LEAF DISEASE DETECTION USING IMAGE SEGMENATION

S.V.Anandhi, M.E.(PhD)
AP/CSE, Dr. Sivanthi Aditanar College of Engineering, Tiruchendur

Abstract

The K-Means clustering technique is a well-known approach that has been applied to solve low-level image segmentation tasks. This clustering algorithm is convergent and its aim is to optimize the partitioning decisions based on a user-defined initial set of clusters that is updated after each iteration. In the first step we identify the mostly green colored pixels. Next, these pixels are masked based on specific threshold values that are computed using Otsu's method, then those mostly green pixels are masked. The other additional step is that the pixels with zeros red, green and blue values and the pixels on the boundaries of the infected cluster (object) were completely removed. The experimental results demonstrate that the proposed technique is a robust technique for the detection of plant leaves diseases

Index Terms: Gaussian filter, Susan algorithm, K-Means Clustering, OTSU threshold

I. INTRODUCTION

India is an agricultural country; wherein about 70% of the population depends on agriculture. Farmers have wide range of diversity to select suitable Fruit and Vegetable crops. However, the cultivation of these crops for optimum yield and quality produce is highly technical. It can be improved by the aid of technological support. The management of perennial fruit crops requires close monitoring especially for the management of diseases that can affect production significantly and subsequently the post-harvest life. In case of plant the disease[3] is defined as any impairment of normal physiological function of plants, producing characteristic symptoms. A symptom is a phenomenon accompanying something and is regarded as evidence of its existence. Disease is caused by pathogen which is any agent causing disease. In most of the cases pests or diseases are seen on the leaves or stems of the plant. Therefore identification of plants, leaves, stems and finding out the pest or diseases, percentage of the pest or disease incidence, symptoms of the pest or disease attack, plays a key role in successful cultivation of crops. In biological science, sometimes thousands of images are generated in a single experiment. These images can be required for further studies like classifying lesion, scoring quantitative traits, calculating area eaten by insects, etc. Almost all of these tasks are processed manually or with distinct software packages. It is not only tremendous amount of work but also suffers from two major issues: excessive processing time and subjectiveness rising from different individuals.

1.1 Existing method

The Existing approach the initial step, the RGB images of all the leaf samples were picked up. That leaves belonging to early scorch, cottony mold, ashen mold and late scorch have significant differences form greasy spot leaves in terms of color and texture. The experimental results demonstrate that the proposed technique is a robust technique for the detection of plant leaves diseases.

1.2 Proposed method

In this first use an effective method to preprocess the image using Gaussian filter to smooth for entire leaf. Then edge has been detected using Susan algorithm to detect the outline of the leaf and remove for remaining area. Then segmentation using K-Means Clustering algorithm[1]. This is a very effective method to provide the correct results. Then green channel masking to be applied and differentiate in the color to convert for RGB into HSI this is very clear view for using leaf affected area extracting. Then extract only affected area using for OTSU threshold method. Then apply background subtraction method used to affected level displayed.

Algorithm steps are,

1.Preprocessing is done by using Gaussian filter method.

2.Edge detection is done by using Susan algorithm.

3.Then segmentation using K-Means Clustering algorithm

4. Then green channel masking to be applied and differentiate in the color to convert for RGB into HSI.

5. Then extract only affected area using for OTSU threshold method.

6. Then apply background subtraction method used to affected level displayed

2. FLOW CHART

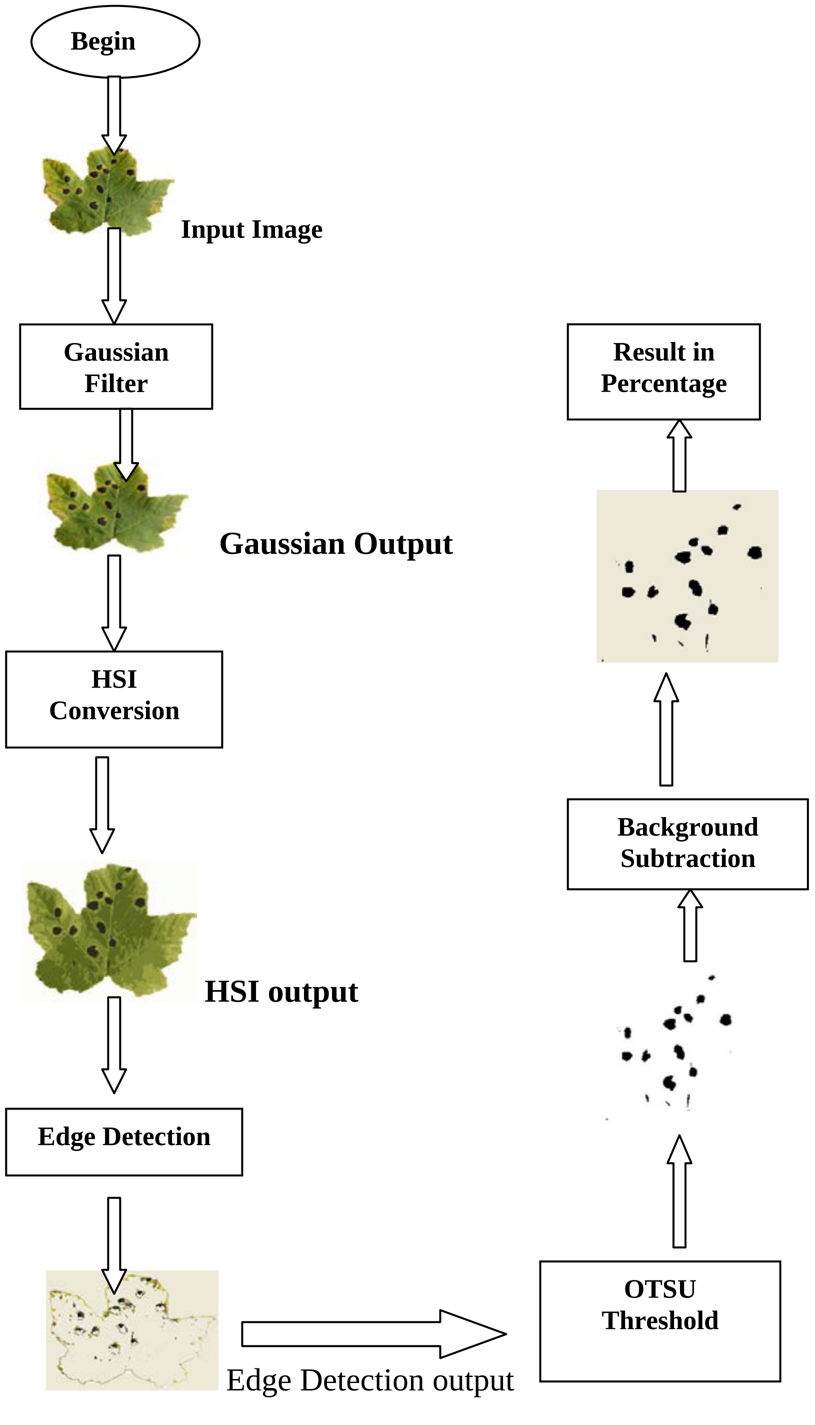


Fig: 2.1 Flow chat

2.1. Preprocessing

Preprocessing use the median filter algorithm. This algorithm is performed to remove the random noise and that can suppress isolated noise without blurring sharp edges.

2.1.1 GAUSSIAN FILTER

The Gaussian Filter block filters the input signal using a Gaussian FIR filter. The block expects the input signal to be up sampled as its input, so that the Input samples per symbol parameter, N, is at least 2. The block's icon shows the filter's impulse response."

Characteristics of the Filter

The impulse response of the Gaussian filter is where and B is the filter's 3-dB bandwidth. The **BT product** parameter is B times the input signal's symbol period. For a given BT product, the Signal Processing Toolbox Gaussian function generates a filter that is half the bandwidth of the filter generated by the Communications Block set Gaussian Filter block.

The **Group delay** parameter is the number of symbol periods between the start of the filter's response and the peak of the filter's response. The group delay and N determine the length of the filter's impulse response, which is 2 * N * **Group delay** + 1.

The **Filter coefficient normalization** parameter indicates how the block scales the set of filter coefficients.

- Sum of coefficients means that the sum of the coefficients equals 1.
- Filter energy means that the sum of the squares of the coefficients equals
- Peak amplitude means that the maximum coefficient equals

After the block normalizes the set of filter coefficients as above, it multiplies all coefficients by the linear amplitude filter gain parameter. As a result, the output is scaled by . If the output of this block feeds the input to the AWGN Channel block, specify the AWGN signal power parameter to be 1/N.

2.1.2 Input and Output Signals

This block accepts scalar, column vector, and *M*-by-*N* matrix input signals. The block filters an *M*-by-*N* input matrix as follows:

- When you set the **Input processing** parameter to Columns as channels (frame based), the block treats each column as a separate channel. In this mode, the block creates *N* instances of the same filter, each with its own independent state buffer. Each of the *N* filters process *M* input samples at every Simulink time step.
- When you set the **Input processing** parameter to Elements as channels (sample based), the block treats each element as a separate channel. In this mode, the block creates *M***N* instances of the same filter, each with its own independent state buffer. Each filter processes one input sample at every Simulink time step.

The output dimensions always equal those of the input signal. For information about the data types each block port supports, see the table on this page.

2.2 Edge Detection

2.2.1Susan algorithm

The edge detection algorithm described here follows the usual method of taking an image and, using a predetermined window centered on each pixel in the image, applying a locally acting set of rules to give an edge response. This response is then processed to give as the output a set of edges.

The SUSAN edge finder has been implemented using circular masks (sometimes known as windows or kernels) to give isotropic responses. Digital approximations to circles have been used, either with constant weighting within them or with Gaussian weighting -- this is discussed further later. The usual radius is 3.4 pixels (giving a mask of 37 pixels), and the smallest mask considered is the traditional three by three mask. The 37 pixel circular mask is used in all feature detection experiments unless otherwise stated.

The mask is placed at each point in the image and, for each point, the brightness of each pixel within the mask is compared with that of the nucleus (the centre point). Originally a simple equation determined this comparison -- see Figure 3.4.2.1 Susan Edge detection;

Fig: 3.4.2.1 Susan Edge detection

2.3 Segmentation

Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image[4]. Each of the pixels in a region is similar with respect to some characteristic or computed property, such as colour, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristics. When applied to a stack of images, typical in medical imaging, the resulting contours after image segmentation can be used to create 3D reconstructions with the help of interpolation algorithms like marching cubes.

2.3.1. K-Means Clustering algorithm

K-means clustering algorithm[5] is simply described as follows: Input: N objects to be cluster (x1, x2, xn), the number of clusters k; Output: k clusters and the sum of dissimilarity between each object and its nearest cluster centre is the smallest: …. (Reference [1])

1. Arbitrarily select k objects as initial cluster centers (m1, m2 ... mk);
2. Calculate the distance between each object Xi and each cluster centre, and then assign each object to the nearest cluster formula for calculating distance as:

$$d(x_i, m_j) = \sqrt{\sum_{j=1}^{d} (x_{i1} - m_{j1})^2} \text{, i=1… N; j=1… k;}$$

3. Calculate the mean of objects in each cluster as the new cluster centres, N,i is the number of samples of current cluster i;

$$m_i = \frac{1}{N_i} \sum_{j=1}^{N_i} x_{ij} \text{, i=1, 2… k;}$$

4. Repeat 2 & 3 until the criterion function E converged, return(m1),m2 . . . mk).

2.3.2Advantages of K-Means Clustering

1. This algorithm is relatively scalable and efficient in processing large data sets because the computational complexity of the algorithm is O (nkt), where n is the number of objects, k is the number of clusters, and t is the number of iterations.

2. It works well when the clusters are compact clouds that are rather well separated from one another.

3. The algorithm is not only simple, but also the results are easily understandable and it can be easily modeled to deal with streaming data. The K-Means algorithm is simple using the data mining tool KNIME especially there are no missing values in the data set.

4. Continual improvements and generalizations of the algorithm have ensured its continued relevance and gradually increased its effectiveness as well.

2.4 Green pixel Masking

A Bayer filter mosaic is a color filter array (CFA) for arranging RGB color filters on a square grid of photo sensors. Its particular arrangement of color filters is used in most single-chip digital image sensors used in digital cameras, camcorders, and scanners to create a color image. The filter pattern is 50% green, 25% red and 25% blue, hence is also called RGBG, GRGB, or RGB.

It is named after its inventor, Bryce Bayer of Eastman Kodak. Bayer is also known for his recursively defined matrix used in ordered dithering.

Alternatives to the Bayer filter include both various modifications of colors and arrangement and completely different technologies, such as color co-site sampling, the Foveon X3 sensor or dichroic mirrors.

2.5. Extraction

2.5.1. OTSU threshold

In computer vision and image processing, Otsu's method[2] is used to automatically perform clustering-based image threshold or the reduction of a gray level image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels or bi-modal histogram (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal. The extension of the original method to multi-level thresholding is referred to as the Multi Otsu's method. Otsu's method is named after Nobuyuki Otsu. ….(Reference [2])

2.5.2Method

In Otsu's method we exhaustively search for the threshold that minimizes the intra-class variance (the variance within the class), defined as a weighted sum of variances of the two classes:

Weights are the probabilities of the two classes separated by a threshold and variances of these classes.

Otsu shows that minimizing the intra-class variance is the same as maximizing inter-class variance: which is expressed in terms of class probabilities and class means.

The class probability is computed from the histogram as :

While the class mean is the value at the center of the histogram bin. Similarly, you can compute and on the right-hand side of the histogram for bins greater than.

The class probabilities and class means can be computed iteratively. This idea yields an effective algorithm.

2.6 Background Subtraction

Background subtraction, also known as Foreground Detection, is a technique in the fields of image processing and vision wherein an image's foreground is extracted for further processing (object recognition etc.). Generally an image's regions of interest are objects (humans, cars, text etc.) in its foreground. After the stage of image preprocessing (which may include image de noising etc.) object localization is required which may make use of this technique. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras. The rationale in the approach is that of detecting the moving objects from the difference between the current frame and a reference frame, often called "background image", or "background model". Background subtraction is mostly done if the image in question is a part of a video stream.

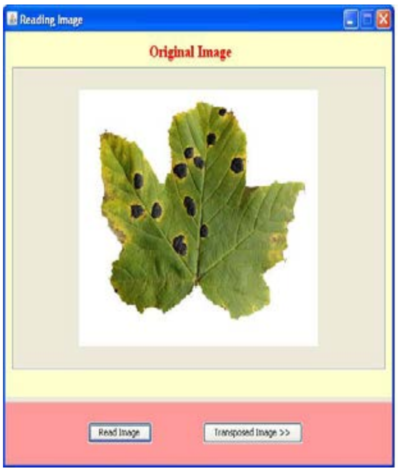
3 EXPERIMENTAL RESULTS

Preparation of the test plays vital role in system testing. After preparing the test data, system under study is tested using the test data. While testing the system using the test data, errors are corrected by using the above-mentioned steps and correction are also for future use.

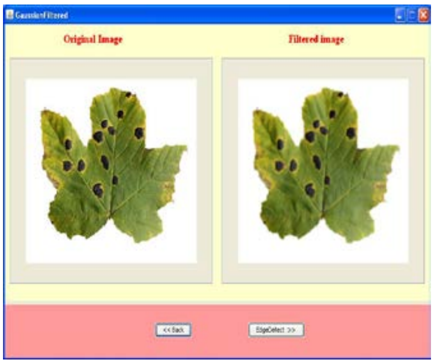
Testing begins at the module level and works towards the integration of the entire computer based system. Testing and debugging are different activities, but any testing includes debugging strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions, against customer requirement. No testing without the verification and validation part.

The goals of verification and validation activities are to access and improve the quality of the work products generated during the development and modification of the software. There are two types of verification life cycle verification and a formal verification. Life cycle verification is the process of determining the degree to which the products of the given phase of the development cycle fulfill the specification established during prior process. Formal verification is a rigorous mathematical demonstration that source code confirms to its specification. Validation is the process of evaluating software at the end of the software development process to determine compilation with the requirements.

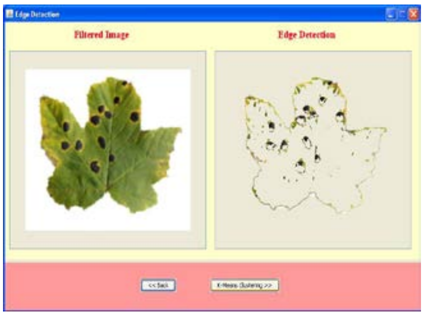
The system has been thoroughly checked. It was tested module by module. First the java file is compiled and then the result is displayed



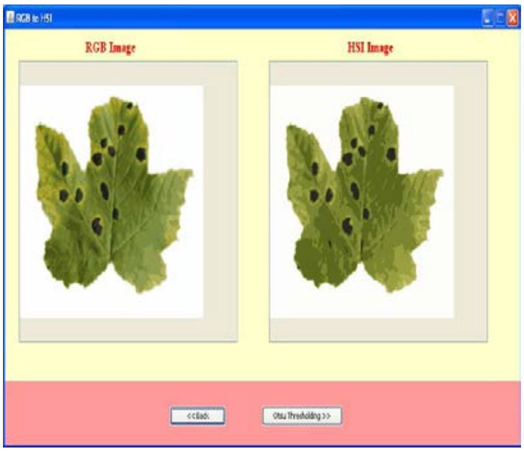
Load Image



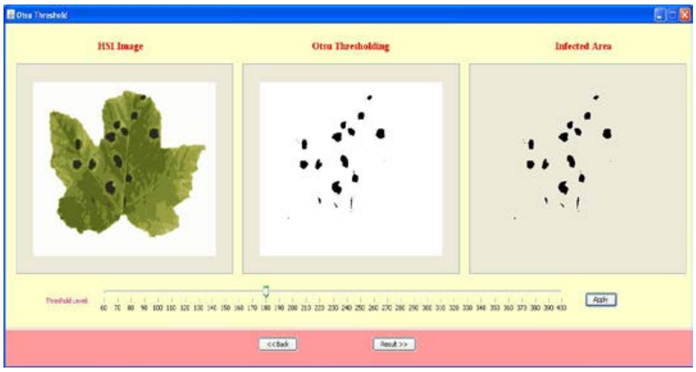
Filtered Image



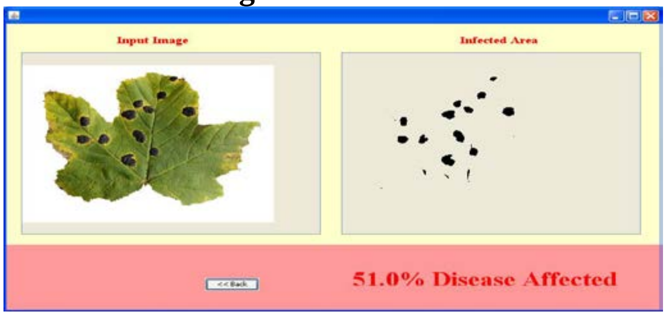
Edge Detection



HSI Conversion



OTSU Thresholding



Result

5 CONCLUSION

A general k-means-based clustering algorithm that can identify natural clusters in datasets, whether they are embedded in the original space or subspaces. Like traditional k-means clustering algorithm, the time complexity of the algorithm is linear with the number of the data points, the dimensionality of the data, and the number of clusters in the dataset. The experiment results show that our algorithm is an efficient algorithm with high clustering accuracy. Clustering analysis method is one of the main analytical methods in data mining; the method of clustering algorithm will influence the clustering results directly.

6 FUTURE SCOPE

Standard versions of k-means algorithms seem be better in finding high fitness solutions. In the same time results obtained in standard and genetic versions of k-means algorithms relative to validity indices are also comparable. During extensive search of solution space, genetic versions of k-means algorithms most often find solutions with slightly worse fitness values but at the same exceptionally good values of individual validity indices. Further investigation into this matter could present starting point into improvement of k-means based image clustering techniques.

REFERENCES

[1] Bashish, D., M. Braik, and S. Bani-Ahmad. 2011. Detection and classification of leaf diseases using K-means-based segmentation. *Information Technology Journal*, 10(2): 267-275.

[2] Otsu, N. (1979). "A threshold selection method from gray-level histograms". IEEE Trans. Sys., Man., Cyber. 9: 62–66. DOI:10.1109.

[3] Weizheng S.,Yachun W.,Zhanliang C.& Hongda W.[2008]Grading Method Of Leaf Spot Disease Based On Image Processing, Proceedings Of 2008 International Conference On Computer Science And Software Engineering, Volume 06.

[4]. Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture feature,s S.Arivazhagan, R. Newlin Shebiah* , S. Ananthi, S. Vishnu Varthini, Vol. 15, No.1.

[5]. Leaf Image Segmentation Based On the Combination of Wavelet Transform and K Means Clustering, N.Valliamma, Dr.S.N.Geethalakshmi, Vol. 1, No. 3, 2012 .