*Real Time Classification Of*

*Mouth Ulcers*

Bachelor of Technology in Electrical Engineering

IIIrd Year

(Session: 2015 – 2016)



## *Submitted in partial fulfilment for the*

## *course credits in the course*

## *“EEN 300: Industry Oriented Problem”*

## **Guided by:** **Submitted By:**

## 

## Dr. Vinod Kumar Anshul Kaim ( 13115022 ) (Deputy Director & Devdoot Ghosh ( 13115044 )

## Professor, Hari Eswar SM ( 13115053 )

## Electrical Engineering Department) Kelkar Sachin Madhav ( 13115063 )

## Kritik Chouhan ( 13115069 )

## Shweta Maurya ( 13115117 )

## Suvansh Kumar ( 13115126 )

**Outline**

1

Abstract

2

Database & Methodology

3

Segmentation & Feature Extraction

4

Classification

5

Performance Analysis

6

Application : GUI and working

7

Conclusion and Future Work

8

References

**Abstract**

Oral Cancer is the most common form of cancer in India. Poor adult villagers from remote areas in their 60s and 70s are the usual victims of oral cancer. In India this is particularly deadly due to the extensive use of tobacco coupled with lack of proper diagnostic facilities.

A biopsy done at right time could save many lives. It is the ignorance regarding the right time attributed to the symptoms appearing to be very much similar to those of a normal mouth ulcer .This causes the consequent sufferings and irreparable damage to the diseased.

This underscores the need for easier methods to identify this villain in disguise at an early stage. This is the central theme of undertaking this project.

The objectives of this work include:

1. Study of different methods for segmentation, feature extraction and classification.

2. Selection of methods suitable for the present purpose.

3. Application of the selected methods on the dataset.

4. Proposing the best system suitable for classification of images into cancerous and non-cancerous. (in terms of a percentage of the “ ulcer under analysis being cancerous “

5. Development of an android app to analyse the images with the analysis being done on a cloud server in real time.

**Database and Methodology**

Dataset : Himalayan Institute of Medical Sciences under HIHT, Doiwala, Dehradun , India.

20 malignant images (20 persons)

51 normal images (10 persons)

1. Sony Cybershot digital still picture camera (4x optical zoom, 10.1MP)
2. Patches of size 32x32 pixels were extracted
3. Contact person: Dr. Sunil Saini, Director, Cancer Research Institute, HIHT
4. Finalized sets of patches (480)

Normal patches : 240

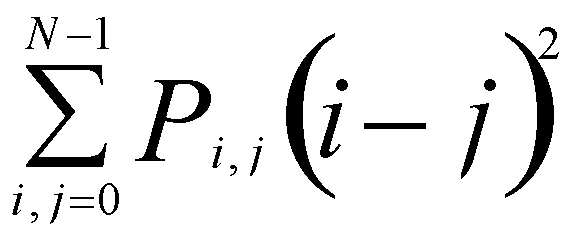
Malignant patches : 240

**Segmentation and Feature Extraction**

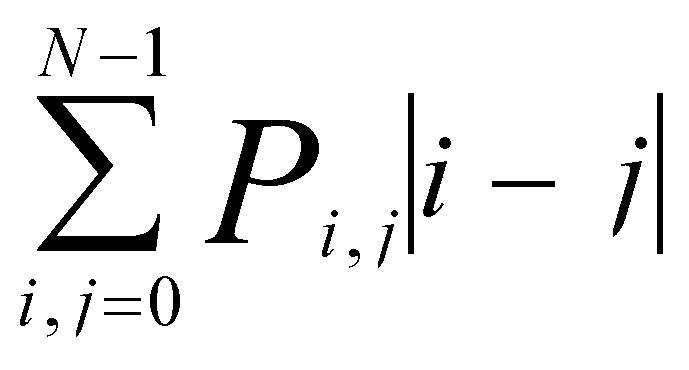
Images acquired by a standard digital camera are given to a segmentation program which identities the ROI. Patches of 32x32 size are extracted from each of the malignant ROI. Similar sized patches are extracted from normal images also. Features based on texture (GLCM), run-length (GLRL) and intensity variation are extracted with the aid of the following equations:

**1. Contrast Group**

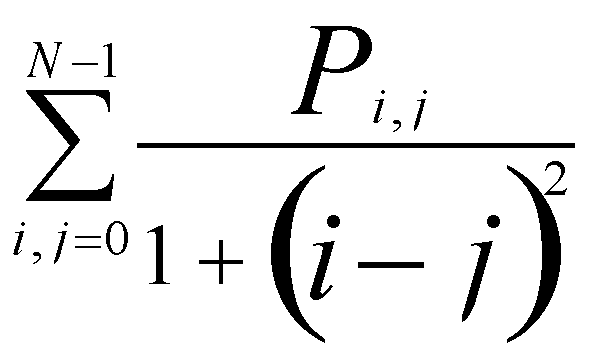
Contrast



Dissimilarity

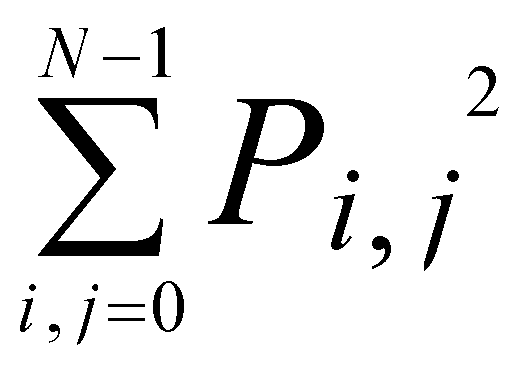


Homogeneity

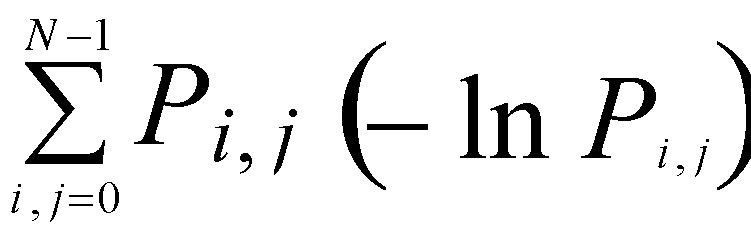


**2. Orderliness Group**

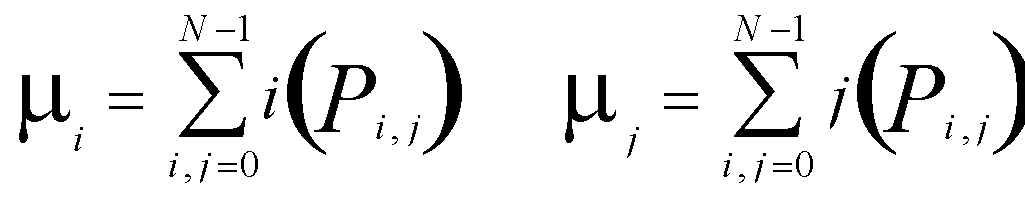
ASM



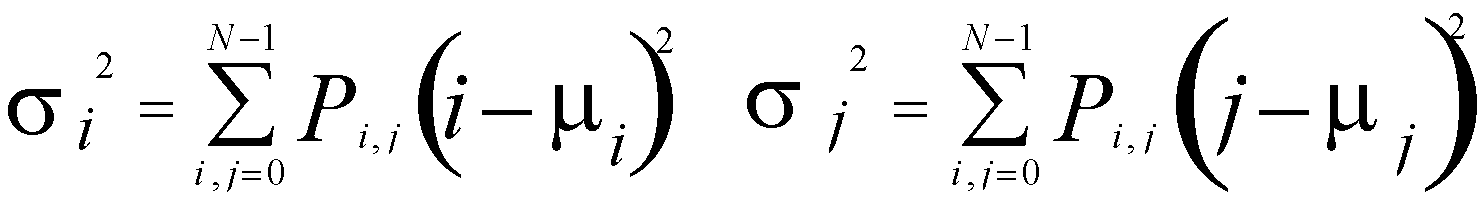
Entropy



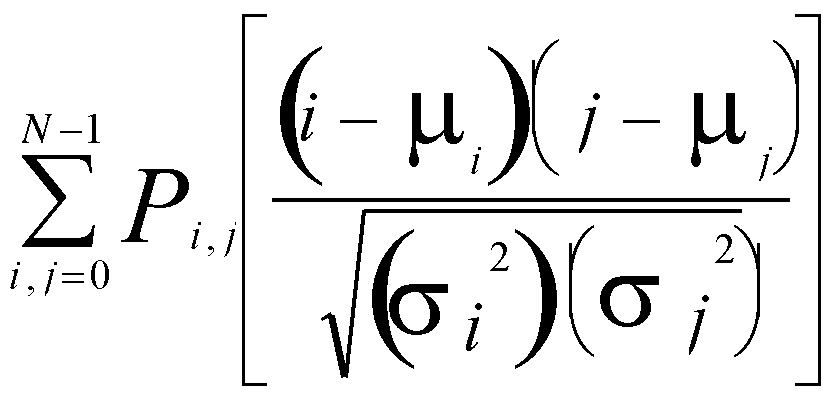
GLCM MEAN



GLCM variance



GLCM Correlation



**Classification**

The dataset operated on by the machine learning algorithm -

an ANN based classifier :

The feed forward type neural network consists of two steps:

1. A forward traversal of computing the net output and error at each layer.

2. A backward traversal where errors are propagated backward and weights are

re-adjusted.

Learning process continues till the error is brought down below a specified threshold. The iterations required for this purpose constitute the total number of epochs. In the current implementation we have used a three layered architecture - one input layer, one hidden layer and one output layer.

The algorithm implemented is discussed below:

Step 1: Initialization of weights with random values and other network parameters.

Step 2: Net output vector calculation for all input training vectors.

Step 3: Network error calculation and computation of sum squared error for all

Input vectors.

Step 4: Calculate new weight matrix of each layer and go to step 2.

Step 5: Continue iterations till sum squared error for all training vectors is less than the specified threshold.

Where,

= Activation function (tan sigmoid for hidden layer and linear for input and output layers)

## x = Input vector

## D = desired output (target)

## V = input to hidden weight vector

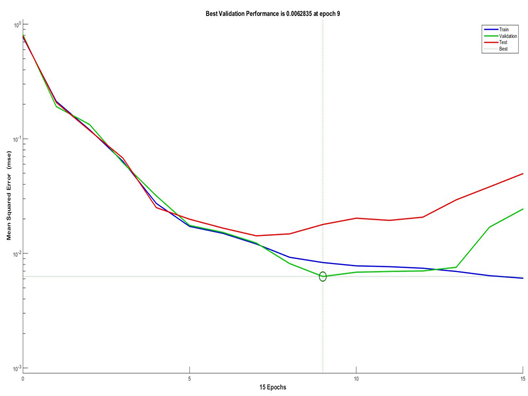
## w = hidden to output weight vector

## n = number of inputs.

**Performance and Analysis**

The training was done by diving the total data set in the ratio 70:15:15. 70%of the dataset was used to train the algorithm (training set). 15% was used as the cross validation set, the cross validation error was used by the Matlab BPANN algorithm to decide the number of layers and the number of features in each layer that gave the minimum error on the cross validation set.

Finally, last 15% of the dataset was used for testing the performance of the algorithm the training set, cross validation set and test set error was plotted by Matlab for each iteration of the BPANN algorithm. The training stopped when cross validation error was minimum which was after 9 iterations. The error on the test set was of the order of 0.01. The plot of the various error is given below.



**Application: GUI and working**

The server side application was made with Python using the Flask web framework.

* Numpy and OpenCV were used for feature extraction of the image.

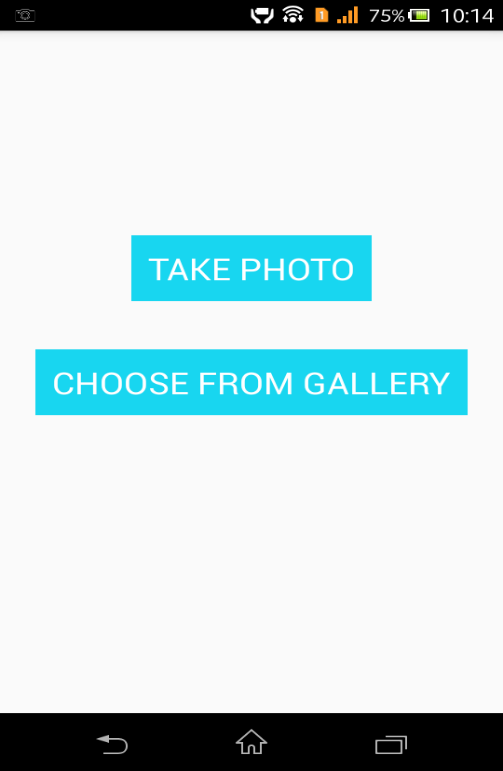
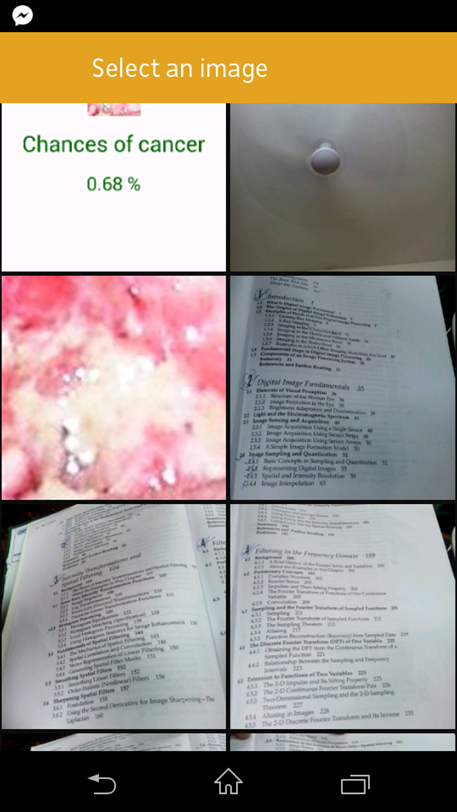
The application accepts a multipart POST request with ‘image’ as a parameter from the client. It then decodes the image to a Numpy array, converting the image to grayscale, and then does feature extraction on the image. A total of 8 features are extracted for an image.

Machine learning training was done offline by backpropagation artificial neural network (BPANN) using MATLAB neural network toolbox (nnet). The weights for the hidden layer were decided from the training and cross validation set. These weights were used with the features extracted from the new image to give the chances of the mouth ulcer being cancerous.

A total of three endpoints were made -

1. The main URL which gives an option to upload images in the browser. It is intended to be used only for quick testing.
2. The ‘/extract’ endpoint which returns the features of the POSTed image in JSON format.
3. The ‘/upload’ endpoint which returns the value i.e. the result of the POSTed image in JSON format.

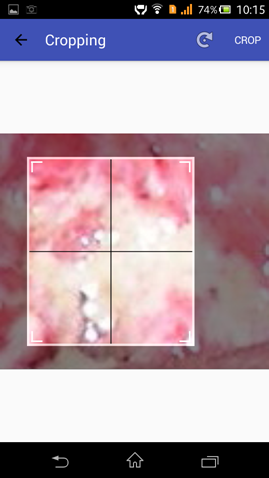
The code can be found at <https://github.com/s4chin/iop>.

**Using the app**

**Step 1:** GUI Home screen **Step2:** Select an image

Select a method for picking (Gallery shown)

the image (Camera or Gallery)

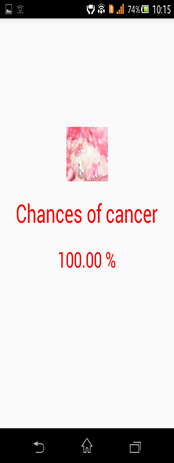
**Step 3:** A window will appear **Step 4:** On changing the size of the

for cropping the image. window a plus sign will appear. The

user must place the plus sign exactly

over the patch to be analysed.

**Step 5**: Click analyse to analyse the image



Application link : <https://github.com/devdoott/RTCoMU-IOP>

**Conclusion and future Work**

The major target of this thesis work is to achieve binary classification of images into cancerous and noncancerous. This elementary goal is reached using various steps as described over the various sub-heads of this report.

In a concise format, the notable contributions made by this work can be enlisted

as given below:

A context specific method of feature selection is proposed, implemented and validated for the purpose of binary classification of oral cavity images into cancerous and noncancerous.

The Graphical User Interface (GUI) developed accepts a camera image as input and outputs an inference on the nature of the image based on analysis and classification performed on the spot. Based on the inference, one can decide whether to go for a biopsy or not.

Addition of rotation and scale invariant features (eg: Gabor wavelet features) to the current feature set can compensate for the possible discrepancies that could creep into the image analysis system due to sloppy acquisition.

The tool developed can be used in mass screening initiatives with required hardware including a common digital camera and a laptop.

Such a tool will act as a supplement to the existing clinical methods and reduce the burden on the medical practitioner in two respects.

1. Any volunteer taking part in a screening programme can use this device to predict within seconds whether the patient has to go for a biopsy and further treatment.

2. It can also be used for quantitative assessment during follow-up to the treatment by comparative studies.

3. Accuracy can be improved with the application of more Machine Learning Algorithms to the dataset.

**References**

1. Android Image Cropper : <http://arthurhub.github.io/Android-Image-Cropper/>
2. GLCM : <http://www.fp.ucalgary.ca/mhallbey/>