 |
| **Optimal Parameter Set** | **(16, (15, 15), 3, 6, 0.75, 0)** | **98.88** | **99.00** |

*Table 5: ViT accuracy comparison with different Gabor parameter sets on the UCI Leaf Dataset*

To validate the selection of the optimal parameter set using SSIM, additional experiments were conducted with various parameter configurations. These included sets with four and eight directions while keeping other parameters constant, as well as three sets where only one parameter was altered at a time. The results presented in Table 5 demonstrate that the ViT method, when applied to images filtered with the optimal SSIM-based parameter set, achieved the highest recognition accuracy compared to manually chosen parameter configurations on the UCI Leaf dataset. This confirms the effectiveness of the OSSGabor filter in improving leaf feature extraction for plant classification. Furthermore, Table 6 presents the K-fold cross-validation results of the OGViT method across four different datasets. The proposed algorithm consistently achieved high performance, with an average accuracy (Mean) across all folds exceeding 0.9751, showcasing strong generalization capabilities and the absence of overfitting. The low standard deviation (STD-DEV) across datasets, ranging from 0.0037 to 0.0167, indicates that the model's performance was stable across different folds, validating the effectiveness of the data partitioning process in K-fold validation. Additionally, the model demonstrated near-perfect precision on both the Swedish Leaf and Flavia datasets. Although performance on the Folio dataset was slightly lower, this can be attributed to the dataset's more complex distribution. The model also maintained strong performance on the UCI Leaf dataset with high average precision.

| **Dataset** | **Fold 1** | **Fold 2** | **Fold 3** | **Fold 4** | **Fold 5** | **Mean** | **STD-DEV** |
|---|---|---|---|---|---|---|---|
| Swedish Leaf | 1 | 1 | 0.9956 | 0.9911 | 1 | 0.9973 | 0.004 |
| Flavia | 1 | 0.9921 | 0.9948 | 1 | 1 | 0.9974 | 0.0037 |
| Folio | 0.9921 | 1 | 0.9842 | 1 | 0.9685 | 0.9889 | 0.0132 |
| UCI Leaf | 1 | 0.9551 | 0.9775 | 0.9773 | 0.9659 | 0.9751 | 0.0167 |

*Table 6: Accuracy of the OGViT method through K-fold cross-validation across four datasets*

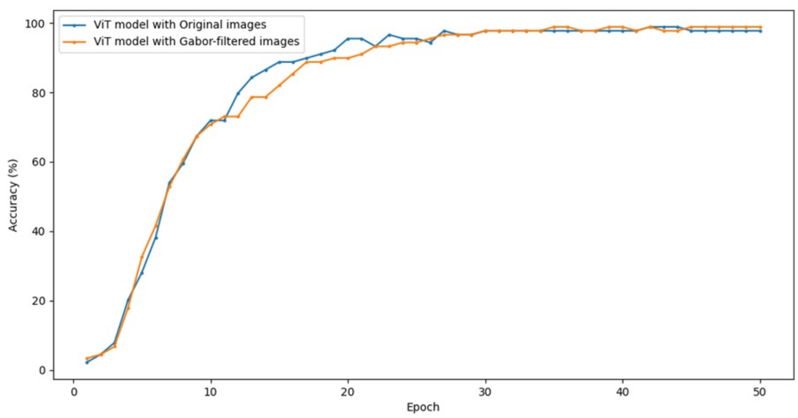
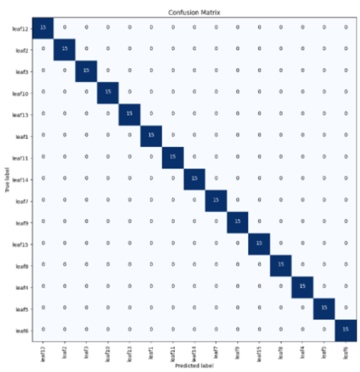


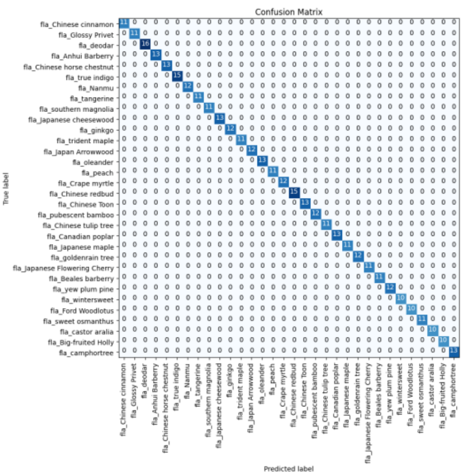

*Figure 15: Accuracies of of ViT model and OGViT model - UCI Leaf*

| **Dataset** | **Accuracy (%)** | **F1 score (%)** | **Test time (second/sample)** |
|---|---|---|---|
| **Swedish Leaf** | | | |
| Original Image | 100 | 100 | 0.08 |
| Gabor Filter images | 100 | 100 | 0.10 |
| **Flavia** | | | |
| Original Image | 100 | 100 | 0.06 |
| Gabor Filter images | 100 | 100 | 0.10 |
| **Folio** | | | |
| Original Image | 99.22 | 99.22 | 0.13 |
| Gabor Filter images | 100 | 100 | 0.16 |
| **UCI Leaf** | | | |
| Original Image | 97.75 | 97.33 | 0.04 |
| Gabor Filter images | 98.88 | 99.00 | 0.07 |

*Table 7: Comparative performance of two models (ViT with original images, ViT with Gabor Filter Images) across datasets*

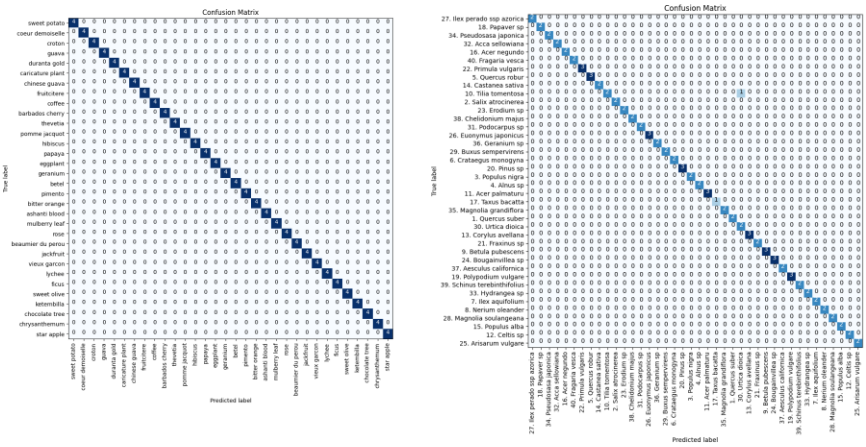


(a)




(b)

*Figure 16: Confusion matrix of the OGViT model on (a) the Swedish Leaf dataset and (b) the Flavia dataset, respectively*



(a) (b)

*Figure 17: Confusion matrix of the OGViT model on (a) the Folio dataset and (b) the UCI Leaf dataset, respectively*

After applying the OSSGabor filter with optimal parameter sets to four datasets, we conducted experiments to compare the performance of the ViT method on the original datasets and the datasets filtered by OSSGabor. Figure 15 presents the accuracy chart of the ViT model on these two datasets during training. The confusion matrices for the OGViT method across all datasets are shown in Figures 16 and Figure 17. The results in Table 7 demonstrate that all four datasets processed with the OSSGabor filter achieved high accuracy when used for model building. For the UCI Leaf dataset, which contains the fewest images per class, the OSSGabor-filtered images achieved an accuracy of 98.88% and an F1 score of 99.00%, outperforming the original images, which reached 97.75% and 97.33%, respectively. Although the proposed method requires slightly more processing time per sample compared to the ViT model on the original dataset, this difference is minor. Given the improved accuracy, the proposed approach remains a strong candidate for plant leaf identification tasks.

### 4.4 Discussion

The primary objective of our study was to evaluate the effectiveness of combining the OSSGabor filter with the ViT model for classifying medicinal plants, a method we call OGViT. Our key finding is a significant improvement in classification accuracy, confirming that the OSSGabor filter enhances the performance of the ViT model. We tested OGViT on four diverse datasets, each presenting distinct challenges, from high- to low-resolution images and varying sample sizes per class. The high accuracy achieved across all datasets highlights the model's versatility, robustness, and adaptability to a range of real-world data conditions, demonstrating its potential for broad application. While OGViT showed strong performance in medicinal plant classification, a limitation of this study is that the evaluation was based on only four datasets. Although these datasets include a wide variety of medicinal plants, the model's performance may vary when applied to different plant species or real-world imaging conditions. Compared with existing research, as detailed in Table 8, our results represent a significant advancement in plant classification. Prior studies have demonstrated the progression of leaf classification techniques, from traditional feature extraction to deep learning, and from single approaches to hybrid methods. While deep learning dominates, traditional methods still show strong performance in specific cases. By combining the strengths of both approaches (traditional feature extraction with deep learning), our study establishes a new benchmark in this field. The integration of the OSSGabor filter with ViT resulted in exceptionally high accuracy, particularly in medicinal plant classification, underscoring the potential of this hybrid approach. In practical terms, precise identification of medicinal plants is crucial in traditional medicine, as it reduces errors and facilitates the discovery of new compounds. Additionally, our research contributes to botany and agriculture by aiding in the conservation of plant diversity and improving crop selection, though its broader impact will depend on ongoing advancements in these fields.

| | **Datasets** | **Best accuracy (%)** | **Method** |
|---|---|---|---|
| Our study | Swedish | 100 | OSSGabor Filter and ViT |
| | Flavia | 100 | |
| | Folio | 100 | |
| | UCI Leaf | 98.88 | |
| [Elhariri et al., 2014] | UCI Leaf | 92.65 | LDA |
| [Arafat et al., 2016] | Flavia | 98.00 | Colored SIFT (CSIFT) |
| [Aakif and Khan, 2015] | Flavia | 96.00 | Fourier descriptors, shape defining-feature and ANN |
| [Lee et al., 2015] | Flavia | 99.40 | CNN, Fine-Tuning |
| [Wick and Puppe, 2017] | Flavia | 99.00 | CNN |
| [Yang, 2021] | Flavia, | 99.10 | Multiscale triangle descriptor and local binary pattern histogram Fourier (LBP-HF) |
| | Swedish Leaf | 98.40 | |
| [Gu et al., 2021] | Folio | 97.90 | VGG16 |
| [Lv and Zhang, 2023] | Flavia<br>Swedish Leaf | 99.30<br>99.52 | LBP, HOG, PCA, and Extreme Learning Machine |

*Table 8: Related works summary –Classification accuracy*

## 5 Conclusions and Future Work

In conclusion, this study presents an innovative approach by combining the OSSGabor filter with the ViT model for plant leaf recognition. The OSSGabor filter serves as a powerful feature extraction technique, leveraging the responses of Gabor filters in 16 directions and optimizing their parameters using the SSIM. This method effectively captures intricate leaf details, such as veins, texture, and frequency variations, which are critical for enabling ViT to maximize its deep learning capabilities for leaf recognition. The significant improvement in classification accuracy across multiple datasets validates the efficacy of this combined approach. While our research focused on integrating OSSGabor with ViT, the potential for OSSGabor to be applied with other deep learning architectures remains largely untapped. Future studies could explore its integration with the state-of-art deep learning models to assess additional performance gains or task-specific advantages. Moreover, the success demonstrated with leaf recognition suggests that OSSGabor could be equally effective in analyzing other botanical structures, such as flowers, stems, roots, or fruits. Expanding the application of OSSGabor could lead to more comprehensive plant identification systems, broadening its utility in botany and beyond.

Future research will explore new deep learning methods and address computational challenges, focusing on larger, diverse plant datasets to refine the approach. This holds promise for advancing botanical research and supporting biodiversity conservation with better plant identification tools.

## References

[Aakif and Khan, 2015] A. Aakif and M. F. Khan, "Automatic classification of plants based on their leaves," Biosystems Engineering, vol. 139, pp. 66-75, 2015.

[Arafat et al., 2016] S. Y. Arafat, M. I. Saghir, M. Ishtiaq, and U. Bashir, "Comparison of techniques for leaf classification," in 2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP), 2016 of Conference: IEEE, pp. 136-141.

[Chaki and Parekh, 2012] J. Chaki and R. Parekh, "Plant leaf recognition using Gabor filter," International Journal of Computer Applications, vol. 56, no. 10, 2012.

[Chaki et al., 2015] J. Chaki, R. Parekh, and S. Bhattacharya, "Plant leaf recognition using texture and shape features with neural classifiers," Pattern Recognition Letters, vol. 58, pp. 61-68, 2015, doi: https://doi.org/10.1016/j.patrec.2015.02.010.

[Chi et al., 2003] Z. Chi, L. Houqiang, and W. Chao, "Plant species recognition based on bark patterns using novel Gabor filter banks," in International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003, 2003 of Conference, vol. 2: IEEE, pp. 1035-1038.

[Conde and Turgutlu, 2021] M. V. Conde and K. Turgutlu, "Exploring Vision Transformers for Fine-grained Classification," ArXiv, vol. abs/2106.10587, 2021.

[Cope et al., 2010] J. S. Cope, P. Remagnino, S. Barman, and P. Wilkin, "Plant texture classification using gabor co-occurrences," in Advances in Visual Computing: 6th International Symposium, ISVC 2010, Las Vegas, NV, USA, November 29–December 1, 2010, Proceedings, Part II 6, 2010 of Conference: Springer, pp. 669-677.

[Dosovitskiy et al., 2021] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," ICLR 2021, doi: 10.48550/arXiv.2010.11929.

[Dyrmann et al., 2016] M. Dyrmann, H. Karstoft, and H. S. Midtiby, "Plant species classification using deep convolutional neural network," Biosystems engineering, vol. 151, pp. 72-80, 2016.

[Elhariri et al., 2014] E. Elhariri, N. El-Bendary, and A. E. Hassanien, "Plant classification system based on leaf features," in 2014 9th International Conference on Computer Engineering & Systems (ICCES), 22-23 Dec. 2014 of Conference, pp. 271-276, doi: 10.1109/ICCES.2014.7030971.

[Goy et al., 2023] S. Y. Goy, Y. F. Chong, T. K. K. Teoh, C. C. Lim, and V. Vijean, "Recognition of plant diseases by leaf image classification using deep learning approach," in AIP Conference Proceedings, of Conference, vol. 2562, no. 1: AIP Publishing.

[Goyal et al., 2019] N. Goyal, K. Gupta, and N. Kumar, "Multiclass Twin Support Vector Machine for plant species identification," Multimedia Tools and Applications, vol. 78, 2019, doi: 10.1007/s11042-019-7588-2.

[Gu et al., 2021] J. Gu, P. Yu, X. Lu, and W. Ding, "Leaf species recognition based on VGG16 networks and transfer learning," in 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2021 of Conference, vol. 5: IEEE, pp. 2189-2193.

[Ishak et al., 2009] A. J. Ishak, A. Hussain, and M. M. Mustafa, "Weed image classification using Gabor wavelet and gradient field distribution," Computers and Electronics in Agriculture, vol. 66, no. 1, pp. 53-61, 2009, doi: https://doi.org/10.1016/j.compag.2008.12.003.

[J. O. Söderkvist, 2001] O. J. O. Söderkvist. Swedish Leaf Dataset, 2001. [Online]. Available: https://www.cvl.isy.liu.se/en/research/datasets/swedish-leaf/

[Joseph et al., 2020] I. A. Joseph, M. A. Khan, and H. Luo, "Orange Leaf Diseases Identification Using Digital Image Processing," Singapore, of Conference: Springer Singapore, in Artificial Intelligence Algorithms and Applications, pp. 360-378.

[Kaya et al., 2019] A. Kaya, A. S. Keceli, C. Catal, H. Y. Yalic, H. Temucin, and B. Tekinerdogan, "Analysis of transfer learning for deep neural network based plant classification models," Computers and Electronics in Agriculture, vol. 158, pp. 20-29, 2019, doi: https://doi.org/10.1016/j.compag.2019.01.041.

[Kayaalp, 2024] K. Kayaalp, "Classification of Medicinal Plant Leaves for Types and Diseases with Hybrid Deep Learning Methods," Information Technology and Control, vol. 53, no. 1, pp. 19-36, 2024.

[Lee et al., 2017] S. H. Lee, C. S. Chan, S. J. Mayo, and P. Remagnino, "How deep learning extracts and learns leaf features for plant classification," Pattern Recognition, vol. 71, pp. 1-13, 2017, doi: https://doi.org/10.1016/j.patcog.2017.05.015.

[Lee et al., 2015] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deep-plant: Plant identification with convolutional neural networks," 2015 of Conference, pp. 452-456, doi: 10.1109/ICIP.2015.7350839.

[Li et al., 2020] H. A. Li et al., "Medical Image Coloring Based on Gabor Filtering for Internet of Medical Things," IEEE Access, vol. 8, pp. 104016-104025, 2020, doi: 10.1109/ACCESS.2020.2999454.

[Lv and Zhang, 2023] Z. Lv and Z. Zhang, "Research on plant leaf recognition method based on multi-feature fusion in different partition blocks," Digital Signal Processing, vol. 134, p. 103907, 2023, doi: https://doi.org/10.1016/j.dsp.2023.103907.

[Mamani Diaz et al., 2019] C. A. Mamani Diaz, E. E. Medina Castaneda, and C. A. Mugruza Vassallo, "Deep Learning for Plant Classification in Precision Agriculture," in 2019 International Conference on Computer, Control, Informatics and its Applications (IC3INA), 2019-10 of Conference, pp. 9-13, doi: 10.1109/IC3INA48034.2019.8949612.

[Naresh and Nagendraswamy, 2016] Y. G. Naresh and H. S. Nagendraswamy, "Classification of medicinal plants: An approach using modified LBP with symbolic representation," Neurocomputing, vol. 173, pp. 1789-1797, 2016, doi: https://doi.org/10.1016/j.neucom.2015.08.090.

[Pudaruth, 2015] S. Pudaruth. Folio, UCI Machine Learning Repository, 2015, doi: 10.24432/C5TW3F.

[Qiu et al., 2023] C. Qiu, F. Ding, X. He, and M. Wang, "Apply Physical System Model and Computer Algorithm to Identify Osmanthus Fragrans Seed Vigor Based on Hyperspectral Imaging and Convolutional Neural Network," Information Technology and Control, vol. 52, no. 4, pp. 887-897, 2023.

[Roopashree and Anitha, 2021] S. Roopashree and J. Anitha, "DeepHerb: A Vision Based System for Medicinal Plants using Xception features," IEEE Access, vol. PP, pp. 1-1, 2021.

[Saleem et al., 2019] G. Saleem, M. Akhtar, N. Ahmed, and W. S. Qureshi, "Automated analysis of visual leaf shape features for plant classification," Comput. Electron. Agric., vol. 157, pp. 270-280, 2019.

[Saud et al., 2020] S. Saud, B. Jamil, Y. Upadhyay, and K. Irshad, "Performance improvement of empirical models for estimation of global solar radiation in India: A k-fold cross-validation approach," Sustainable Energy Technologies and Assessments, vol. 40, p. 100768, 2020.

[Shiddiq et al., 2024] M. Shiddiq, F. Candra, B. Anand, and M. F. Rabin, "Neural network with k-fold cross validation for oil palm fruit ripeness prediction," TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 22, no. 1, pp. 164-174, 2024.

[Silva, 2014] P. a. M. Silva, Andr. Leaf

[Su et al., 2020] J. Su, M. Wang, Z. Wu, and Q. Chen, "Fast Plant Leaf Recognition Using Improved Multiscale Triangle Representation and KNN for Optimization," IEEE Access, vol. 8, pp. 208753-208766, 2020, doi: 10.1109/ACCESS.2020.3037649.

[Sun et al., 2017] Y. Sun, Y. Liu, G. Wang, and H. Zhang, "Deep Learning for Plant Identification in Natural Environment," Computational Intelligence and Neuroscience, vol. 2017, 2017, doi: 10.1155/2017/7361042.

[Van Hieu et al., 2023] N. Van Hieu, N. L. H. Hien, L. Van Huy, N. H. Tuong, and P. T. K. Thoa, "PlantKViT: A Combination Model of Vision Transformer and KNN for Forest Plants Classification.," JUCS: Journal of Universal Computer Science, vol. 29, no. 9, 2023.

[Van et al., 2025] H. T. Van, P. T. Khuat, T. Van, T. T. Tuan, and Y. S. Chung, "A deep learning method for differentiating safflower germplasm using optimal leaf structure features," Ecological Informatics, vol. 85, p. 102998, 2025.

[Venkatesh et al., 2021] Venkatesh, N. Y, S. U. Hegde, and S. S, "Fine-tuned MobileNet Classifier for Classification of Strawberry and Cherry Fruit Types," in 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021-01 of Conference, pp. 1-8, doi: 10.1109/ICCCI50826.2021.9402444.

[Wick and Puppe, 2017] C. Wick and F. Puppe, "Leaf identification using a deep convolutional neural network," arXiv preprint arXiv:1712.00967, 2017.

[Wu et al., 2007] S. G. Wu, F. S. Bao, E. Y. Xu, Y. X. Wang, Y. F. Chang, and Q. L. Xiang, "A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network," in 2007 IEEE International Symposium on Signal Processing and Information Technology, 15-18 Dec. 2007 of Conference, pp. 11-16, doi: 10.1109/ISSPIT.2007.4458016.

[Yang, 2021] C. Yang, "Plant leaf recognition by integrating shape and texture features," Pattern Recognition, vol. 112, p. 107809, 2021, doi: https://doi.org/10.1016/j.patcog.2020.107809.

[Zhang and Tao, 2015] H. Zhang and X. Tao, "Leaf Image Recognition Based on Wavelet and Fractal Dimension ⋆," of Conference.

[Zhang et al., 2020] S. Zhang, W. Huang, Y.-a. Huang, and C. Zhang, "Plant species recognition methods using leaf image: Overview," Neurocomputing, vol. 408, pp. 246-272, 2020, doi: https://doi.org/10.1016/j.neucom.2019.09.113.