International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-11, September 2019

Leaf Identification using Harris Corner Detection, SURF Feature and FLANN Matcher

Juby George, Gladston Raj S

Abstract: ***Leaf Recognition is very important in agriculture for identification of plants. Leaves of various plants have unique characteristics which are to be used for categorization. Out of the different features, leaf vein is one of the prominent biometric feature. Extracting leaf vein and perform classification based on these features leads to more accurate identification of plants. In practice, due to change in various lighting conditions and orientations, the extraction of leaf vein becomes difficult. This work focuses on extracting veins using ridge orientation and frequency estimation using region mask which brings out good quality vein structure under different conditions. The vein structure thus obtained is used for identifying keypoints using Harris corner detector. Features are extracted from the keypoints using SURF feature extraction method and finally the trained and query images are compared to identify the correct leaf species using FLANN matcher. Flavia leaf image database with 32 different species are used and an accuracy of 98.75% was resulted. The proposed methodology can be used for plant leaf identification in real world for identifying medicinal plants and other category of plants. This method can be used for identifying veins of dry leaves which can further extract the features and identify the species.***

Keywords: ***Frequency estimation, Region mask, Harris Corner detector, SURF Feature Descriptor, FLANN matcher.***

I. INTRODUCTION

Plant recognition has got broad applications in agriculture and medicine, which are significant to the bio diversity research. Its importance is that many plants are at the risk of extinction just because of not having proper identification mechanism so that they can be conserved. Leaf recognition plays an important role in plant classification. Shape and vein of leaves helps to characterize and recognize plant species more accurately. Leaf veins are normally two-dimensional in nature and are suitable for machine processing.

The present leaf identification system mainly focus on vein and other morphological features. For any leaf image, the region of interest are categorized into two category:

- *Well-defined region*, where primary veins and secondary veins are clearly differentiated from one another so that edge detection algorithm is able to extract the veins of leaf image.
- *Distorted Regions*, where veins are distorted by a small amount of shade and lighting condition.

Manuscript published on 30 September 2019.
*****Correspondence Author(s)
Juby George, Research Scholar, School of Computer Science, Mahatma Gandhi University, Kottayam, 686560 Email: jubygeorgek@rediffmail.com
Gladston Raj S, Department of Computer Science, Govt. College Nedumangad, Thiruvananthapuram, Kerala Email: gladston@rediffmail.com

This paper presents how leaf veins are segmented and extracted using frequency estimation and region mask. The enhanced leaf vein image is then used to identify keypoints using Harris corner detection. The keypoints thus extracted is used for retrieving features by SURF feature detector and the obtained feature is used by FLAAN matcher. The matching score thus obtained is used for identifying leaf species. A minimum matching score accurately predict the correct leaf species.

Proposed Leaf Identification Algorithm.

Step 1: Pre-processing and enhancement of leaf vein image using ridge frequency estimation and region mask.

Step 2: Keypoint identification in vein image using Harris corner detection.

Step 3: The number of keypoints extracted from the Standard image and query images are analysed and if it falls within the limit then go to step 4 otherwise a mismatch.

Step 4: Extract Leaf feature descriptors from the keypoints using SURF feature descriptor.

Step 5: Match the feature descriptors of Standard image and query image using FLANN matcher.

Figure 1 illustrate the flow diagram of proposed algorithm.

II. PREPROCESSING AND LEAF VEIN ENHANCEMENT

Algorithm 1: Algorithm for leaf Vein extraction and enhancement

Step 1: Normalize the input image
Step 2: Output the orientation image from normalized input image.
Step 3: Estimate the frequency using gradient magnitude.
Step 4: Apply the region mask by forming meshX and meshY.
Step 5: Perform Postprocessing filtering to avoid the outliers.
Step 6: Result out the Enhanced Leaf Vein Image.

Leaf species of 32 different categories from Flavia Database is used for this work.

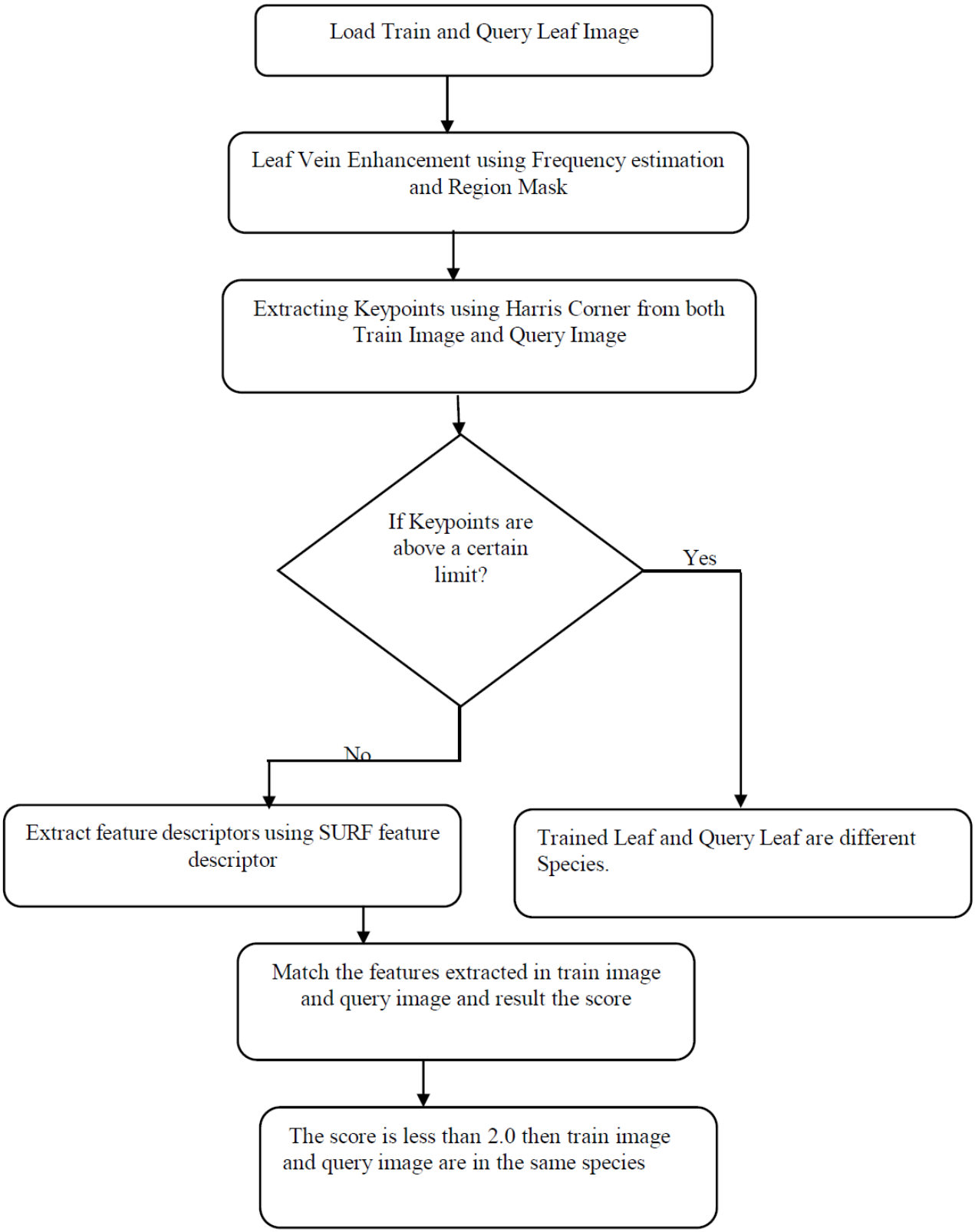


Fig. 1. Flow Diagram of the Proposed Algorithm.

Retrieval Number: K20160981119/19©BEIESP
DOI: 10.35940/ijitee.K2016.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

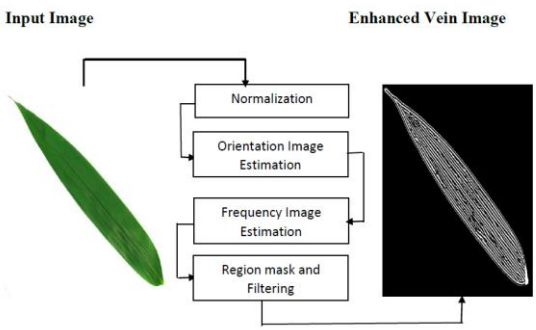


Fig 2. Leaf vein enhancement process

A. *Normalization*

The first step in the leaf vein enhancement process is image normalisation. Normalisation is used to standardise the intensity values in an image by adjusting the range of grey-level values so that it lies within a desired range of values [2]. Let $I(i,j)$ represent the grey-level value at pixel (i,j), and $N(i,j)$ represent the normalised grey-level value at pixel (i,j). The grey-level value for each pixel is then compared with the average grey level value for the host block. For a pixel $I(i,j)$ belonging to a block of average grey-level value of M_0, the result of comparison produced a normalized grey-level value $G(i,j)$ [3]. The normalised image is defined as:

$$G(i,j) = \begin{cases} M_0 + \sqrt{\frac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{if } I(i,j) > M \\ M_0 - \sqrt{\frac{VAR_0(I(i,j)-M)^2}{VAR}} & \text{otherwise} \end{cases} \quad (1)$$

Where M_0 and VAR_0 are the estimated mean and variance of $I(i,j)$, respectively,

Steps to Normalize the Input Image

- Reading the image
- Smoothing the image using medianBlur.
- Converting to grayscale.
- Perform normalization by calculating Mean and variance.
- Adjust the gray level value to the normalized mean and variance.
- Returning a normalized image as output.

B. *Orientation image estimation and Frequency image estimation*

After normalization the estimation of the orientation of the image is carried out. Given a normalized image $G(i,j)$, the algorithm works by dividing the whole image into blocks of size 16×16 The steps are [3]:

1. Compute the gradients at each pixel (i, j).
2. Estimate the local orientation of each block centered at pixel (i, j) is computed by

$$\begin{aligned} X[k] &= \frac{1}{w}\sum_{d=0}^{w-1} \mathcal{G}(u,v), \quad k = 0,1,\ldots,l-1, \\ u &= i + (d - \frac{w}{2})\cos\mathcal{O}(i,j) + (k - \frac{l}{2})\sin\mathcal{O}(i,j), \\ v &= j + (d - \frac{w}{2})\sin\mathcal{O}(i,j) + (\frac{l}{2} - k)\cos\mathcal{O}(i,j). \end{aligned} \quad (2)$$

Where $\theta(i,j)$ is the least square estimate of the local ridge orientation at the block centered at pixel (i, j). Let $G(i,j)$ be the normalized image and $\theta(i, j)$ be the orientation image, the steps for frequency estimation are as follows:

a. Divide $G(i,j)$ into block size of 16 X 16.
b. For each block centered at pixel(i, j), compute an oriented window size l x w
c. The angles between the blocks are then smoothened by passing the image through a low pass filter as follows.

$$\begin{aligned} \mathcal{V}_x(i,j) &= \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2\partial_x(u,v)\partial_y(u,v), \\ \mathcal{V}_y(i,j) &= \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} (\partial_x^2(u,v) - \partial_y^2(u,v)), \\ \theta(i,j) &= \frac{1}{2}tan^{-1}(\frac{\mathcal{V}_y(i,j)}{\mathcal{V}_x(i,j)}), \end{aligned}$$

(3)

The following method is adopted for the calculation of the frequency of the local blocks. X-signatures of each block are computed along the direction perpendicular to the orientation angle in each block. The window used for this purpose is of size 16×32. The frequency is then computed by the distance between the peaks obtained in the X-signatures [3].

Steps for calculating the frequency of gradient magnitude

- Input is the normalized image with gradientsigma, blocksigma and orient smoothsigma values.
- Define the Gaussian kernel using SetGaussiankernal function.
- Perform filtering using Filter2D function and Gaussian kernel.
- Gradient of x and y calculated (horizontal and vertical).
- The steps to be repeated by using Blocksigma as well as Orientsmoothsigma to calculate the covariance and the magnitude.
- The Gradient magnitude of the image is calculated and as a result orientation image will be produced as the output.

Region mask is obtained by classifying each block in the normalized image. The frequency values for this blocks need to be interpolated from the frequency of the neighbouring blocks which have a well-defined frequency. meshX and meshY used for interpolation is calculated using the meshgrid function. A low pass filter is then used to remove the outliers. The enhanced leaf image is outputted as the result.

Result of Pre-processing and Enhancement

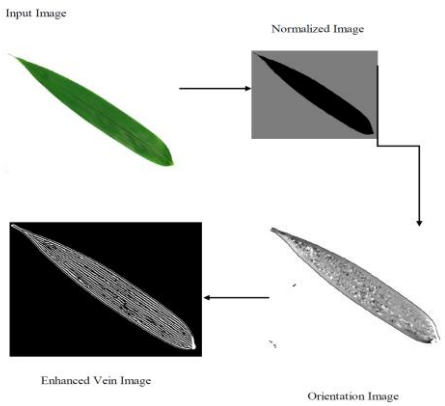


Fig. 3. Example of Pre-processing and Enhancement.

Input Image

III. KEYPOINT EXTRACTION USING HARRIS CORNER DETECTION

The search for discrete image point in an image correspondence can be divided into three main steps.

1. Interest points are selected at distinctive locations in the image such as corners, blobs and T-junctions [7]. The most valuable property of an interest point detector is its repeatability. The repeatability expresses the reliability of a detector for finding the same physical interest points under different viewing conditions.
2. The neighbourhood of every interest point is represented by a feature vector. This descriptor has to be distinctive and at the same time robust to noise, detection displacements and geometric and photometric deformations [7].
3. Finally the descriptor vectors are matched between different images. The matching is based on a distance between the vectors. The dimension of the descriptor have a direct impact on the time, and less dimension are desirable for fast interest point matching [7].

Keypoint detection using Harris Corner detection method identifies the points of interest in the following way.

- Compute the ***Harris matrix H*[8]** for (the window around) that point, defined as

$$H = \sum_p w_p \nabla I_p (\nabla I_p)^T = \sum_p w_p \begin{pmatrix} I_{x_p}^2 & I_{x_p} I_{y_p} \\ I_{x_p} I_{y_p} & I_{y_p}^2 \end{pmatrix} =$$
$$= \sum_p \begin{pmatrix} w_p I_{x_p}^2 & w_p I_{x_p} I_{y_p} \\ w_p I_{x_p} I_{y_p} & w_p I_{y_p}^2 \end{pmatrix} = \begin{pmatrix} \sum_p w_p I_{x_p}^2 & \sum_p w_p I_{x_p} I_{y_p} \\ \sum_p w_p I_{x_p} I_{y_p} & \sum_p w_p I_{y_p}^2 \end{pmatrix} \quad (4)$$

Where the summation is over all pixels p in the window. I_{x_p} Is the x derivative of the image at point p, the notation is similar for the y derivative. The weights w_p are chosen to be circularly symmetric, a 9×9 ***Gaussian kernel*** with 0.5 sigma is chosen to achieve this. ***H*** is a *2×2 matrix.* To find *interest points*, the ***corner strength function*** [6]

$$c(H) = \det(H) - 0.1.\left(trace(H)\right)^2 \quad (5)$$

Once c is computed for every point in the image, choose points where c is above a ***threshold***. Make c to be a ***local maximum*** in a 9×9 *neighbourhood (with* ***non-maximum suppression****)*. In addition to compute the ***feature locations***, it calculates a *canonical orientation* for each feature, and then store this orientation (in degrees) in each feature element. To compute the canonical orientation at each pixel, the gradient of the blurred image is found and the angle of the gradient is used as orientation.

Algorithm 6:1 Algorithm for extracting Key points using Harris Corner

Step 1: Compute x and y derivatives of image

$I_x = G_\sigma^x * \mathrm{I} \quad I_y = G_\sigma^y * \mathrm{I}$

Step 2: Compute products of derivatives of image

$I_{x2} = I_x * I_x I_{y2} = I_y * I_y I_{xy} = I_x * I_y$

Step 3: Compute the sums of the products of derivatives at each pixel

$S_{x2} = G_{\sigma'} * I_{x2} S_{y2} = G_{\sigma'} * I_{y2} S_{xy} = G_{\sigma'} * I_{xy}$

Step 4: Define at each pixel (x, y) the matrix

$$\mathrm{H}(x, y) = \begin{matrix} S_{x2}(x,y) & S_{xy}(x,y) \\ S_{xy}(x,y) & S_{y2}(x,y) \end{matrix}$$

Step 5: Compute the response of the detector at each pixel

$R = Det(H) - k(Trace(H))^2$

Step 6: Threshold on value of R. Compute nonmax suppression[10].

After identifying the keypoints of the trained image and query image, the number of keypoints generated are compared. If the difference of the number of keypoints in the train image and query image is greater than 700 then automatically both leaves are treated as different species. If the difference of keypoints identified are less than 700 keypoints then the algorithm proceed by extracting the features from trained image and query image using SURF feature extraction method and proceeds to matching. Fig. 5 shows the example of keypoint extraction using Harris Corner Detection using Flavia Leaf Database.

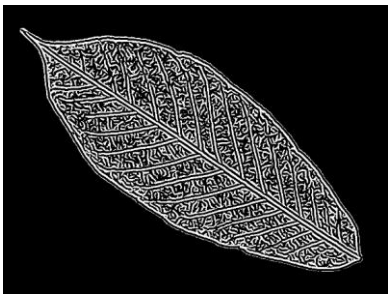
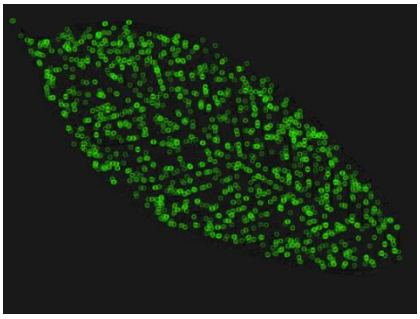
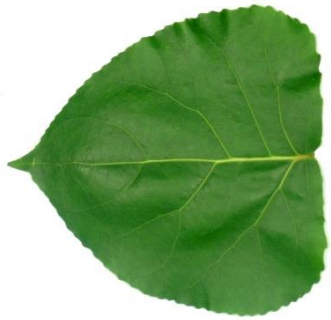
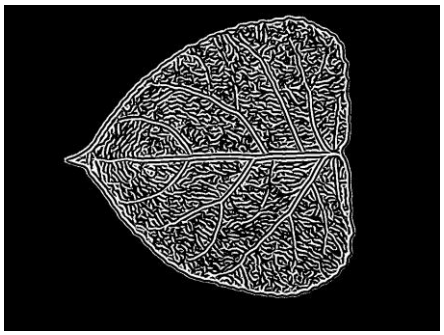
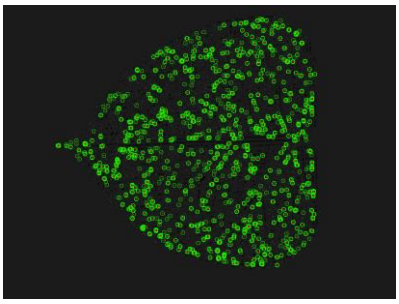
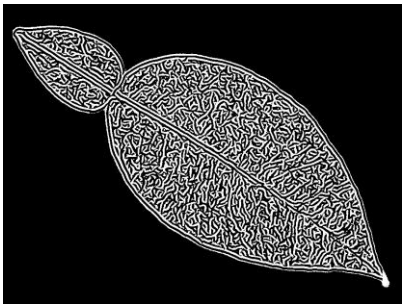
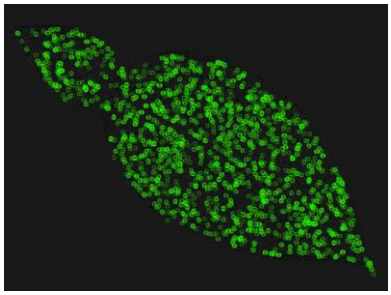
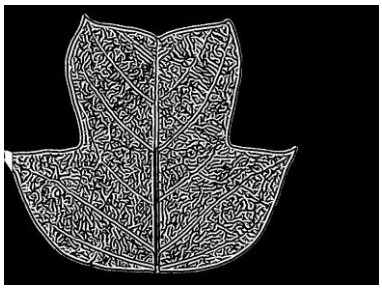
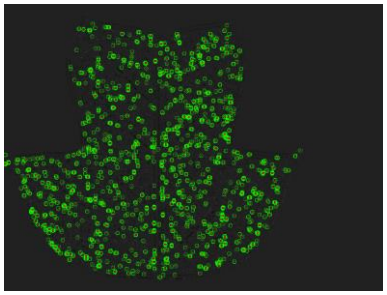
IV. FEATURE EXTRACTION USING SURF [SPEED UP ROBUST FEATURE]

After identifying the ***keypoints***, the next step is to come up with a ***descriptor*** for the ***feature*** *centered* at each ***keypoint***. This descriptor is the representation used to compare the ***features*** in **different images** to see if they ***match or not***. The Speed-Up Robust Features (SURF) detector-descriptor scheme developed by Bayetal [7] is designed as an efficient alternative to SIFT. It is much faster, and more robust as opposed to SIFT. Its basic idea is to approximate the second order Gaussian derivatives in an efficient way with the help of integral images using as setoff box filters. The idea of SURF is based on sums of 2D Haar wavelet responses and makes an efficient use of integral images. For features, it uses the sum of the Haar wavelet response around the point of interest with the aid of integral image.

Retrieval Number: K20160981119/19©BEIESP
DOI: 10.35940/ijitee.K2016.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

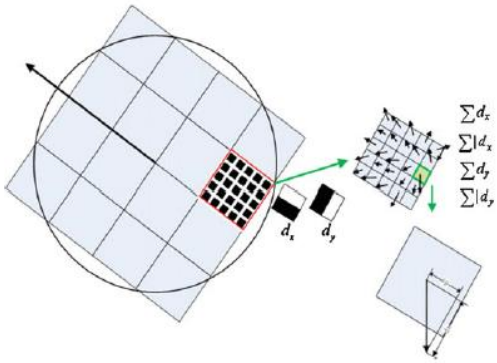
Fig. 4: Example of keypoint extraction using Harris Corner Detection using Flavia Leaf Database

Input Image	Enhanced Vein Image	Keypoints Extracted using Harris Corner
		
		
		
		

The SURF [7, 8] descriptor starts by constructing a square region centered around the detected interest point, and oriented along its main orientation. The size of this window is 20s, where s is the scale at which the interest point is detected. Then, the interest region is further divided into smaller 4×4 sub-regions and for each sub region the Harr wavelet responses in the vertical and horizontal directions (denoted dx and dy, respectively) are computed at a 5×5 sampled points as shown in Fig. These responses are weighted with a Gaussian window centered at the interest point to increase the robustness against geometric deformations and localization errors. The wavelet responses dx and dy are summed up for each sub-region and entered in a feature vector v, where

$$v = \left(\sum d_x, \sum |d_x|, \sum d_y, \sum |d_y|\right) \quad (6)$$

Fig. 5. Dividing the interest region into 4×4 sub-regions for computing the SURF descriptor



Computing this for all the 4×4 sub-regions, resulting a feature descriptor of length 4×4×4=64 dimensions. Finally, the feature descriptor is normalized to a unit vector in order to reduce illumination effects. The main advantage of the SURF [7] descriptor compared to SIFT is the processing speed as it uses 64 dimensional feature vector to describe the local feature, while SIFT uses 128.

V. FEATURE MATCHING WITH FLANN (FAST LIBRARY FOR APPROXIMATE NEAREST NEIGHBORS)

After unique keypoints and descriptors are extracted from both leaf images, a matching should be done. The FLANN [9] library contains a collection of algorithms optimized for fast nearest neighbour search in large datasets and for high dimensional features, and it works faster for large datasets. Two main optimized algorithm used are Randomized k-d Tree Algorithm and Priority k-means algorithm.

VI. RESULTS AND CONCLUSION

Flavia dataset is used for identification of leaves. Entire work is implemented using opencv and Qt. At first 4 classes of leaf species, each species contains 3 leaves are tested by extracting veins and keypoints and all the leaves are correctly classified. The number of keypoints and their differences are given in Table1. The matching score of different leaf species are retrieved. From the Table2 it was observed that a score of less than 2.00 or a minimum score predicts the same class of leaf species. The identified four Flavia species result is given in Table3. Then 7 and 10 classes of leaf species, each containing 3 leaves are tested and the algorithm correctly identifies the leaf species. The keypoints are identified for query image and test image and if the difference of keypoints above 700 assumes as different category of species. In that case no need to proceed with feature extraction and matching and automatically treated as different category of leaves. Otherwise the features are extracted and matching scores are calculated.

Finally all the 32 Species of Flavia leaf species are tested by taking 10 leaf images of each species. The matching score of 32 different leaf species result a matching score less than 2.0. From the result it is clear that all samples or query images under the same leaf species have a matching score less than 2 for Flavia leaf database. An accuracy of 98.75% resulted using the proposed methods. The accuracy of recognition is improved when compared to other recognition methods.

From the results after classifying 32 different leaf species of Flavia Database, it can be observed that the percentage of identification is 98.75%. This methodology can be used for different Plant Leaf identification especially in the field of medicinal plant leaf identiifcation by taking the leaf image photos and testing those images with a standard leaf image of the same species. The matching certanity score clearly set a limit to predict the leaf species. This work can be further explored by automating the suitable number of the limits of keypoints used for feature extraction and identification.

Retrieval Number: K20160981119/19©BEIESP
DOI: 10.35940/ijitee.K2016.0981119
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Table 1.Keypoints and Differences 4 Species [Flavia Leaf Image Database]

Leaf Image	Keypoints	1050	1051	1053	1062	1064	1065	1123	1125	1128	1198	1199	1200
1050	247	0	60	259	1427	1932	1696	3237	2616	3154	58	10	34
1051	307	60	0	199	1367	1872	1636	3177	2556	3094	118	70	26
1053	506	259	199	0	1168	1673	1437	2978	2357	2895	317	269	225
1062	1674	1427	1367	1168	0	505	269	1810	1189	1727	1485	1437	1393
1064	2179	1932	1872	1673	505	0	236	1305	684	1222	1990	1942	1898
1065	1943	1696	1636	1437	269	236	0	1541	920	1458	1754	1706	1662
1123	3484	3237	3177	2978	1810	1305	1541	0	621	83	3295	3247	3203
1125	2863	2616	2556	2357	1189	684	920	621	0	538	2674	2626	2582
1128	3401	3154	3094	2895	1727	1222	1458	83	538	0	3212	3164	3120
1198	189	58	118	317	1485	1990	1754	3295	2674	3212	0	48	92
1199	237	10	70	269	1437	1942	1706	3247	2626	3164	48	0	44
1200	281	34	26	225	1393	1898	1662	3203	2582	3120	92	44	0

Table 2 Matching score of 4 different Flavia leaf species.

Leaf Image	Keypoints	1050	1051	1053	1062	1064	1065	1123	1125	1128	1198	1199	1200
1050	247	0	0.390	1.502	OC	OC	OC	OC	OC	OC	2.460	6.437	2.997
1051	307	0.390	0	1.761	OC	OC	OC	OC	OC	OC	6.590	12.214	9.584
1053	506	1.502	1.761	0	OC	OC	OC	OC	OC	OC	7.490	5.902	3.819
1062	1674	OC*	OC	OC	0	0.008	0.324	OC	OC	OC	OC	OC	OC
1064	2179	OC	OC	OC	0.008	0	0.418	OC	OC	OC	OC	OC	OC
1065	1943	OC	OC	OC	0.324	0.418	0	OC	OC	OC	OC	OC	OC
1123	3484	OC	OC	OC	OC	OC	OC	0	0.054	0.406	OC	OC	OC
1125	2863	OC	OC	OC	OC	OC	OC	0.054	0	1.070	OC	OC	OC
1128	3401	OC	OC	OC	OC	OC	OC	0.406	1.070	0	OC	OC	OC
1198	189	2.460	6.590	7.490	OC	OC	OC	OC	OC	OC	0	0.440	0.814
1199	237	6.437	12.214	5.902	OC	OC	OC	OC	OC	OC	0.440	0	0.686
1200	281	2.997	9.584	3.819	OC	OC	OC	OC	OC	OC	0.814	0.686	0

OC- Other Species of Leaf category

Table 3: Classification of four different species

Leaf Image Class	Falvia1	Flavia2	Flavia3	Flavia4
Flavia1 pubescent bamboo	3	0	0	0
Flavia2 Chinese horse chestnut	0	3	0	0
Flavia3 Chinese redbud	0	0	3	0
Flavia4 true indigo	0	0	0	3

REFERENCES

1. R.Gonzalez, R, Woods, and S.Eddins, *Digital image processing*. Prentice Hall.
2. R.Gonzalez, R, Woods, and S.Eddins, *Digital image processing using MATLAB*. Prentice Hall.
3. Lin Hong,Yifei Wan,Anil Jain. Fingerprint Image Enhancement: Algorithm and Performance Evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence. Volume 20 Issue 8, August 1998. Page 777-789
4. Schimid C, Mohr R, Bauckhane C. Evaluation of interest point detectors [J]. International Journal of Computer Vision, 2000, 37(2):151-172.
5. H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In European Conference on Computer Vision, May 2006.
6. https://bitbucket.org/coldvisionio/coldvision library/src/7507fa8c875be203cfbcf2edef9340b82d2f80a8/samples/2_imaging/surf_features/
7. Marius Muja, David G. Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration, 2009
8. https://github.com/alexanderb14/Harris-Corner-Detector
9. Chris Harris and Mike Stephens (1988). "A Combined Corner and Edge Detector". Alvey Vision Conference. 15.
10. Konstantinos G. Derpanis (2004). The harris corner detector. York University.

AUTHORS PROFILE

Ms. Juby George : Completed **MCA and MPhil** and qualified UGC-NET. Presently working as Assistant Professor, Department of Computer Applications, Marian college kuttikkanam [Autonomous]. Presently registered as a research scholar at Mahatma Gandhi University Kottayam at School of Computer Science under the guidance of Dr,Gladston Raj S. Having 14 years of teaching experience, published research papers in two conference and journals and won best research paper award organized by IFERP at VOC College Thoothukudi in the year 2018. Serving as member board of studies at Fatima Matha College Kollam and question paper setter at Kerala University. Served as organizing committee member for international conferences.

Dr.Gladston Raj S: He took **MSc**, **MTech** and **PhD** in **Computer Science** from University of Kerala and having qualified with **UGC-NET** and **PGDCH**. Presently serving as Associate Professor & Head, Department of Computer Science @ Government College Nedumangad. Having guideship at MG University Kottayam, Bharathiar University Coimbatore and University of Kerala. He produced seven PhDs, submitted two more thesis and guiding research scholars of Mahatma Gandhi University, Kottayam. With 20 years of teaching and research, he published more than 45 research papers in reputed journals. He is an Academic Council member and Chairman, Board of Studies (Computer Science) of University of Kerala and also chairman and member of various Universities/autonomous colleges in Kerala. He served as resource person for more than 100 National/International conferences/workshops sessions. He is the subject expert at SCERT, Trivandrum and Technical advisor to File Flow Management System (FFMS) of DCE, Govt. of Kerala.