ABSTRACT

Quantitative techniques for prediction and classification in soil survey are developing rapidly. Classification of soil is the dissolution to soil sets to particular group having a like characteristics and similar manners. Almost all countries do product exporting, in which those countries exporting higher agricultural product are very much depend on the soil characteristics. Thus, soil characteristics identification and classification is very much important. Identification of the soil type helps to avoid agricultural product quantity loss. A classification for engineering purpose should be based mainly on mechanical properties. This paper explains support vector machine based classification of the soil types. Paper introduces application of Support Vector Machines in the estimate of values of soil properties and soil type classification based on known values of physical properties, texture features and colour moments in sampled profiles. Soil classification includes steps like image acquisition, image pre-processing, feature extraction and classification. The texture features of soil images are extracted using the low pass filter, Gabor filter and using colour quantization technique. Mean amplitude, HSV histogram, Standard deviation are taken as the statistical parameters.

TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **DESCRIPTION** | **PAGE NO.** |
|  | ACKNOWLEDGEMENT | i |
|  | ABSTRACT | ii |
|  | TABLE OF CONTENTS | iii |
| **1** | **INTRODUCTION** | **1** |
| **2** | **LITERATURE SURVEY** | **2** |
| **3** | **PROBLEM STATEMENT OF PROJECT** | **4** |
| **4** | **OBJECTIVE AND SCOPE OF PROJECT** | **5** |
| **5** | **PROPOSED METHODOLOGY** | **6** |
| **6** | **SIGNIFICANCE OF PROJECT FOR SOCIETY** | **8** |
|  | **REFERENCES** | **9** |

CHAPTER 1

# INTRODUCTION

Soil is the term which has different meaning for different people: for a geologist it represents the products of past surface processes. To a penologist it represents physical and chemical processes occurring currently. For an engineer soil is the solid thing up on which foundation for houses, factories, building, roads, etc can be built. Soil study means the knowing of externally identifiable patterns seen on soil. Grouping of soil is particularly basic for reasonable agricultural business. Recognizing the characteristics of soil is the key feature to reduce the product quantity losses. Knowing the type of soil is very useful for cultivation, construction etc. As far as plant is concerned plantation according to the soil characteristics is very much important for its success.

Measuring and assessment of soil components and properties is generally a time-consuming and costly procedure. Lack of sampling data is often compensated by results of predictions or modeling. Various modeling procedures, known as predictive soil mapping, are specially developed to estimate spatial distribution of soil variables. Most of them are based on numerical or statistical models of relationship among other environmental variables and soil properties, applied to geographic databases in order to create a predictive map or to derive the values of soil properties at unmeasured sites from field collected data.

The nature of soil is influenced by many factor, some of them are power of hydrogen (PH), Exchangeable sodium percentage, moisture content…etc. depending on their amount in soil they show different characteristics and that varies for different region. In preparation manual segmentation and classification method is monitored. This is time consuming, requires efficient people and expensive also. The main task is to automate the procedure. With the emerging of image processing and machine learning we can efficiently classify the soil sample in to groups which it belong to.

There are many examples of supervised learning methods used in agriculture, especially in precision agriculture. Demonstrated the capability of SVM method, analysing hyperspectral data for identification of weed and nitrogen stresses in early growth stage of a cornfield. Machine learning methods could also be used to predict the quantity forward in time, based on training sets which use the past data. Paper describes classification of the found segments using Machine Learning (ML) method Support Vector Machines (SVM).

In this method two parts important. First part is training part. In which database formation is important work. Second one is testing part. In which actual classification takes place.

For testing purpose select any soil sample image. Which will undergoes processes like pre-processing, Feature extraction, feature preparation. Develop a SVM algorithm to find class of soil. Implementation of algorithm may be by using MATLAB.

There are various methods for soil classification using different algorithms. But still soil classification with high accuracy and low cost is challenging task. So, keeping in view, proposed algorithm is useful to soil classification using SVM(Support Vector Machine).We can also use ANN method for classification. Here main objective is to classify soil sample picture with high accuracy and low cost.

**DOMAIN INTRODUCTION**

**DIGITAL IMAGE PROCESSING**

* 1. **GENERAL**

The term digital image refers to processing of a two dimensional picture by a digital computer. In a broader context, it implies digital processing of any two dimensional data. A digital image is an array of real or complex numbers represented by a finite number of bits. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

**1.1.1THE IMAGE PROCESSING SYSTEM**

**DIGITIZER**

**MASS STORAGE**

**HARD COPY DEVICE**

**DISPLAY**

**IMAGE PROCESSOR**

**DIGITAL COMPUTER**

**OPERATOR CONSOLE**

**FIG 1.1 BLOCK DIAGRAM FOR IMAGE PROCESSING SYSTEM**

* **DIGITIZER**

A digitizer converts an image into a numerical representation suitable for input into a digital computer. Some common digitizers are

* Microdensitometer
* Flying spot scanner
* Image dissector
* Videocon camera
* Photosensitive solid- state arrays.
* **IMAGE PROCESSOR**

An image processor does the functions of image acquisition, storage, preprocessing, segmentation, representation, recognition and interpretation and finally displays or records the resulting image. The following block diagram gives the fundamental sequence involved in an image processing system.

**PROBLEM DOMAIN**

**KNOWLEDGE**

**BASE**

**SEGMENTATION**

**PREPROCESSING**

**IMAGE ACQUISITION**

**RECOGNITION & INTERPRETATION**

**REPRESENTATION & DESCRIPTION**

**RESULT**

**FIG 1.2 BLOCK DIAGRAM OF FUNDAMENTAL SEQUENCE INVOLVED IN AN IMAGE PROCESSING SYSTEM**

As detailed in the diagram, the first step in the process is image acquisition by an imaging sensor in conjunction with a digitizer to digitize the image. The next step is the preprocessing step where the image is improved being fed as an input to the other processes. Preprocessing typically deals with enhancing, removing noise, isolating regions, etc. Segmentation partitions an image into its constituent parts or objects. The output of segmentation is usually raw pixel data, which consists of either the boundary of the region or the pixels in the region themselves. Representation is the process of transforming the raw pixel data into a form useful for subsequent processing by the computer. Description deals with extracting features that are basic in differentiating one class of objects from another. Recognition assigns a label to an object based on the information provided by its descriptors. Interpretation involves assigning meaning to an ensemble of recognized objects. The knowledge about a problem domain is incorporated into the knowledge base. The knowledge base guides the operation of each processing module and also controls the interaction between the modules. Not all modules need be necessarily present for a specific function. The composition of the image processing system depends on its application. The frame rate of the image processor is normally around 25 frames per second.

* **DIGITAL COMPUTER**

Mathematical processing of the digitized image such as convolution, averaging, addition, subtraction, etc. are done by the computer.

* **MASS STORAGE**

The secondary storage devices normally used are floppy disks, CD ROMs etc.

* **HARD COPY DEVICE**

The hard copy device is used to produce a permanent copy of the image and for the storage of the software involved.

* **OPERATOR CONSOLE**

The operator console consists of equipment and arrangements for verification of intermediate results and for alterations in the software as and when require. The operator is also capable of checking for any resulting errors and for the entry of requisite data.

**IMAGE PROCESSING FUNDAMENTAL**

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form. They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

**IMAGE PROCESSING TECHNIQUES**

This section gives various image processing techniques.

Image Enhancement

Image Restoration

Image Analysis

Image Compression

Image Synthesis

* **IMAGE ENHANCEMENT**

Image enhancement operations improve the qualities of an image like improving the image’s contrast and brightness characteristics, reducing its noise content, or sharpen the details. This just enhances the image and reveals the same information in more understandable image. It does not add any information to it.

* **IMAGE RESTORATION**

Image restoration like enhancement improves the qualities of image but all the operations are mainly based on known, measured, or degradations of the original image. Image restorations are used to restore images with problems such as geometric distortion, improper focus, repetitive noise, and camera motion. It is used to correct images for known degradations.

* **IMAGE ANALYSIS**

Image analysis operations produce numerical or graphical information based on characteristics of the original image. They break into objects and then classify them. They depend on the image statistics. Common operations are extraction and description of scene and image features, automated measurements, and object classification. Image analyze are mainly used in machine vision applications.

* **IMAGE COMPRESSION**

Image compression and decompression reduce the data content necessary to describe the image. Most of the images contain lot of redundant information, compression removes all the redundancies. Because of the compression the size is reduced, so efficiently stored or transported. The compressed image is decompressed when displayed. Lossless compression preserves the exact data in the original image, but Lossy compression does not represent the original image but provide excellent compression.

* **IMAGE SYNTHESIS**

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

**1.1.3 Image types**

There are several ways of encoding the information in an image.

1. Binary image
2. Grayscale image
3. Indexed image
4. True color or RGB image

* **Binary image**

Each pixel is just blackor white. Since there are only two possible values for each pixel (0, 1), we only need one bitper pixel.

* **Grayscale image**

Each pixel is a shade of gray, normally from 0 (black) to 255(white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power of 2.

* **Indexed image**

An indexed image consists of an array and a color map matrix. The pixel values in the array are direct indices into a color map. By convention, this documentation uses the variable name X to refer to the array and map to refer to the color map.

* **True Color or RGB image**

Each pixel has a particular color; that color is described by the amount of red, greenand bluein it. If each of these components has a range 0–255, this gives a total of 2563different possible colors. Such an image is a “stack” of three matrices; representing the red, greenand bluevalues for each pixel. This means that for every pixel there correspond 3 values.

**1.1.4 APPLICATIONS of image processing**

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

* **DOCUMENT PROCESSING**

It is used in scanning, and transmission for converting paper documents to a digital image form, compressing the image, and storing it on magnetic tape. It is also used in document reading for automatically detecting and recognizing printed characteristics.

* **Medicine**

Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyotypes. In medical applications, one is concerned with processing of chest X-rays, cineangiograms, projection images of transaxial tomography and other medical images that occur in radiology, nuclear magnetic resonance (NMR) and ultrasonic scanning. These images may be used for patient screening and monitoring or for detection of tumors’ or other disease in patients.

* **Industry**

Automatic inspection of items on a production line, inspection of paper samples.

* **DEFENSE/INTELLIGENCE**

It is used in reconnaissance photo-interpretation for automatic interpretation of earth satellite imagery to look for sensitive targets or military threats and target acquisition and guidance for recognizing and tracking targets in real-time smart-bomb and missile-guidance systems.

* **RADAR IMAGING SYSTEM**

Radar and sonar images are used for detection and recognition of various types of targets or in guidance and maneuvering of aircraft or missile systems.

* **Agriculture**

Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables distinguishing good and fresh produce from old.

CHAPTER 2

**LITERATURE SURVEY AND OUTCOME OF EXISTING LITERATURE**

B.Bhattacharya, [1]uses the concept of segmentation , feature extraction and classification. The signals which are measured segmented using segmentation algorithms. Boundary energy method is used for extracting features from the input data. Depending on these features classifiers such as ANN, SVM and decision trees are employed and satisfactory results are obtained.

I.T. Young, [2] introduces a concept for analysing the technique for biological shape based on bending energy. It finds out the amount of work used for typical biological shape in addition to this it explains sampling theorem for contours which are connected and closed and a fast algorithm for calculating the bending energy.

Zhongjie Zhang and M.T. Tumay, [3] this paper explains that there will be an uncertainty between the soil composition correlation and mechanical behaviour of soil while deriving from CPT. This uncertainty leads to overlapping of different soil classes. The existing method available for this is point and region estimation. The author introduces a new fuzzy approach here that is independent of CPT.

V. Sudarshan and B. Yamuna, [4] introduces the concept of SVM decoder (support vector machine), it is a margin based classification and regression technique. This paper makes use of machine learning algorithm for training of the data set. And specific decoders have been designed by making use of training size efficiently. Svm is being controlled by an equation that is being formulated as quadratic programming problem.

Branislav Bajat, [5] uses the concept of soil type classification and estimation of soil properties using support vector machines. This paper introduces the concept based on support vector machine to estimate the value of soil properties and map the soil properties. Mapping of the soil properties is based on colour and texture of the soil. Soil properties and soil type classification can also be based on known particular chemical data values.

A.Coerts, [6] uses the concept of cone penetration test(CPT) it is one of the soil investigation method. It is used in the role of sub surface soil and for depth information from the collected soil image samples. This paper is based on constraint classification and consists of various algorithms and properties of classes and topologies of decision tree diagrams. This paper makes use of few parameters to represent geological models using principal component analysis (PCA). It is used for reducing gradient based approaches and to improve the matching processes.

C.H. Juang, X.H. Huang, and R.D. Holtz,[7] introduces the concept of fuzzy measure. It is used to measure integrals of correlation density. It is also used for a base for aggregating the correlation between relative densities and CPT(cone penetration test). There are three levels high, medium and low, sands are selected by the friction ratio. Based on the differences present between these levels the compressibility is measured. And based on fuzzy c-means and integrals correlation density is measured and the results are compared.

R. Webster, [8] introduces two methods for Optimally partitioning soil transects. One method based on the use of window called spilt moving window. And the other method is based on maximum level variance. The transect is examined through spilt moving window and the midpoint is compared. maximum level variance examines every possible region. And minimizing them within the square variance.

CHAPTER 3

**PROBLEM STATEMENT OF PROJECT**

The main task is to automate the procedure of soil type detection. With the emerging of image processing and machine learning we can efficiently classify the soil sample in to groups which it belong to. This paper describes classification of the found segments using Machine Learning (ML) method Support Vector Machines (SVM).

Soil classification includes steps like image acquisition, image preprocessing, feature extraction and classification.

**MOTIVATION**

Agriculture plays a significant role in the Indian economy. At present the population is statistically increasing whereas the ratio agriculture is getting decreased. To ensure this drawback people should encouraged towards agricultural sector. The main challenge is to know the type of soil suitable for different type of agriculture.

The manual identification of soil type requires complete knowledge about the soil. The experienced farmer can make the task easy but accuracy is not ensured. Farmers might not have a complete knowledge of the basic soil problems and applying inappropriate methods causing considerable loss in the products. Hence, automating the system to predict the type of soil helps in reducing the product loss and improving the agricultural products. The machine learning technique on the other hand, makes it less time consuming and effective in classifying the soil into different types.

**OBJECTIVES AND SCOPE OF PROJECT**

* Studying the soil properties like Sandy soil, Silty soil, Clay soil, based on the physical properties.
* The main task is to automate the procedure. With the emerging of image processing and machine learning we can efficiently classify the soil sample in to groups which it belong to.
* Analysing the 2D image to estimate the types of soil and its properties.

**CHAPTER 4**

**EXISTING SYSTEM AND PROPOSED SYSTEM METHODOLOGY**

**EXISTING SYSTEM**

**CONE PENETRATION TEST**

To identification of the soil classes by drilling boreholes and testing soil samples is very expensive. Cone Penetration Testing (CPT) is cheaper alternative for soil classification. In CPT, a metallic cone is pushed into the soil and an indication of the soil strength is obtained by measuring the force needed to let it advance at a constant rate. A CPT recording is a quasi-continuous picture of the subsurface at the test location. It contains the vertical variations of the mechanical characteristics of the subsoil. These variations in turn indicate variations in geological layers and their properties.

During a test, two primary signals are recorded:

1) the cone tip resistance stress (qc),

2) the frictional stress (f1) which is used to derive the more widely used friction ratio Rf = f s\*100/qc.

Additionally, information is available from borehole drilling in the proximity of CPTs typically with the frequency of 1 borehole for 10 CPTs . Observing the variations of qc and Rf and using the nearby borehole information, an expert firstly segments the logs i.e., finds boundaries of layers (class boundaries), and secondly, using the domain knowledge assigns a soil class C, to each segment (where I =, 2, ..., I and I = number of classes).

Also in other papers uses few parameters for representing complex geological models using principle component analysis (PCA). Normal PCA works by performing multiplication using basis matrix and makes high dimensional model. Here optimization is used for mapping (O-PCA) which have non-Gaussian characteristics and enhance the features. Thus it is used for reducing gradient based approaches and to improve the matching process.

**EXISTING SYSTEM DRAWBACKS**

* In preparation manual segmentation and classification method is monitored. This is time consuming, requires efficient people and expensive also.
* Less effective feature extraction methods used in existing systems.
* The existing systems are less efficient and time-consuming.

**PROPOSED SYSTEM**

The main task is to automate the procedure. With the emerging of image processing and machine learning we can efficiently classify the soil sample in to groups which it belong to. This paper describes classification of the found segments using Machine Learning (ML) method Support Vector Machines (SVM). This paper explains support vector machine based classification of the soil types. Soil classification includes steps like image acquisition, image pre-processing, feature extraction and classification. The texture features of soil images are extracted using the low pass filter, Gabor filter and using color quantization technique. Mean amplitude, HSV histogram, Standard deviation are taken as the statistical parameters. With more data and soil science domain-specific tricks, the potential for applying machine learning to soil property prediction would surely be maximized.

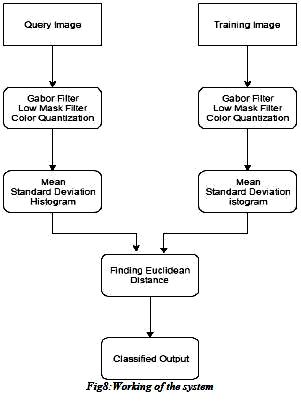
In order to classify the soil there are certain steps to be followed

1. Apply the transformation (low mask filter, color quantization , histogram ) to the original image

2. Use statistical measures to analyses the colour, texture, shape

3. Finding the distance with Euclidean distance formula.

**PROPOSED SYSTEM BLOCK DIAGRAM**

****

**PROPOSED SYSTEM ADVANTAGES**

* Our proposed method shows better performance compared to existing.
* With more data and soil science domain-specific tricks, the potential for applying machine learning to soil property prediction would
* From a set of seven classes the classifier has to identify the appropriate class in this area.
* Accuracy rate is high.

**PROPOSED SYSTEM TECHNIQUES**

**Methods and Algorithms**

1. Image pre - processing
   1. Filtering - Median Filter.
2. Feature Extraction
   1. Color features 🡪 Color moment

(Mean, Standard deviation)

* 1. Texture features 🡪 HSV-Histogram.

1. Classification 🡪 Support vector Machine (SVM).

**CHAPTER 4**

**SYSTEM REQUIREMENT SPECIFICATION**

System requirement specifications gathered by extracting the appropriate information to implement the system. It is the elaborative conditions which the system need to attain. Moreover, the SRS delivers a complete knowledge of the system to understand what this project is going to achieve without any constraints on how to achieve this goal. This SRS not providing the information to outside characters but it hides the plan.

**4.1 Hardware Requirements**

The necessary hardware regarding private PC that comprises configuration as specified as follows:-

1. Processor: Intel core i5.

2. Disk capability: 1GB for MATLAB only.

3. RAM: 2GB.

**4.2 Software Tool used**

The necessary program regarding private PC that comprises configuration as specified as follows:-

1. Windows 7(64-bit) operating system.

2. MATLAB 7.14 Version R2012a

.

**4.2.1 Overview of MATLAB**

MATLAB is built up by means of math works intended for fourth-generation programming language. A variety of process approved within MATLAB contains control concerning the matrix, purpose as well as plotting of data, execution regarding algorithms, design of user interface, as well as integrating by means of programs formed within other languages like C, C++, and java. Despite mathematical calculation, MATLAB can be meant for representational calculation as well. MATLAB can be meant for embedded methods and by the guide regarding extra package known as simulink. Specifically MATLAB permit intended for matrix estimation as well as thus can be intended for image processing. MATLAB is simple towards gaining knowledge of a variety of device boxes used for it; an illustration is image processing toolbox.

MATLAB interfaces programming surroundings, calculation as well as mental picture. This contains integrated correcting, data compositions as well as object-oriented correcting devices. These integrated tasks create MATLAB appropriate used for education as well as do research. To resolve scientific trouble MATLAB includes other benefits than usual programming language like c plus java. MATLAB arrived into promotion in 1984 in addition to now it is employed globally. Additional graphical instructions are offered within MATLAB that builds the visual effects obtainable right away. A variety of device box contains signal processing, simulation, control theory as well as some former that are employed extensively in science and technology. The lone disadvantage regarding MATLAB is expenditure worry.

**4.2.2 Image Processing Toolbox**

Image processing device box permits carrying out image improvement, deblurring of image, characteristic identification, decreasing of noise, image segmentation, arithmetical alteration, as well as registration of image. Image processing device intended for the execution regarding methods proposed are specified below:-

1. Fundamental import as well as export

2. Display

3. Thresholding

**Fundamental import as well as export functions**

Fundamental import as well as export functions permits images obtained by means of image accomplishment plans for example, digital cameras, medical imaging devices such as CT and MRI, microscopes, satellite and airborne sensors, telescopes, and other scientific instruments. Hence those images can be observed; analyzed, as well as process these images into numerous data types, together with single-accuracy as well as double-accuracy floating-point in addition to signed as well as unsigned 8-bit, 16-bit, plus 32-bit integers. Import as well as export functions are accustomed to carry out read as well as write process on images.

**Display function**

Display purpose is accustomed to illustrate the images that are interpreted by means of the import purpose. This purpose permits towards making displays by means of graphics as well as wording, images within a particular window as well as specific displays for example outline plot, histogram and so on.

**Thresholding**

Thresholding is a simple system concerning image segmentation. As of a gray scale image, thresholding can be able to be accustomed to generate binary images. Within thresholding section, the intensity not more than an appropriate value within input image will be displayed as black (intensity is zero) as well as the left out intensities will be made white (intensity is one) then displayed. This procedure is done in the direction of obtaining the segmented image.

**4.2.3 Features of Matlab**

* Interactive background meant for aim investigation as well as resolving the difficulty.
* MATLAB is a sophisticated language intended for creating, calculating as well as building up a purpose.
* It contains numerical tasks such as figures, calculus, sorting out, developments, mathematical integration, as well as working out equations.
* Graphics integrated intended for visualization.
* Intended for generating traditional plot integrated equipments is accessible.
* Troubles as well as way outs are given in well-known numerical symbol.

**4.2.4 SOFTWARE REQUIREMENTS**

* MATLAB 8.3 Version R2014a

**MATLAB**

The MATLAB high-performance language for technical computing integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

* Data Exploration ,Acquisition, Analyzing &Visualization
* Engineering drawing and Scientific graphics
* Analyzing of algorithmic designing and development
* Mathematical functions and Computational functions
* Simulating problems prototyping and modeling
* Application development programming using GUI building environment.

Using MATLAB, you can solve technical computing problems faster than with traditional programming languages, such as C, C++, and Fortran.

**CHAPTER 5**

**SYSTEM DESIGN**

**PROPOSED SYSTEM BLOCK DIAGRAM**

**Database**

**Soil\_Type**

**Classification(SVM)**

TRAIN

IMAGE\_PRE-PROCESSING

FEATURE EXTRACTIONA

**Contrast Enhancement**

**Resize\_Image**

**Image Filter**

**1.Gabor Filter 2.Colour Movement 3. HSV Histogram**

**Image Acquisition**

**Contrast Enhancement**

**Resize\_Image**

**Image Filter**

**1.Gabor Filter 2.Colour Movement 3. HSV Histogram**

TEST

IMAGE\_PRE-PROCESSING

FEATURE EXTRACTIONA

**FIG: BLOCK DIAGRAM OF PROPOSED SYSTEM**

In this method two parts important. First part is training part. In which database formation is important work .Second one is testing part. In which actual classification takes place.

1. **Training Part**

In training part, first part is to collect different types of soil sample picture. Then preprocesses apply on that soil images. Then develop an algorithm to extract the features of soil sample images using features extraction methods ,likeHsvhistogram ,Color moments-mean sd, Gabor features ,Colour auto correlogram. Then develop database with feature preparation of sample images. Develop an SVM algorithm for deciding class.

1. **Testing Part**

For testing purpose select any soil sample image. Which will undergoes processes like pre-processing , Feature extraction, feature preparation. Develop an SVM algorithm to find class of soil. Implementation of algorithm may be by using MATLAB.

**FLOW CHART:**

Feature Parameters

(Mean, Standard deviation , histogram )

clayey sand

Humus clay

Clayey Peat

Peat

Silty Sand

sandy clay

SVM

Image Pre-processing

Query Image

Image Filtering & Enhance Contrast

Feature Extraction

Classifications

Clay

Median Filter

(Gabor Filter , low pass filter , color Momentts)

Soil images

**FIG: DATA\_FLOW DIAGRAM OF PROPOSED SYSTEM.**

**5.1 Data Flow Diagram**

A data flow diagram (DFD) is an illustration in a graphical form. It is an illustration concerning the "run" of information all the way through in a sequence, representing its procedure part. Frequently they are a beginning move employed towards building a general idea regarding the method that will be detailed later on. A DFD demonstrates the variety of input data in sequence to with output as of the method, in which the information will approach as of plus exit towards, in addition to in what the information will be accumulated. It doesn’t demonstrate in sequence regarding the instance of procedures, or else in sequence regarding whether the procedures will work sequentially or else separately.

In order to classify the soil there are certain steps to be followed

1. Apply the transformation (low mask filter, color quantization , histogram ) to the original image

2. Use statistical measures to analyses the color ,texture, shape

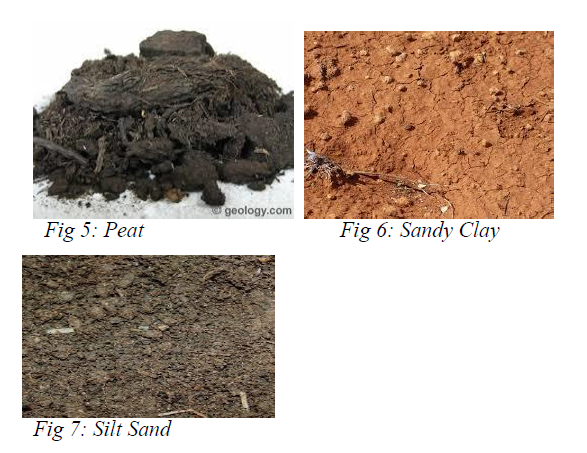
3. Finding the distance with Euclidean distance formula.

In training and testing part following procedure is followed

**DATASET**

The dataset consisted of a collection of soil sample measures. The dataset contain Silty Sand, Sandy Clay,Peat, Humus Clay, Clayey Sand, Clayey Peat, Clay.

Some of the dataset images is as follows:



**PRE-PROCESSING**

This is an essential part of human action recognition based on skeleton joints. In this step the image is made ready for the actual matching. The input of this phase is the original human action image and the final output of this step is the minutiae of that image.

**RESCALING**

The size of the input image was rescaled to [256, 256] or [512, 512], since the input image given for the segmentation and feature extraction should be in square matrix form.

**IMAGE ENHANCEMENT**

imadjust([I](https://in.mathworks.com/help/images/ref/imadjust.html" \l "budqw0o-1-I)) maps the intensity values in grayscale image I to new values in J. By default, imadjust saturates the bottom 1% and the top 1% of all pixel values. This operation increases the contrast of the output image J.

**MEDIAN FILTER**

Median Filter is a non-linear digital filtering technique which is used to remove noise from an image. Such a noise reduction is a pre-processing step that improves the results of later processing. Median filtering is often used in digital image processing as the edges are preserved under certain conditions while removing noise.

**FEATURE EXTRACTION**

**1.Low pass Filter**

A low-pass filter passes frequency below the cut-off frequency and attenuates the higher frequency. The attenuated frequency depends on the filter design. In audio applications this also called as high-cut filter or treble –cut filter.

**2.Color Moment**

Color moments are procedures that characterise color distribution in an image similar to the central moments uniquely describe a probability distribution. Color moments are used for color indexing purposes as features in image retrieval applications to compare the two images based on color. Usually one image is compared to a database of digital images with pre-computed features in order to find and retrieve a similar Image. Each comparison between images results a similarity score, and the lower this score is the more identical the two images are supposed to be.

**B. Feature Parameters**

1. Mean

2. Standard Deviation

**3. Histogram**

A histogram is an precise representation of the distribution of numerical data. It is an estimation of the probability distribution of a continuous variable. To construct a histogram, the first step is to "bin" the range of values which is to divide the entire range of values into a series of intervals, and then count how many values drop into each interval. The bins are usually quantified as consecutive, non-overlapping intervals of a variable. The bins must be head-to-head, and are often of equal size.

**SVM CLASSIFICATION**

Finally, SVM classifier is made using the whole model of feature subset selection. The pattern classification is defined as the task to categorize any object within a given category called class. For this paper, the classification stage was made using a support vector machine (SVM). It classifies the given image is cancerous or healthy.

CHAPTER 6

**FUNCTIONAL MODULES**

**MODULE 1**

**INPUT IMAGE CONVERSION**

**Preprocessing:**

In this section enhancement like processes carried out. Which will help for feature extraction process.

**MODULE 2**

**IMAGE ENHANCEMENT**

**a.Enhancement**

Image enhancement is the process of adjusting digital images so that the results are more suitable for display or further image analysis. We can use enhancement process to remove noise, sharpen, or brighten an image, making it easier to identify key features.

In computer graphics, the process of improving the quality of a digitally stored image by manipulating the image with software. It is easy processes , like to make an image lighter or darker, or to increase or decrease contrast. Advanced image enhancement software also supports many filters for altering images in various ways. Programs specialized for image enhancements are sometimes called image editors. This processing helps in maximizing clarity, sharpness and details of features of interest towards in format extraction and further analysis The objective of image enhancement is to process a given image so that the result is more suitable than the original image for a specific application. It sharpens image features such as edges, boundaries, or contrast to make a graphic display more helpful for display and analysis. The enhancement doesn't increase the inherent information content of the

data, but it increases the dynamic range of the chosen features so that they can be detected easily.

Methods for enhancement is divided into two category.

1.Spacial domain method

2.Frequency domain method

**a.1 Spatial domain methods**

Spatial Domain isAggregate of pixels composing an image. Spatial Domain Methods equal to procedures that operate directly on these pixels, denoted by g(x,y)=T[f(x,y)], T Operator on Image f(x,y) Processed Image. g(x,y) is output Image, T can also operate on a set of Images.

Examples are Negation, Contrast stretching , Gray level slicing.

**a.2 Frequency domain methods**

Image enhancement in the frequency domain is straightforward. Here simply compute the Fourier transform of the image to be enhanced, multiply the result by a filter (rather than convolve in the spatial domain), and take the inverse transform to produce the enhanced image.The idea of blurring an image by reducing its high frequency components, or sharpening an image by increasing the magnitude of its high frequency components is intuitively easy to understand.

However, computationally, it is often more efficient to implement these operations as convolutions by small spatial filters in the spatial domain. Understanding frequency domain concepts is important, and leads to enhancement techniques that might not have been thought of by restricting attention to the spatial domain. Examples are Low pass filter ,High pass filter.

**MODULE 3**

**FEATURE EXTRACTION**

**C.Feature Extraction**

In this section features extraction methods ,likeHsvhistogram ,Color moments-mean sd, Gabor features will use to extract features of soil.

**C.1. HSV-Histogram**

In image processingand photography, a color histogram is a representation of the distribution of colors in an image. For digital images, a color histogram represents the number of pixels that have colors in each of a fixed list of color ranges, that span the image's color space, the set of all possible colors.The color histogram can be built for any kind of color space, although the term is more often used for three dimensional spaces like RGB or HSV. For monochromatic images, the term intensity histogram may be used instead. For multi-spectral images, where each pixel is represented by an arbitrary number of measurements (for example, beyond the three measurements in RGB), the color histogram is N-dimensional, with N being the number of measurements taken. Each measurement has its own wavelength range of the light spectrum, some of which may be outside the visible spectrum. If the set of possible color values is sufficiently small, each of those colors may be placed on a range by itself; then the histogram is merely the count of pixels that have each possible color. Most often, the space is divided into an appropriate number of ranges, often arranged as a regular grid, each containing many similar color values. The color histogram may also be represented and displayed as a smooth function defined over the color space that approximates the pixel counts. Like other kinds of histograms, the color histogram is a statistic that can be viewed as an approximation of an underlying continuous distribution of colors values.

**C.2 Color moment**

Color moments are measures that characterize color distribution in an image in the same way that central moments uniquely describe a probability distribution. Color moments are mainly used for color indexing purposes as features in image retrieval applications in order to compare how similar two images are based on color. Usually one image is compared to a database of digital images with pre-computed features in order to find and retrieve a similar image. Each comparison between images results in a similarity score, and the lower this score is the more identical the two images are supposed to be.

**1) Mean**

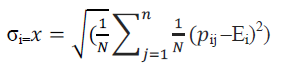
The first color moment can be interpreted as the average color in the image, and it can be calculated by using the following formula



where N is the number of pixels in the image and Pi is the value of the j-th pixel of the image at the i-th color channel.

**2) Standard Deviation**

3) The second color moment is the standard deviation, which is obtained by taking the square root of the variance of the color distribution.

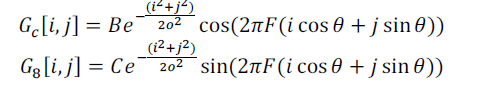


where Ei is the mean value, or first color moment, for the i-th color channel of the image.

**C.3 Gabor Filter**

In image processing, a Gabor filter, named after Dennis Gabor, is a linear filter used for edge detection. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination. Simple cells in the visual cortex of mammalian brains can be modeled by Gabor functions. Thus, image analysis with Gabor filters is thought to be similar to perception in the human visual system.

A set of Gabor filters with different frequencies and orientations may be helpful for extracting useful features from an image. In the discrete domain, two-dimensional Gabor filters are given by,

B and C are normalizing factors to be determined. 2-D Gabor filters have rich applications in image processing, especially in feature extraction for texture analysis and segmentation. f defines the frequency being looked for in the texture. By varying θ, we can look for texture oriented in in a particular direction. By varying σ, change the support of the basis or the size of the image region being analyzed.

**D.Feature Database:**

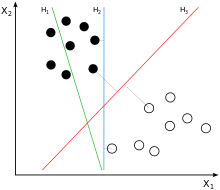
In this section with the help of previous two blocks feature database formation is carried out.

**MODULE 4**

**SVM CLASSIFICATION**

**E.Support Vector Machine:**

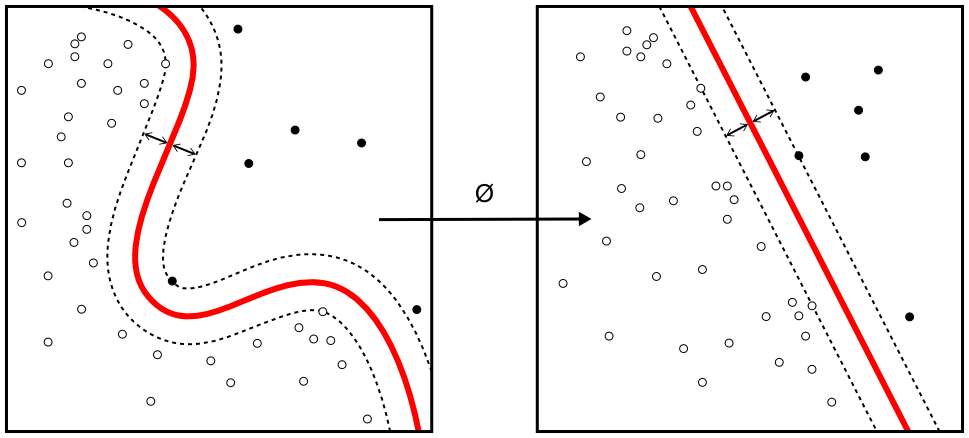
For actual classification SVM i.e. support vector machine method is useful. In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. When data are not labeled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The clustering algorithm which provides an improvement to the support vector machines is called support vector clustering and is often used in industrial applications either when data is not labeled or when only some data is labeled as a preprocessing for a classification pass.



**fig. 2 Example of separation of classes**

From the figure 2 H1 does not separate the classes,H2 does, but only with a small margin, H3 separates them with the maximum margin. Classifying data is a common task in machine learning. Suppose some given data points each belong to one of two classes, and the goal is to decide which class a new data point will be in. In the case of support vector machines, a data

point is viewed as a p-dimensional vector (a list of p numbers), and we want to know whether it can separate such points with a (p-1)dimensional hyperplane. This is called a linear classifier. There are many hyperp lanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the maximum-margin hyperplaneand the linear classifier it defines is known as a maximum margin classifier; or equivalently, the perceptron of optimal stability. More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.



**Fig. 3 kernel machine**

Whereas the original problem may be stated in a finite dimensional space, it often happens that the sets to discriminate are not linearly separable in that space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space making the separation easier in that space. To keep the computational load reasonable, the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function k(x, y) selected to suit the problem .The hyperplanes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant. The vectors defining the hyperplanes can be chosen to be linear combinations with parameters αi of images of feature vectors xi that occur in the data base. With this choice of a hyperplane, the points x in the feature space that are mapped into the hyperplane are defined by the relation: αi k(xi, x)=Constant. The fact that the set of points x mapped into any hyperplane can be quite convoluted as a result, allowing much more complex discrimination between sets which are not convex at all in the original space.

**CHAPTER 7**

**IMPLEMENTATION**

The execution stage regarding the task is that the complete aim is essentially changed keen on running code. Intend regarding the stage is towards interpreting the aim keen on a finest likely result within an appropriate programmed language. In this section, it covers up the execution phase concerning the task, providing particulars regarding the programmed language as well as improvement background employed. It as well provides a general idea about the important sections regarding the task by means of its bit by bit course.

The execution phase involves the following tasks:-

* Cautious scheduling.
* Examination regarding structure as well as constraints.
* Aim concerning the techniques towards accomplishing the conversion.
* Assessment concerning the conversion technique.
* Accurate judgment about the choosing of the proposal.
* Suitable choosing regarding the language intended for function growth.

**7.3 INTRODUCTION**

**MATLAB** (**mat**rix **lab**oratory) is a [numerical computing](http://en.wikipedia.org/wiki/Numerical_analysis) environment and [fourth-generation programming language](http://en.wikipedia.org/wiki/Fourth-generation_programming_language). Developed by [Math Works](http://en.wikipedia.org/wiki/MathWorks), MATLAB allows [matrix](http://en.wikipedia.org/wiki/Matrix_(mathematics)) manipulations, plotting of [functions](http://en.wikipedia.org/wiki/Function_(mathematics)) and data, implementation of [algorithms](http://en.wikipedia.org/wiki/Algorithm), creation of [user interfaces](http://en.wikipedia.org/wiki/User_interface), and interfacing with programs written in other languages, including [C](http://en.wikipedia.org/wiki/C_(programming_language)), [C++](http://en.wikipedia.org/wiki/C%2B%2B), [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), and [Fortran](http://en.wikipedia.org/wiki/Fortran).

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) [symbolic engine](http://en.wikipedia.org/wiki/Computer_algebra_system), allowing access to [symbolic computing](http://en.wikipedia.org/wiki/Symbolic_computing) capabilities. An additional package, [Simulink](http://en.wikipedia.org/wiki/Simulink), adds graphical multi-domain simulation and [Model-Based Design](http://en.wikipedia.org/wiki/Model_based_design) for [dynamic](http://en.wikipedia.org/wiki/Dynamical_system) and [embedded systems](http://en.wikipedia.org/wiki/Embedded_systems).

In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of [engineering](http://en.wikipedia.org/wiki/Engineering), [science](http://en.wikipedia.org/wiki/Science), and [economics](http://en.wikipedia.org/wiki/Economics). MATLAB is widely used in academic and research institutions as well as industrial enterprises.

MATLAB was first adopted by researchers and practitioners in [control engineering](http://en.wikipedia.org/wiki/Control_engineering), Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of [linear algebra](http://en.wikipedia.org/wiki/Linear_algebra) and [numerical analysis](http://en.wikipedia.org/wiki/Numerical_analysis), and is popular amongst scientists involved in [image processing](http://en.wikipedia.org/wiki/Image_processing). The MATLAB application is built around the MATLAB language. The simplest way to execute MATLAB code is to type it in the Command Window, which is one of the elements of the MATLAB Desktop. When code is entered in the Command Window, MATLAB can be used as an interactive mathematical [shell](http://en.wikipedia.org/wiki/Shell_(computing)). Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a [script](http://en.wikipedia.org/wiki/Shell_script) or encapsulated into a [function](http://en.wikipedia.org/wiki/Functional_programming), extending the commands available.

MATLAB provides a number of features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

**7.4 FEATURES of matlab**

* High-level language for technical computing.
* Development environment for managing code, files, and data.
* Interactive tools for iterative exploration, design, and problem solving.
* Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration.
* 2-D and 3-D graphics functions for visualizing data.
* Tools for building custom graphical user interfaces.
* Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++, Fortran, Java™, COM, and Microsoft Excel.

MATLAB is used in vast area, including signal and image processing, communications, control design, [test and measurement](http://www.mathworks.in/applications/t_m), financial modeling and analysis, and computational. Add-on toolboxes (collections of special-purpose MATLAB functions) extend the MATLAB environment to solve particular classes of problems in these application areas.

MATLAB can be used on personal computers and powerful server systems, including the [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha" \o "Cheaha) compute cluster. With the addition of the Parallel Computing Toolbox, the language can be extended with parallel implementations for common computational functions, including for-loop unrolling. Additionally this toolbox supports offloading computationally intensive workloads to [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha" \o "Cheaha) the campus compute cluster.MATLAB is one of a few languages in which each variable is a matrix (broadly construed) and "knows" how big it is. Moreover, the fundamental operators (e.g. addition, multiplication) are programmed to deal with matrices when required. And the MATLAB environment handles much of the bothersome housekeeping that makes all this possible. Since so many of the procedures required for Macro-Investment Analysis involves matrices, MATLAB proves to be an extremely efficient language for both communication and implementation.

**7.4.1 INTERFACING WITH OTHER LANGUAGES**

MATLAB can call functions and subroutines written in the [C programming language](http://en.wikipedia.org/wiki/C_(programming_language)) or [FORTRAN](http://en.wikipedia.org/wiki/Fortran). A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "[MEX-files](http://en.wikipedia.org/wiki/MEX_file)" (for **M**ATLAB **ex**ecutable).

Libraries written in [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), [ActiveX](http://en.wikipedia.org/wiki/ActiveX) or [.NET](http://en.wikipedia.org/wiki/.NET_Framework) can be directly called from MATLAB and many MATLAB libraries (for example [XML](http://en.wikipedia.org/wiki/XML) or [SQL](http://en.wikipedia.org/wiki/SQL) support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with MATLAB extension, which is sold separately by Math Works, or using an undocumented mechanism called JMI (Java-to-Mat lab Interface), which should not be confused with the unrelated Java that is also called JMI.

As alternatives to the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) based Symbolic Math Toolbox available from Math Works, MATLAB can be connected to [Maple](http://en.wikipedia.org/wiki/Maple_(software)) or [Mathematica](http://en.wikipedia.org/wiki/Mathematica" \o "Mathematica).

Libraries also exist to import and export [MathML](http://en.wikipedia.org/wiki/MathML" \o "MathML).

* **Development Environment**
* Startup Accelerator for faster MATLAB startup on Windows, especially on Windows XP, and for network installations.
* [Spreadsheet Import Tool](http://www.mathworks.in/videos/matlab/new-spreadsheet-import-tool-in-r2011b.html?type=shadow) that provides more options for selecting and loading mixed textual and numeric data.
* Readability and navigation improvements to warning and error messages in the MATLAB command window.
* [Automatic variable and function renaming](http://www.mathworks.in/videos/matlab/new-automatic-variable-and-function-renaming-in-r2011b.html?type=shadow) in the MATLAB Editor.
* **Developing Algorithms and Applications**

MATLAB provides a high-level language and development tools that let you quickly develop and analyze your algorithms and applications.

* **The MATLAB Language**

The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory. In many cases, MATLAB eliminates the need for ‘for’ loops. As a result, one line of MATLAB code can often replace several lines of C or C++ code.

At the same time, MATLAB provides all the features of a traditional programming language, including arithmetic operators, flow control, data structures, data types, [object-oriented programming](http://www.mathworks.in/products/matlab/object_oriented_programming.html) (OOP), and debugging features.

MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution. For fast execution of heavy matrix and vector computations, MATLAB uses processor-optimized libraries. For general-purpose scalar computations, MATLAB generates machine-code instructions using its JIT (Just-In-Time) compilation technology.

This technology, which is available on most platforms, provides execution speeds that rival those of traditional programming languages.

* **Development Tools**

MATLAB includes development tools that help you implement your algorithm efficiently. These include the following:

**MATLAB Editor**

Provides standard editing and debugging features, such as setting breakpoints and single stepping

**Code Analyzer**

Checks your code for problems and recommends modifications to maximize performance and maintainability

**MATLAB Profiler**

Records the time spent executing each line of code

**Directory Reports**

Scan all the files in a directory and report on code efficiency, file differences, file dependencies, and code coverage

**Designing Graphical User Interfaces**

By using the interactive tool GUIDE (Graphical User Interface Development Environment) to layout, design, and edit user interfaces. GUIDE lets you include list boxes, pull-down menus, push buttons, radio buttons, and sliders, as well as MATLAB plots and Microsoft ActiveX® controls. Alternatively, you can create [GUIs](http://www.mathworks.in/discovery/matlab-gui.html) programmatically using MATLAB functions.

**5.5 The MATLAB System**

The MATLAB system consists of five main parts:

* **Development Environment**.

This is the set of tools and facilities that help you use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, and browsers for viewing help, the workspace, files, and the search path.

* **The MATLAB Mathematical Function Library**.

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

* **The MATLAB Language**.

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

* **Handle Graphics**.

This is the MATLAB graphics system. It includes high-level commands for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

* **The MATLAB Application Program Interface (API).**

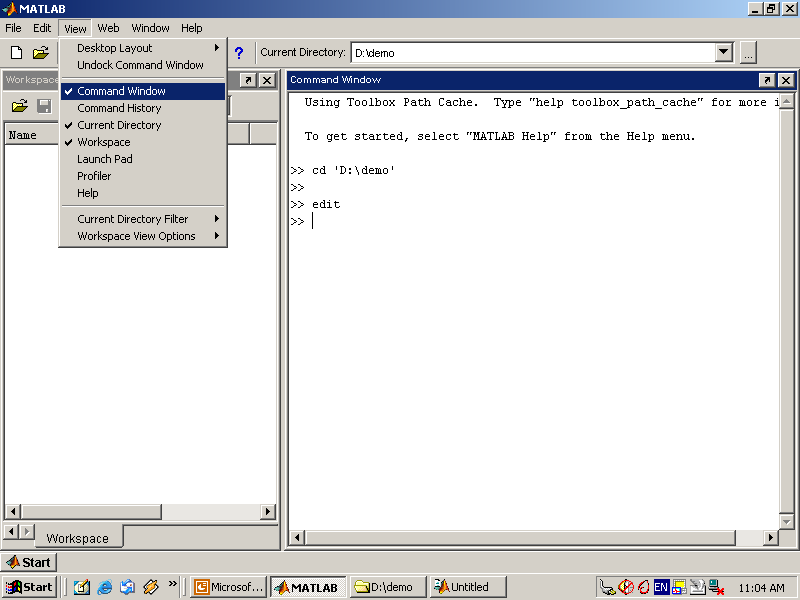
This is a library that allows you to write C and FORTRAN programs that interact with MATLAB. It include facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

**7.5.1 DESKTOP TOOLS**

This section provides an introduction to MATLAB's desktop tools. You can also use MATLAB functions to perform most of the features found in the desktop tools. The tools are:

* Current Directory Browser
* Workspace Browser
* Array Editor
* Editor/Debugger
* Command Window
* Command History
* Launch Pad
* Help Browser

**Command Window**



Use the Command Window to enter variables and run functions and M-files.

* **Command History**

Lines you enter in the Command Window are logged in the Command History window. In the Command History, you can view previously used functions, and copy and execute selected lines. To save the input and output from a MATLAB session to a file, use the diary function.

* **Running External Programs**

You can run external programs from the MATLAB Command Window. The exclamation point character! is a shell escape and indicates that the rest of the input line is a command to the operating system. This is useful for invoking utilities or running other programs without quitting MATLAB. On Linux, for example,!emacs magik.m invokes an editor called emacs for a file named magik.m. When you quit the external program, the operating system returns control to MATLAB.

* **Launch Pad**

MATLAB's Launch Pad provides easy access to tools, demos, and documentation.

* **Help Browser**

Use the Help browser to search and view documentation for all your Math Works products. The Help browser is a Web browser integrated into the MATLAB desktop that displays HTML documents.

To open the Help browser, click the help button in the toolbar, or type helpbrowser in the Command Window. The Help browser consists of two panes, the Help Navigator, which you use to find information, and the display pane, where you view the information.

* **Help Navigator**

Use to Help Navigator to find information. It includes:

* **Product filter**

Set the filter to show documentation only for the products you specify.

* **Contents tab**

View the titles and tables of contents of documentation for your products.

* **Index tab**

Find specific index entries (selected keywords) in the MathWorks documentation for your products.

* **Search tab**

Look for a specific phrase in the documentation. To get help for a specific function, set the Search type to Function Name.

* **Favorites tab**

View a list of documents you previously designated as favorites.

* **Display Pane**

After finding documentation using the Help Navigator, view it in the display pane. While viewing the documentation, you can:

* **Browse to other pages**

Use the arrows at the tops and bottoms of the pages, or use the back and forward buttons in the toolbar.

* **Bookmark pages**

Click the Add to Favorites button in the toolbar.

* **Print pages**

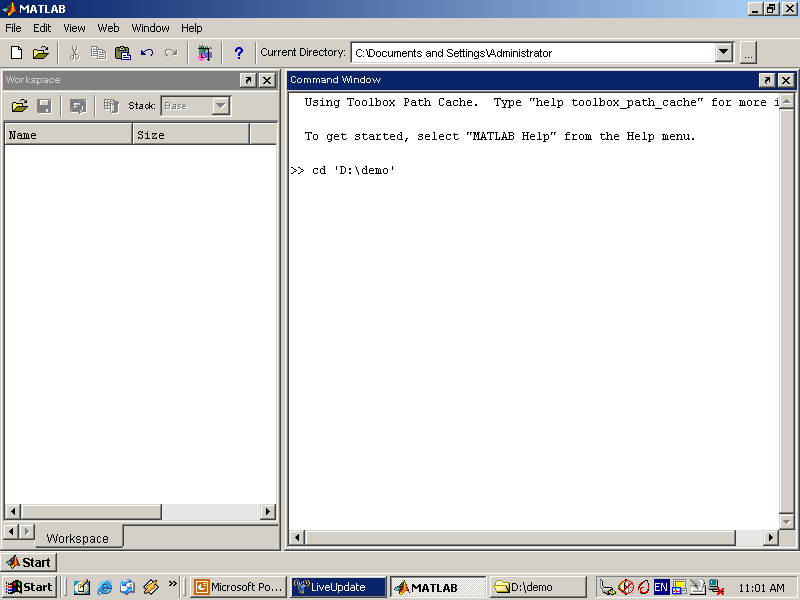
Click the print button in the toolbar.

* **Find a term in the page**

Type a term in the Find in page field in the toolbar and click Go.

Other features available in the display pane are: copying information, evaluating a selection, and viewing Web pages.

**Current Directory Browser**

****

MATLAB file operations use the current directory and the search path as reference points. Any file you want to run must either be in the current directory or on the search path.

**Search Path**

To determine how to execute functions you call, MATLAB uses a search path to find M-files and other MATLAB-related files, which are organized in directories on your file system. Any file you want to run in MATLAB must reside in the current directory or in a directory that is on the search path. By default, the files supplied with MATLAB and MathWorks toolboxes are included in the search path.

* **Workspace Browser**

The MATLAB workspace consists of the set of variables (named arrays) built up during a MATLAB session and stored in memory. You add variables to the workspace by using functions, running M-files, and loading saved workspaces.

To view the workspace and information about each variable, use the Workspace browser, or use the functions who and whos.

To delete variables from the workspace, select the variable and select Delete from the Edit menu. Alternatively, use the clear function.

The workspace is not maintained after you end the MATLAB session. To save the workspace to a file that can be read during a later MATLAB session, select Save Workspace As from the File menu, or use the save function. This saves the workspace to a binary file called a MAT-file, which has a .mat extension. There are options for saving to different formats. To read in a MAT-file, select Import Data from the File menu, or use the load function.

* **Array Editor**

Double-click on a variable in the Workspace browser to see it in the Array Editor. Use the Array Editor to view and edit a visual representation of one- or two-dimensional numeric arrays, strings, and cell arrays of strings that are in the workspace.

* **Editor/Debugger**

Use the Editor/Debugger to create and debug M-files, which are programs you write to run MATLAB functions. The Editor/Debugger provides a graphical user interface for basic textediting, as well as for M-file debugging.

You can use any text editor to create M-files, such as Emacs, and can use preferences (accessible from the desktop File menu) to specify that editor as the default. If you use another editor, you can still use the MATLAB Editor/Debugger for debugging, or you can use debugging functions, such as dbstop, which sets a breakpoint.

If you just need to view the contents of an M-file, you can display it in the Command Window by using the type function.

**7.5.2 ANALYZING AND ACCESSING DATA**

MATLAB supports the entire data analysis process, from acquiring data from external devices and databases, through preprocessing, visualization, and numerical analysis, to producing presentation-quality output.

* **Data Analysis**

MATLAB provides interactive tools and command-line functions for data analysis operations, including:

* Interpolating and decimating
* Extracting sections of data, scaling, and averaging
* Thresholding and smoothing
* Correlation, Fourier analysis, and filtering
* 1-D peak, valley, and zero finding
* Basic statistics and curve fitting
* Matrix analysis

**Data Access**

MATLAB is an efficient platform for accessing data from files, other applications, databases, and external devices. You can read data from popular file formats, such as Microsoft Excel; ASCII text or binary files; image, sound, and video files; and scientific files, such as HDF and HDF5. Low-level binary file I/O functions let you work with data files in any format. Additional functions let you read data from Web pages and XML.

**Visualizing Data**

All the graphics features that are required to visualize engineering and scientific data are available in MATLAB. These include 2-D and 3-D plotting functions, 3-D volume visualization functions, tools for interactively creating plots, and the ability to export results to all popular graphics formats. You can customize plots by adding multiple axes; changing line colors and markers; adding annotation, Latex equations, and legends; and drawing shapes.

**2-D Plotting**

Visualizing vectors of data with 2-D plotting functions that create:

* Line, area, bar, and pie charts.
* Direction and velocity plots.
* Histograms.
* Polygons and surfaces.
* Scatter/bubble plots.
* Animations.

**3-D Plotting and Volume Visualization**

MATLAB provides functions for visualizing 2-D matrices, 3-D scalar, and 3-D vector data. You can use these functions to visualize and understand large, often complex, multidimensional data. Specifying plot characteristics, such as camera viewing angle, perspective, lighting effect, light source locations, and transparency.

3-D plotting functions include:

* Surface, contour, and mesh.
* Image plots.
* Cone, slice, stream, and isosurface.

**7.5.3 PERFORMING NUMERIC COMPUTATION**

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed by experts in mathematics, are the foundation of the MATLAB language. The core math functions use the LAPACK and BLAS linear algebra subroutine libraries and the FFTW Discrete Fourier Transform library. Because these processor-dependent libraries are optimized to the different platforms that MATLAB supports, they execute faster than the equivalent C or C++ code.

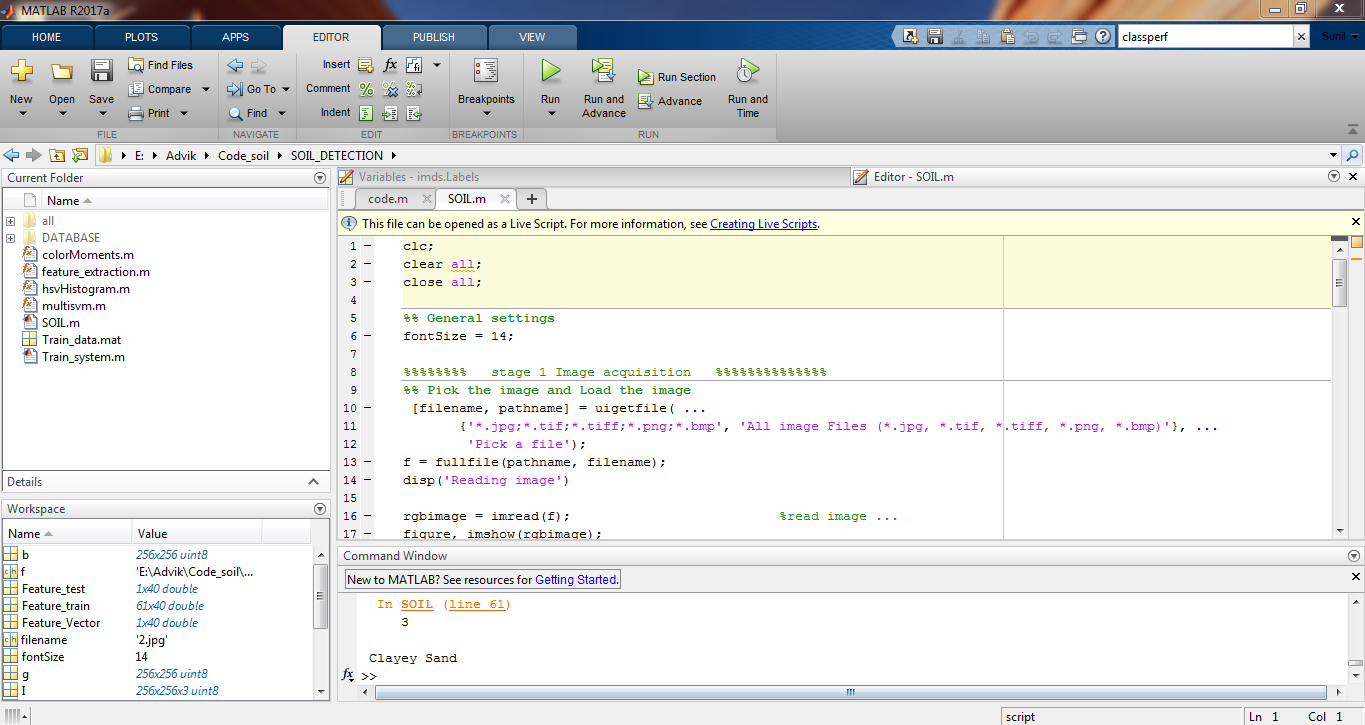
MATLAB provides the following types of functions for performing mathematical operations and analyzing data:

* Matrix manipulation and linear algebra.
* Polynomials and interpolation.
* Fourier analysis and filtering.
* Data analysis and statistics.
* Optimization and numerical integration.
* Ordinary differential equations (ODEs).
* Partial differential equations (PDEs).
* Sparse matrix operations.

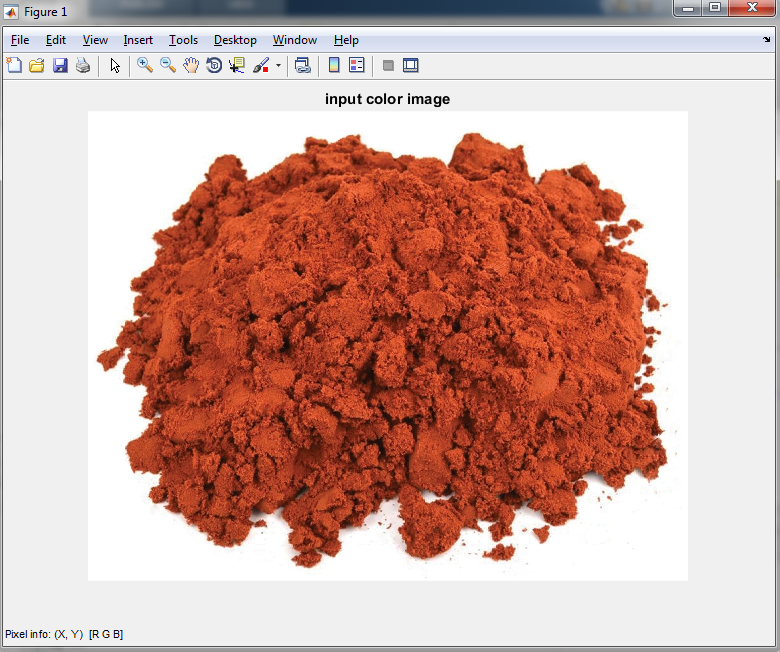
MATLAB can perform arithmetic on a wide range of data types, including doubles, singles, and integers.

# Chapter 8

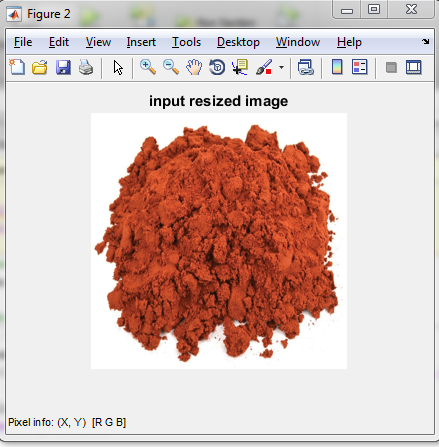
## RESULTS



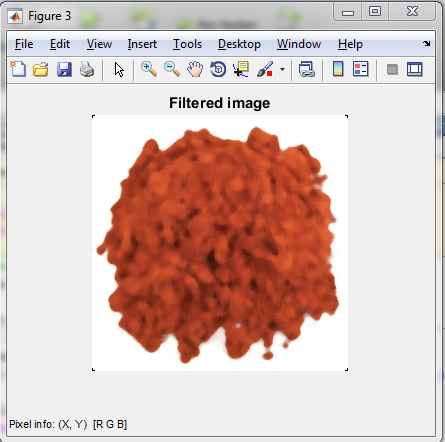
**FIG: MATLAB CODE**



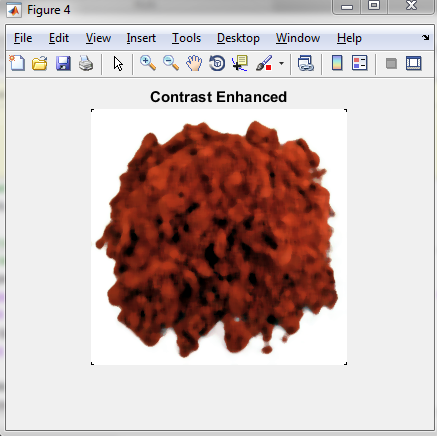
**FIG: INPUT COLOUR IMAGE**



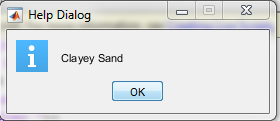
**FIG: INPUT RESIZED IMAGE .**



**FIG: MEDIAN FILTER IMAGE .**

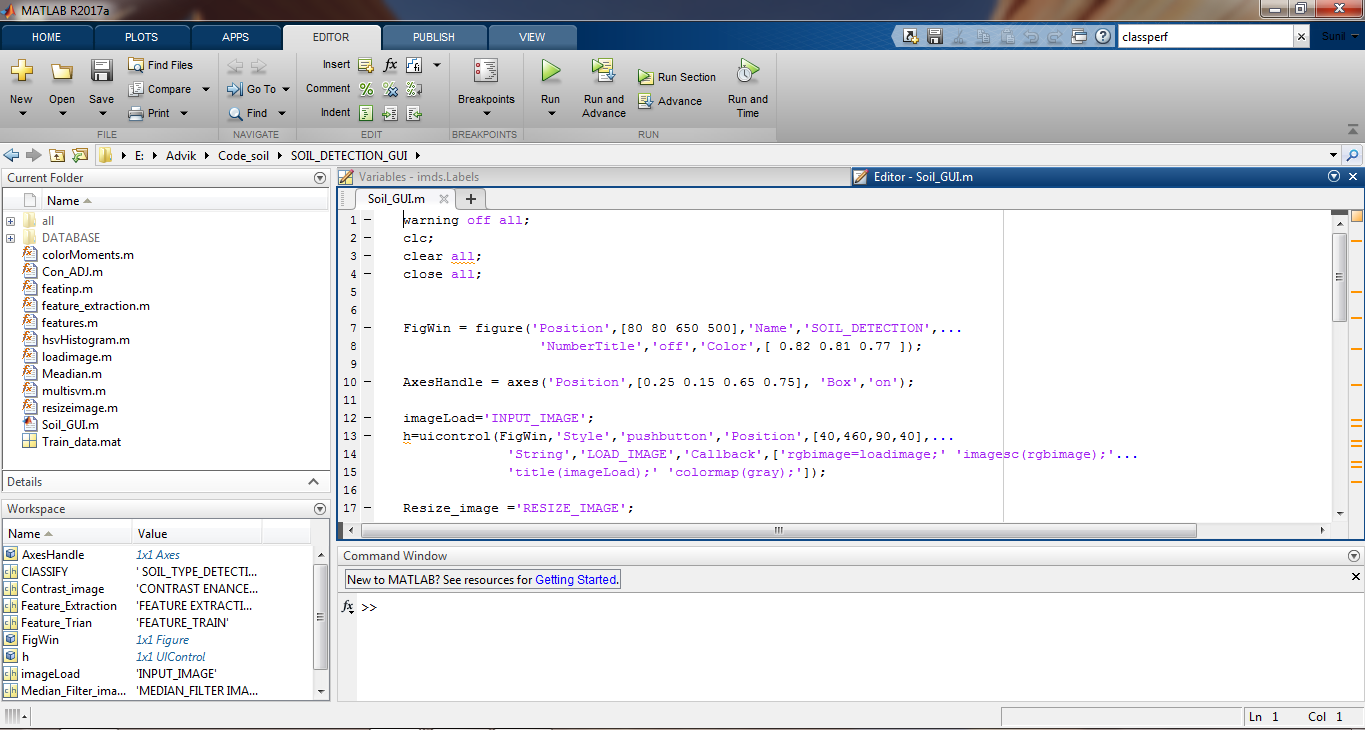


**FIG:CONTRAST ENHANCEMENT IMAGE .**

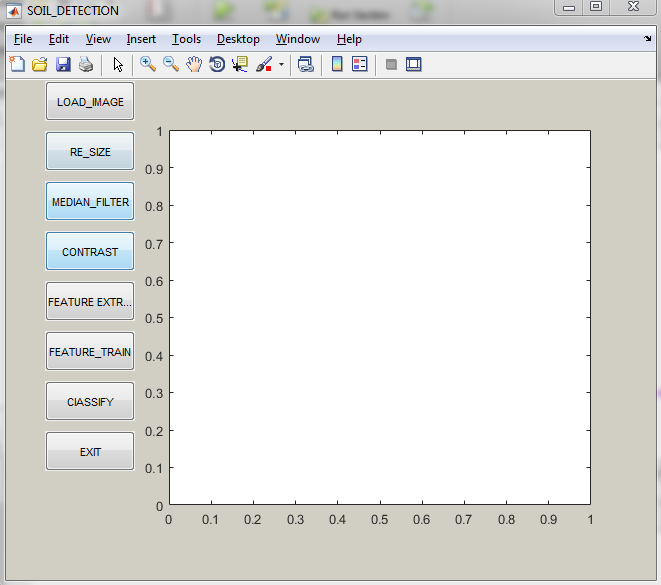


**FIG: ClASSIFICATION RESULT**

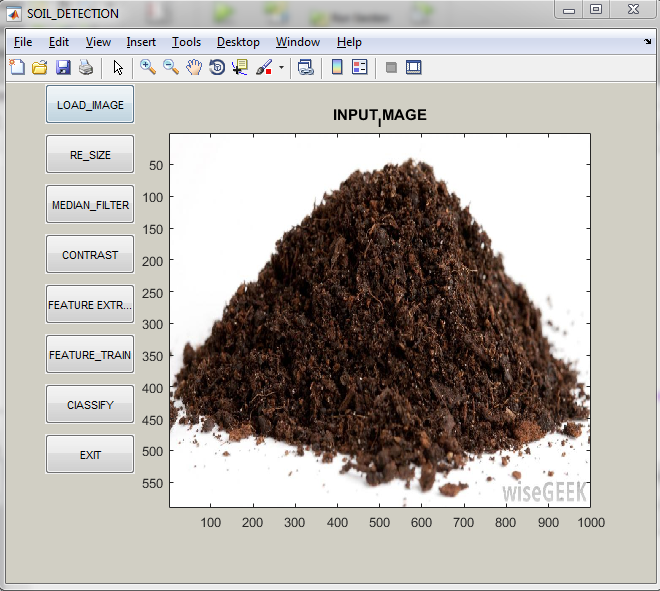
**MATLAB\_GUI RESULTS**

****

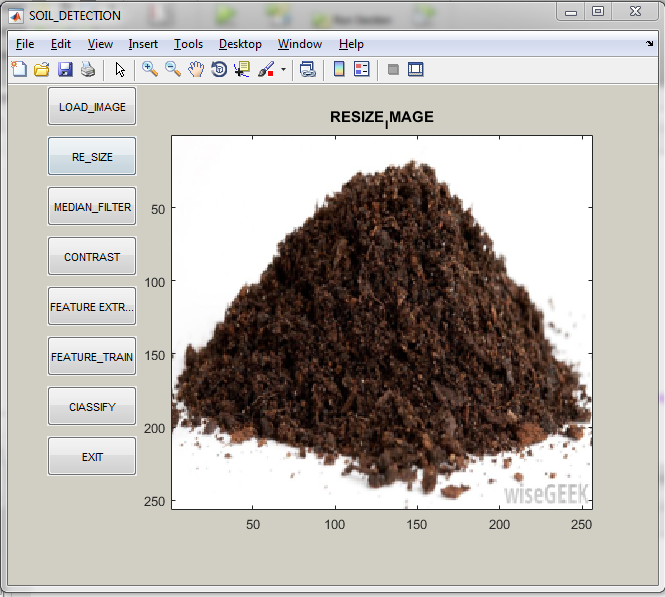
**FIG: MATLAB\_ GUI CODE**

****

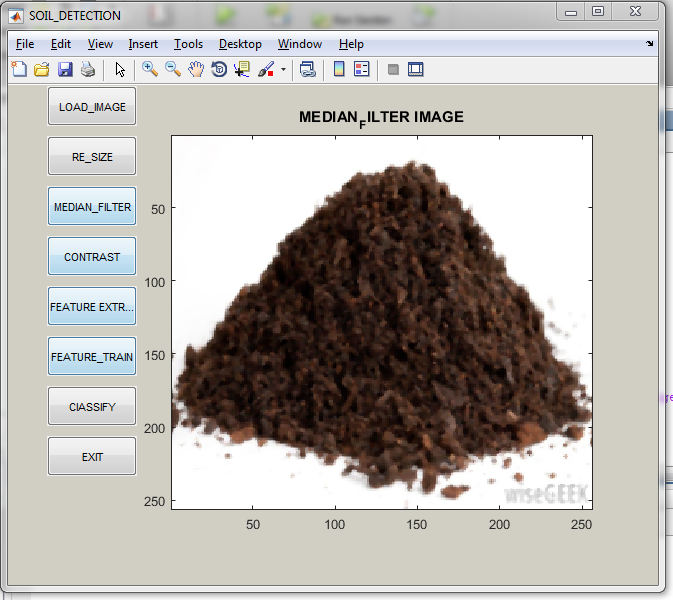
**FIG: GUI\_DISPLAY.**

****

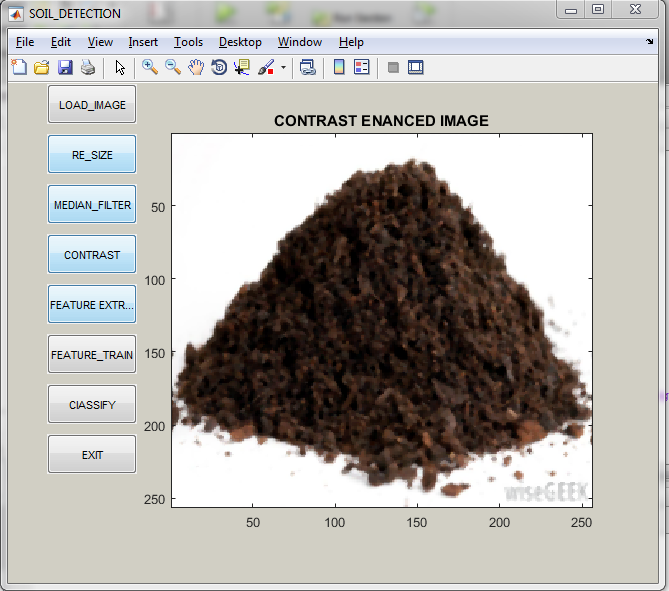
**FIG : LOAD INPUT IMAGE**

****

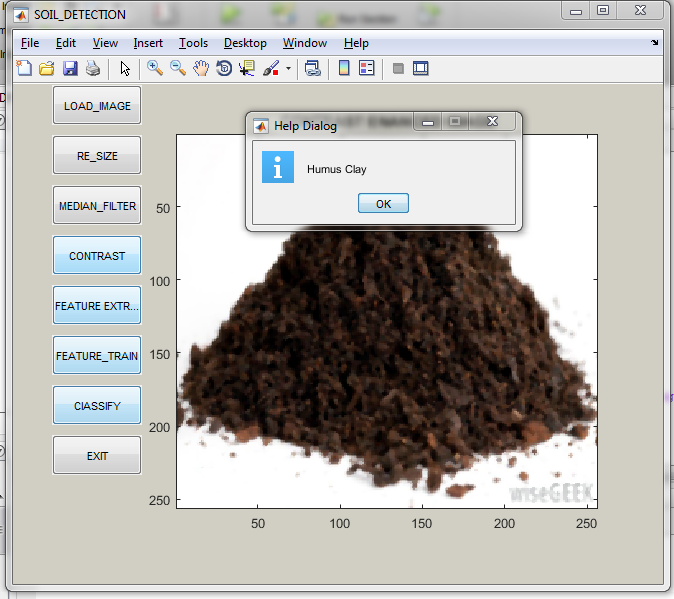
**FIG : RESIZE IMAGE.**

****

**FIG: MEDIAN\_FILTER\_IMAGE**

****

**FIG : CONTRAST\_ENANCED\_IMAGE.**

****

**FIG: CLASSIFICATION\_RESULT.**

# Chapter 9

## CONCLUSION AND FUTURE SCOPE

There are various methods for soil classification using different algorithms. But still soil classification with high accuracy and low cost is challenging task. So, keeping in view, proposed algorithm is useful to soil classification using SVM (Support Vector Machine).We can also use ANN method for classification. Here main objective is to classify soil sample picture with high accuracy and low cost.

Almost all misclassified objects are relayed near to the segment line. Near the segment boundary Measurements spotted as often noisy and thus can be decided that the enactment of classifiers was excellent. With more data and soil science domain-specific tricks, the potential for applying machine learning to soil property prediction would surely be maximized. It is able to achieve a 95% accuracy rate for classifying.

REFERENCES

\

[1]. V. Sudharsan and B. Yamuna “Support Vector Machine based Decoding Algorithm for BCH Codes” Journal of Telecommunication and Information Technology 2016.

[2]. Unmesha Sreeveni.U .B, Shiju Sathyadevan “ADBF Integratable Machine Learning Algorithms –Map reduce Implementation” Second International Symposium on computer vision and the Internet(VisionNet’15).

[3]. B. Bhattacharya, and D.P. Solomatine "An algorithm for clustering and classification of series data with constraint of contiguity", Proc. 3T"d nt. Conf: on Hybrid and Intelligent Systems, Melboume, Australia, 2003, pp. 489-498.

[4]. A.Coerts, Analysis of Static Cone Penetration Test Data for Subsurface Modelling - A Methodology (PhD Thesis),Utrecht University, The Netherlands, 1996.

[5]. L.F. Costa, and R.M. Cesar, Shape Analysis and Classification: Theory and Practice, Boca Raton, Florida: CRC Press, 2001.

[6]. Gordon, A.D. "A survey of constrained classification", Computational Statistics & Data Analysis, vol. 21, pp. 17-29, 1996.

[7]. S. Haykin, Neural Networks: A Comprehensive Foundation, New Jersey: Prentice Hall, 1999.

[8]. D.M. Hawkins, and D.F. Merriam, "Optimal zonation of digitized sequential data", Mathematical Geology, vol. 5, pp. 389-395, 1973.

[9]. G.P. Huijzer, Quantitative Penetrostratigraphic Classification (PhDThesis), Free University of Amsterdam, The Netherlands, 1992.

[10]. C.H. Juang, X.H. Huang, R.D. Holtz, and J.W. Chen, "Determining relative density of sands from CPT using fuzzy sets", J. of Geotechnical Engineering, vol. 122(1), pp. 1-6, 1996.

[11]. M.G. Kerzner, Image Processing in Well Log Analysis, Dordrecht,The Netherlands: Reidel Pub., 1986.

[12]. J. K. Kumar, M. Konno, and N. Yasuda,"Sub surface soil-geology interpolation using fizzy neural network", J. of Geotechnical and Geoenvironmental Engineering, ASCE, vol. 126(7), pp. 632-639, 2000.

[13]. Neural Machine, http:/ivww.data-machine.com/, 28.1.2005.

[14]. Neuro Soluions, http://www.nd.com/, 28.1.2005.

[15]. RHUL , Computer Learning Research Centre, Royal Holloway University of London, (http://www.clrc.rhul.ac.uk, 26/1/2005)

[16]. L.J. van Vliet, and P.W. Verbeeck, "Curvature and bending energy in digitised 2D and 3D images", in: K.A. Hogda, B. Braathen and K.Heia (Eds), Proc. 8" Scandinavian Confon Image Analysis, Norway,1993, vol. 2, pp. 1403-1410.

[17]. K. Wagstaff, Intelligent Clustering with Instance-Level Constraints(PhD thesis), Cornell University, USA, 2002.

[18]. R. Webster, "Optimally partitioning soil transects", Journal of Soil Science, vol. 29, pp. 388402, 1978.

[19]. H.J.T. Weerts, Complex Confining Layers, Utrecht University, The Netherlands, 1996.

[20]. I.H. Witten, and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufimann, 2000.

[21]. I.T. Young, and T.W. Calvert, "An analysis technique for biological shape", Information and Control, vol. 25, pp 357-370, 1974.

[22]. Z. Zhang, and M. T. Tumay, "Statistical to fuzzy approach toward CPT soil classification", J of Geotechnical and Geo environmental Engineering, vol. 125(3), pp. 179-186, 1999.