IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

Efficient Leaf Recognition Algorithm for Plant Classification

Somu Parande

Department of Electronics and Communication,

Basaveshwar Engineering College, Bagalkot, Karnataka, India

Abstract: The research found that services provided by Kerala Bank, as well as those based on artificial intelligence and other developing technologies, were noteworthy. Artificial intelligence (AI), according to the research, has substantial impacts on and links with developing technologies, perceived service quality, customer happiness, and consumer trust. A close second to the correlation between customer trust and satisfaction was the correlation between customers' perceptions of the quality of the service they received. Customers are most impacted by technical revolutions when they promise convenient and high-quality services. The dependability, privacy, security of transactions, and confidence in customer service of Kerala Bank's clients should be guarantyd by the bank's ongoing technical advancements. Increased customer satisfaction in the digital banking age is a result of technology developments that include client-centered service delivery. Customers feel confident when these innovations are implemented.

INTRODUCTION

The function of plants in nature is crucial. Planet Earth's ecosystems could not function in the absence of plants. On the other hand, a number of plant species have lately been threatened with extinction. It is crucial to have a plant database in order to record plant species diversity and to save plants. All around the globe, there is an enormous variety of plant life. Research into the creation of a quick and accurate categorization method has recently stepped up in response to the need to deal with such massive data sets [1]. Recognizing plants is not only important for conservation purposes, but also for harnessing their medical capabilities and turning them into biofuel and other alternative energy sources. A plant may be identified in a number of ways, including by its flowers, roots, leaves, fruit, and so on. A growing number of automated plant detection systems are using computer vision and pattern recognition algorithms [2]. Plants may be found in many kinds of environments, including those with and without humans. There is important data about human civilization's progress in several of them. There is an immediate threat to plant life since many species are in danger of becoming extinct. Therefore, creating a database for the purpose of plant preservation is crucial.

Botany, the tea and cotton industries, and others rely on the categorization of plant leaves [3, 4]. In addition, plant categorization and early disease detection both make use of leaf morphological traits [5]. Identifying plants is a difficult but necessary process. An essential part of plant classification is leaf identification, and the main challenge is determining whether the selected traits are stable and can effectively distinguish between different types of leaves. A lot of time is needed for the recognition process. Due to inadequate models and inefficient representation methodologies, computer-aided plant detection remains a very demanding job in computer vision. For plant identification, the most important criteria are geometrical morphological and Fourier moment based characteristics of leaves. When it comes to classifying plants, this information is crucial. Physiological leaf characteristics such as length, width, diameter, perimeter, area, smooth factor, aspect ratio, and Fourier moments were used in the study by Ji Xiang, Huang, and Xiao Feng [6] to identify known plant species. The technique of plant identification relies heavily on the extraction of leaf characteristics [7, 8]. Pattern recognition faces a new obstacle as a result of this feature extraction procedure [9] [10]. There has been no implementation of the computer-assisted data gathering from live plants. Recognizing and classifying plants is crucial to studies of biological variety and has far-reaching potential uses in agriculture and medicine.

The phrase "artificial immune system" (AIS) describes problem-solving adaptive systems that draw from theoretical and experimental immunology. In order to address real-world issues, they include any computational tool or system that draws inspiration and analogies from the biological immune system [12–15]. A wide variety of learning and recalling capabilities, as well as pattern matching, are built into artificial immune systems. To effectively adapt the AIS paradigm to many areas, researchers are actively working on it [12–14]. An artificial immune system may be implemented using the following algorithms: the immune network, the clone selection algorithm, and the negative

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

selection algorithm [12, 13]. Dendritic Cell Algorithm (DCA), a theory-based new paradigm, was recently designed [14]. Data categorization is only one area where this immune-inspired method has found widespread use.

Instead of differentiating between self and non-self proteins, the danger hypothesis posits that the immune system reacts upon detection of host harm [15, 16]. It comes down to differentiating between self-harming invaders (also called antigens) and non-self-harming ones in this situation. A differentiation would be useful if interesting and non-interesting data were to replace the terms self and non-self. The AIS is being used as a classifier in this instance [16].

In any case, without the right models or representation approaches, plant identification is a difficult but crucial endeavor. Classification based on leaf image is the preferred way for plant classification, as opposed to other approaches like cell and molecular biology methods. It is easy and inexpensive to photogene leaves. On top of that, you can find and gather leaves just about anywhere. It is feasible to identify various plants with relative ease by calculating a few efficient leaf attributes and using an appropriate pattern classifier.

Objectives of the Project

The automated method of plant identification has made use of computer vision and pattern recognition tools. Further uses for leaf morphology include plant taxonomy and the early detection of certain plant diseases. Identifying plants is a difficult but necessary process. An essential part of plant classification is leaf identification, and the main challenge is determining whether the selected traits are stable and can effectively distinguish between different types of leaves. A lot of time is needed for the recognition process. Due to inadequate models and inefficient representation methodologies, computer-aided plant detection remains a very demanding job in computer vision. For plant identification, the most important criteria are geometrical morphological and Fourier moment based characteristics of leaves. When it comes to classifying plants, this information is crucial.

Pattern recognition and classification are two areas where Support Vector Machines (SVM) have lately shown their worth [17]. This paper's objective is to assess SVM's capabilities for picture identification and picture classification jobs.

The basic idea behind a linear support vector machine (SVM) is that it takes a collection of points that can be classified into two classes and uses them to determine the hyperplane that maximizes the distance between the two classes while keeping the maximum number of points from the same class on the same side.

Image classification problems are particularly challenging for traditional classification algorithms due to the large dimensionality of the feature space. In this research, we demonstrate that support vector machines (SVMs) perform admirably when faced with challenging picture classification tasks that only include high-dimensional histograms as features.

Optimal Separating Hyperplane

Let $(x_i, y_i) 1 \leq i \leq N$ be a set of training examples, $x_i \in R^n$ and is a member of the group indicated by the symbol.In order to ensure that all points with the same label are on the same side of the hyperplane, it is necessary to carry out the equation of a hyperplane that splits the set of instances. Seek for w and b in such a way that

$$yi(w \cdot xi + b) > 0, i = 1, \ldots, N \qquad (2.1)$$

A set is considered linearly separable if and only if a hyperplane exists that satisfies equation (2.1). Here, rescaling w and b is always an option, allowing for

$$\min_{1 \leq i \leq n} yi(w \cdot xi + b) \geq 1, i = 1, \ldots, N$$

meaning that there is a distance of from the nearest point to the hyperplane $1/||w||$. Therefore, equation (3.1) is transformed into

$$yi(w \cdot xi + b) \geq 1 \qquad (2.2)$$

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

The term "optimal separating hyperplane" (OSH) is used to describe the separating hyperplane that has the shortest distance to the nearest point $1/||w||$. Because the distance to the nearest point is, determining the OSH is equivalent to solving the following problem: minimize within limitations. (2.2)

$$\frac{1}{2}w.w \qquad (2.3)$$

The quantity $2/||w||$ is called the margin and thus, The OSH optimizes the margin by acting as a separating hyperplane. The complexity of the task is proportional to the size of the margin, therefore a smaller margin indicates a more challenging problem. As seen in figure 2.1, the generality is projected to be better with a wider margin.

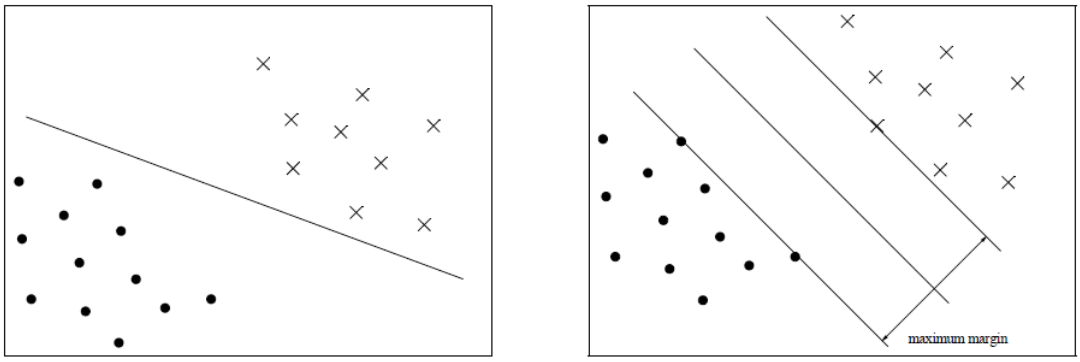


Figure 2.1: Both hyperplane separate correctly the training examples. But the Optimal Separating Hyperplane on the right hand side has a larger margin and is expected to give better generalization

Since w^2 is convex, and we may minimize equation (2.3) under linear constraints (2.2) by using Lagrange multipliers; the N non-negative Lagrange multipliers linked to constraints (2.2) are by $\alpha = (\alpha_1, \ldots, \alpha_N)$. Finding the saddle point of the Lagrange function is equivalent to minimizing eq (2.3):

$$L(w, b, \alpha) = \frac{1}{2}w.w - \sum_{i=1}^{N} \alpha_i [y_i(w.x_i + b) - 1] \qquad (2.4)$$

Locating this spot requires maximizing the function over the Lagrange multipliers and minimizing it over w and b, $\alpha_i \geq 0$, Here are the requirements that the saddle point must meet:

$$\frac{\partial L(w, b, \alpha)}{b} = \sum_{i=1}^{N} y_i \alpha_i = 0 \qquad (2.5)$$

$$\frac{\partial L(w, b, \alpha)}{w} = w - \sum_{i=1}^{N} y_i \alpha_i x_i = 0 \qquad (2.6)$$

Our optimization issue amounts to maximizing by substituting equations (2.5) and (2.6) into (2.4).

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

$$w(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j x_i . x_j \qquad (2.7)$$

With $\alpha_i \geq 0$ and although limited in scope (2.5). Common quadratic programming techniques may do this [20].

Once the vector $\alpha^0 = (\alpha^0{}_1, \ldots, \alpha^0{}_N)$ After considering the answer to the maximizing issue (2.7), the following extension of the OSH (w_0, b_0) is obtained:

$$w_0 = \sum_{i=1}^{N} \alpha^0{}_1 \, y_i x_i \qquad (2.8)$$

While b_0 can be determined from the Kuhn-Tucker conditions

$$\alpha^0{}_i [y_i (w_0 . x_i + b_0) - 1] = 0 \qquad (2.9)$$

Note that from equation (2.9), the points for which $\alpha^0{}_i > 0$ hold true when solving equation (3.2) for x. In geometric terms, it indicates that these locations are quite near to the ideal hyperplane (see to figure 2.1).Given that the formulation of the OSH requires just these points (refer to equation (2.8)), their significance cannot be overstated. They are called support vectors to point out the fact that they "support" the expansion of w_0.

Given a support vector x_i, the parameter b_0 becomes possible by using the matching Kuhn-Tucker criterion as

$$b_0 = y_i - w_0 \cdot x_i$$

The problem of classifying a new point x is solved by looking at the sign of

$$w_0 \cdot x + b_0$$

If we take into account the expansion (2.8) of w0, we may express the hyperplane choice function as

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{N} \alpha_i y_i x_i . x + b \right) \qquad (2.10)$$

2.2 Linearly non-separable case

The challenge of identifying the OSH loses all value if the data are not linearly separable. To make it possible for instances to violate (2.2), one can introduce slack variables $(\xi_1, \ldots, \xi_N)$ with $\xi_i \geq 0$ [18] such that

$$y_i (w \cdot x_i + b) \geq 1 - \xi_i, i = 1, \ldots, N \qquad (2.11)$$

The purpose of the variables ξ_i is to allow misclassified points. Misclassified points are paired with their $\xi_i > 1$, so $\sum \xi_i$ sets a maximum allowable number of training mistakes. In this case, the generalized OSH is seen as the answer to the following issue: cut down

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

$$\frac{1}{2}\mathrm{w.w} + \mathrm{c}\sum_{\mathrm{i}=1}^{\mathrm{N}} \xi_i \qquad (2.12)$$

Subject to constraints (2.11) and $\xi_i \geq 0$. As in the separable situation, the first term is reduced to limit the capacity learning, while the second term is used to keep the number of misclassified points under control. The user decides on the parameter C; a bigger C means that a heavier penalty will be applied to mistakes.

Using the same reasoning as for the separable example, we may solve the following optimization issue by making use of Lagrange multipliers: attain full potential

$$w(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j x_i x_j$$

Subject to: $\sum_{i=1}^{N} \alpha_i y_i = 0$ and $0 \leq \alpha_i \leq C$. When compared to the separable case, α_i the only change is the presence of an upper constraint of C.

2.3 Nonlinear Support Vector Machines

The basic premise of SVMs is to use non-linear mappings that are predetermined to transform the input data into a high-dimensional feature space [19]. Figure 2.2 shows the area where the OSH was built.

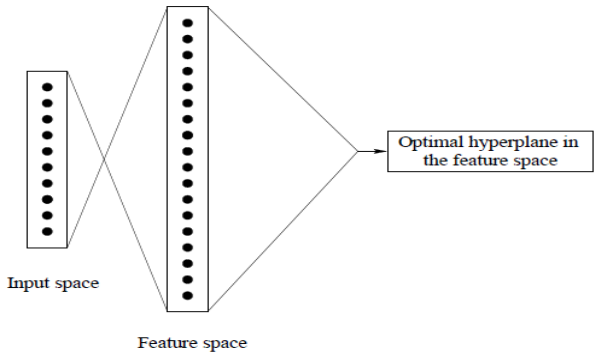


Figure 2.2

After creating a high-dimensional feature space from the input space, the SV machine builds an optimal hyperplane in that space.

METHODOLOGY

Here is the methodology that was employed to bring the system to life. The Irvin Watson Carpenter, Jr. Herbarium, housed at Appalachian State University's Department of Biology, is the principal repository for leaf photographs. There are images that come from the University of North Carolina Herbarium and the Robert K. Godfrey Herbarium at Florida State University. Images of many plant components, including leaves and flowers, are stored in the specimen databases of these herbaria. There are three steps to this method: preprocessing the images, extracting features, and classification.

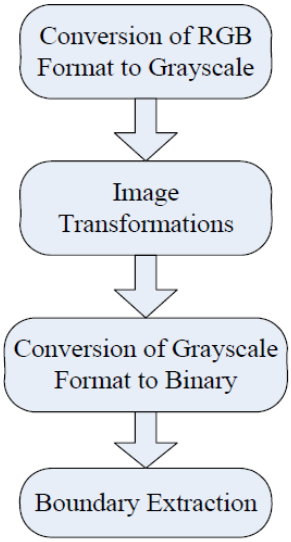
Image Pre-processing: Due to their significant size (up to 17 Megabytes) and RGB color mode, the photos obtained from the Irvin Watson Carpenter, Jr. Herbarium are unsuitable for image processing. Processing a 17 MB color picture requires a lot of computing power. All the pixel values in a binary picture may be either zero or one, making calculations simpler. This makes feature extraction using binary images easier. Consequently, all morphological (shape-related) information is preserved throughout the preprocessing and binary format conversion of the photos, resulting in reduced file sizes. Flow Chart 3.1 shows the stages of picture preparation.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115



Flow Chart 3.1 Four Steps of Methodology

1 Conversion of RGB Format to Gray scale

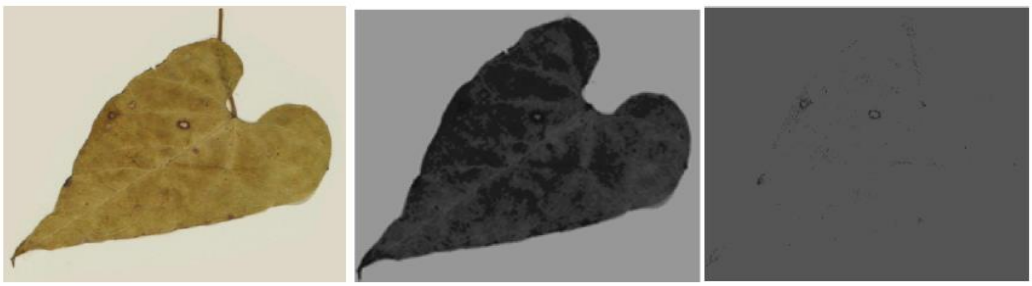
There are two ways to get the grayscale value for each pixel in an RGB digital picture: averaging and weighted averaging. The averaging technique would determine the grayscale intensity, GI, based on (3.1) if R, G, and B were the corresponding intensities for the red, green, and blue channels of a pixel.

$$GI = \frac{R+G+B}{3} \tag{3.1}$$

While both methods use averages, weighted averaging gives different intensities of colors a different relative importance. The research weights are specified in equation (3.2).

$$GI = 0.2989 * R + 0.5870 * G + 0.1140 * B \tag{3.2}$$

Fig. 3.1 shows both the original RGB picture and the grayscale versions that were created using various averaging and weighting methods. Since the weights applied can only provide an ideal outcome, the weighted averaging approach yielded a picture with improved foreground-background pixel separation.



a) b) c)

Figure 3.3 a) An arbitrary RGB image, b) Grayscale equivalent using weighted averaging, c) Grayscale equivalent using averaging

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

Image Transformations: The generic form for the modification of picture coordinates or pixels by operations like scaling, rotation, translation, and shear is shown in (3.3).

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = T \begin{bmatrix} v \\ w \\ 1 \end{bmatrix} \quad (3.3)$$

Where (v, w) is a pixel in the source picture, (x, y) is the coordinate of the corresponding pixel in the edited image, and T is the matrix of transformations. To make the picture file smaller, the scale transformation is used. Image resizing makes use of the inverse mapping process. At any given point in space (x, y), the output pixel is calculated by adding up the contributions of the nearby pixels that are relevant inputs. By calculating the average of the values of the closest input pixels, we can get the value of the output pixels.

Conversion of Greyscale Format to Binary:

Converting a reduced-size grayscale picture to binary format is the next step. Using Otsu's [21] automated thresholding approach, a grayscale picture may be transformed into a binary image. When thresholding a picture, pixels are classified as either foreground or background depending on whether their values are greater than a certain threshold. Automatic thresholding is based on the histogram's form, with the goal of minimizing the interclass variation of the black and white pixels.

Boundary Extraction: For edge identification and area of interest extraction, the Canny edge detector is used. There are four main components to the Canny edge detection algorithm:

1. Using a Gaussian filter to smooth out the input picture
2. Calculating the angle pictures and gradient magnitude
3. Identifying and connecting edges via the use of connectivity analysis and double thresholding
4. Reducing the gradient magnitude picture using a method other than maximum suppression

Step 1: To apply a Gaussian filter to the input picture and smooth it out, we utilize a 2-dimensional Gaussian function, G, which is defined by equation (3.4).

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2a^2}} \quad (3.4)$$

Where, $a > 0$ is a constant, σ is Euler's constant, and $f(x,y)$ denotes the input image. If $G(a,b)$ denotes a discrete Gaussian filter of size m by n, the smoothened image, $fs(i,j)$ is formed by convolving G and f and the intensity value at pixel (i,j) is given by (3.5).

$$f_s(i,j) = \sum_{a=1}^{m} \sum_{b=1}^{n} f(i-a, j-b) G(a,b) \quad (3.5)$$

The impulse response of a Gaussian filter is also Gaussian. The study used a discrete Gaussian filter, as shown in Table 3.1.

$\frac{1}{273}\times$	1	4	7	4	1
	4	16	26	16	4
	7	26	41	26	7
	4	16	26	16	4
	1	4	7	4	1

Table 3.1 A Gaussian filter

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

Step 2: The next step in picture smoothing is to calculate the gradient magnitude and angle for each pixel (x, y) in the image. The gradient and angle are computed using (3.6). Where, $G(x,y)$ is the magnitude at pixel (x,y) and $\alpha(x,y)$ is the gradient angle at pixel (x,y).

$$G(x,y) = e^{\frac{x^2+y^2}{2a^2}}$$

$$\alpha(x,y) = \tan^{-1}\left[\frac{g_x}{g_y}\right] \qquad (3.6)$$

Step 3: Finding and connecting edges with double thresholding

Lastly, we need to decrease the amount of spurious edge points. In contrast to previous algorithms that rely on a single threshold to reduce the number of false edges, the Canny edge detector [22] makes use of two thresholds. Hysteresis thresholding with a low threshold TL and a high threshold TH is utilized because using only one threshold will lead to inaccurate results (a high threshold will identify some bogus edges and a low threshold would delete some legitimate edges). In contrast, pixels are preserved if their magnitude falls within the range of TL ≤ D < TH, while those with a gradient magnitude D < TL and D > TH are promptly deleted.

Step 4: Applying non-maxima suppression to the gradient image

There is no pixel thickness to the edges since they are identified using a gradient. The local maxima surrounding the margins are used to make them one pixel thick. To thin the edges, non-maxima suppression is used. Pixels on the edge with the largest gradient magnitude are the only ones retained by non-maxima suppression. Each pixel (x, y) is checked for the following orientations, as shown in figure 4.4: If the direction of (x, y) is between $-22.5^0$to 22.5^0or from -157.5^0 to 157.5^0, then it lies in the vertical border.If the direction of (x, y) is between 67.5^0 to112.5^0 or -67.5^0 to-112.5^0, then it's lying on the border that's horizontal. Each pixel's vertical neighbors are checked for whether it's located in the boundary. We look at the pixels directly next to the pixel on the horizontal axis if it is on the border. If the magnitude of (x, y) is larger than the magnitude of all the neighbors that were considered, then it is retained on the edge; otherwise, it is eliminated.

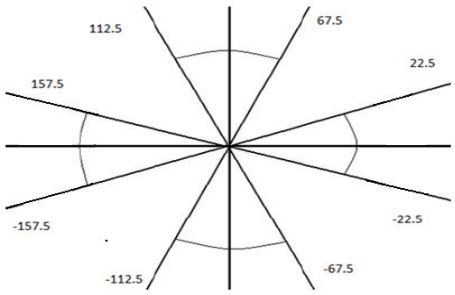


Figure 3.4 Separation of angles for determining a vertical/horizontal neighbor in Non-maxima suppression

Fundament Fundamental Steps in Image Processing In Digital Domain

Image Acquisition: Learn where images come from and how to obtain them with the help of Image Acquisition. Scaling and other preprocessing steps are a part of the image capture stage. Reducing or enlarging the image's physical size by adjusting the pixel count is called scaling. Digital images are created via the process of picture acquisition.

Image Enhancement: The purpose of an enhancement approach is to either make previously hidden details more visible or to draw attention to certain parts of a picture. Elevation is a subjective endeavor. To improve the picture, mathematical instruments are used.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 1, January 2024

DOI: 10.17148/IJIREEICE.2024.12115

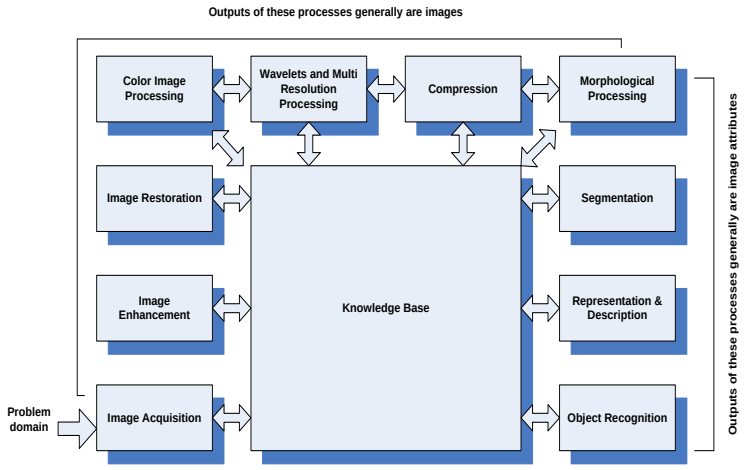


Image Restoration: recovery refers to the process of acquiring something once again. Restoration of images is a method that is unbiased. Eliminating background noise is an important part of picture restoration. Image degradation models, whether mathematical or probabilistic, provide the basis of restoration strategies.

Wavelets: Image data compression and pyramidal representation, in which pictures are broken into smaller sections, are two applications of wavelets. pictures may be represented using wavelets at several resolutions (multi resolution).

Compression: Image compression is a method for reducing the amount of space needed to save or transmit a picture. Internet services that transport large amounts of visual material benefit from compression. JPEG files are a kind of picture compression.

Morphological Processing: The field of morphological image processing focuses on methods for obtaining shape-representative picture components. Beginning with the processing of the image's morphology, the next step is to handle the image's properties.

Segmentation: Segmentation processes divide a picture into its individual elements. Object identification is the end result of autonomous or rough segmentation.

Representation and Description: Raw pixel data representing the area's boundaries or all of the points within the region are typically what segmentation returns. When looking at the outside of a form, boundary representation is the way to go. When looking at internal features like texture, regional representation is the way to go. In order to prepare raw data for further processing by computers, selecting a representation is only one component of the solution. The process of describing an item, also known as feature selection, entails collecting characteristics that provide quantitative data for object identification.

Recognition: A procedure called recognition uses an object's description to give a label to it.

Knowledge base: In the context of knowledge management, a knowledge base refers to a subset of databases. The image processing system's issue area may be better understood with the help of a knowledge database.

Additionally, it directs how each processing module should be operated. Additionally, it regulates how modules communicate with one another. al Steps in Image Processing In Digital Image.

CONCLUSION

This article presents an effective machine learning method for plant leaf identification and proposes a novel method of plant categorization based on leaves recognition. The procedure had three stages: preprocessing, feature extraction, and classification. When digital cameras or scanners capture photographs of leaves, the computer may automatically identify the species. During the feature extraction process, the more fundamental characteristics are employed to extract widely used digital morphological features (DMFs). Because of its straightforward structure, quick training time, and high accuracy, the support vector machine classifier is chosen for the classification method. In order to create the input vector for SVM, features are extracted and processed via PCA. Precision and runtime are the metrics used to assess the suggested method's efficacy. The suggested technique outperforms the k-NN method in terms of accuracy while requiring much less processing time during execution. Improving the classifier's performance is possible in future studies by using efficient kernel functions.

DOI: 10.17148/IJIREEICE.2024.12115

REFERENCES

[1] J. S. Cope, D. Corney, J. Y. Clark, P. Remagnino, and P. Wilkin, "Plant species identification using digital morphometrics: A review," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7562–7573, 2012.
J. Chaki and R. Parekh, "Plant Leaf Recognition using Gabor Filter," *International Journal of Computer Applications*, vol. 56, no. 10, pp. 26–29, Oct. 2012. ijcaonline.org+1
[2] 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.
[3] J. Chaki, R. Parekh, and S. Bhattacharya, "Recognition of Whole and Deformed Plant Leaves using Statistical Shape Features and Neuro-Fuzzy Classifier," in *Proc. IEEE International Conference on Recent Trends in Information Systems (ReTIS)*, 2015, pp. 189–194.
[4] J. Chaki, R. Parekh, and S. Bhattacharya, "Plant Leaf Recognition Using a Layered Approach," in *Proc. IEEE International Conference on Microelectronics, Computing and Communications (MicroCom)*, 2016.
[5] J. Chaki, R. Parekh, and S. Bhattacharya, "Recognition of Plant Leaves with Major Fragmentation," in *Proc. International Conference on Computer and Systems Engineering (ICCSE)*, 2016.
[6] C. Arun Priya, T. Balasaravanan, and A. Thanamani, "An efficient leaf recognition algorithm for plant classification using Support Vector Machine," in *Proc. Int. Conf. Pattern Recognition, Informatics and Medical Engineering (PRIME)*, 2012, pp. 428–432.
[7] B. Yanikoglu, E. Aptoula, and C. Tirkaz, "Automatic plant identification from photographs," *Machine Vision and Applications*, vol. 25, no. 6, pp. 1369–1383, 2014.
[8] 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.
[9] A. Aakif and M. F. Khan, "Automatic classification of plants based on their leaves," *Biosystems Engineering*, vol. 139, pp. 66–75, 2015.
[10] J. Wäldchen and P. Mäder, "Plant species identification using computer vision techniques: A systematic literature review," *Archives of Computational Methods in Engineering*, vol. 25, no. 2, pp. 507–543, 2018.
[11] M. M. Islam, A. S. A. Rabby, M. H. R. Arfin, and S. A. Hossain, "PataNet: A convolutional neural network for plant identification from leaf images," in *Proc. IEEE Int. Conf. Image, Vision and Pattern Recognition*, 2019.
[12] K. K. Thyagharajan and I. Kiruba Raji, "A review of visual descriptors and classification techniques used in leaf species identification," 2020.
[13] S. Sachar and A. Kumar, "Survey of feature extraction and classification techniques to identify plant through leaves," *Expert Systems with Applications*, vol. 167, Art. no. 114181, 2021.
[14] P. S. Kanda, K. Xia, and O. H. Sanusi, "A deep learning-based recognition technique for plant leaf classification," *IEEE Access*, vol. 9, pp. 162590–162613, 2021.
[15] S. Veni, R. Anand, D. Mohan, and P. Sreevidya, "Leaf recognition and disease detection using content-based image retrieval," in *Proc. 7th Int. Conf. Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2021, pp. 243–247.
[16] B. Yanikoglu, E. Aptoula, and C. Tirkaz, "Automatic Plant Identification from Photographs," *Machine Vision and Applications*, vol. 25, no. 6, pp. 1369–1383, 2014. This extended recognition from scanned leaf images to natural photographs.
[17] S. H. Lee, C. S. Chan, P. Remagnino, and S. J. Mayo, "How Deep Learning Extracts and Learns Leaf Features for Plant Classification," *Pattern Recognition*, vol. 71, pp. 1–13, 2017. This marked the shift from handcrafted features and traditional neural networks to convolutional neural networks (CNNs).
[18] J. Wäldchen and P. Mäder, "Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review," *Archives of Computational Methods in Engineering*, vol. 25, no. 2, pp. 507–543, 2018. This comprehensive review documents the evolution from early ANN-based approaches to modern deep learning methods.
[19] K. K. Thyagharajan and I. Kiruba Raji, "A review of visual descriptors and classification techniques used in leaf species identification," 2020.
[20] P. S. Kanda, K. Xia, and O. H. Sanusi, "A deep learning-based recognition technique for plant leaf classification," *IEEE Access*, vol. 9, pp. 162590–162613, 2021.
[21] S. Sachar and A. Kumar, "Survey of feature extraction and classification techniques to identify plant through leaves," *Expert Systems with Applications*, vol. 167, Art. no. 114181, 2021.