A Mini Project Report on

Automated Crop Statistics

Submitted to the Department of Computer Science & Engineering, GNITS in the partial fulfillment of the academic requirement for the award of B.Tech (CSE) under JNTUH

By

B. SAI TULASI - 16251A05C3

G. ALEKHYA - 16251A05D3

V. ANUSHA - 17255A0528

under the guidance of

Dr. D. V. Lalitha Parameswari

Sr. Asst. Professor

Department of Computer Science & Engineering
G. Narayanamma Institute of Technology & Science
(Autonomous) (for Women)
Shaikpet, Hyderabad- 500 104.

Affiliated to
Jawaharlal Nehru Technological University Hyderabad
Hyderabad – 500 085
Nov, 2019

Department of Computer Science & Engineering
G. Narayanamma Institute of Technology & Science
(Autonomous) (for Women)
Accredited by NBA & NAAC &Affiliated to JNTUH
Hyderabad-500104,T.S,INDIA

Certificate

This is to certify that the Project report on "**AUTOMATED CROP STATISTICS**" is a bonafide work carried out by B. Sai Tulasi - 16251A05C3, G.Alekhya - 16251A05D3, V. Anusha - 17255A0528 in the partial fulfillment for the award of B.Tech degree in Computer Science & Engineering, G. Narayanamma Institute of Technology & Science, Shaikpet, Hyderabad, affiliated to Jawaharlal Nehru Technological University, Hyderabad under our guidance and supervision.

The results embodied in the project work have not been submitted to any other University or Institute for the award of any degree or diploma.

Internal Guide
Dr. D. V. Lalitha Parameswari
Sr. Asst. Professor

Head of the Department
Dr. M. Seetha
Professor & Head
Department of CSE

External Examiner

Acknowledgements

We would like to express our sincere thanks to **Dr. K. Ramesh Reddy, Principal** , GNITS, for providing the working facilities in the college.

Our sincere thanks and gratitude to **Dr. M. Seetha, Professor and HOD**, Dept. of CSE, GNITS for all the timely support and valuable suggestions during the period of our miniproject.

We are extremely thankful to **Dr. D. V. Lalitha Parameswari, Sr. Asst. Prof, Mrs. P. Neelima Reddy, Asst. Prof** Dept. of CSE, GNITS, the miniproject coordinators **and Dr. N. Kalyani, Professor,** Dept. of CSE, GNITS for their encouragement and support throughout the project.

We are extremely thankful and indebted to our internal guide, **Dr. D. V. Lalitha Parameswari, Sr. Asst. Prof,** Dept. of CSE, GNITS for her constant guidance, continuous advice, encouragement and moral support throughout the miniproject.

Finally, we would also like to thank all the faculty and staff of CSE Department who helped us directly or indirectly, parents and friends for their cooperation in completing the miniproject work.

B. Sai Tulasi - 16251A05C3

G. Alekhya - 16251A05D3

V. Anusha - 17255A0528

ABSTRACT

Automated Crop Statistics is an Artificial Intelligence (AI) Project. With the advancement of Technology, one can find solution to any difficult problem. Thus came the idea of AI. The primary function of this project is to help the Government to plan before hand to overcome the problem of crisis in crop production. Often there are situations where there is a sudden rise in the price of vegetables or fruits due to less production of that particular crop. This brings the burden on normal people.

Automated Crop Stats, thus comes as a one stop solution to all such problems. It is useful in determining the important needs of a country. For instance, if crop area of Onion, is relatively less in an year, then the Government can plan for importing onions, when onions price is low in the world market. This project aims at analyzing the satellite images, and calculating the crop areas. This project has various modules. CAI(Crop Area Image) module is used to provide the satellite images as input to the system. AD(Area determination) module is used to analyze the image, and determine areas of different types of crops. RG(Report Generation) module is used to generate a complete report regarding the crop statistics of that particular satellite image. Calculation module is used to calculate the different crop statistic

Table of Contents

1. INTRODUCTION

Automated Crop Statistics, is useful in determining the import needs of a country. For instance, if the crop area of onion is relatively less in an year, then the government can plan for importing onions, when onions price is low in the world market. This project aims at analyzing the satellite images, and calculating the crop areas.

For instance, if crop area of Onion, is relatively less in an year, then government can plan for importing onions, when onions price is low in the world market. This project aims at analyzing the satellite images, and calculating the crop areas.
This Project uses PyQt tool to create the needed Graphical User Interfaces, PyUIC module to automatically generate the automated code and python image Processing libray(Pillow).

This project uses the pixel based segmentation algorithm to calculate the areas of different types of crops , and image cascading algorithm to identify different types of crops . The images of various types of crops are stored in an image file system. The file system will be stored at a specific location in windows directory structure. The location will be fed as in input to the system through a GUI screen. The GUI screens in this project are created by a python tool called PyQt.

Project Advantages:

The project is very useful to the agriculture department in tracking the statistics of different types of crops . The project is also useful to the government in controlling the prices, by timely importing those things whose crop area is relatively less. This project finally leads to the improvement of the quality of life for common man.

Technical advantages:

Using Python, which is chosen as the best programming language, by the Programming Community. More Functionality can be implemented with less no.of lines of code in Python. PyQt tool is used to create the Graphical User interfaces. All the Front end code is generated automatically by PyUIC.

1.1 Objectives

- This project entitled "Automated Crop Statistics" aims at developing a tool for accurately calculating the crop statistics of various crops of a particular region by processing their remote satellite maps/images.
- This tool is developed by using Python along with its layout toolkit PyQt & PyUIC.
- This project is very useful to the agriculture department in tracking the statistics of different types of crops.
- The project is also useful to the government in controlling the prices, by timely importing those things whose crop area is relatively less
- Thereby it finally leads to the improvement of the quality of life for common man.

1.2 Methodology

This Project uses PyQt tool to create the needed Graphical User Interfaces, PyUIC module to automatically generate the automated code and python image Processing library(Pillow). This project uses the pixel based image template matching algorithm to calculate the areas of different types of crops , and to identify different types of crops. Then a detailed report is generated on the statistics of various crops in that particular satellite image. The satellite images of various types of crops are stored in an current

working folder. The file names will be fed as in input to the system through a GUI screen. The GUI screens in this project are created by a python tool called PyQt.

PyQt is a Python binding of the cross-platform GUI toolkit Qt, implemented as a Python plug-in. PyQt is developed by the British firm Riverbank Computing. PyQt supports Microsoft Windows as well as various flavors of UNIX, including Linux and MacOS. PyQt implements different classes and methods including classes for accessing SQL databases (ODBC, MySQL, Oracle, SQLite), Scintilla-based rich text editor widget, data aware widgets that are automatically populated from a database, an XML parser and SVG support. Scalable Vector Graphics (SVG) is an XML-based vector image format for graphics with support for interactivity.

All the above mentioned features of PyQt, are extensively used in this project, to create the needed Graphical User Interfaces. PyUic tool is used to automatically generate the code for the Front end user interfaces created by PyQt. All the front end python code is automatically generated by this tool, by converting the user interface(.ui) files into .py files.

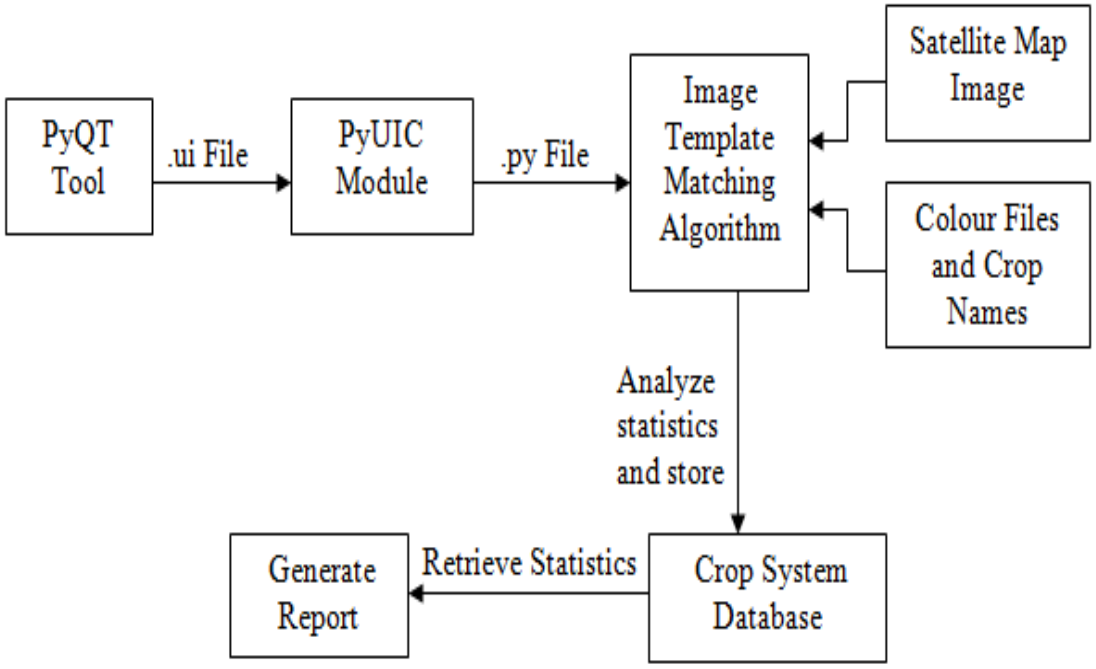


Fig 1.1 Block Diagram of the project

1.3 Organization of the project

This project has three modules. CAI(Crop Area Image) module is used to provide the satellite images of crops, as input to the system. AD(Area determination) module is used to analyze the image, and determine the areas of different crops in that particular satellite image. Report Generation module is used to generate the report regarding crop statistics of that particular satellite image.

Chapter-1 deals with Analysis of Automated Crop Type Statistics:

Analysis is going to give brief understanding about the Requirements needed and about Feasibility study.

Requirement analysis gives an overview about Hardware and Software requirements required for the implementation of the project.

Feasibility study is going to explain the Economic feasibility and Technical feasibility which confirms the strategy ,plan or design possible and tells whether the project is worth the investment

Chapter-2 gives Brief Understanding about Literature Survey:

Literature survey is going to give a overview of Crop Seasons in India
Categories of crops in India, Major crops in India which shows the various analyses and research made in the field of one's interest and the results already published, taking into account the various parameters of the project and the extent of the project.

Chapter-3 describes Automated Crop Statistics

This section proposes the **Architecture of the whole system** which is nothing but is structure of application and explains the layers are used to retrieve the data from the Database. This section also illustrates about the modules which high –level

description of a functional area, consisting of a group of processes describing the functionality of the module and a group of packages implementing the functionality

Chapter-4 demonstrates the Implementation of the proposed system

Implementation module is going to describe the technology and tools used, Namely PyQt tool and Python Qt designer . It also briefs about

- Languages used in the project are also explained Namely Python.
- Steps involved in installing required packages in python are described briefly.
- Database connectivity, Namely about SQlite
- UML Diagrams

Chapter-5 Shows the Results of the Implemented System

System tested results are explained in this section. System tested results: project is thoroughly tested by testing the each and every text box and push buttons of the GUI Screens, and verifying the corresponding results in the Data Base.

2. DESKTOP STUDY ON CROPS

`MarcG´orriz Blanch et .al, in their research, dealt with the application of image segmentation to compute the size of images. Segmentation gives accurate results, when the size of the segment is less. In this project, pixel based segmentation, of different types of crops in the image, is used to compute the areas of different types of crops accurately.

A crop is a large area dominated by plants. Hundreds of more precise definitions of crops are used throughout the world, incorporating factors such as plant density, plants average height, land use, legal standing and ecological function.

Crop production forecasts/estimates are generally portrayed as the product of two components: area (to be) harvested and (expected) yield per unit area. The accurate forecasting/estimation of both harvested area and yield are equally important in ensuring the accurate determination of their product. Although the yield part of the equation gets most of the attention, there are many complexities to the estimation of area that might not be readily apparent

Certainly the most significant challenge in estimating crop production is in doing so early in the season. Area estimation can create problems especially in countries prone to drought or flooding problems. However, very severe events in any country can take planted area out of production, since it either can't be harvested or it's economically infeasible to do so. These unforeseen, one-time conditions may cause a need for a special, one-time measurement of area.

Throughout the years there have been many approaches to crop area estimation. Agricultural statistics are collected both by censuses, which require enumeration of the total population of interest, and by samples requiring enumeration of only a small part of the population. Early types involved expert opinion of voluntary crop reporters by local area, such as villages, communes, or parishes. This is one of the cheapest forms, but not always accurate

Statistical methods have made clear that the enumeration of small samples can greatly reduce the cost of the collection of agricultural statistics while increasing their accuracy. A well designed sample, for which the data are carefully collected, can provide much cheaper statistics than a census and provide more timely information on current condition

Area sampling evolved as another approach; area defined populations are complete and their estimates are unbiased. However they have their own set of costs and problems, such as poor small area estimates for minor crops and the initial cost of creating the area sampling frame. Combining area and list frames, known as multiple frame sampling, has some very good qualities and solves some of the problems with each of them individually.

Other sources of crop area estimation may be derived from administrative sources or as by-products of administrative data. For example, farm program sign-ups or mail surveys may ask for changes from last year, leading to ratio estimates based on the prior year's numbers. Processors of raw crop inputs (such as cotton gins) provide information. Legislative actions limiting area planted for specific crops may be present

Remote sensing has evolved from sporadic aerial over-flights to frequent repeat high resolution coverage and from black and white film coverage to multispectral digital scanners. Computer processing has progressed from just being able to add up survey sampling information to digitally classifying points on the ground to crop types and other land covers. Geographic Information Systems (GIS's) have evolved from marks hand drawn on photo mosaics to devices showing digital locations down to centimeters. These technology advances have greatly increased the ease of area frame construction, maintenance, and sampling; decreased associated costs and have reduced sampling variation both through better stratification and later through regression estimates

Ground Data Only Systems for Crop Area Estimation:

Ground surveys have long been the underpinning of estimates of crop area and production. The documented sampling and estimation strategies for crop forecasting and estimation is still very relevant today. There is a rich literature history of systems for crop area estimation based solely upon ground gathered data systems. The focus of this section will be to discuss those methods further.[3]

There are Censuses of Agriculture conducted around the world. The results, based upon farmer reported data, are quite thorough and detailed. The results are usually well after harvest though and not timely for a crop production forecasting and estimation system. They are a great source of small area estimates and can serve as a bench mark for ratio estimates in periods between censuses.

Role of Remote Sensing in Crop Area Estimation:

Area estimation throughout the crop season is typically accomplished through ground surveys or ground surveys supplemented with remotely sensed data. The remote sensing imagery is generally used for stratification and is often used directly in estimation as well. Remote sensing techniques have become very popular in area estimation over the past few decades, as the technology and methodologies have matured. An excellent document on the issues of area estimation in general as well as those from the earth observation (EO) perspective was published by the Group on Earth Observations (GEO) following a June 2008 conference on the topic [3]

Remote sensing has a long history of use in crop acreage estimation and assessment dating back to the 70s. With technology improvements in sensor quality and availability and processing advances, it has become an even stronger player in countries' efforts to estimate crop area and production. Issues, especially in regard to timing, efficiency, and assurances of continuing availability remain with its use. According to GEO Ag Task 07 03 [2], the timing or schedule of crop area estimation or early estimation depends on the following elements

1) The number of days after sowing a crop can be detected by a remote sensor

2) The spatial variability in sowing practices of the region

3) The crop calendars of competing crops

4) The characteristics of remote sensors (revisiting time)

5) The date in which the crop can be reliably recognized in the field

6) The time needed for the ground survey, and

7) The time needed for ground data processing.

2.1 Crop Seasons in India

India is the top producer of many crops in the world. There can be many ways to divide the types of crops (based on area, season, economic value etc.). Based on seasons, the crops in India are divided into three types; Rabi, Kharif and Zaid.[4]

Kharif Crops

- Sown in June-July when rains first begin (Monsoon crop).
- Harvested in September-October.
- Requires lot of water and hot weather to grow
- These crops are dependent on the quantity of rain water as well its timing. Too much, too little or rain at the wrong time may lay waste to the whole year's efforts.
- Sowing of seeds begins in the rainy season around July
- Kharif crops stand in contrast to the rabi crops, which are cultivated during the dry season.
- **Common karif Crops :** Rice is the most important kharif crop of India. It is grown in rain fed areas with hot and humid climates, especially the eastern and southern parts of India. Rice requires a temperature of 16–20 °C (61–68 °F) during the growing season and 18–32 °C (64–90 °F) during ripening[4]. It needs

rainfall from 150–200 centimetres (59–79 in) and needs a flooded field during the growth period.

Cereals : Bajra, Jowar, Maize (corn), Millet, Rice (paddy and deepwater)

Fruits : Muskmelon, Sugarcane, Watermelon, Orange

Seed plants : Arhar (tur), Black gram (urad), Cotton, Cowpea (chavala), Green gram (moong), Groundnut, Guar, Linseed (flax), Moth bean, Mung bean, Sesame (til), Urad bean

Vegetables : Bitter gourd (karela)Bottle, gourd Brinjal, Chili, lady fingers, Sponge gourd, Tinda, Tomato, Turmeric, French bea

Rabi Crops

- Sown in October-November
- Harvested in April-May.
- Requires warm climate for germination of seeds and maturation and cold climate for the growth.
- Require cold and relatively dry conditions to grow
- The seeds are sown at the beginning of autumn, which results in a spring harvest.
- Though, these crops are grown in large parts of India, states from the north and north-western parts such as Punjab, Haryana, Himachal Pradesh, Jammu and Kashmir, Uttarakhand and Uttar Pradesh are important for the production of wheat and other rabi crops.
- Availability of precipitation during winter months due to the western temperate cyclones helps in the success of these crops.
- However, the success of the green revolution in Punjab, Haryana, western Uttar Pradesh and parts of Rajasthan has also been an important factor in the growth of the above-mentioned rabi crops.[4]
- **Example:** Wheat, Oat, Gram, Pea, Barley, Potato, Tomato, Onion, Oil seeds (like Rapeseed, Sunflower, Sesame, Mustard) etc.

Zaid Crops

- Grown between March-June between Rabi and Kharif crop seasons.
- Early maturing crops.
- These crops are grown on irrigated lands and do not have to wait for monsoons.
- **Example:** Cucumber, Bitter Gourd, Pumpkin, Watermelon, Muskmelon, Moong Dal etc.

2.2 Categories of Crops in India

The major crops can all be divided into four main categories depending on their usage.

1. Food Crops (Wheat, Maize, Rice, Millets and Pulses etc.)
2. Cash Crops (Sugarcane, Tobacco, Cotton, Jute and Oilseeds etc.)
3. Plantation Crops (Coffee, Coconut, Tea, and Rubber etc.)
4. Horticulture crops (Fruits and Vegetables)

2.3 Major Crops in India

Now let us look at the major crops in India in detail.

Rice

Rice is a tropical crop that can be grown almost throughout the year. It depends on atmospheric moisture and rainfall for irrigation. India is the 2nd largest producer of rice in the world. India has largest area in world under rice cultivation. Productivity is low compared to wheat because Green Revolution primarily boosted wheat production in India. The traditional rice fields are known as paddy fields and require to be flooded with 10-12 cm deep water in the early stages.[4]

- **Soiltype :** Clay/ Loamy
- **Major Producers :** West Bengal, Uttar Pradesh, Andhra Pradesh, Punjab, Bihar, Orissa, Chhattisgarh, Assam, Tamil Nadu, Haryana
- **Highest Producer :** West Bengal

- **Highest per Hectare Yield :** Punjab
- **Research Centres :** Cuttack, Odisha
- **Highest Producing Country :** China

Wheat

It is the 2nd most important food crop in India. It is a Rabi crop. India stands second in production of wheat worldwide. It is more flexible in terms of climactic and other conditions of growth.[4]

- **Soil Type :** Clay loam, Sandy loam
- **Major Producers :** Uttar Pradesh, Punjab, Madhya Pradesh, Haryana, Rajasthan, Bihar, Gujarat, Maharashtra, West Bengal, Uttarakhand
- **Highest Producing State :** Uttar Pradesh
- **Highest per Hectare Yield :** Punjab
- **Research Centres :** Karnal, Haryana
- **Highest Producing Country :** China

Cotton

Cotton is a tropical and subtropical Kharif crop. It is a fibre crop and is known as 'White gold'. India ranks 3rd in the production of cotton worldwide. It is a dry crop but roots need timely supply of water at maturity.

- **Soil Type :** Black soil (Highly water retentive soil)
- **Major Producers :** Gujarat, Maharashtra, Andhra Pradesh, Haryana, Madhya Pradesh, Punjab, Rajasthan, Karnataka, Tamil Nadu, Orissa
- **Leading Producer :** Gujarat (According to the 2015 report)
- **Research Centres :** Nagpur, Maharashtra
- **Highest Producing Country :** China

Jute

Jute is a tropical plant that requires hot and humid climate. It is one of the most important natural fibres in terms of cultivation and usage. Almost 85% of the world's jute is cultivated in the Ganges Delta.[4]

- **Soil Type :** Sandy and Clay Loam
- **Major Producers :** West Bengal, Bihar, Assam, Andhra Pradesh, Orissa, Meghalaya, Nagaland, Tripura, Uttar Pradesh
- **Largest Producer :** West Bengal
- **Highest per Hectare Yield :** West Bengal
- **Research Centres :** Kolkata & Nilgunj, West Bengal
- **Highest Producing Country :** India (but highest exporter is Bangladesh)

3. ANALYSIS OF AUTOMATED CROP STATISTICS

3.1 Requirement Analysis

3.1.1 Hardware Requirements

- It requires a minimum of 2.16 GHz processor.
- It requires a minimum of 4 GB RAM.
- It requires 64-bit architecture.
- It requires a minimum storage of 500GB.

3.1.2 Software Requirements

- It requires a 64-bit Windows Operating System.
- Python Qt Designer for designing user interface.
- SQLite server for storing database Entities.
- PyUIC for converting the layout designed user interface (UI) to python code.
- Python language for coding.

3.2 Feasibility analysis

As the name implies, a feasibility study is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment—in some cases, a project may not be doable. There can be many reasons for this, including requiring too many resources, which not only prevents those resources from performing other tasks but also may cost more than an organization would earn back by taking on a project that isn't profitable.

3.2.1 Economical Feasibility

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision makers determine the positive economic benefits to the organization that the proposed project will provide. This project is economically feasible because this project uses “WINDOWS”, “PYTHON”, “PYQT” designer tool and “PYUIC” which are all available as an open source.

3.2.2 Technical Feasibility

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity. Technical feasibility also involves evaluation of the hardware, software, and other technology requirements of the proposed system. A prototype of the tool was developed to verify the technical feasibility. The prototype is working successfully and hence the project is feasible.

4. AUTOMATED CROP STATISTICS

4.1. Architecture

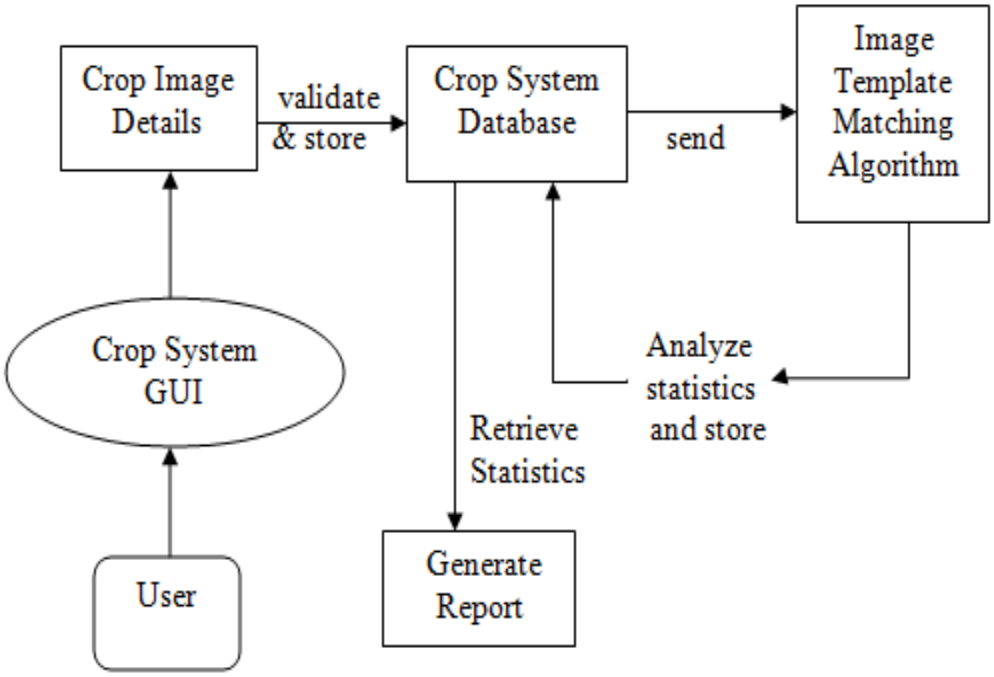


Fig 4.1 : Flow Diagram of the project

In the above fig 4.1 i.e., architecture diagram one can see that first the user have to log into the system with the provided credentials and then the system will validate the entered details. On successful login the user is allowed to enter the crop image details and they are stored in the database. Crop areas of that particular crop image are calculated using the algorithm and are stored in the database. These details are used to generate a detailed report on crop statistics

4.2. Module description

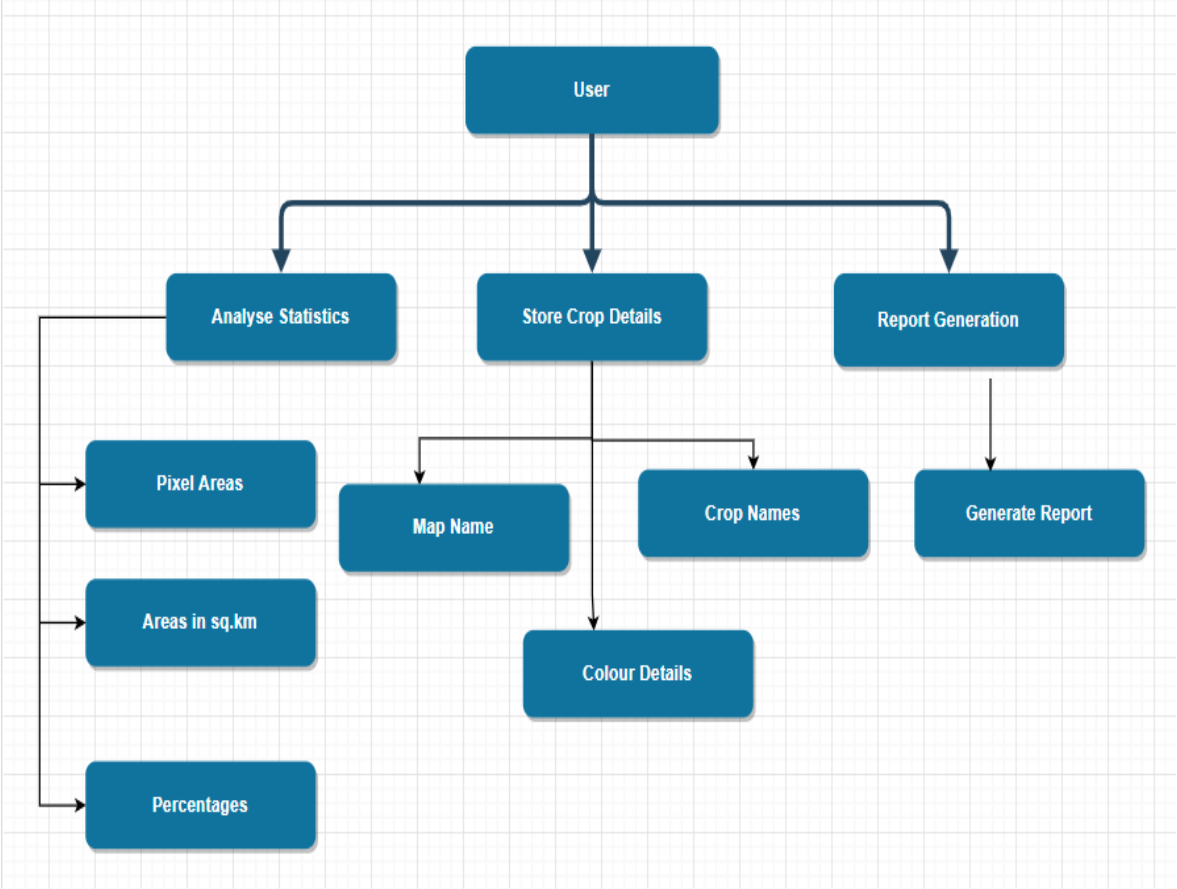


Fig 4.2 Modules of the project

Fig 4.2 depicts the details like name of the map image, names of the crop and names of the color files that are to be provided as input to the system, using the corresponding user interfaces.

This project has three modules :

Crop Details module is used to provide the satellite images of crops, image name, crop names and colour file details as input to the system.

Analyse Statistics module is used to analyze the image, and determine the areas of different crops in that particular satellite image.

Report Generation module is used to generate a detailed report regarding crop statistics of that particular satellite image.

4.3. Algorithm used

Algorithm : Image Template Matching Algorithm

Template Matching is a high-level machine vision technique that allows to identify the parts of an image (or multiple images) that match the given image pattern. It can be used in manufacturing as a part of quality control, a way to navigate a mobile robot, or as a way to detect edges in images. In this one can see all various methodologies which is used for implementing Template Matching. Template Matching is a technique for finding areas of an image that match (are similar) to a template image (patch).[1] It requires two primary components :

Source image (I): The image in which one expect to find a match to the template image. **Template image (T):** The patch image which will be compared to the template image.

The goal is to detect best technique for the highest matching area. Template matching is one of the areas of profound interests in recent times [2]. It has turned out to be a revolution in the field of computer vision. Template matching provides a new dimension into the image- processing capabilities, although there have been many attempts to resolve different issues in this field. Template Matching is a high-level machine vision technique that allows to identify the parts of an image (or multiple images) that match the given image pattern. Advanced template matching algorithms allow finding the pattern occurrences regardless of their orientation and local brightness. It is basically a method for searching and finding the location of a template image in a larger image.

- The idea here is to find identical regions of an image that match a template one has provided, giving a threshold
- The threshold depends on the accuracy with which one want to detect the template in the source image.

- In cases where almost identical templates are to be searched, the threshold should be set high.(t>=0.8)

How Template Matching Works?

- The template image simply slides over the input image (as in 2D convolution)
- The template and patch of input image under the template image are compared.
- The result obtained is compared with the threshold.
- If the result is greater than threshold, the portion will be marked as detected.
- In the function cv2.matchTemplate (img_gray, template, cv2. TM_ CCOEFF_ NORMED) the first parameter is the main image, second parameter is the template to be matched and third parameter is the method used for matching.

Template Matching Approaches

The choice of matching depends on the nature of the image and the problem to be solved. General classifications of template or image matching approaches are: Template or Area based approaches and Feature-based approaches.

a) Featured-based approach:

Featured-based approach is well suited when both reference and template images had more correspondence with respect to features and control points. Features include points, curves, or a surface model that have to be matched. Here, the aim is to locate the pair wise connection between reference and template using their spatial relations or descriptors of features. Subcategories of the above approach are spatial relations, invariant descriptors, pyramids and wavelets and relaxation methods.

b) Area-based approach:

Area-based methods are sometimes called as correlation-like methods or template matching methods. Fonseca and Manjunath (1996) [3], which is the combination of feature detection and feature matching Motion tracking and occlusion handling: For the templates which may not provide a direct match, then Eigen spaces are used, which

gives the detail of matching image under various conditions, as illumination, color contrast or acceptable matching poses.

Limitation

Template matching techniques applicability is limited mostly by the available computational power, as the identification of big image patterns is time-consuming.

1. Pattern occurrences have to preserve the orientation of the reference pattern image(template)
2. As a result, it does not work for rotated or scaled versions of the template as a change in shape/size/shear etc. of object with respect to template will give a false match.
3. The method is inefficient when calculating the pattern correlation image for medium to large images as the process is time consuming.

To avoid the issue caused by the different sizes of the template and original image, one can use multiscaling. In case where, just because the dimensions of the template do not match the dimensions of the region in the image you want to match, does not mean that you cannot apply template matching.

Multiscaling mechanism in Template Matching

The process of Multi scaling is as follows:

1. Loop over the input image at multiple scales (i.e. make the input image progressively smaller and smaller).
2. Apply template matching using cv2.matchTemplate and keep track of the match with the largest correlation coefficient (along with the x, y-coordinates of the region with the largest correlation coefficient).
3. After looping over all scales, take the region with the largest correlation coefficient and use that as the "matched" region.

5. IMPLEMENTATION

5.1. Description of the technologies used

This Project uses PyQt tool to create the needed Graphical User Interfaces, PyUIC module to automatically generate the automated code and python image Processing library(Pillow). PyQt is a Python binding of the cross-platform GUI toolkit Qt, implemented as a Python plug-in. PyQt is developed by the British firm Riverbank Computing. PyQt supports Microsoft Windows as well as various flavors of UNIX, including Linux and MacOS. PyQt implements different classes and methods including classes for accessing SQL databases (ODBC, MySQL, Oracle, SQLite)

PyUIC tool is used to automatically generate the code for the Front end user interfaces created by PyQt. All the front end python code is automatically generated by this tool, by converting the user interface(.ui) files into .py files.

5.1.1. Python Qt Designer

The PyQt installer comes with a GUI builder tool called Qt Designer. Using its simple drag and drop interface, a GUI interface can be quickly built without having to write the code. It is however, not an IDE such as Visual Studio. Hence, Qt Designer does not have the facility to debug and build the application. Creation of a GUI interface using Qt Designer starts with choosing a top-level window for the application. You can then drag and drop required widgets from the widget box on the left pane. You can also assign value to properties of widget laid on the form. The designed form is saved as demo.ui. This ui file contains XML representation of widgets and their properties in the design. This design is translated into Python equivalent by using pyuic4 command line utility. This utility is a wrapper for ui module.

5.1.2. Languages (PYTHON)

Python was conceived in the late 1980s, and its implementation began in December 1989 by Guido van Rossum at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to the ABC language (itself inspired by SETL) capable of exception handling and interfacing.

Python is an interpreted high-level programming language for general-purpose programming. Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python interpreters are available for many operating systems. Python is an open source software and has a community-based development model, as do nearly all of its variant implementations. Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by meta programming and meta objects (magic methods)). Many other paradigms are supported via extensions, including design by contract and logic programming.

Python uses dynamic typing, and a combination of reference counting and a cycle detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Steps for installing required packages in Python :

1) Install Python by down loading it from the following site.

https://www.python.org/downloads/release/python-370/

Download Windows x86-64 executable installer...if the laptop is a 64-bit machine.

During installation, Check the radio box(with a tick mark) corresponding to Add path...

Note: Anaconda's python is not enough. You should have a separate Python.

2) After successful installation of python, open the command prompt(CMD), as administrator, and give the command **python -V**. If python version is displayed, then the installation is successful. If the installation is not successful, then skip the following 3,4,5,6 commands.

3) Execute the command from command prompt(CMD): **pip3 install wheel** you need to type 'y', if the system asks you to choose y/n. If the above command don't work, then use: **pip install wheel**

4) Execute the command from command prompt(CMD): **pip3 install pyqt5**. If the above command don't work, then use: **pip install pyqt5**

5) Execute the command from command prompt(CMD): **pip3 install pillow**. If the above command don't work, then use: **pip install pillow**

6) Execute the command from command prompt(CMD): **pip3 install opencv-python**. If the above command don't work, then use: **pip install opencv-python**

7) Installing Anaconda : Install Anaconda by downloading it from:

https://www.anaconda.com/download/

5.1.3 Database Connectivity

- SQLite is an open-source database management system, commonly installed as part of the popular LAMP (Linux, Apache, MySQL, PHP/Python/Perl) stack. It uses a relational database and SQL (Structured Query Language) to manage its data.
- The short version of the installation is simple:

Steps for installing SQLite

On Windows 8.1, only the latest version of SQLite is included in the APT package repository by default. Install sqlite by downloading it from the following site:

https://www.sqlite.org/download.html

5.2 UML Diagrams

5.2.1. Use Case Diagram :

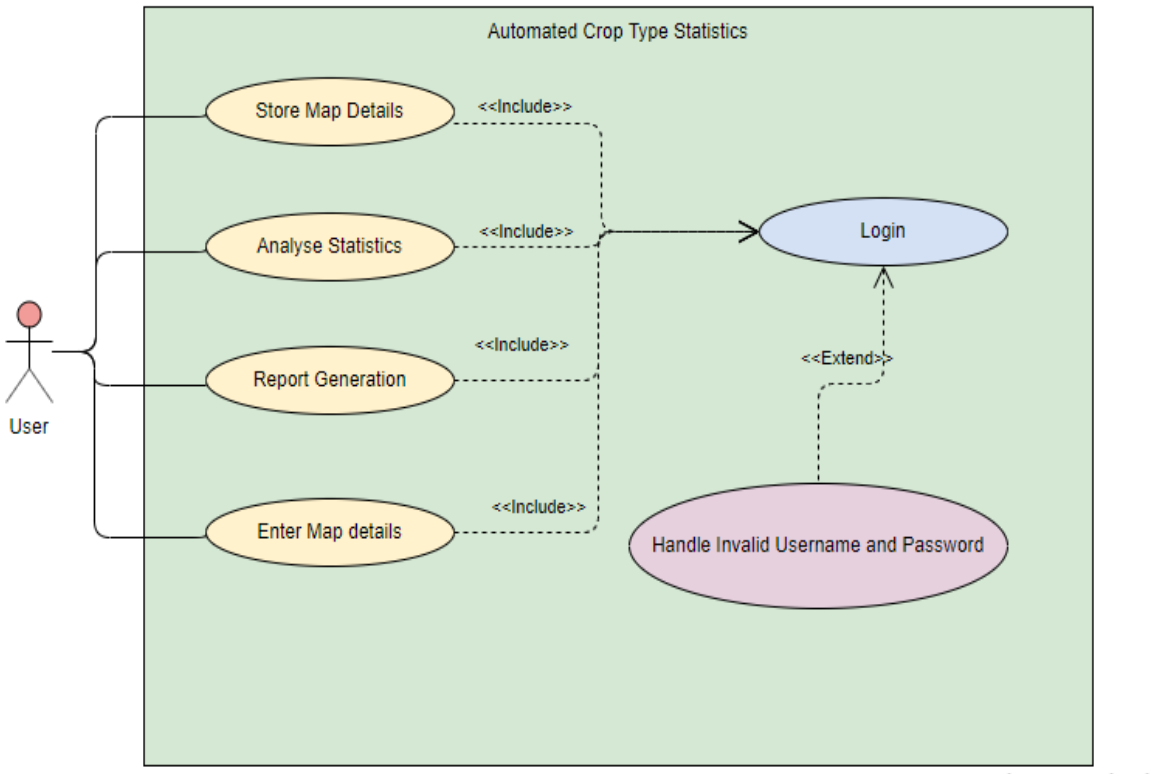


Fig 5.1 : Use Case Diagram

Description of Use case diagram

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

As one can see the user is interacting with system by a UI through which the user can perform above mentioned operations like entering the BGR Code details, scale of the map, name of the image map & Color files details

5.2.2. Sequence diagram :

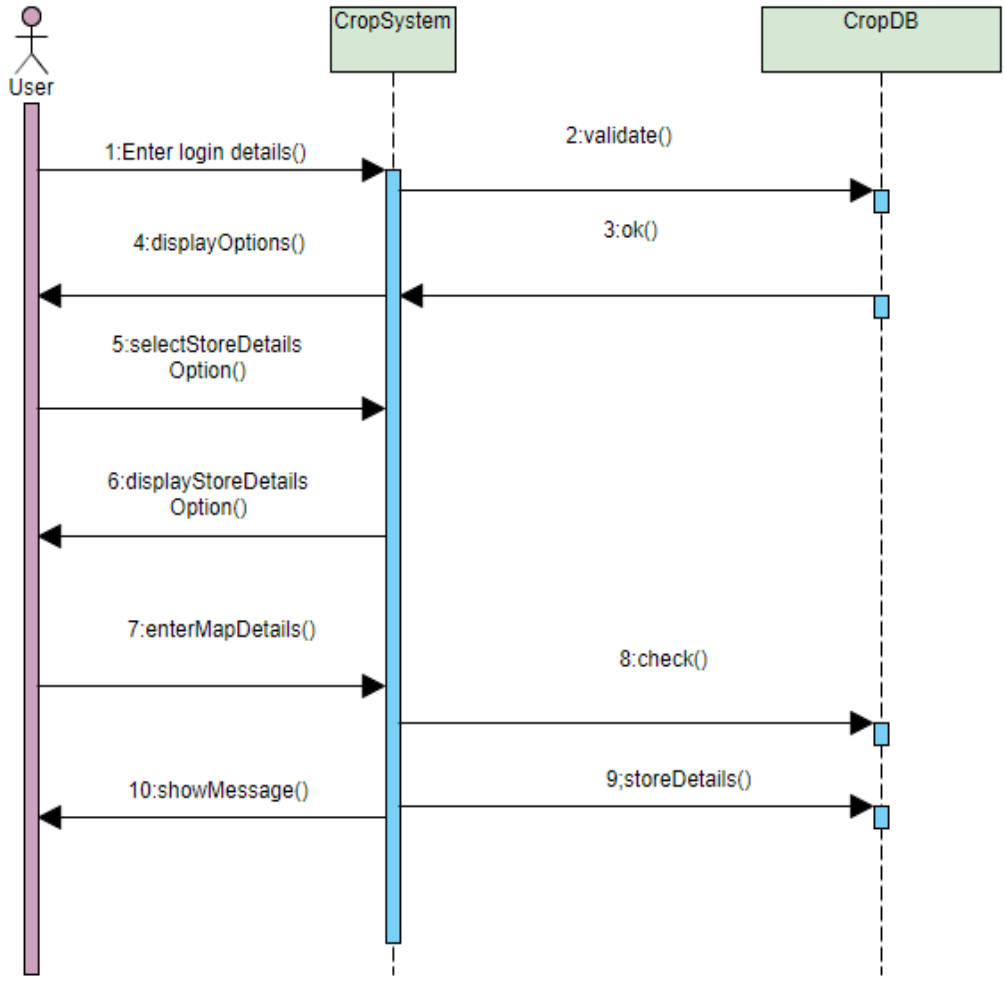


Fig 5.2 : Sequence Diagram for Store Details Option

Description of sequence diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence.

Fig 5.2 depicts the step by step procedure one have to follow in sequence: Enter the needed details as shown in the above figure, Provide the map details as Input to store them in the database for future purpose

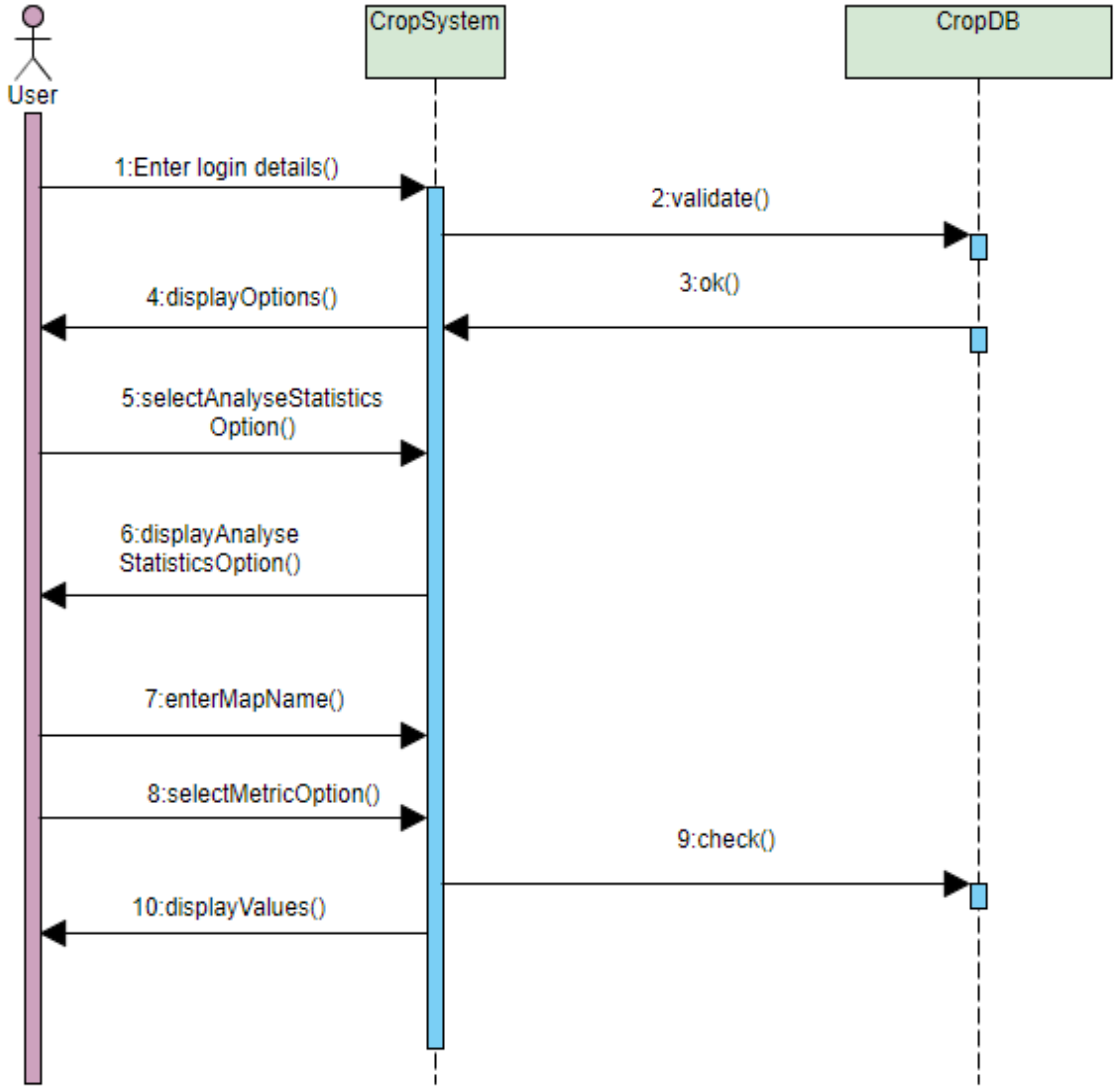


Fig 5.3 : Sequence Diagram to Analyze Statistics

Description of sequence diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence.

Fig 5.3 depicts the step by step procedure one have to follow in sequence: Enter the needed details as shown in the above figure, Provide the map name as input to analyze the image, Identify & Calculate the areas.

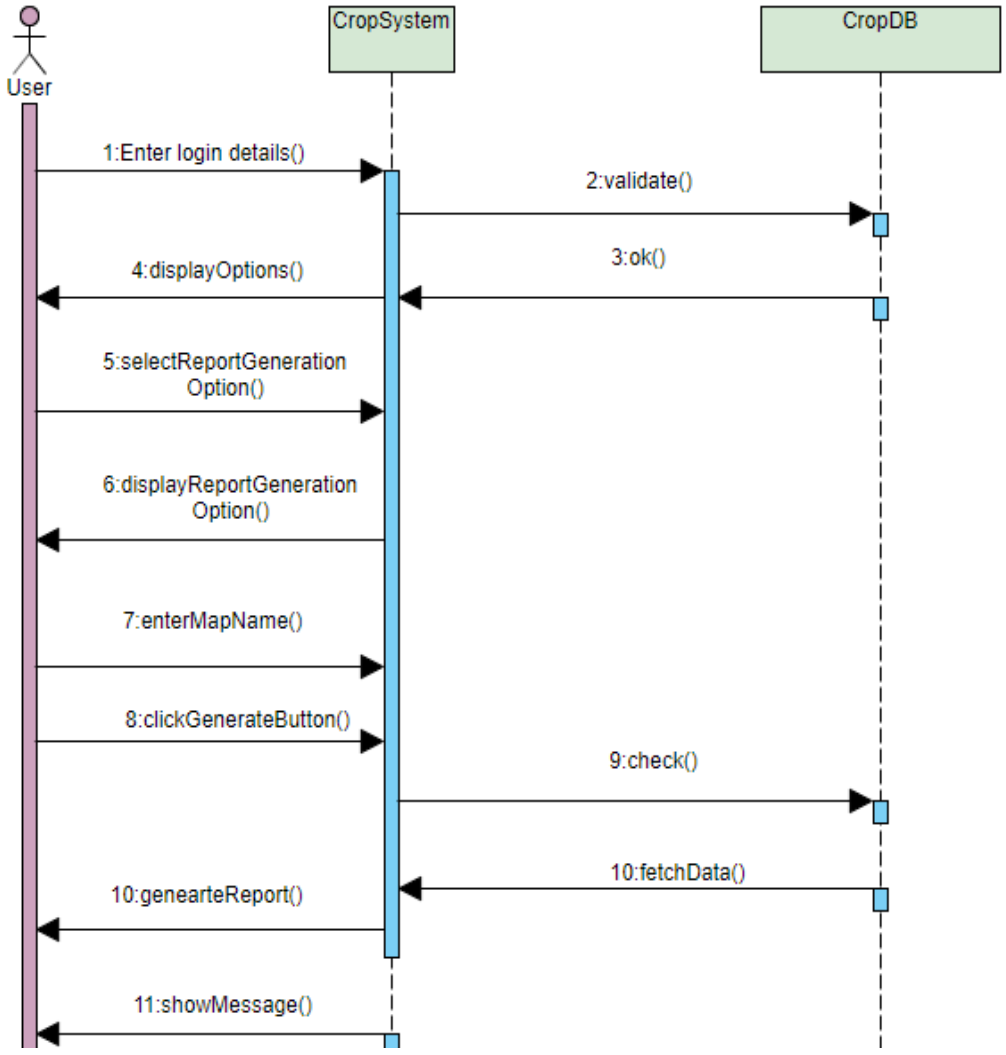


Fig 5.4 : Sequence Diagram to Generate Report

Description of sequence diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence.

Fig 5.4 describes the step by step procedure one have to follow in sequence: Enter the needed details as shown in the above figure, Provide the map name as input to fetch details from database and generate a report showing the crop statistics

6. RESULTS

6.1. System Testing Results

The project is thoroughly tested by testing the each and every text box and push buttons of the GUI Screens, and verifying the corresponding results in the Data Base.

Fig 6.1: Login Page

Fig 6.1 allows the user to login to the system with the given credentials i.e., Username and Password. These credentials are tested against those stored in that of database. Based on the result the user is allowed to proceed with the application. If the credentials are wrong the user is displayed with the alert message regarding incorrect username or password

Fig 6.2 : Alert Messages

Fig 6.2 i.e., Alert Messages are displayed to the user regarding the navigation of screens in the application and also regarding the successful or unsuccessful login. Alert messages to warn against the entry fields are also displayed to the user so that only valid and correct data is stored in the database.

Fig 6.3: Home Page

Upon successful login the user is navigated to fig 6.3 i.e., home page where the user is provided with various options such as to store crop details in the database, analyse the statistics of various crops in the provided satellite image and to generate a detailed report on the statistics of various crops in that particular satellite image.

Fig 6.4 Screen for Storing details in database

User when chooses the store details option presented to him in the home page, he/she will be navigated to fig 6.4 i.e., file locations page where the user is allowed to store the details related to crop image such as map name, different colours of crops and their names in the satellite image provided. Entered details are validate and are stored in the database which are used in the future to calculate areas of various crops.

Fig 6.5 Report Generation Page

User when chooses the report generation option presented to him in the home page, he/she will be navigated to fig 6.5 i.e., report generation page where the user is allowed to enter the name of the map file and to generate report. Based on the map name provided the data is fetched from the database and the detailed report is generated regarding the crop statistics in the current working directory.

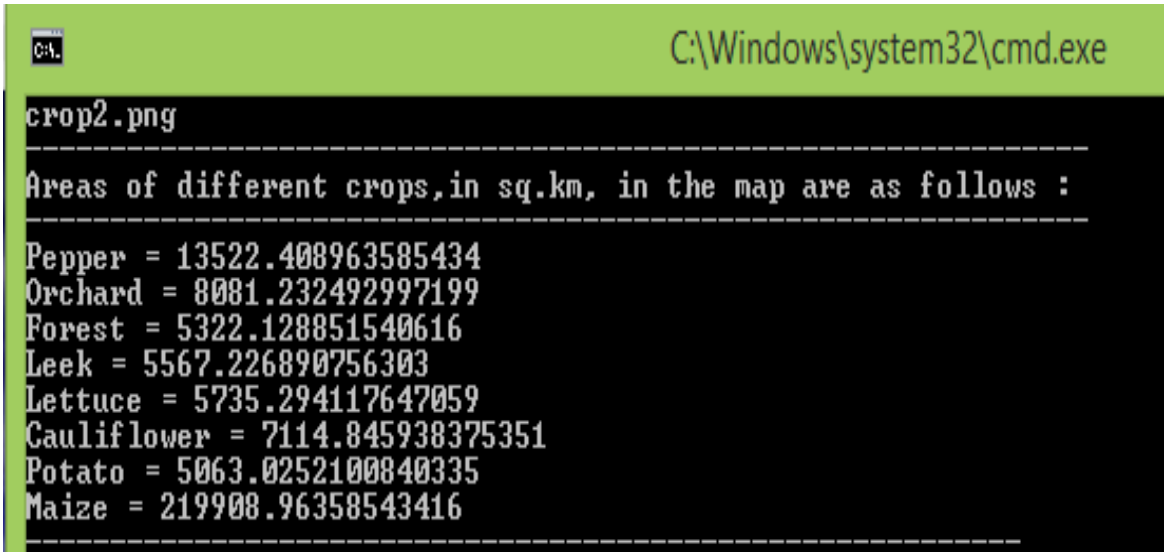
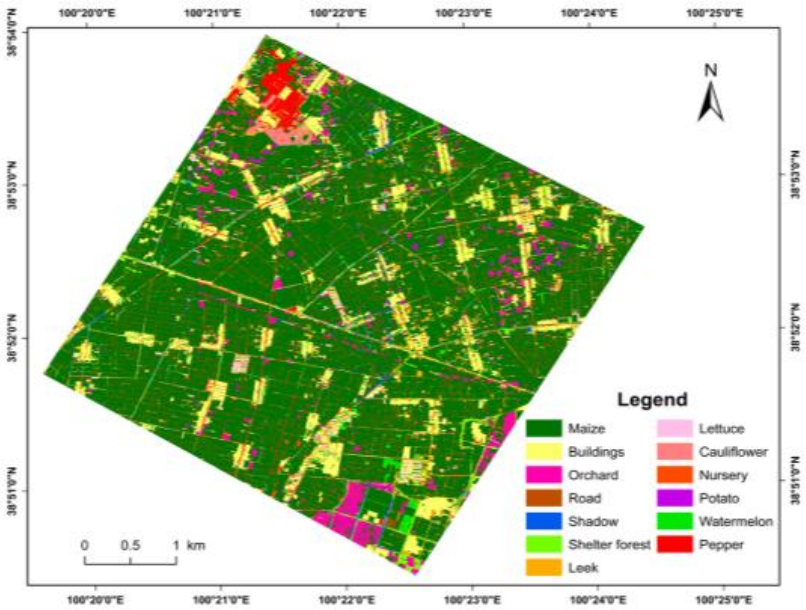


Fig 6.6 Results displayed on command line

Fig 6.7 Analyse Statistics Page

Crop Statistics(in percentage)

File Name : crop2.png



Crop Stat Report

CROP NAME	AREA(in %)
Maize	81.3528146939198
Pepper	5.00246107613792
Orchard	2.98955985596228
Cauliflower	2.63205616434807
Lettuce	2.12170669153649
Leek	2.05953213647315
Forest	1.96886091033911
Potato	1.87300847128313

Fig 6.8 : Statistics Report

7. CONCLUSIONS AND SCOPE FOR FUTURE ENHANCEMENTS

7.1. Conclusion

This project entitled **"Automated Crop Statistics"** has presented an approach to identify and calculate areas of different types of crops, using image processing techniques. The project is very useful to the agriculture department in tracking the statistics of different types of crops. The project is also useful to the government in controlling the prices, by timely importing those things whose crop area is relatively less. This project finally leads to the improvement of the quality of life for common man. Ability to make yield prediction before harvest using satellite remote sensing is important in many aspects of agricultural decision-making.

In this project, after observing the results of all generated models for crop yield prediction and the validation analysis for the generated models, it could be concluded that using image template matching algorithm under normal environmental conditions and common agricultural practices during the period of the maximum vegetative growth could be the best methodology of crop yield forecasting using satellite images. Using high resolution satellite imagery is necessary to be able to calculate crop cultivation especially in the intensive agriculture lands is necessary to apply these models over national scale. All generated models are empirical models limited to environmental conditions and applicable under similar conditions.

7.2. Future scope

The project is currently working accurately for calculating the areas of different types of crops in a given region of India. The applicability of this project to other countries, is yet to be explored. The current project requires manual entry of details related to the colours in the map. It can be extended to automatically detect the different colours in the map and calculate area. This project can also be extended to suggest the various type of crops that can be grown in that particular area based on soil conditions to increase the productivity in that area.

REFERENCES

1. Boryan, Claire and Yang, Zhengwei, "Deriving Crop Specific Covariate Data Sets From Multi-Year NASS Geospatial Cropland Data Layers", Proceedings of the IEEE International Geoscience and Remote S ensing Symposium, IGARSS 13, Melbourne, Australia, July 2013.
2. Heipke C, "Overview of image matching techniques". OEEPE Official Publication 33, 173–189 (1996).
3. Gallego, F. J., "The use of area frame surveys and remote sensing", Agriculture unit, IPSC, JRC, 2011
4. Michael Craig and Dale Atkinson, A Literature Review of Crop Area Estimation, July 2013
5. Sensing Symposium, IGARSS 13, Melbourne, Australia, July 2013.
6. NilaminiBhoi, MihirMohanty, "Template Matching Based Eye Detection in Facial Image", International Journal Of Computer applications(0974-8887), Vol12, Issue No. 5.
7. Sainis J. K., Molecular Biology and Agriculture Division, R. Rastogi, Computer Division, and V. K. Chadda, Electronics Systems Division, application of image processing in biology and agriculture.
8. https://testbook.com/blog/crops-in-india-gk-notes-pdf/
9. https://www.python.org/
10. https://github.com/baoboa/pyqt5/blob/master/pyuic/uic/pyuic.py
11. https://www.numpy.org/
12. https://riverbankcomputing.com/software/pyqt/intro
13. https://www.pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/
14. https://www.sciencedirect.com/science/article/pii/S1110982313000094