DETECTION OF MARSH FEVER IN BLOOD IMAGES USING NEURAL NETWORK

Ms. Deepa Saibannavar[1], Mrs. Sahana S.Bisalapur[2]

[1]BE (CSE), Lecturer, S. G. Balekundri Institute of Technology, Belgaum, Karnataka, (India)

[2]M.Tech (CSE), Lecturer, S. G. Balekundri Institute of Technology, Belgaum, Karnataka, (India)

ABSTRACT

Malaria is a serious global health problem, and rapid, accurate diagnosis is required to control the disease. An image processing algorithm to automate the diagnosis of malaria in blood images is developed in this paper. The image classification system is designed to positively identify malaria parasites present in thin blood smears, and differentiate the species of malaria. The implemented new approach to low-level image processing -SUSAN (Smallest Univalue Segment assimilating Nucleus) Principle, performs Edge and Corner Detection. Images are acquired using a charge-coupled device camera connected to a light microscope. Morphological and novel threshold selection techniques are used to identify erythrocytes (red blood cells) and possible parasites present on microscopic slides. Image features based on colour, texture and the geometry of the cells and parasites are generated, as well as features that make use of a priori knowledge of the classification problem and mimic features used by human technicians. The first order features provides the basic mathematical ranges for different types of parasites. A two-stage tree classifier distinguishes between true and false positives, and then diagnoses the species of the infection. Malaria samples obtained from the various biomedical research facilities are used for training and testing of the system.

***Keywords*: SUSAN, Malaria Parasite, Edge Detection, Image Analysis.**

1. INTRODUCTION

Malaria [1] is a serious global health disease caused by a blood parasite named plasmodium spp. Malaria is caused by Plasmodium parasites. The parasites are spread to people through the bites of infected Anopheles mosquitoes, called "malaria vectors", which bite mainly between dusk and dawn.

There are four parasite species [2] that cause malaria in humans:

- Plasmodium falciparum
- Plasmodium vivax
- Plasmodium malariae
- Plasmodium ovale

1.1 Background
1.2

A survey has been done for the different types of methods available for each and every stage of proposed design. Following are the stages with survey information:

1.1.1 Image Analysis

There are many methods available for Image Analysis for edge and corner detection. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. Comparision of 3 methods of Image Analysis namely: Canny Edge Detection, Median Filtering, and SUSAN [3] [4] Algorithm has been done using SUSAN algorithm

1.1.2 Canny Edge Detection

The Canny edge detection [5] operator was developed by John F. Canny in 1986 and uses a multi-stage algorithm to detect a wide range of edges in images. Most importantly, Canny also produced a computational theory of edge detection explaining why the technique works.

Advantages of Canny Edge Detection

- Using probability for finding error rate.
- Localization and response.
- Improving signal to noise ratio.
- Better detection especially in noise conditions.
- Results are stable.

Disadvantages of Canny Edge Detection

- Complex Computations.
- False zero crossing.
- Time consuming – This algorithm is slower than SUSAN approach.
- The edge connectivity at junction is poor and corners are rounded.

1.1.3 Median Filtering

It is the most popular and simple nonlinear low pass filter. It reduces noise without blurring. In microscopic image processing, it is usually necessary to perform high degree of noise reduction in an image before performing higher-level processing steps, such as edge detection. The median filter is a non-linear digital filtering technique, often used to remove noise from images or other signals. The idea is to examine a sample of the input and decide if it is representative of the signal. This is performed using a window consisting of an odd number of samples. The values in the window are sorted into numerical order; the median value, the sample in the center of the window, is selected as the output. The oldest sample is discarded, a new sample acquired and the calculation repeats.

Algorithm

The median filter considers each pixel in the image in turn and looks at its nearby neighbors to decide whether or not it is representative of its surroundings. Instead of simply replacing the pixel value with the mean of neighboring pixel values, it replaces it with the median of those values. The median is calculated by first sorting all the pixel values from the surrounding neighborhood into numerical order and then replacing the pixel being considered with the middle pixel value. (If the neighborhood under consideration contains an even number of pixels, the average of the two middle pixel values is used).

Advantages of Median Filtering

- It's simple to understand.
- The median filter preserves brightness differences resulting in minimal blurring of regional boundaries.
- Preserves the positions of boundaries in an image, making this method useful for visual examination and measurement.
- Median computer algorithm can be customized.

Disadvantages of Median Filtering

- Less effective in removing Gaussian or random-intensity noise "The median filter can remove noise only if the noisy pixels occupy less than one half of the neighborhood area."
- Repeating will remove noise but at the expense of detail (posterization occurs) where pixel brightness values are leveled across regions "group of pixels having similar brightness values ".
- High computational cost (for sorting N pixels, the temporal complexity is O(N·log N), When the median filter must be carried out in real time, the software implementation in general-purpose processors does not usually give good results and FPGAs are a good alternative (Field Programmable Gate Array -hardware)
- Some median algorithms are not good for real time processing.

1.1.4 Susan Approach (A Non-linear Filtering)

For a real time system using time varying image sequences, speed is an important criterion to be considered. Also there has to be a compromise between maximizing signal extraction and minimizing output noise: the so called "Uncertainty Principle" of edge detection. In this paper a new approach to low-level image processing - SUSAN (Smallest Univalue Segment assimilating Nucleus) Principle is implemented, which performs Edge and Corner Detection and Structure Preserving Noise Reduction.

Advantages of SUSAN Approach:

- Detection of image edges and lines is done accurately and quickly.
- The localization of the features is independent of the mask size used and hence the noise suppression is good.
- Connectivity of edges and lines at junctions is good.
- The SUSAN principle can be viewed as an efficient way of finding features using local information from a pseudo-global view point.

II METHODOLOGY

The objective of the paper is:

- To develop a fully automated image classification system to positively identify malaria parasites present in thin blood smears, and differentiate the species.
- The algorithm generated will be helpful in the area where the expert in microscopic analysis may not be available. The effort of the algorithm is to detect presence of parasite at any stage. One of the parasites grows in body for 7 to 8 days without any symptoms. So if this algorithm is incorporated in routine tests, the presence of malaria parasite can be detected.

The design follows the same steps as that of a pattern recognition problem. But the best part of the algorithm is the usage of the most appropriate algorithm for each stage. The test algorithms illustrated above give an insight about the algorithm to be used for each stage. The process given below is shown in Fig 1.

1. Image Acquisition and database collection
2. Image Analysis
3. Image Segmentation
4. Feature Generation
5. Classification of Parasite and result verification

2.1 Image Acquisition and Database Collection

Oil immersion views (10x1000), of Giemsa stained blood films [6] were captured using a binocular microscope mounted with a digital camera. Captured images were 430 pixels X 360 pixels jpeg images.

2.1.1 Image analysis

For a real time system using time varying image sequences [7], speed is an important criterion to be considered. Also there has to be a compromise between maximizing signal extraction and minimizing output noise: the so-called "Uncertainty Principle" of edge detection. I have implemented a new approach to low-level image processing - SUSAN (Smallest Univalue Segment assimilating Nucleus) Principle, which performs Edge and Corner Detection and has become one of the most widely used edge finding algorithms, is found to be ten times slower than this SUSAN approach. The results are stable for Canny but the edge connectivity at junction is poor Structure Preserving Noise Reduction. Canny edge detector, which and corners are rounded.

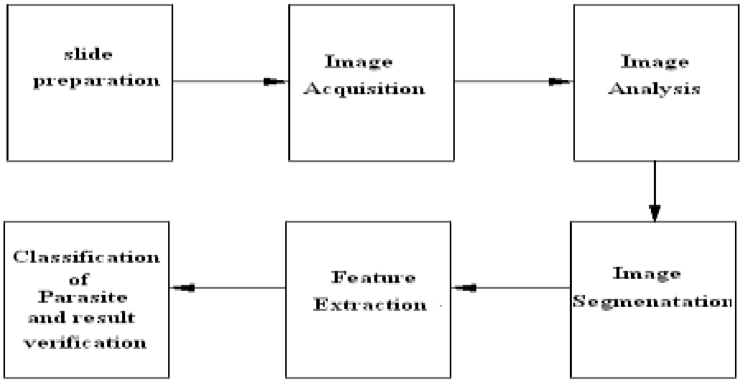


Fig. 1 Block Schematic of the Parasite Detection System

The fact that SUSAN edge and corner enhancement uses no image derivative, explains why the performance in the presence of noise is good. The integrating effect of the principal together with its non-linear response gives strong noise rejection.

An RGB image is selected and a predefined 2D filter is applied for contrast enhancement and a N-D filtering for multidimensional images is applied.

An RGB filtered image is firstly converted into a gray scale image. (1) is the formula used to convert RGB value of a pixel into its gray scale value.

$$:Gray = 0.2989*R+0.5870*G+0.1140*B \qquad (1)$$

Where R, G, B correspond to the colors of the pixel, respectively. This gray scale image is given as input to SUSAN process for Edge and Corner Detection.

2.1.2 Image segmentation

Techniques have been proposed earlier that make use of thresholding or morphology to segment an image. In this section the presented technique takes benefit of morphological operation [9] at appropriate positions in the whole process to maximize the productivity of the algorithm. In order to use morphological methods for image segmentation, the shape and size of the objects in the image must be known. The size and eccentricity of the erythrocytes are also required for the calculation of some feature values (as these can be indicative of infection). The shape of the objects (circular erythrocytes) is known a priori, but the image must be analyzed to determine the size distribution of objects in the image and to find the average eccentricity of erythrocytes present.

The next stage of the process identifies and segments potential parasites and erythrocytes from the image background. To extract the infected erythrocytes, it is first necessary to identify them from the combination of parasites and erythrocytes in the image, and then segment them from the background.

A SUSAN binary image is applied with dilation for bridging gaps in an image and erosion is applied to eliminate unwanted detail. This dilated image is input for bw-labeling. Bw-label is Label connected components in binary image. The erythrocytes that have been identified as possibly infected are then extracted from the image and passed to the next stage of the algorithm for feature generation. The binary mask of the erythrocyte, as well as a binary mask of parasite -like objects present in the cell, is also passed to the next stage.

2.1.3 Feature Generation and Classification

Feature Generation: Two sets of features are used for development. The first set will be based on image characteristics that have been used previously in biological cell classifiers, which include geometric features (shape and size), colour attributes and grey-level textures.

It will be advantageous to apply expert, a priori knowledge to a classification problem. This will be done with the second set of features, where measures of parasite and infected erythrocyte morphology that are commonly used by technicians for manual microscopic diagnosis are used. It’s desirable to focus on these features, because it is already known that they are able to differentiate between species of malaria.

Feature Classification: The final classification of an erythrocyte as infected with malaria or not, and if so, the species of the parasite, falls to the classifier. The classifier is a two-stage tree classifier, with an infection classified as positive or negative at the first node, and the species assigned at the second node.

The design of a tree classifier has the following steps: the design of a tree structure (which has already been assigned), the selection of features to be used at every node, and the choice of decision rule at each node. The same type of classifier is used at both nodes.

The features selected for the first classifier are those that describe the colour and texture of the possible parasites. The features used by microscopists to differentiate malaria species are selected for the second classifier. The training goal is to minimize squared errors, and training is stopped when the error of a validation set increased. This is done to avoid overtraining.

2.1.4 Results

The performance and accuracy of the algorithm are analyzed using 4 measures:

- **Sensitivity**, the ability of the algorithm to detect a parasite present;
- **Specificity**; **accuracy** and **positive predictive value** (PPV), the success of the algorithm at excluding non-infected cells. These values are expressed in terms of true positives (TP), false positives (FP) and false negatives (FN): The algorithm has been tested on various malaria malarial parasites

III IMPLEMENTATION

This chapter will discuss the implementation processes of this research. There are six stages in the implementation process viz., Slide Preparation, Image Acquisition, Image Analysis, Image segmentation, Feature Extraction, and Classification of parasite and Result Verification.

3.1 Image Processing

Image processing involves changing the nature of an image in order to either

- Improve its pictorial information for human interpretation, or
- Render it more suitable for autonomous machine

perception. It may include,

- Enhancing the edges of an image to make it appear sharper.
- Removing noise from an image; noise being random errors in the image.
- Removing motion blur from an image.

3.1.1 Aspects of image processing

It is convenient to subdivide different image processing algorithms into broad subclasses. There are different algorithms for different tasks and problems, and often would like to distinguish the nature of the task at hand.

Image Enhancement: This refers to processing an image so that the result is more suitable for a particular application. Examples include:

- Sharpening or de-blurring an out of focus image,
- Highlighting edges,
- Improving image contrast, or brightening an image,
- Removing noise.

Image Restoration: This may be considered as reversing the damage done to an image by a known cause. Examples include:

- Removing of blur caused by linear motion,
- Removal of optical distortions,
- Removing periodic interference.

Image Segmentation: This involves subdividing an image into constituent parts, or isolating certain aspects of an image.

- Finding lines, circles or particular shapes in an image.

3.2 Types of the images

3.2.1 Binary Image

Each pixel is just black or white. Since there are only two possible values for each pixel, only need is one bit per pixel. Such images can therefore be very efficient in terms of storage. Images for which a binary representation may be suitable include text, fingerprints or architectural plans.

3.2.2 Grayscale Image

Each pixel is a shade of grey, normally from 0(black) to 255(white).this range means that each pixel can be represented by eight bits or exactly one byte. This is a very natural range for image file handling.

True color or RGB.

Here each pixel has a particular color; that color being described by the amount of red, green and blue in it. If each of these components has a range 0-255 this gives a total of 255^3=16,77,216 different possible colors in the image. Since the total number of bits required for each pixel is 24, such images are also called 24-bit color images. Such an image may be considered as consisting of a stack of three matrices representing the red, green and blue values for each pixel.

3.3 Slide Preparation

Anonymized thin blood film images were obtained from the laboratories. The samples obtained mostly had low number of parasites in early stages (rings) of their life cycle. Such features are often hard to detect. In addition several samples did not have any parasites (negative controls). The samples were stained using a fast Giemsa protocol to highlight the diagnosis. Slide images were acquired using a charge coupled device (CCD) camera with different range of magnification. Some images had variable stain characteristics making computer-based detection more challenging as shown in fig 2.

3.4 Image Acquisition

Oil immersion views (10x1000), of Giemsa stained blood films were captured using a binocular microscope mounted with a digital camera. Captured images were 430 pixels X 360 pixels jepeg images. The database consists of 300 images as shown in fig 3.

Fig. 2 Slide Preparation

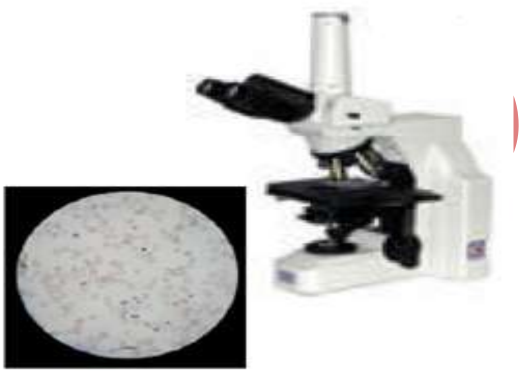


Fig. 3 Image Acquisition

3.5 Image Analysis

Image analysis usually starts with a pre-processing stage, which includes operations such as noise reduction. The purpose of pre-processing is to remove unwanted objects and noise from the image to facilitate image segmentation into meaningful regions. The steps required to carry out image pre-processing were implemented on low resolution images are as follows:

i) Load colored (RGB) or gray scale image, the colored image is converted to gray scale image. The contrast of the gray scale image is enhanced using local histogram equalization to enhance the visibility of the parasites and RBC.

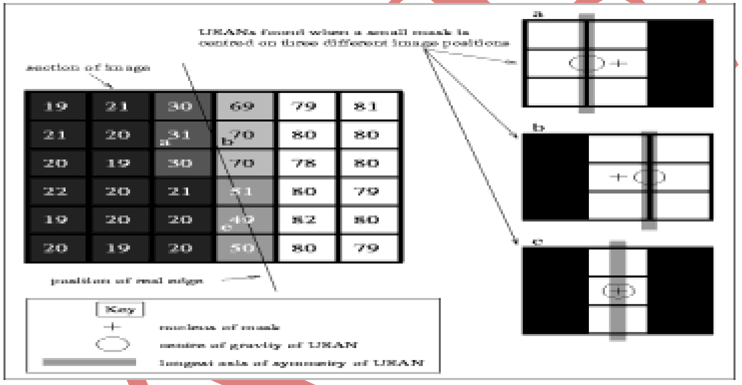
ii) The next and important step in image segmentation is to extract meaningful regions, or in other words, distinguish objects from background. The common way described in the literature is to use edge detection algorithms.

Non-Linear Filtering: SUSAN

For a real time system using time varying image sequences, speed is an important criterion to be considered. Also there has to be a compromise between maximizing signal extraction and minimizing output noise: the so called “Uncertainty Principle” of edge detection. The implemented approach to low-level image processing – SUSAN (Smallest Univalue Segment assimilating Nucleus) Principle, performs Edge and Corner Detection and Structure Preserving Noise Reduction. Canny edge detector, which has become one of the most widely used edge finding algorithms, is found to be ten times slower than this SUSAN approach. The results are stable for Canny but the edge connectivity at junction is poor and corners are rounded. The fact that SUSAN edge and corner enhancement uses no image derivative, explains why the performance in the presence of noise is good. The integrating effect of the principal together with its non-linear response gives strong noise rejection.

Algorithm: The following steps are performed at each image pixel:

- Place a circular mask around the pixel in question.
- Calculate the number of pixels within the circular mask which have similar brightness to the nucleus. These define the USAN.
- Subtract USAN size from geometric threshold to produce edge strength image.
- Use moment calculations applied to the USAN to find edge direction.
- Apply non-maximum suppression thinning and sub-pixel estimation, if required.


USANs found when a small mask is centred on three different image positions
section of image
a
b
c
position of real edge
Key
nucleus of mask
centre of gravity of USAN
longest axis of symmetry of USAN
a
b
c

19	21	30	69	79	81
21	20	31	70	80	80
20	19	30	70	78	80
22	20	21	51	80	79
19	20	20	49	82	80
20	19	20	50	80	79

Fig. 4 The Two Main Edge Types

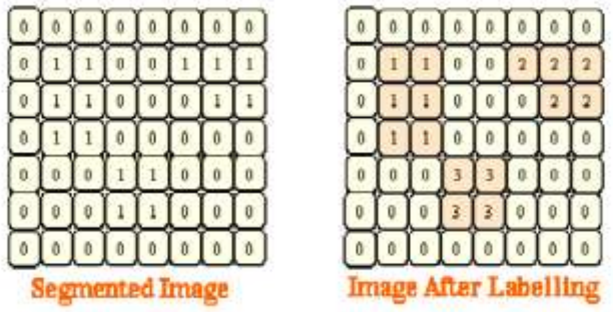
Fig. 4 represents a typical straight edge in a section of a real image, with brightness indicated by the numerical text as well as the shading of the pixels. The SUSANs for three points of interest are shown as the white regions of a small (3 by 3) mask. Points (a) and (b) are standard edge points, lying definitely on one side of the edge or the other. Point (c) lies on a band of brightness half way between the brightnesses of the two regions generating the edge. It therefore has a differently shaped USAN, and a centre of gravity coinciding with the nucleus.

Hence SUSAN allows image edges, corners and junctions to be accurately and quickly found and a related method for reducing noise whilst preserving image structure. The localization of the features is independent of the mask size used and noise suppression is shown to be good.

3.6 Image Segmentation

For the actual recognition stage, *segmentation* should be done before it to extract out only the part that has useful information. The goal of the segmentation process is to define areas within the image that have some properties that make them homogeneous. The definition of those properties should satisfy the general condition that the union of neighboring regions should not be homogeneous if considered the same set of properties. After segmentation, the discontinuities in the image correspond to boundaries between regions can be easily established**.** In this section the presented technique takes benefit of morphological operation and thresholding at appropriate positions in the whole process to maximize the productivity of the algorithm. In order to use morphological methods for image segmentation, the shape and size of the objects in the image must be known. The commonly used method is connected component labelling. The size and eccentricity of the erythrocytes are also required for the calculation of some feature values (as these can be indicative of infection). The shape of the objects (circular) the image and to find the average eccentricity of erythrocytes present.

The binary image of statistically similar region generated after Susan approach distinguishing RBC and background [7], but because of biconcave shape of the RBC, the central pallor is assigned the same features as the background as shown in . It is required to remove the central pallor in the RBC [8] and to perform this task a 'hole filling' algorithm was designed. The technique to fill the holes in the binary digital image is to extract the largest connected component among all the connected components. Connected component analysis extracts the information on pixel connectivity in a two-dimensional image by labelling connected pixels possessing same intensities. All the connected components were extracted as shown in and intersection of the largest connected component i.e. the background was performed with the original image shown in. The reason for explicitly designing a hole-filling method is to be able to obtain an average diameter of the RBCs. To do this there is a need to obtain a count of the RBCs. The central pallor could easily get construed as an RBC and, therefore, it had to be masked or the hole had to be filled to capture the total number of RBCs and their size distribution. Secondly, the hole-filling algorithm provides an extra piece of information about the size distribution of central pallor which can be useful in detecting other blood diseases.



0	0	0	0	0	0	0	0
0	1	1	0	0	1	1	1
0	1	1	0	0	0	1	1
0	1	1	0	0	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0

Segmented Image

0	0	0	0	0	0	0	0
0	1	1	0	0	2	2	2
0	1	1	0	0	0	2	2
0	1	1	0	0	0	0	0
0	0	0	3	3	0	0	0
0	0	0	3	3	0	0	0
0	0	0	0	0	0	0	0

Image After Labelling

Fig. 5 Component Labelling

Bwlabel Algorithm

1. L = bwlabel(BW, n) returns a matrix L, of the same size as BW, containing labels for the connected objects in BW. The variable n can have a value of either 4 or 8, where 4 specifies 4-connected objects and 8 specifies 8-connected objects. If the argument is omitted, it defaults to 8.
2. The elements of L are integer values greater than or equal to 0. The pixels labeled 0 are the background. The pixels labeled 1 make up one object; the pixels labeled 2 make up a second object; and so on.
3. [L, num] = bwlabel(BW, n) returns in num the number of connected objects found in BW.

4. The functions bwlabel, bwlabeln, and bwconncomp all compute connected components for binary images. bwconncomp replaces the use of bwlabel and bwlabeln. It uses significantly less memory and is sometimes faster than the other functions.

3.7 Feature Extraction

Feature selection is one of the most important tasks in data mining area, with methods which allows determining the most relevant features for pattern recognition. A suitable subset of features is found when it permits synthesizing the similarity of the pattern within its class and dissimilarity amongst other different classes. The goal of feature selection is to reduce the dimensionality of vectors associated to patterns selecting a subset of attributes smaller than the original. The classifier performance is often improved eliminating redundant features. Hence the purpose of feature generation is to compute new variables from the image array that concentrate information to separate classes. The classifier has two functions: it must determine whether or not a detected cell is truly positive for malaria, and what the species of the infection is. Features are created with these functions in mind. They provided information with which the classifier distinguished between parasites and other artifacts in the blood, and allowed the classifier to differentiate between parasites of different species. The final performance of the classifier directly depended on the success of the feature generation stage.

Two sets of features had been chosen for development. The first set is based on image characteristics that have been used previously in biological cell classifiers, which include geometric features (shape and size), colour attributes and grey-level textures.

3.7.1 First Order Features

Texture is generated from the grayscale image matrices of the red, green and blue components, as well as the intensity component from the hue-saturation-intensity image space.

1. **Average gray level or Mean:** It is the average or mean value of the array.

$: \quad \bar{} \Sigma$ **(2)**

2. **Standard Deviation:** It shows how much variation or dispersion exists from the average.

$[\Sigma \quad ^{=} \,]$ **(3)**

3. **Kurtosis:** It is a measure of shape and peakedness of probability distribution.

$— \; \Sigma \quad ^{-}$ **(4)**

4. **Skewness:** It is a measure of asymmetry of data around the sample mean.

$:S_S =— \Sigma \quad ^{-}$ **(5)**

5 Entropy: It is a statistical measure of randomness that is used to characterize the texture of the input image.

(6)

$:S_E = - \Sigma$

3.7.2 Probabilistic Neural Network

A **probabilistic neural network (PNN)** [10] [11] is a feedforward neural network, which was derived from the Bayesian network and a statistical algorithm called Kernel Fisher discriminant analysis. In a PNN, the operations are organized into a multilayered feedforward network with four layers:

- Input layer
- Hidden layer
- Pattern layer/Summation layer
- Output layer

Architecture of a PNN Network

In 1990, Donald F. Specht proposed a method to formulate the weighted-neighbor method described above in the form of a neural network. He called this a "*Probabilistic Neural Network*". Here is a diagram of a PNN network as illustrated in Fig 6:

All PNN networks have four layers:

Input layer: There is one neuron in the input layer for each predictor variable. In the case of categorical variables, *N*-1 neurons are used where *N* is the number of categories. The input neurons (or processing before the input layer) standardize the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

Hidden layer: This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the *x* vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function using the sigma value(s). The resulting value is passed to the neurons in the pattern layer.

Pattern layer / Summation layer: The next layer in the network is different for PNN networks. For PNN networks there is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron's category. The pattern neurons add the values for the class they represent (hence, it is a weighted vote for that category).

Decision layer: The decision layer is different for PNN networks. For PNN networks, the decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

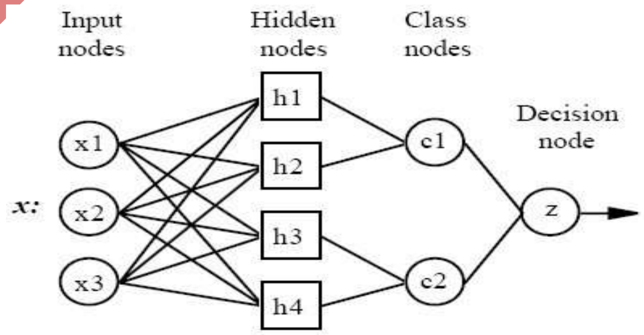


Fig. 6 Layers of PNN

Advantages

There are several advantages and disadvantages using PNN instead of multilayer perceptron.

- PNNs are much faster than multilayer perceptron networks.
- PNNs can be more accurate than multilayer perceptron networks.
- PNN networks are relatively insensitive to outliers.
- PNN networks generate accurate predicted target probability scores.
- PNNs approach Bayes optimal classification.

Disadvantages

- PNN are slower than multilayer perceptron networks at classifying new cases.
- PNN require more memory space to store the model.

Applications based on PNN

- Probabilistic neural networks in modelling structural deterioration of storm water pipes.
- Probabilistic neural networks method to gastric endoscope samples diagnosis based on FTIR spectroscopy.
- Probabilistic Neural Networks in Solving Different Pattern Classification Problems.
- Application of probabilistic neural networks to population pharma cokineties.
- Probabilistic Neural Networks to the Class Prediction of Leukemia and Embryonal Tumor of Central Nervous System.
- Ship Identification Using Probabilistic Neural Networks.
- Probabilistic Neural Network-Based sensor configuration management in a wireless AD-HOC network.
- Probabilistic Neural Network in character recognizing.

3.8 Feature Classification and Result Verification

The final classification of an erythrocyte as infected with malaria or not, and if so, the species of the parasite, falls to the classifier. The classifier is a two-stage tree classifier, with an infection classified as positive or negative at the first node, and the species assigned at the second node.

The design of a tree classifier has the following steps: the design of a tree structure [8] (which has already been assigned), the selection of features to be used at every node, and the choice of decision rule at each node. The same type of classifier is used at both nodes

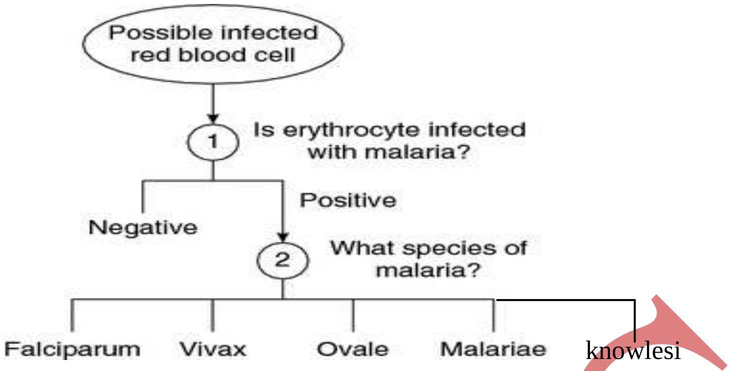


Fig. 7 Two Stage Classifier

3.9 Results

The performance and accuracy of the algorithm are analyzed using 4 measures: **sensitivity**, the ability of the algorithm to detect a parasite present; **specificity**; **accuracy** and **positive predictive value** (PPV), the success of the algorithm at excluding non-infected cells. These values are expressed in terms of true positives (TP), false positives (FP) and false negatives (FN).

Independent of the test result (classification) a test sample can be *positive* (infected) or *negative* (not infected), as determined by another (reference) reliable method. There can be four different outcomes after the classification: (*tp*) true positive (the classification result is positive for a positive sample); (*tn*) true negative (the classification result is negative for a negative sample); (*fp*) false positive (the classification result is positive for a negative sample); (*fn*) false negative (the classification result is negative for a positive sample). The number of occurrences of these conditions can be used to analyse the diagnosis performance:
TP; TN; FP; FN for *tp; tn; fp; fn*, respectively.

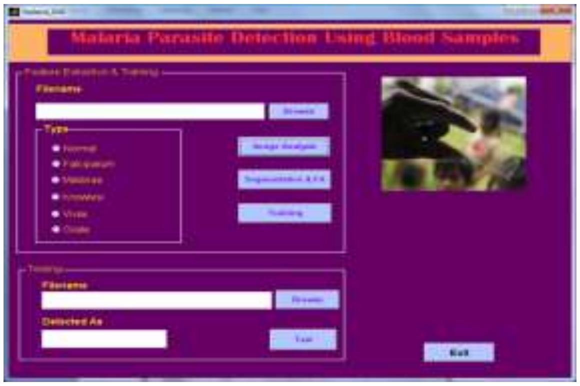
Sensitivity (SE) is the proportion of the samples that are classfiied as positive among all the positive samples, which is usually called the true detection rate (in the pattern recognition community) for binary pattern detection tasks. The higher the sensitivity, the more likely an infected person to be diagnosed as positive (i.e. sick).

:SE = _____ **(7)**

Specificity (SP) is the proportion of the samples that are classified as negative among all the negative samples. It is the probability of a negative result for a negative object (regular blood component, e.g.RBC, WBC, platelet, and artifact).

:SP = _____ **(8)**

The higher the specificity, the less likely that a healthy blood component will be classified as a parasite. Consequently, a healthy person is more likely to be diagnosed as healthy. Sensitivity and specificity values of a diagnosis test should be interpreted together. In theory, for an ideal diagnosis method the values are independent and both can be as high as 1:0. It should be noted that assigning all the queries to one of the classes (consider the positive class) can simply achieve (*SE* = 1:0) but the specificity in this case would be *SP* = 0:0 which is not desirable. In general pattern detection terms, the value (1 - *SP*) is usually called the false detection rate.



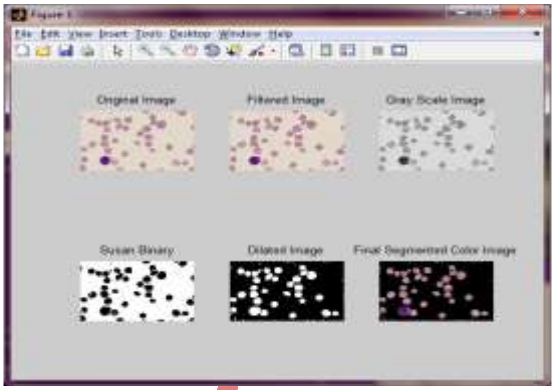
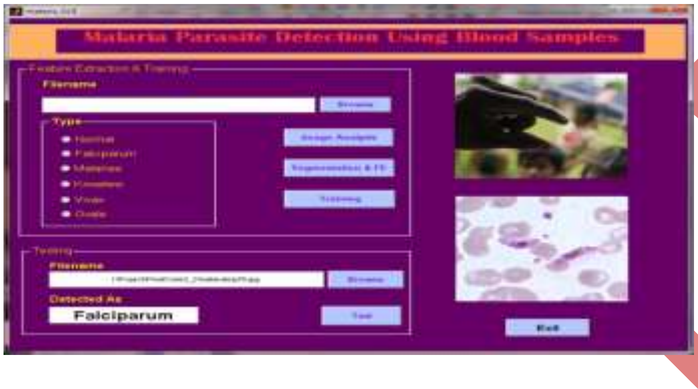


Fig. 8 GUI and Validation of Filename Textbox Fig. 9 Output of Image Segmentation and Feature Extraction



Falciparum

Fig. 10 Detected Parasite Name Displayed

IV CONCLUSION

The proposed automated parasite detection algorithm has many advantages compared to other diagnostic techniques. It avoids the problems associated with rapid methods, such as being species-specific and having high per-test costs, while retaining many of the traditional advantages of microscopy, viz. species differentiation, determination of parasite density, explicit diagnosis and low per-test costs.

Apart from overcoming the limitations of conventional methods of parasite detection, the proposed algorithm is optimized to overcome limitations of image processing algorithms used in the past. Among the tested test algorithms, 'SUSAN edge detection technique' gave good localization of edges but formed a thick border making cell separation difficult. Component labeling gave proper segmented image of the parasites and helped in extracting features from each of the labeled cells separately. Dilation and erosion are two morphological operations they are used for bridging gaps and eliminating irrelevant detail from a binary image. The first order features provide the mathematical ranges for simple RBC and parasite affected RBC. Results prove that the algorithm developed in this paper has better execution speed than all the previous work done in this field. It is applicable to many other blood cell abnormalities other than malaria. This is because the percentage of pathological differences in various diseases has very less effect on this robust algorithm. The algorithm detects the species of parasite with a sensitivity of 92.77%, specificity of 88%, accuracy of 91%, and positive predictive value of 95%.

References

[1]World Health Organization. What is malaria? Facts sheet no 94. http://www.who.int/mediacentre/factsheets/fs094/en/.

[2] F. Castelli, G.Carosi, *Diagnosis of malaria*, chapter 9, Institute of Infectious and Tropical Diseases, University of Brescia (Italy).

[3] Jigyasha Soni *Advanced Image Analysis based system for Automatic Detection of Malarial Parasite in Blood Images Using SUSAN Approach.* IJEST.

[4] S. M. Smith, J. M. Bardy, *SUSAN—A New Approach to Low Level Image Processing, International Journal of Computer Vision*, Volume 23, and Issue 1 Pages: 45 – 78, may 1997.

[5] Rafeal C. Gonzalez, Richard E. Woods, *Digital Image Processing*, 2nd Edition, Prentice Hall, 2006.

[6] Hitech Lab, Pathology Department, KLE Hospital, Belgaum.

[7] Selena W.S. Sio, Malaria Count: *An image analysis-based program for the accurate determination of parasitaemia*, Laboratory of Molecular and Cellular Parasitology, Department of Microbiology, Yong Loo Lin School of Medicine, National University of Singapore. May 2006.

[8] Mui JK, Fu K-S, *Automated classification of nucleated blood cells using a binary tree classifier*. IEEE Trans Pattern Anal Machine Intell 2(5):429–443, 1980

[9] Di Ruberto C, Dempster A, Khan S, Jarra B, *Analysis of infected blood cell images using morphological operators*. Image Vis Comput 20(2):133–146, 2002.

[10] J.M. Zurada, Introduction to Artificial Neural Systems.

[11] J.Hertz, A.Krogh, and R. G. Palmer: Introduction to the Theory of Neural Computation,(Addison Wesley, Redwood City,1991).