Revista Brasileira de Computação Aplicada, November, 2025

DOI: 10.5335/rbca.v17i3.16412
Vol. 17, Nº 3, pp. 23–40
Homepage: seer.upf.br/index.php/rbca/index

ORIGINAL PAPER

Comparative analysis of the Support Vector Machine and Convolutional Neural Network algorithms applied in classifying *Acer Palmatum* plant subspecies

Patrick de Souza Sagioratto[1], Rúbia Eliza de Oliveira Schultz Ascari [1], Dalcimar Casanova [1]

[1]Federal University of Technology (UTFPR) - Pato Branco, Paraná, Brazil

patricksagioratto@alunos.utfpr.edu.br;*rubia@utfpr.edu.br; dalcimar@utfpr.edu.br

Received: 2024-10-21. **Revised:** 2025-07-25. **Accepted:** 2025-10-28.

Abstract

Background: Plant identification is an essential task as it provides valuable information about plant characteristics and helps determine the population and distribution of species. This paper presents an artificial intelligence solution developed to automatically classify subspecies of the plant species *Acer palmatum* based on images. A database containing subspecies of *Acer palmatum* was created, and supervised learning algorithms such as Support Vector Machine (SVM) and Convolutional Neural Network (CNN) were used to classify them. The experiments included different scenarios, such as original images, feature extraction, and data augmentation techniques. **Results:** The results showed that the CNN with transfer learning and data augmentation performed best, standing out as the best model tested regardless of the dataset evaluated. **Conclusions:** These findings suggest that advanced Machine Learning techniques can be highly effective in classifying subspecies of *Acer palmatum*, providing a valuable biodiversity monitoring and mapping tool.

Keywords: Artificial Intelligence; Computer Vision; Convolutional Neural Network; Support Vector Machine.

Resumo

Contexto: A identificação de plantas é uma tarefa essencial, pois fornece informações valiosas sobre suas características e ajuda a determinar a população e a distribuição das espécies. Este artigo apresenta uma solução de inteligência artificial desenvolvida para classificar automaticamente subespécies da espécie vegetal Acer palmatum com base em imagens. Um banco de dados contendo subespécies de Acer palmatum foi criado e algoritmos de aprendizado supervisionado, como Support Vector Machine (SVM) e Convolutional Neural Network (CNN), foram utilizados para classificá-las. Os experimentos incluíram diferentes cenários, como imagens originais, extração de características e técnicas de aumento de dados. **Resultados:** Os resultados mostraram que a CNN com aprendizado de transferência e aumento de dados apresentou o melhor desempenho, destacando-se como o melhor modelo testado, independentemente do conjunto de dados avaliado. **Conclusões:** Essas descobertas sugerem que técnicas avançadas de Aprendizado de Máquina podem ser altamente eficazes na classificação de subespécies de Acer palmatum, fornecendo uma valiosa ferramenta de monitoramento e mapeamento da biodiversidade.

Palavras-Chave: Inteligência Artificial; Visão Computacional; Rede Neural Convolucional; Máquinas de Vetores de Suporte.

1 Introduction

Plant identification is essential since it provides valuable information about their characteristics, such as their

natural habitat, cultivation requirements, medicinal properties, and potential toxicity (Mulugeta et al., 2024). This information serves as a basis for various areas, such as medicine, which uses information on medicinal and toxic properties for application in traditional medicine or studies of new drugs (Bateman et al., 1998). Areas such as agriculture and the food industry take advantage of information about cultivation and habitat to create beneficial conditions for food (Jayanthi et al., 1999). Identifying plant species also helps to determine the population and distribution of plants in the environment where they are located. This is necessary to recognize species at risk of extinction and plants that have medicinal potential (Sharifalillah et al., 2008).

Each plant species has specific characteristics, which may be similar to the characteristics of plants of other species and which may also vary between the different subspecies of the same species due to their different evolutionary lines (Weber and Agrawal, 2014). These characteristics may be linked to the specificities of the leaves, petioles, trunks, flowers, fruits, and inflorescences, which vary in shape, texture, margin, size, and color, with some of these traits being evident only in certain seasons of the year (i.e., the color of the flower) or may vary between seasons (i.e., leaf color) (Rippy et al., 2021). Thus, identifying and classifying plant species is not always trivial, especially for non-specialized people.

According to Heredia (2017), work done in the biodiversity community relies heavily on hand-labeled image data from a community of experts. More recent studies indicate that, in large biodiversity centers, modern technologies are being implemented in the analysis of new data. However, human observations still constitute a portion of this record, carried out during field expeditions or in citizen scientific initiatives (Gadelha Jr et al., 2021). The dependence on these manual observations can make it difficult for new users to enter, who, having access to appropriate technology, could help monitor biodiversity (Heredia, 2017).

The rapid development of image processing and pattern recognition technology has enabled automatic computer recognition of plant species based on image processing (Wang et al., 2017). Different Machine Learning and Computer Vision techniques have been explored to achieve this goal.

Plants in the genus *Acer*, commonly known as maple, contain approximately 200 species with leaves that vary in size, shape, and color. Among the Acer species, *Acer palmatum* stands out, which is subdivided into several subspecies, some of which have characteristics that differ greatly from each other, such as size, appearance, and color of leaves and petioles (Trees and Online, 2024). On the other hand, some subspecies have very similar traits, making accurate classification difficult.

In this context, the proposal of this study corresponds to the development of an Artificial Intelligence solution for the automatic classification of subspecies of plants of the *Acer palmatum* species based on images. To this end, we evaluated different Machine Learning algorithms in experiments with varied data sets.

2 *Acer palmatum*

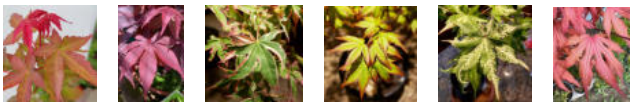
Plants of the genus *Acer* are widely distributed in the northern hemisphere and cultivated as ornamental plants worldwide (Ji et al., 1992), in addition to being traditionally used to treat a wide range of illnesses in East Asia and North America (Bi et al., 2016).

The *Acer palmatum* species is original and popular on the Asian continent (China, South Korea, and Japan) but is currently widespread in several parts of the world and is subdivided into several subspecies. The botanical classification of *Acer palmatum*, also known as Japanese Maple, is presented in Table 1.

Table 1: Botanical Classification of *Acer palmatum* (USDA, NRCS, 2024)

Common name	Japanese maple	**Gender**	*Acer*
Kingdom	*Plantae*	**Subkingdom**	*Tracheobionta*
Class	*Magnoliopsida*	**Order**	*Sapindales*
Division	*Magnoliophyta*	**Species**	*Acer palmatum*
Family	*Sapindaceae*	**Scientific name**	*Acer palmatum*

The leaves of *Acer palmatum* are palmate, measuring five to 10 cm in length and width, with five or seven lobes and a rounded contour (Trees and Online, 2024). This leaf morphology is one of the reasons why the tree is highly valued ornamentally. The lobes are oblong to lanceolate, finely tapered, and jagged to dissected edges, with a fine and smooth texture (Pollock and Griffiths, 2005). Figure 1 presents some examples of leaf images from different subspecies of this plant.



(a) (b) (c) (d) (e) (f)

Figure 1: Examples of *Acer palmatum* leaves corresponding to the subspecies (a) *deshojo*, (b) *atropurpureum*, (c) *beni schichihenge*, (d) *komachi hime*, (e) *orido nishiki*, (f) *amber ghost* (Ascari, 2023)

The colors of the leaves of the different subspecies of *Acer palmatum* throughout the seasons stand out as a distinctive characteristic of this species, which can display leaves in shades of green, dark red-purple, and brownish-red in early spring. In autumn, they transform into purple, crimson, and crimson-red colors. During the summer, some subspecies have greenish-gold or green leaves shaded in reddish-orange, while others display a medium green. This color variation reflects the adaptability and genetic diversity of this tree species (Trees and Online, 2024).

Acer palmatum produces small flowers and paired samaras (˜ 1.9 cm) (Figure 2) (Garden, 2019). It grows best

in well-drained, slightly acidic to neutral, organic-rich soils (Toolbox, 2020).

Figure 2: Flowers from a *Acer palmatum* beni kawa (left), and samaras from a *Acer palmatum* yamamomiji (right) (Ascari, 2023)

Due to its high adaptability in different situations, according to Dirr et al. (1990), probably one of the most versatile maple species when it comes to landscape uses, it is necessary to understand the particularity of the planting site and climate to benefit the health and vitality of the *Acer palmatum* (Phillips, 2003). Therefore, the correct identification of the subspecies can be helpful in the search for information on the proper management to be given to the plant, especially by less specialized users.

It is not known to the authors of this paper, and based on the systematic mapping described by (Ascari et al., 2023), the existence of automated solutions for the classification of different subspecies of *Acer palmatum*. Nor were any datasets composed of samples referring to the various subspecies of this species identified, as described in Ascari et al. (2023).

Thus, this research presents the creation of a new dataset composed of specific samples of this plant and the implementation of a system that uses Computer Vision and Machine Learning to classify different subspecies of plants of the species *Acer palmatum*.

3 Computer vision

The idea of an intelligent automaton comes from the beginnings of technology, which several enthusiasts in artificial intelligence have sought to grant computer capabilities comparable to biological organisms. One of the first goals of Artificial Intelligence is to grant computers the ability to deal with sensory input, in other words, the ability to "see" (Dana H. Ballard, 1982).

In object recognition, the human brain processes visual information to extract semantically meaningful features like line segments, boundaries, and shapes. Computers need to process visual information in a data space of strongly detectable but less significant characteristics, such as colors and textures (Zhang, 2010).

Dana H. Ballard (1982) defines Computer Vision as the initiative to automate and integrate a wide range of processes and representations used for vision perception. This includes techniques such as image processing, statistical pattern classification, geometric modeling, and cognitive processes. Forsyth and Ponce (2012) define Computer Vision as an initiative that uses statistical methods to separate data using models built with the help of geometry, physics, and learning theory.

Existing studies that use Computer Vision on *Acer palmatum* images generally perform classification between different species and not between different subspecies of the same species. Furthermore, based on a systematic mapping conducted by the authors of this paper (Ascari et al., 2023), in the datasets in which *Acer palmatum* is included, the images generally refer to the most common subspecies, "*Acer palmatum thumb.*", represented by plant leaves in a shade of green, as shown in Figure 3, referring to a sample available in the Flavia Dataset dataset.

Figure 3: Leaf sample of *Acer palmatum* (Wu et al., 2007)

However, like many deciduous plants, the *Acer palmatum* changes the color of its leaves and is not just related to autumn. Therefore, the color of the leaves can vary greatly, taking on shades of red and orange, among others. These variations in leaf color make classifying species based on images even more challenging, and, in this context, color is a characteristic that cannot be considered by the algorithms in isolation. Thus, other important characteristics need to be considered, such as the shape and texture of the leaves.

4 Machine Learning

Some of the most significant transformations in recent decades occurred due to computers and digital technology. Initially, programmers defined through a programming language what the computer should do; currently, some tasks are not performed by writing programs but by collecting data (Alpaydin, 2016).

According to Faceli et al. (2011), Machine Learning is the process of inducing a hypothesis (or function approximation) from experience. For Alpaydin (2014), Machine Learning is programming computers to optimize a performance criterion using example data or experience.

Machine Learning Algorithms can be organized into different criteria, which can be divided into predictive and descriptive. The algorithms are classified as predictive models in tasks that aim to find a function or hypothesis from training data so that a label or value that characterizes a new example can be predicted based on its input attributes. These must have input and output attributes. Such models follow the supervised learning paradigm, where an external supervisor knows the inputs and their appropriate outputs and can evaluate the induced hypothesis's ability to predict the output value for new examples. The algorithms classified as descriptive models are used in tasks that follow the unsupervised learning paradigm. These models do not have an output attribute, so they aim to find similar groups in the data set.

Classification algorithms are commonly used in Computer Vision solutions, and their objective is to classify something, whether items or samples, into a distinct set of classes or categories according to the characteristics observed by the supervisor. This research used supervised learning classification algorithms:

Support Vector Machine (SVM) and Convolutional Neural Network (CNN) via Transfer Learning. The classifiers used were chosen based on the good results presented in the literature for image classification. SVM-based classifiers offer strong generalization capability, simple architecture, and the ability to classify a few samples (Liu et al., 2018). CNN-based classifiers have recently shown explosive popularity, in part due to their success in image classification and other fields of Computer Vision (Wang et al., 2018). Transfer Learning can leverage the knowledge gained from a deep CNN pre-trained on a large dataset for a specific task, improving subspecies classification performance (the task of this research) with a small dataset composed of a limited number of samples (the context of this research).

4.1 Support Vector Machine (SVM)

In cases where classes are not linearly separable in the original input space, SVM non-linearly transforms the original input space into a higher-dimensional feature space (Géron, 2022). This transformation is done through a mapping using a kernel, the most common of which are Linear, Polynomial, RBF (Radial Basis Function), and Sigmoid.

After the nonlinear transformation step, finding an optimized linear hyperplane to separate the classes in the feature space is relatively trivial. For cases where classes are overlapping and not linearly separable, even in a higher dimensional feature space, the classifier needs to be changed to find the maximum margin, allowing some data not to be classified or to be classified on the wrong side of the decision margin. This maneuver is called soft margin, and all data within this margin is neglected (Kecman, 2001).

In multiclass classification, it is common to construct a set of binary classifiers, each trained to separate one class from the rest, and then combined to perform multiclass classification according to the maximum output (Schölkopf et al., 2001). In this research, multiclass classification was used, aiming to classify images of plants *Acer palmatum* between different subspecies.

4.2 Convolutional Neural Network (CNN)

Neural Networks were developed to simulate the human nervous system for Machine Learning tasks in a similar way to human neurons (Szeliski, 2022). Neural Networks can theoretically learn any mathematical function with sufficient training data (Albarghouthi, 2021). CNN is a type of neural network designed to work with inputs structured in grids, which have strong spatial dependencies in local regions of the grid. An example of a structured grid is a two-dimensional image (Géron, 2022). In an image, dependence on local regions is easily noticeable, as the colors of nearby pixels often have similar color values to an individual pixel. A three-dimensional image captures different colors. This new dimension means a new entry, which represents the volume.

CNN can be single-layer or multilayer. In single-layer CNN, the input is mapped directly to the output using a generalized variation of a linear function. In multilayer CNN, neurons are organized into layers in which hidden layers separate the input and output (Goodfellow et al., 2016). The states of each layer are organized according to the spatial grid structure. These spatial relationships are inherited from one layer to the next because the value of each feature is based on a small local spatial region of the previous layer. Each layer of a CNN is a 3-dimensional grid structure, which has height, width, and depth (Goodfellow et al., 2016).

4.3 Transfer Learning

Transfer learning is a technique adopted in Machine Learning, specifically in Neural Networks. Its main objective is to transfer knowledge from a model trained on a specific task to a new model, usually on a related task (Torrey and Shavlik, 2010). The motivation behind transfer learning lies in the idea that knowledge acquired in one task can be helpful in other tasks, especially when data sets are small or when there are few computational resources available to train models from scratch (Pan and Yang, 2010).

One of the main advantages of transfer learning is the ability to leverage weights learned in previous tasks, which can result in faster training and better model performance. This is particularly relevant in scenarios where training datasets are limited, which is common in many real-world applications. There are several approaches to implementing transfer learning. The three most common are feature extraction, fine-tuning, and domain adaptation.

Feature extraction is the most common way to apply transfer learning. A pre-trained model is tuned to the new task, typically training just a few final layers of the model while keeping the initial layers frozen (Stevens et al., 2020). This allows the model to adapt to specific patterns in the new dataset while retaining the general knowledge learned in previous tasks.

Fine-tuning is usually applied after common transfer learning. The pre-trained model is fine-tuned for the new task by unfreezing all model layers. Then, an extremely low learning rate is applied. This allows the model to carefully adjust to the patterns of the new dataset while maintaining the ability to converge more smoothly and effectively (Zhuang et al., 2021). By allowing all model layers to be updated, fine-tuning is particularly useful when patterns in the new task differ substantially from those learned in previous (Chollet, 2021) tasks. This means the model can adapt more flexibly and accurately to new nuances and complexities in the data.

5 Related Works

Computer vision and machine learning techniques have been extensively applied to plant species classification. However, we didn't find any studies with the same objective as this research. Even so, approaches successfully employed for different plant identification using digital image analysis can be used as a basis for our performance analysis.

Azadnia et al. (2022) developed a vision-based system using a custom CNN model for real-time identification of medicinal plants. The model, tested with different image resolutions, achieved over 99.3% accuracy across all configurations. Jiménez et al. (2023) developed a method to determine the sex (female/hermaphrodite) of papaya using computer vision and machine learning, considering that Hermaphrodite papayas are preferred commercially. The authors created a dataset of 520 images of pre-anthesis flowers (FaP), collected nine weeks after sowing. The best-performing CNN model, using image contour features, achieved 80% accuracy.

Trivedi et al. (2023) proposed a hybrid deep learning model combining You Only Look Once (YOLO) and CNN for Thai cannabis plant classification, achieving 95.37% accuracy. Applications of the study include monitoring growth, ensuring product quality, and assisting law enforcement in detecting illegal cultivation. Erkan et al. (2023) introduced a hyperparameter optimization method using the Artificial Bee Colony (ABC) algorithm to enhance CNN architectures for leaf image classification. Their Optimal Deep CNN (ODC) classifier achieved an accuracy of 98.99%. Blazakis et al. (2024) used machine learning to discriminate 14 olive cultivars based on spectral data from fruits, leaves, and endocarps across two growing seasons. Their meta-classifier approach, combining basic classifiers, reached 95% accuracy.

Furthermore, Barhate et al. (2024) conducted a systematic review of over 150 studies, presenting an overview of effective machine learning, image processing, and deep learning methods for plant species recognition. The analysis shows that machine learning and deep learning algorithms outperform traditional approaches in species identification. Key challenges include performance degradation, misclassification, lack of unified evaluation criteria, limited quantitative comparisons, high computational demands, and inaccurate predictions.

6 Methodology

Figure 4 presents the development steps used in this research, which include the organization of different data sets, the implementation of the algorithms, and the analysis of the results obtained. Each step is described in detail below.

6.1 Organization of datasets

The dataset *Japanese Maple Dataset* (Ascari, 2023) was used, which comprises images of 59 subspecies of the plant *Acer palmatum*. This dataset was built collaboratively by people who grow plants of the subspecies *Acer Palmatum*, as described in Ascari et al. (2023), and covers various images, including trunks, entire plants, leaves, or combinations of these elements. These images can present both a white background and a more complex background, referred to as non-white backgrounds. These backgrounds can include elements such as sunlight, buildings, objects, or other plants, as demonstrated in Figure 5, where samples of the subspecies *atropurpureum* exemplify each of these types of images in a representative way.

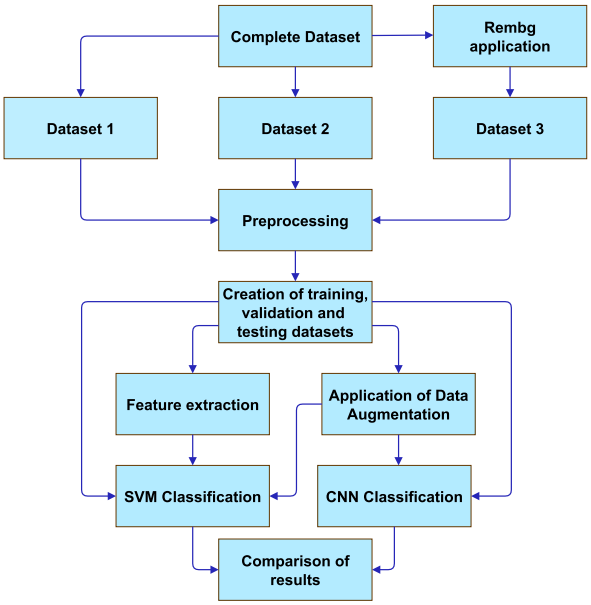


Figure 4: Steps in the image-based plant classification process

The organization of new datasets from the Japanese Maple Dataset aims to create different environments for evaluating the Machine Learning algorithms used as classifiers. One proposed approach to data analysis is to find relationships between botanical data objects, such as leaf characteristics and color patterns, among others. Then, the remaining analysis will be performed using these relationships instead of the data objects. This approach aims to reveal patterns and hidden information in the data, allowing different insights into the results of analysis algorithms (Tan et al., 2016). This approach was used to extract color, texture, and shape features, to feed the SVM classifier since the CNN performs feature extraction automatically.

Generating different sets from the original Dataset makes it possible to observe the behavior of extracting these characteristics in each scenario. The Table 2 describes the logic of organizing each set. It is essential to highlight that the process adopted in each set is identical, but the databases are treated independently, without any interconnection between them.

6.2 Data preprocessing

Machine Learning algorithms tend to perform worse when input attributes have very different scales (Géron, 2022). Because of this reason, data preprocessing is a crucial step in preparing data before feeding it into Machine Learning models. One widely used technique to address this problem is scale normalization, known as min-max normalization (Goodfellow et al., 2016).

Min-max normalization resizes attribute values so that they vary between 0 and 1, taking into account the minimum and maximum values of the attribute to be

(a) (b) (c) (d) (e)

Figure 5: Examples of images of *Acer palmatum* corresponding to the subspecies *atropurpureum* composed of (a) trunk, (b) entire plant, (c) leaves with a complex background, (d) leaves with a white background, (e) mixed (Ascari, 2023)

Table 2: Datasets created from the original

Dataset	Description
1 - All images (10+)	All subspecies of *Acer palmatum* with 10 or more images in the Japanese Maple Dataset, considering images with complex, and white backgrounds.
2 - White background	Only white background images of all subspecies of *Acer palmatum* available on the Japanese Maple Dataset.
3 - Background removed	All *Acer palmatum* images from the Japanese Maple Dataset that went through the *rembg* procedure and did not lose their essential identification characteristics (shape, texture, margin, among others).

normalized (Eq. (1)), where A' is the normalized value, A is the value of the attribute to be normalized, *Amin* is the minimum value of the attribute to be normalized and *Amax* is the maximum value of the attribute to be normalized (Goldschmidt and Passos, 2005).

$$A' = \frac{A - Amin}{Amax - Amin} \tag{1}$$

A practical example of where data normalization is commonly applied is in digital image processing. In this context, pixels in images generally range from 0 to 255 due to the most frequently used encoding format, such as 8 bits per pixel (bpp) or 24 bpp format (Pratt, 2007). In the 8 bpp format, each pixel is represented by a single byte of data, with 256 combinations, each representing a light intensity. The value 0 represents the absence of light (black color), while the value 255 represents the maximum light intensity (white color). In the 24 bpp format, each pixel is represented by 3 bytes of data, divided into 3 primary color channels: red, green, and blue (RGB) (Gonzalez and Woods, 2008).

To normalize this image data, the maximum and minimum value of the pixels is applied, as shown in Eq. (2), where A' represents the normalized pixel and A is the value of the original pixel.

$$A' = \frac{A}{255} \tag{2}$$

In the context of Machine Learning applied to image processing, it is essential to resize the data so that each row represents an image and each column represents a specific characteristic. Each pixel is treated as a characteristic of the image in question (Nixon and Aguado, 2012).

The resizing of images is carried out using the vector V, as shown in Eq. (3). In this equation, n represents the position of the image in the vector, while H and W denote the height and width dimensions of the image, respectively. Furthermore, C represents the number of color channels present in the image, where $C = 1$ for black and white images and $C = 3$ for color images.

$$V[n] = (H * W * C) \tag{3}$$

With data normalization and appropriate resizing, data can be used in Machine Learning algorithms, contributing to better performance and more accurate results. All images used in this research were normalized and resized, as explained in this section.

6.3 Feature extraction

Each tool used to extract features is applied independently, using the initial database specific to each approach. Various methods were employed, including a color histogram for color extraction, the local binary pattern for texturing, and Zernike moments for shape analysis.

6.3.1 *Color extraction*

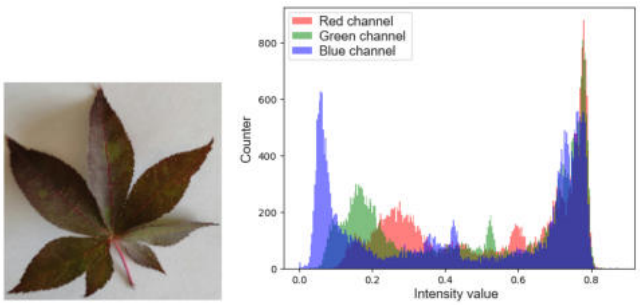
To obtain the colors, the histogram tool from Scikit-learn was used, in which only the *nbins* parameter was modified. This parameter represents the number of bins used in calculating the histogram. We chose to use the maximum *bins* available (256 *bins*), a choice justified by the variety of shades present in the different species of *Acer palmatum*, some of which exhibit subtle color differences.

This approach resulted in a vector containing the histogram values for each image. Figure 6 shows the image that served as the basis for the histogram and the corresponding histogram itself.

6.3.2 *Texture Extraction*

Texture extraction was performed using the local binary pattern tool from Scikit-learn. This technique requires a grayscale image as input, along with the following parameter settings:

- P: The number of neighboring set points arranged circularly. The value of P was selected after iterations, resulting in $P = 27$.



(a) (b)

Figure 6: Example image of *Acer palmatum atropurpureum* (a) original; (b) its respective histogram

- *R*: The radius of the neighborhood circle. The value of *R* was adjusted by experiment, adopting *R* = 3.
- *method*: The method used was "*uniform*", chosen for its invariance to rotation.

After texture extraction, the histogram approach was also employed, which has been shown to improve results. Figure 7 presents the results generated by texture extraction in its three stages.

6.3.3 Format extraction

Format extraction was conducted using the Zernike moments tool from Mahotas, which requires a grayscale image as input, along with the following parameters:

- *radius*: Maximum radius for Zernike polynomials, in pixels. The region outside the circle (centered on the center of mass) defined by this radius is disregarded. We decided to adopt *radius* = 120 after experimentation.
- *degree*: Maximum degree used for the analysis. The value of the degree was set to 13, also after experimentation.

This approach results in a vector containing numerical values representing each processed image's absolute Zernike moments per degree. The techniques described in the previous sub-sections, including the extraction of color, texture, and shape features, were used to generate the input data for the SVM classifier.

6.4 Data Augmentation

Data Augmentation, which was proposed initially by Tanner and Wong (1987), consists of creating additional samples from existing data, being useful in a scenario of insufficient training data, to improve the performance of the classifier employee. Therefore, if necessary and feasible, the training data set can be expanded by including new samples generated from variations of the original samples.

Several operations can be applied during the Data Augmentation process, enriching the training data set. Geometric transformations, such as rotations, reflections, and shear, can introduce different perspectives of images. Additionally, intensity adjustments, such as brightness and contrast changes, can simulate variations in lighting conditions. Other techniques, such as random cropping, adding noise, and even elastic deformations, can increase the model's diversity and generalization ability. This variety of operations, applied in a controlled and realistic manner, aims to improve the robustness and learning capacity of the classifier in the face of different scenarios and conditions (Zhang et al., 2021).

For this study, horizontal and vertical rotation operations were carried out on each image, random cropping procedures, and brightness and contrast adjustments to the original images and their modified versions, thus generating 15 variations from each original image. The results of these processes can be seen in Figure 8, which displays both the original image and its variations resulting from these manipulations.

6.5 Construction of the models

To perform image classification, it is necessary to separate the data set into training, validation, and test data. Several sampling methods can be used to perform this step, for example, Bootstrap (Efron, 1992), Holdout (Kohavi et al., 1995), K-fold cross-validation (Geisser, 1975), and Leave-one-out (Nguyen and Poo, 2017). Each strategy separates the data for training, validation, and testing differently. Still, a classification model is induced from the training and validation data set, and its performance is evaluated using a test data set.

The stratified K-fold cross-validation method with K=3 was used for this study, especially useful when the data set presents an unequal distribution between classes. In this process, the dataset is partitioned into *K* subsets (folds), as illustrated in Figure 9. Of the *K* subsets, *K-1* are allocated for training, while the remaining fold is reserved for validation. This operation is repeated *K* times, with each fold serving as a validation set in a different iteration. Thus, the results are combined to provide a more reliable assessment of the model's performance (Hastie et al., 2009).

In scenarios where the distribution of classes is uneven, stratified K-fold cross-validation is a common approach. By maintaining the proportion of classes in each fold, this strategy not only aims to mitigate the effects of imbalances but also guarantees a more accurate and fair evaluation of the model. The representation of all classes during the assessment is carefully maintained, which helps to minimize the risk of biased results towards majority or minority classes (He and Garcia, 2009).

The division of the datasets was organized as illustrated in Table 3, displaying information regarding the total number of subspecies in each set and the total number of images present.

The decision to adopt the ratio of 80% for training-validation and 20% for testing was based on considerations regarding the optimal amount of data to train the model effectively while ensuring a robust assessment of model performance. More unequal ratios, such as 90/10, could result in insufficient data to test, leading to a less reliable evaluation of the model on unseen data. On the other hand, ratios like 70/30 could reduce the amount of data available for training, limiting the

(a) (b) (c)

Figure 7: Example image of *Acer palmatum atropurpureum* (a) original, (b) grayscale, (c) extracted texture

(a) (b) (c) (d) (e) (f) (g)

Figure 8: Data augmentation applied to (a) original image, (b) horizontally rotated, (c) vertically rotated, (d) horizontally and vertically rotated, (e) random brightness, (f) contrast random, (g) random clipping

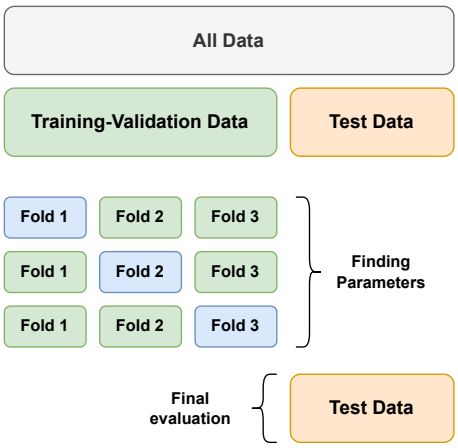


Figure 9: K-fold cross-validation with K=3, representing the division of the training-validation set into three parts, two used for training and one for validation (Adapted from Scikit-learn (2023))

model's ability to learn complex patterns in the training data. Therefore, the 80/20 ratio was considered a suitable compromise between these observations, allowing for good model generalizability and a reliable assessment of its performance.

After carefully deliberating the most suitable Machine Learning algorithms to classify *Acer palmatum* images, it was decided to incorporate the Support Vector Machine and Convolutional Neural Network. The choice of these algorithms was based on a review of specialized literature, which highlighted their effectiveness and ability to deal with the complexity of the characteristics of *Acer palmatum* leaves. SVM is recognized for its generalization capacity and robustness in classification problems (Géron, 2022). CNN excels in extracting relevant features from images, especially in domains with complex visual patterns (Chollet, 2021), such as foliage.

The search for the best hyperparameters for each scenario was conducted systematically, using grid search with the library GridSearchCV (Buitinck et al., 2013) for

Table 3: Allocation of images from each set into training-validation and testing, with a proportion of 80% for training-validation and 20% for testing

Dataset	Subspecies	Dataset division	Samples
Dataset 1	13	Training-validation	350
		Test	88
		Total	438
Dataset 2	7	Training-validation	274
		Test	69
		Total	343
Dataset 3	7	Training-validation	212
		Test	54
		Total	266

SVM and KerasTuner (O'Malley et al., 2019) for CNN. This automated approach provided an efficient way to explore the hyperparameter space, allowing for a more accurate selection of the best parameters for each model. This methodology stands out for its ability to simplify the search for the ideal configurations of each classifier, thus ensuring a more comprehensive and reliable optimization.

Hyperparameter optimization, a crucial step in developing Machine Learning models, aims to find the optimal combination of hyperparameters to maximize model performance. This search seeks to balance generalization and precision, avoiding excessive fitting (overfitting) to the training data and ensuring greater generalization capacity about new unobserved data (Lakshmanan et al., 2021).

When experimenting, feature extraction was performed only in the SVM classifier, as the CNN classifier has its way of integrating features into its training. This decision was motivated by the intrinsic nature of the CNN, which can automatically learn relevant features directly from the input data during the training process. In this way, by focusing only on feature extraction for the SVM, one can explore its classification capacity using previously extracted features while allowing the CNN to take advantage of its architecture to learn and extract features adaptively during training, thus maximizing both models' efficiency and generalization capacity.

The Soft Voting technique from the Scikit-learn library was applied to integrate all the features. This approach combines the predictions of multiple individual classifiers in a weighted way, resulting in a more robust and generally more accurate final decision. In Soft Voting, each classifier assigns a probability score to each output class, and the class with the highest weighted average score is chosen as the final prediction (Géron, 2022). The classifiers adopted were the SVMs trained with a specific extracted feature.

This technique is beneficial when you have a variety of models with different strengths and weaknesses. By combining its predictions, Soft Voting takes advantage of the diversity of classifiers, resulting in an aggregated model that is more resilient to individual errors and capable of capturing subtle nuances in the data.

For the CNN, a transfer learning approach was adopted, employing the pre-trained model *MobileNetV3Large* (Howard et al., 2019), initialized with the weights from the *ImageNet* dataset (Deng et al., 2009). This strategic choice allows you to leverage the model's prior knowledge of various image classes, providing a solid foundation for further training. Transfer learning simplifies and accelerates the training process, resulting in better performance than training from scratch.

Due to most of the convolutional parameters being within the pre-trained model *MobileNetV3Large*, the parameters considered for CNN optimization were:

- Dense layer units: It refers to the number of neurons in this layer. The more units used, the more parameters the model will have and, therefore, the more complexity it can learn.
- *Dropout*: It is a regularization technique used to mitigate overfitting in neural network models. During training, a fraction of the neurons in the dropout layer are randomly turned off (set to zero). This helps prevent neurons from over-relying on other specific neurons, making the network more robust and less susceptible to overfitting training data.
- Base learning rate: It is the factor that multiplies the gradient calculated during training to update the neural network weights. It is the initial value of the learning rate. Too high a learning rate can cause the model to oscillate around the global minimum, while too low a rate can result in slow training or stopping before reaching an optimal solution.
- *fine-tuning* layers: This term refers to the network layers that will be adjusted during this process. Generally, during initial training, convolutional layers are frozen. Subsequently, a part of these layers is unfrozen for fine-tuning the new dataset, adapting to the specific characteristics of the new task.

The parameters used for SVM optimization were:

- *Kernel*: This parameter specifies the type of kernel to be used in the SVM algorithm. Available kernel types include linear, polynomial, sigmoid, and RBF.
- C: This parameter controls the penalty for wrong classifications. A smaller value of C indicates a more minor penalty, which can result in a softer decision boundary, allowing for more misclassifications in the training set. On the other hand, a more significant value of C increases the penalty, leading to a tighter decision boundary and possibly overfitting the training data.
- Gamma: This parameter influences how training points affect the margin decision. A low gamma value means that distant points have a high influence, resulting in a softer decision margin. On the other hand, a high gamma value causes the closest training points to have greater weight, resulting in a more complex decision margin adjusted to the details of the training data.
- *Coefo*: This term is only meaningful for polynomial and sigmoidal kernels. In polynomials, it controls how much the model is influenced by higher-degree interactions versus lower-degree interactions. Sigmoidal acts as a compensation term when the activation function is not symmetric around zero.

6.6 Performance Measures

Classification model evaluations were conducted using a set of fundamental measures, including precision, recall, and F1-score. They allow you to measure the model's ability to correctly identify instances of each class and quantify the balance between the rate of true positives and the rate of false negatives, in addition to combining precision and recall into a single value. The joint interpretation of these measures provides a comprehensive understanding of the model's effectiveness in terms of assertiveness and comprehensiveness.

6.6.1 *Precision*

Precision is a fundamental performance measure that evaluates the proportion of correctly classified instances relative to the total instances predicted as positive (true positives and false positives). In other words, precision measures the rate of true positives relative to instances classified as positive by the model (Murphy, 2012). The formula for calculating precision is presented in Eq. (4):

$$Precisão = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4}$$

6.6.2 *Recall*

Recall, or true positive rate, measures the proportion of positive instances correctly identified by the model in relation to the number of real positive instances, as presented in Eq. (5). Recall is significant when focusing on minimizing false negatives (Murphy, 2012).

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{5}$$

6.6.3 *F1-Score*

The F1-score is a performance measure that combines precision and recall into a single value, balancing the two measures. It is especially useful when there is an imbalance between classes or when precision and recall are important (Murphy, 2012). The F1-score is calculated by the harmonic mean between precision and recall (Eq. (6)):

$$F1\text{-}Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (6)$$

6.7 Statistical measure

The confidence interval is a statistical measure that estimates the range in which a population parameter is likely based on a sample of the data. In other words, it gives an idea of the uncertainty associated with a point estimate, such as the mean, proportion, standard deviation, among others.

In the context of Machine Learning models, the confidence interval is a tool that helps evaluate the stability and reliability of estimates made by models. For example, when calculating the accuracy of a classifier on a test data set, one can provide not only a single estimate of the accuracy but also a confidence interval that indicates the expected variation in accuracy due to randomness in the training and test data.

The Student's t confidence interval was developed based on the Student's t distribution, created by William Sealy Gosset, a British statistician who used the pseudonym 'Student'. This method has become fundamental for the statistical analysis of small samples, allowing the estimation of more precise confidence intervals when the population variance is unknown (Student, 1908).

The Student's t confidence interval is particularly relevant in Machine Learning models, where there is often interest in evaluating the model's average performance on a given measure, such as precision, recall or F1-score. By calculating the Student's t confidence interval for the measure of interest, it is possible to better understand the variability of model performance across different data sets (Ara et al., 2003).

The general formula for Student's t confidence interval (Eq. (7)) is used when the sample size is small (generally considered $n < 30$) and/or the population standard deviation is unknown. In this equation $\bar{x}$ is the sample mean, t is the critical value of the Student t distribution corresponding to the desired confidence level and degrees of freedom, s is the standard deviation of the sample and n is the sample size.

$$\bar{x} \pm t\frac{s}{\sqrt{n}} \quad (7)$$

With this process, it is possible to obtain a confidence interval around the mean of the model performance measure, indicating the range in which the true population mean is likely to be, with a specified confidence level. This range measures the uncertainty associated with estimating model performance, helping to more robustly and reliably interpret the results obtained.

This study adopted a sample size of 10 observations ($n = 10$) to evaluate the performance of the Machine Learning models. Additionally, a 95% confidence interval was used. With the aid of the Student t distribution table, for a confidence level of 95% and 9 degrees of freedom ($n - 1$), the corresponding critical value is $t = 2.262$. Therefore, the Eq. (8) presents the Student t confidence interval formula for the mean, with these specific values.

$$\bar{x} \pm 2,262\frac{s}{\sqrt{10}} \quad (8)$$

The decision to use 10 samples in this study was motivated by practical and feasibility considerations. Experiments evaluating the performance of Machine Learning models can be computationally and temporally intensive, especially when run on complex datasets or in restricted computational environments. Furthermore, the cost of computing infrastructure, including processing time, can be significant. Therefore, a more economical approach was chosen, using smaller samples to represent the population while still seeking a valid and informative assessment of the model's performance. It is worth noting that this approach is consistent with previous studies that also adopted similar sample sizes (Reis et al., 2020; Medeiros et al., 2023; Lima et al., 2023).

7 Results and Discussions

In this section, the results obtained from the execution of the different sets of experiments are presented. Performance evaluations of the tested models and discussions of the implications of these findings for improving the method used are described. The results were generated based on the datasets previously described in Table 2, each one tested with the following approaches depending on the classifier:

- SVM
  - Dataset without changing images
  - Dataset with data augmentation
  - Dataset with color extraction with color histogram
  - Dataset with texture extraction with LBP
  - Dataset with format extraction with Zernike moments
  - Dataset with all extracted features (colors, texture, and shape)
- CNN
  - Dataset with transfer learning
  - Dataset with transfer learning and data augmentation

Were tested, as well as the CNN without transfer learning (both with data augmentation and without). However, due to the unsatisfactory performance of these methods, it was decided to omit their detailed results from this analysis.

For the experiment involving the SVM with data augmentation, the technique was applied only to the model with the best parameters identified after the optimization process. Therefore, the images generated by data augmentation were only used in the final training phase, which preceded the tests. This approach was adopted due to the limitations of the experimental

environment, which did not allow enough time to apply data augmentation at all stages of the process.

7.1 Dataset 1 - All images (10+)

For Dataset 1, subspecies with at least 10 images in the *Japanese Maple Dataset* (Ascari, 2023) were selected. This selection ensured that each class had a minimum amount of data, providing a solid foundation for training and evaluating classification models. Table 4 shows the distribution of subspecies in this set and the number of images available for each one.

Table 4: Subspecies present in Dataset 1

Subspecies	Images	Images (%)
atropurpureum	78	17,8%
beni kawa	11	2,5%
bihou	35	8,0%
bloodgood	18	4,1%
yamamomiji (or "common")	47	10,7%
deshojo	92	21,0%
dissectum	37	8,4%
dissectum atropurpureum	20	4,6%
dissectum rubrum	37	8,4%
hogyoku	13	3,0%
katsura	19	4,3%
orange dream	18	4,1%
sango kaku	13	3,0%

The Table 5 presents the results obtained for the classification model applied to Set 1. The precision, recall, and f1-score values are provided for each combination of model and process, along with their respective confidence intervals. The model with the best f1-score was the CNN with transfer learning and data augmentation, reaching a value of 75.55%; that is, the balance between precision and recall of predictions positives is 75.55% on average. The margin of error of 2.13%, with a 95% confidence level, suggests that this equilibrium varies around this average. In contrast, the worst f1-score was 37.57% with a margin of error of 3.35% and a confidence level of 95%, coming from the SVM model with shape extraction.

Figure 10 displays a visual representation of the confidence intervals for the f1-score. The y-axis presents, in percentage, the f1-score values, and the x-axis shows the different machine learning models used in the experiment. Each bar represents the f1-score value for a specific model, while the vertical lines above the bars correspond to the confidence intervals, indicating that the values may vary between the range shown. In this way, the graph allows you to visually compare the performance of the models, taking into account the statistical precision of the results presented.

In the general overview, it was noticeable that feature extraction in images with complex backgrounds did not prove to be effective in SVM, with the best result obtained from color extraction. One of the possible causes of this fact is the environment in which the leaves are located, which made it difficult to extract the texture and shape, showing that the model had difficulty

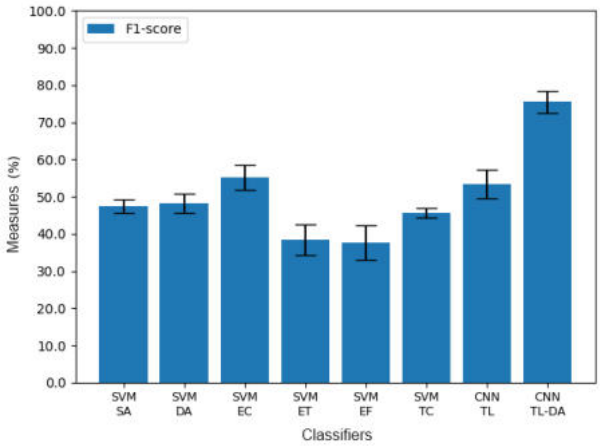


Figure 10: Dataset 1 Confidence Interval

distinguishing between the leaf and the environment. However, as leaves occupy most of the image in most of the dataset, color extraction did not present much difficulty for classification.

On the other hand, CNN presented better results, abstracting more details in the image, which allowed a higher success rate in classification.

Figure 11 presents specimens of the different subspecies present in set 1, making it possible to observe the various colors and shapes of the plants, assuming shades of red, orange, yellow, and green, in different intensities, which highlights the challenges faced by classification models when dealing with this variability.

7.2 Dataset 2 - White background

Dataset 2 used all images with a white background present in the *Japanese Maple Dataset* (Ascari, 2023), still fulfilling the condition of having at least 10 images of the subspecies, similar to Dataset 1. Table 6 displays the subspecies that were used in this set.

The Table 7 presents the results obtained for the classification model applied to Set 2. The precision, recall, and f1-score values and their respective confidence intervals are provided for each combination model and process. The highest f1-score reached 86.54%, with a margin of error of 1.81% and a confidence level of 95%, which was obtained again by the CNN model with transfer learning and data augmentation.

The lowest f1-score was 48.29%, with a margin of error of 4.92% and the same level of confidence obtained by the SVM model with texture extraction.

Figure 12 presents a graphical visualization of the confidence intervals for the models' f1-scores. Each bar in the graph represents the f1-score of a specific model, while the vertical lines over the bars illustrate confidence intervals at a 95% confidence level.

Compared to the results obtained in Dataset 1, an increase in the effectiveness of the models in Dataset 2 was evident. This increase could be attributed to the more favorable conditions for classifying images in Set 2, which did not present complex backgrounds that could influence

Table 5: Weighted average of results referring to Dataset 1

Model	Procedure	Precision	Recall	F1-Score
SVM	No change	52,90% ± 2,16%	49,20% ± 1,38%	47,38% ± 1,33%
	With data augmentation	50,87%± 2,12%	49,89% ± 1,89%	48,13% ± 1,84%
	Color extraction	62,20% ± 1,17%	56,36% ± 2,46%	55,19% ± 2,47%
	Texture extraction	45,89% ± 3,45%	40,34% ± 2,14%	38,38% ± 2,89%
	Format extraction	38,39% ± 3,55%	40,34% ± 3,84%	37,57% ± 3,35%
	All features	48,88% ± 1,13%	51,14% ± 0,66%	45,68% ± 0,94%
CNN	Transfer learning	71,02% ± 3,88%	48,18% ± 2,80%	53,46% ± 2,80%
	Transfer learning and data augmentation	81,90% ± 1,94%	75,11% ± 2,53%	75,55% ± 2,13%

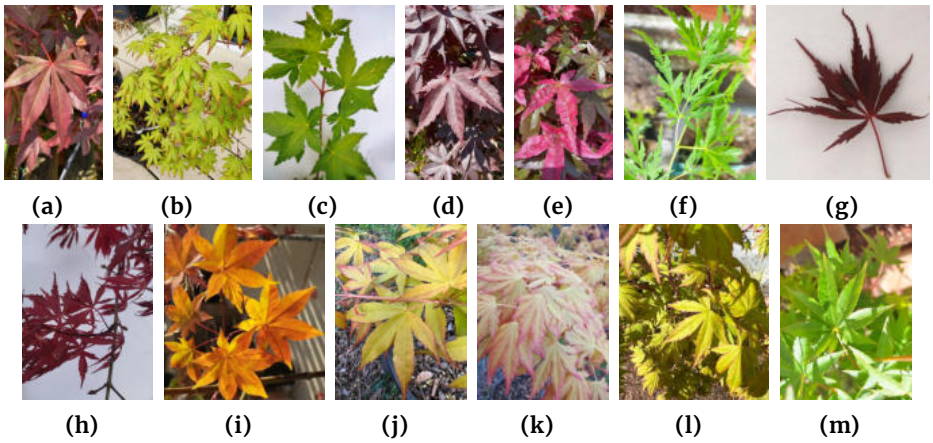


Figure 11: Representative specimens of *Acer palmatum* of the subspecies (a) *atropurpureum*, (b) *beni kawa*, (c) *bihou*, (d) *bloodgood*, (e) *deshojo*, (f) *dissectum*, (g) *dissectum atropurpureum*, (h) *dissectum rubrum*, (i) *hogyoku*, (j) *katsura*, (k) *orange dream*, (l) *sango kaku*, (m) *yamamomiji* (Ascari, 2023)

Table 6: Subspecies present in Dataset 2

Subspecies	Images	Images (%)
atropurpureum	61	17,8%
bihou	34	9,9%
yamamomiji	99	28,9%
deshojo	72	21,0%
dissectum	23	6,7%
dissectum atropurpureum	20	5,8%
dissectum rubrum	34	10,0%

the decisions of the classifier. Furthermore, it is worth highlighting that Set 2 included only 7 subspecies, almost half the number compared to Dataset 1, which may have contributed to a more accurate and effective classification.

The Figure 13 displays specimens of the subspecies present in Dataset 2, and it is possible to notice that the variety of colors has decreased, with variations in red and green being visible.

7.3 Dataset 3 - Background removed

For Dataset 3, subspecies with at least 10 images in the *Japanese Maple Dataset* (Ascari, 2023) were selected for background removal using the *rembg* algorithm. Only images that preserved their essential identifying characteristics were kept, these being only a few images with a white background, while no images with a complex background could meet this criterion. Table 8 presents the subspecies used in this set and the total number of images of each one.

Table 9 illustrates the results obtained by the classification model applied to Dataset 3. The maximum f1-score was 90.44% with a margin of error of 1.31% and a level of confidence of 95%, achieved by the CNN model with transfer learning and data augmentation. The lowest f1-score recorded was 56.29% with a margin of error of 3.64% and a confidence level of 95%, originating from the SVM model with data augmentation.

Figure 14 illustrates the confidence intervals of the f1-scores visually. In the graph, each bar corresponds to the f1-score of a specific model, with vertical lines

Table 7: Weighted average of the results for Dataset 2

Model	Procedure	Precision	Recall	F1-Score
SVM	No change	65,43% ± 2,30%	64,49% ± 2,36%	63,20% ± 2,63%
	With data augmentation	64,58% ± 2,05%	62,61% ± 2,43%	62,65% ± 2,39%
	Color extraction	64,42% ± 1,33%	61,16% ± 1,82%	61,30% ± 1,89%
	Texture extraction	48,03% ± 5,55%	50,87% ± 3,40%	48,29% ± 4,92%
	Format extraction	56,29% ± 1,82%	52,61% ± 2,29%	51,07% ± 1,73%
	All features	58,54% ± 1,91%	59,57% ± 1,33%	55,09% ± 1,54%
CNN	Transfer learning	71,42% ± 4,34%	61,59% ± 5,20%	65,80% ± 5,02%
	Transfer learning and data augmentation	87,33% ± 1,67%	85,94% ± 1,96%	86,54% ± 1,81%

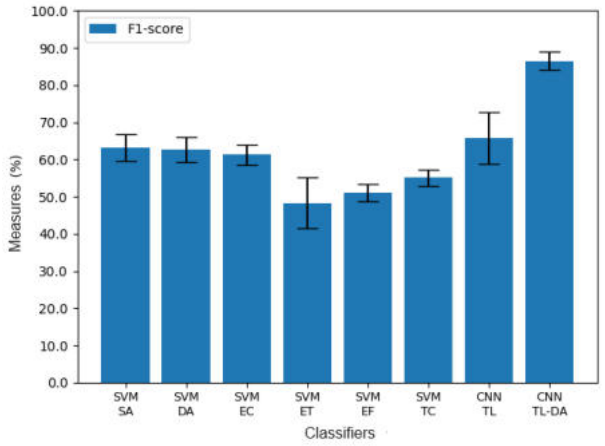


Figure 12: Dataset 2 Confidence Interval

Table 8: Subspecies present in Dataset 3

Subspecies	Images	Images (%)
atropurpureum	57	21,4%
bihou	29	10,9%
yamamomiji	71	26,7%
deshojo	48	18,0%
dissectum	19	7,1%
dissectum atropurpureum	19	7,1%
dissectum rubrum	23	8,6%

over the bars denoting the confidence intervals at a 95% confidence level. In this way, the graph allows a clear visual comparison between the models, considering the statistical accuracy of the results presented.

Dataset 3 presented the best results among all the datasets tested for practically all models. A considerable improvement was observed in the results of the SVM that performed the texture and shape extraction. Removing the background from the images made leaf recognition easier for these models, a problem noted in the other sets. The CNN that applied only transfer learning did not show a considerable difference in its results compared to Dataset 2, indicating that this model had no difficulty differentiating the leaf from the background.

Figure 15 presents a specimen of each subspecies that makes up the dataset with background removed, the same present in Set 2.

8 Model comparison

Since we didn't find studies with similar objectives to those proposed in this work, a direct comparison of the classifiers' performance with related work was not performed. However, similar technologies were employed and presented positive results in other approaches, which were also observed in this study.

In this research, it was possible to notice that the f1-score obtained does not correspond to the value found by the harmonic mean between precision and recall. This is due to the value presented in the results was calculated individually for each sample via code. When the f1-score is calculated directly using the precision and recall averages, a value is obtained that does not consider variation between runs. On the other hand, the value received via code reflects the average of the individual f1-scores of each execution. This performance measure penalizes the difference between precision and recall on each run, resulting in a generally smaller value. Therefore, the average of individual f1-scores is often lower than the f1-score calculated from the precision and recall averages, as each run considers the specific disparity between these measurements.

Concerning the SVM models, data augmentation did not prove effective, given the number of images in each set, presenting a result superior to the set without changing only Set 1 and by a slight difference. Nevertheless, color extraction was more efficient in distinguishing subspecies in images with a complex background, while texture extraction showed better results in images with the background removed. Shape extraction did not obtain the worst results among the characteristics analyzed and did not stand out significantly. Therefore, when the peculiarities of the images are known, using specific feature extraction can be a more practical approach. However, when these properties are unknown, using soft voting with all the features may be a safer choice.

Due to the use of neural networks, CNN models were expected to present better results than SVM. As observed, the CNN with transfer learning and data augmentation demonstrated greater ease in identifying features in the images, allowing us to more accurately differentiate the subspecies of *Acer palmatum*. Data augmentation proved to be a precious technique in image classification, compared to CNN without data augmentation. This model stood out as the best among those tested in this experiment, surpassing all others, regardless of the set evaluated.

(a) (b) (c) (d) (e) (f) (g)

Figure 13: Representative specimens of *Acer palmatum* of the subspecies (a) *atropurpureum*, (b) *bihou*, (c) *deshojo*, (d) *dissectum*, (e) *dissectum atropurpureum*, (f) *dissectum rubrum*, (g) *yamamomiji* (Ascari, 2023)

Table 9: Weighted average of results referring to Dataset 3

Model	Procedure	Precision	Recall	F1-Score
SVM	No change	70,55% ± 0,93%	67,22% ± 0,89%	66,96% ± 1,03%
	With data augmentation	59,61% ± 4,35%	58,15% ± 4,38%	56,29% ± 3,64%
	Color extraction	71,48% ± 2,54%	70,56% ± 2,53%	67,91% ± 2,67%
	Texture extraction	78,76% ± 3,77%	76,85% ± 4,29%	76,86% ± 4,27%
	Format extraction	64,92% ± 2,60%	62,59% ± 2,85%	62,08% ± 2,97%
	All features	60,92% ± 2,79%	67,59% ± 0,70%	61,25% ± 1,40%
CNN	Transfer learning	73,96% ± 2,87%	63,89% ± 2,67%	67,07% ± 2,77%
	Transfer learning and data augmentation	91,88% ± 1,39%	90,00% ± 1,42%	90,44% ± 1,31%

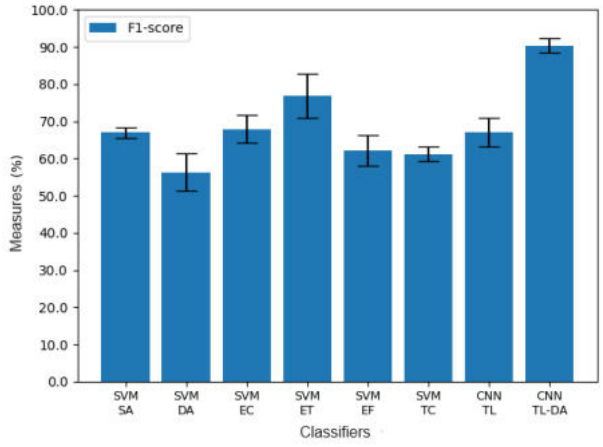


Figure 14: Dataset 3 Confidence Interval

9 Conclusion

Plant classification plays a crucial role in monitoring and mapping biodiversity. Throughout this research, Machine Learning models, such as SVM and CNN, were explored in different scenarios. These experiments highlighted the challenges in classifying *Acer palmatum* subspecies. During the development of this study, a dataset was created involving both the selection of existing images and the collection of new images of different subspecies of *Acer palmatum* from the *Japanese Maple Dataset* (Ascari, 2023). This dataset was fundamental for applying Machine Learning techniques in creating an automatic subspecies classifier of *Acer palmatum*.

Results considered promising were presented, demonstrating that, with the appropriate combination of Machine Learning techniques, it is possible to significantly advance the task of classifying plants, contributing to the knowledge and preservation of biodiversity. The experiments revealed that, although SVM showed some effectiveness, CNN, especially when combined with transfer learning and data augmentation, demonstrated superior performance, achieving an f1-score of 90.44%±1,31% in its best-case scenario.

This study highlighted the importance of adapting and combining different approaches to face the specific challenges of each data set, reinforcing the continuous need for innovation and application of advanced technologies in biodiversity.

10 Difficulties and Challenges

The variability in lighting conditions, capture angles, and complex backgrounds of *Acer palmatum* images available in the dataset used represented a significant challenge to the consistency of the classification models. Removing the background with *rembg* presented difficulties, especially in images where the leaf was not well highlighted from the background. This resulted in the loss of essential features in several images, reducing the removed background dataset by around 30%, considering that there were 438 images in set 1 and only 266 left in set 3. Furthermore, the number of images available for some subspecies was limited, which made it difficult to obtain statistically significant results.

Some feature extractions were tested with SVM, the results of which were not presented, such as edge extraction and leaf venation. The main reason for the absence of these results was that many leaves were not well defined for the extraction of the edges, and, in most images, the photo was not taken close enough for the venation to be visible for the correct classification of subspecies. Collecting new images and focusing on capturing these specific features could have made these strategies viable, potentially improving the performance of the classification model.

The limitation of computational resources was

(a) (b) (c) (d) (e) (f) (g)

Figure 15: Representative specimens of *Acer palmatum* of the subspecies (a) *atropurpureum*, (b) *bihou*, (c) *deshojo*, (d) *dissectum*, (e) *dissectum atropurpureum*, (f) *dissectum rubrum*, (g) *yamamomiji* (Ascari, 2023)

another critical obstacle. CNN models, especially those with transfer learning, require powerful hardware and considerable processing time. The lack of constant access to such resources limited the number of experiments that could be carried out in a reasonable time.

11 Future works

For future experiments, collecting more images for subspecies with few samples could help improve the accuracy and robustness of classification models. Furthermore, including images from different lighting conditions, angles, and backgrounds can contribute to the generalization of models. This would allow models to learn to identify subspecies in various contexts, increasing their applicability in real-world situations.

Considering the model with the best results, the CNN with transfer learning and data augmentation, a more strategic choice of the different variations of data augmentation could help the model obtain superior results while reducing the computational cost. Selecting fewer but more effective variations can optimize the training process. This lower computational cost would allow a more robust choice of the transfer learning model, considering that *MobileNetV3Larg* was partially chosen due to its low computational cost.

Another promising area of research is the investigation of more advanced image segmentation and background removal techniques. Methods that can better preserve the essential characteristics of leaves, regardless of background and capture conditions, are crucial. Improved segmentation can reduce classification errors caused by complex backgrounds and ensure that models focus more accurately on leaf features.

Furthermore, creating mobile applications or web-based systems that utilize the developed classification models can be extremely useful for monitoring biodiversity. These applications could help identify subspecies of *Acer palmatum* in the field, making the technology accessible to researchers and enthusiasts. This would involve the community in conservation efforts, allowing for broader and more collaborative data collection.

Acknowledgements

The authors thank CAPES, CNPq, and UTFPR for supporting this research. The authors also would like to express their great thanks to all the growers who contributed to constructing the *Japanese Maple Dataset*.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

Albarghouthi, A. (2021). Introduction to neural network verification, *Foundations and Trends® in Programming Languages* **7**(1–2): 1–157. Available at http://dx.doi.org/10.1561/2500000051.

Alpaydin, E. (2014). *Introduction to Machine Learning*, The MIT Press, Cambridge, MA, USA.

Alpaydin, E. (2016). *Machine Learning: The New AI*, The MIT Press, Cambridge, MA, USA.

Ara, A. B., Musetti, A. V. and Schneiderman, B. (2003). *Introdução à estatística*, Editora Blucher, São Paulo.

Ascari, R. E. d. O. S. (2023). Japanese maple dataset. Available at https://drive.google.com/drive/folders/1jpddFRKKHRltAFs65UScqArKbCdRRbQe.

Ascari, R. E. d. O. S., Casanova, D., Fávero, E. M. D. B., Sagioratto, P. d. S. and Braun, V. C. (2023). Developing an automatic classifier of different plant genera of the subspecies acer palmatum, pp. 166–173. https://doi.org/10.14210/cotb.v14.p166-173.

Azadnia, R., Al-Amidi, M. M., Mohammadi, H., Cifci, M. A., Daryab, A. and Cavallo, E. (2022). An ai based approach for medicinal plant identification using deep cnn based on global average pooling, *Agronomy* **12**(11): 2723. https://doi.org/10.3390/agronomy12112723.

Barhate, D., Pathak, S., Singh, B. K., Jain, A. and Dubey, A. K. (2024). A systematic review of machine learning and deep learning approaches in plant species detection, *Smart Agricultural Technology* **9**: 100605. https://doi.org/10.1016/j.atech.2024.100605.

Bateman, J., Chapman, R. and Simpson, D. (1998). Possible toxicity of herbal remedies, *Scottish Medical Journal* **43**(1): 7–15. https://doi.org/10.1177/003693309804300104.

Bi, W., Gao, Y., Shen, J., He, C., Liu, H., Peng, Y., Zhang, C. and Xiao, P. (2016). Traditional uses, phytochemistry, and pharmacology of the genus acer (maple): A review, *Journal of Ethnopharmacology* **189**: 31–60. https://doi.org/10.1016/j.jep.2016.04.021.

Blazakis, K. N., Stupichev, D., Kosma, M., El Chami, M. A. H., Apodiakou, A., Kostelenos, G. and Kalaitzis, P. (2024). Discrimination of 14 olive cultivars using morphological analysis and machine learning algorithms, *Frontiers in Plant Science* **15**: 1441737. https://doi.org/10.3389/fpls.2024.1441737.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B. and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project, *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122. Available at https://hal.univ-grenoble-alpes.fr/INRIA/hal-00856511.

Chollet, F. (2021). *Deep learning with Python*, Manning Publications, Shelter Island, NY.

Dana H. Ballard, C. M. B. (1982). *Computer Vision*, Prentice Hall, Englewood Cliffs, New Jersey.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848.

Dirr, M. A. et al. (1990). *Manual of woody landscape plants: their identification, ornamental characteristics, culture, propagation and uses*, number Ed. 4, Stipes Publishing Co, Champaign.

Efron, B. (1992). Bootstrap methods: another look at the jackknife, *Breakthroughs in statistics*, Springer, New York, NY, pp. 569–593. https://doi.org/10.1007/978-1-4612-4380-9_41.

Erkan, U., Toktas, A. and Ustun, D. (2023). Hyperparameter optimization of deep cnn classifier for plant species identification using artificial bee colony algorithm, *Journal of Ambient Intelligence and Humanized Computing* **14**(7): 8827–8838. https://doi.org/10.1007/s12652-021-03631-w.

Faceli, K., Lorena, A. C., Gama, J. and Carvalho, A. C. P. d. L. F. d. (2011). *Inteligência artificial: uma abordagem de aprendizado de máquina*, LTC, Rio de Janeiro.

Forsyth, D. A. and Ponce, J. (2012). *Computer Vision - A Modern Approach, Second Edition.*, Pearson, Boston.

Gadelha Jr, L. M. R., de Siracusa, P. C., Dalcin, E. C., da Silva, L. A. E., Augusto, D. A., Krempser, E., Affe, H. M., Costa, R. L., Mondelli, M. L., Meirelles, P. M., Thompson, F., Chame, M., Ziviani, A. and de Siqueira, M. F. (2021). A survey of biodiversity informatics: Concepts, practices, and challenges, *WIREs Data Mining and Knowledge Discovery* **11**(1): e1394. https://doi.org/10.1002/widm.1394.

Garden, M. B. (2019). Acer palmatum. Available at https://www.missouribotanicalgarden.org/PlantFinder/PlantFinderDetails.aspx?taxonid=275408.

Geisser, S. (1975). The predictive sample reuse method with applications, *Journal of the American statistical Association* **70**(350): 320–328. https://doi.org/10.1080/01621459.1975.10479865.

Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*, " O'Reilly Media, Inc.", Gravenstein Highway North Sebastopol, CA.

Goldschmidt, R. and Passos, E. (2005). *Data mining: um guia prático*, Gulf Professional Publishing, Rio de Janeiro.

Gonzalez, R. C. and Woods, R. E. (2008). Digital image processing, prentice hall, *Upper Saddle River, NJ* .

Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*, MIT Press, Cambridge, MA, USA. Available at http://www.deeplearningbook.org.

Hastie, T., Tibshirani, R., Friedman, J. H. and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, Vol. 2, Springer, New York.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data, *IEEE Transactions on Knowledge and Data Engineering* **21**(9): 1263–1284. https://doi.org/10.1109/TKDE.2008.239.

Heredia, I. (2017). Large-scale plant classification with deep neural networks, *Proceedings of the Computing Frontiers Conference*, CF'17, Association for Computing Machinery, New York, NY, USA, p. 259–262. https://doi.org/10.1145/3075564.3075590.

Howard, A., Pang, R., Adam, H., Le, Q. V., Sandler, M., Chen, B., Wang, W., Chen, L.-C., Tan, M., Chu, G., Vasudevan, V. K. and Zhu, Y. (2019). Searching for mobilenetv3, pp. 1314–1324. https://doi.org/10.1109/ICCV.2019.00140.

Jayanthi, S., Kocha, B. and Sinha, K. K. (1999). Competitive analysis of manufacturing plants: An application to the us processed food industry, *European Journal of Operational Research* **118**(2): 217–234. https://doi.org/10.1016/S0377-2217(99)00022-3.

Ji, S.-B., Yokoi, M., Saito, N. and Mao, L.-S. (1992). Distribution of anthocyanins in aceraceae leaves, *Biochemical Systematics and Ecology* **20**(8): 771–781. https://doi.org/10.1016/0305-1978(92)90036-D.

Jiménez, J. C. D., Valverde, J. A. M., Martínez-Arroyo, M., Bravo, J. M. H. and Hernández Hernández, J. L. (2023). Automatic identification of hermaphrodite papaya applying computer vision and machine learning, *International Conference on Technologies and Innovation*, Springer, pp. 207–219. https://doi.org/10.1007/978-3-031-45682-4_15.

Kecman, V. (2001). *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*, MIT press, Cambridge, MA, USA.

Kohavi, R. et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection, *Ijcai*, Vol. 14, Montreal, Canada, pp. 1137–1145. https://dl.acm.org/doi/10.5555/1643031.1643047.

Lakshmanan, V., Görner, M. and Gillard, R. (2021). *Practical machine learning for computer vision*, " O'Reilly Media, Inc.", Gravenstein Highway North Sebastopol, CA.

Lima, D. L. S. et al. (2023). Classificação de imagens de exames de endoscopia por cápsula utilizando transformers.

Liu, Y., He, Y. and Cui, W. (2018). An improved svm classifier based on multi-verse optimizer for fault diagnosis of autopilot, *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, IEEE, pp. 941–944. https://doi.org/10.1109/IAEAC.2018.8577808.

Medeiros, T. A., Junior, R. G. S., de Souza, G., de Oliveira Nascimento, F. A., de Carvalho, J. L. A. et al. (2023). Radiômica e aprendizado de máquina-uma assinatura para prever o status da codelecao 1p19q em gliomas de baixo grau, *Concilium* **23**(1): 41–56.

Mulugeta, A. K., Sharma, D. P. and Mesfin, A. H. (2024). Deep learning for medicinal plant species classification and recognition: a systematic review, *Frontiers in Plant Science* **14**. https://doi.org/10.3389/fpls.2023.1286088.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*, MIT press, Cambridge, MA, USA.

Nguyen, H. D. and Poo, D. C. C. (2017). Unified structured framework for mhealth analytics: Building an open and collaborative community, *International Conference on Social Computing and Social Media*, Springer, pp. 440–450. https://doi.org/10.1007/978-3-319-58562-8_34.

Nixon, M. S. and Aguado, A. S. (2012). *Feature extraction & image processing for computer vision*, Academic Press, Cambridge, MA, USA.

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L. et al. (2019). Kerastuner. Available at https://github.com/keras-team/keras-tuner.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning, *IEEE Transactions on Knowledge and Data Engineering* **22**(10): 1345–1359. https://doi.org/10.1109/TKDE.2009.191.

Phillips, G. (2003). Culture and propagation of japanese maple.

Pollock, M. and Griffiths, M. (2005). *Illustrated dictionary of gardening*, DK Publishing (Dorling Kindersley), London.

Pratt, W. K. (2007). *Digital image processing: PIKS Scientific inside*, Vol. 4, Wiley Online Library, Hoboken, New Jersey.

Reis, T., Teixeira, M., Almeida, J. and Paiva, A. (2020). A recommender for resource allocation in compute clouds using genetic algorithms and svr, *IEEE Latin America Transactions* **18**(06): 1049–1056. https://doi.org/10.1109/TLA.2020.9099682.

Rippy, M., Krauss, L., Pierce, G. and Winfrey, B. (2021). Plant functional traits and viewer characteristics co-regulate cultural services provisioning by stormwater bioretention, *Ecological Engineering* **168**: 106284. https://doi.org/10.1016/j.ecoleng.2021.106284.

Schölkopf, B., Smola, A. J., Bach, F. et al. (2001). *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, Cambridge, MA, USA.

Scikit-learn (2023). Cross-validation: evaluating estimator performance. Available at https://scikit-learn.org/stable/modules/cross_validation.html.

Sharifalillah, N., Mohd, S. B. and Khairuddin, I. (2008). Plant checklist information model for species identification, *2008 2nd IEEE International Conference on Digital Ecosystems and Technologies*, pp. 38–43. https://doi.org/10.1109/DEST.2008.4635142.

Stevens, E., Antiga, L. and Viehmann, T. (2020). *Deep learning with PyTorch*, Manning Publications, Shelter Island, NY.

Student (1908). The probable error of a mean, *Biometrika* pp. 1–25. https://doi.org/10.2307/2331554.

Szeliski, R. (2022). *Computer vision: algorithms and applications*, Springer Nature, Berlin/Heidelberg, Germany.

Tan, P.-N., Steinbach, M. and Kumar, V. (2016). *Introduction to data mining*, Pearson Education India, New Delhi, India.

Tanner, M. A. and Wong, W. H. (1987). The calculation of posterior distributions by data augmentation, *Journal of the American statistical Association* **82**(398): 528–540. https://doi.org/10.2307/2289457.

Toolbox, N. C. E. G. P. (2020). Acer palmatum. Available at https://plants.ces.ncsu.edu/plants/acer-palmatum/.

Torrey, L. and Shavlik, J. (2010). Transfer learning, *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, IGI global, Hershey, PA, pp. 242–264.

Trees and Online, S. (2024). Acer palmatum. Available at https://treesandshrubsonline.org/articles/acer/acer-palmatum/.

Trivedi, N. K., Maheshwari, H., Tiwari, R. G., Gautam, V. and Agarwal, A. K. (2023). Hybrid deep learning for thai cannabis plant classification: Yolo+ cnn approach, *2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, IEEE, pp. 695–700. https://doi.org/10.1109/ICIMIA60377.2023.10426583.

USDA, NRCS (2024). PLANTS Database. Available at https://plants.sc.egov.usda.gov/.

Wang, Y., Xu, C., Xu, C. and Tao, D. (2018). Packing convolutional neural networks in the frequency domain, *IEEE transactions on pattern analysis and machine intelligence* **41**(10): 2495–2510. https://doi.org/10.1109/TPAMI.2018.2857824.

Wang, Z., Li, H., Zhu, Y. and Xu, T. (2017). Review of plant identification based on image processing, *Archives of Computational Methods in Engineering* **24**(3): 637–654. https://doi.org/10.1007/s11831-016-9181-4.

Weber, M. G. and Agrawal, A. A. (2014). Defense mutualisms enhance plant diversification, *Proceedings of the National Academy of Sciences* **111**(46): 16442–16447. https://doi.org/10.1073/pnas.1413253111.

Wu, S. G., Bao, F. S., Xu, E. Y., Wang, Y.-X., Chang, Y.-F. and Xiang, Q.-L. (2007). A leaf recognition algorithm for plant classification using probabilistic neural network, *2007 IEEE International Symposium on Signal Processing and Information Technology*, pp. 11–16. https://doi.org/10.1109/ISSPIT.2007.4458016.

Zhang, A., Lipton, Z. C., Li, M. and Smola, A. J. (2021). Dive into deep learning, *arXiv preprint arXiv:2106.11342* .

Zhang, B. (2010). Computer vision vs. human vision, *9th IEEE International Conference on Cognitive Informatics (ICCI'10)*, pp. 3–3. https://doi.org/10.1109/COGINF.2010.5599750.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H. and He, Q. (2021). A comprehensive survey on transfer learning, *Proceedings of the IEEE* **109**(1): 43–76. https://doi.org/10.1109/JPROC.2020.3004555.