ENHANCED IMAGE PROCESSING MODEL FOR WEED CONTROL IN AUTOMATED FARMING

OBIDIKE E. O.; ONUODU F.E.; AND OKENGWU, U.A.
Department of Computer Science University of Port Harcourt, Nigeria

ABSTRACT

Crops are very important for the survival of the human race. It is s source of food to animals, humans and even other plants directly or indirectly. However, when a crop is attacked by weeds, it competes with the crop for essential nutrients and space and thereby inhibits the growth of the crops. Automated farming is a term who has increasingly gained popularity in recent times, in which farming practices are automated using powerful computerized tool, weed detection and classification is one of such processes which can be improved using machine learning. Several attempts have been made by researchers in previous years to achieve this feat however, there have been drawbacks which have limited the performance and accuracy of the existing models. In this work, an enhanced image processing model for weed control in automated farming have been proposed and implemented. Convolutional Neural Network (CNN) was used to train the image initially and define features of the image. CNN algorithm was used to train the classification network to different the weed prone segment of the image from the actual crops. The Rapid Application Development methodology was adopted in this approach. The system was implemented using Python programming language. The model had a performance accuracy of 95% which outperforms other existing models. Other parameters such as classification time (49 sec.), Learning rate (0.002) etc. were also used to evaluate the model. This study could be beneficial to Ministry of Agriculture, to Commercial Farmers, to Agricultural Institutions, to Farm Implement developers and to the research community.

INTRODUCTION

The agricultural sector, like so many other sectors have grown increasing automated. The application of artificial intelligence (AI) based approaches and models have help speed up this automation. The incorporation of these advanced techniques has led to increased crop yield, faster cultivation, and eased the farming process generally from ploughing to harvesting (Kaarthik & Vivek, 2018). A major area that has benefitted from this is the aspect of weed control.

Weed are the unwanted plants that grow amidst the relevant crops. They compete for nutrient with the crops and space for expansion, and in most cases, if not properly controlled (and on time too) could overpower the crops, slow its growth and yield and even lead to the death of that crop, thereby decreasing the crop yield. Traditionally, weed control involved handpicking the weeds or using farming tools like hoes to remove them. Herbicides were also applied manually around the farm to kill the weeds. However, for large commercial farms, this will be a tedious task and will require a lot of manpower and manhours. Another challenge is that the herbicides may also affect the real crops and kill them as well with the weeds. This form of application also leads to wastage of the herbicide, since the target areas are not covered due to lack of information about the weed distribution (Sonawane, et al, 2021). This has led to the adoption of modern automated methods which include the use of optical sensors to detect varying weed densities and species, which can be mapped using GPS data (Weis & Gerhards, 2018). The images can also be captured using the conventional camera. The weeds are extracted from images using image processing techniques and described by shape features. With this information, the herbicides can be applied directly to that area, without harming the crops as well. The result of this image processing task is then fed to a robotic sprayer, which then applies the herbicide around the farm for improved efficiency. This is what is known as automation in weed control using image processing. Several machine learning and neural networks can be used for image processing, as experimented by researchers over the years, however, we are adopting the convolutional neural network (CNN) for this design.

Herbicides help in the control of weeds in an agricultural setting. However, misapplication can have the exact opposite effect on the crops. Researchers have attempted to develop models for identifying weeds in fields and controlling them using herbicides, however, weed control still remains a major challenge yet to be completely tackled. Three major issues have been identified as the possible cause of these drawbacks. Firstly, the manual activation of the weed detector application does not really reduce the amount of manpower as desired. Secondly, identification and classification of weeds using scanty parameters such as only colour, or size, is not sufficient enough to guaranty accurate classification results, and misclassification can lead to wrong application and other adverse results. Lastly, for very large farms, focusing on only a specific area which is suspected to have more weed presence, and programming the spray bot to that section alone, may not yield the desired result. Our research will attempt to tackle these challenges, especially the weed detection drawback.

The aim of this work is to develop an enhanced image processing model for weed control in automated farming. The specific objectives are to:

i. identify and classify weeds from plant images using convolutional neural network (CNN).
ii. implement our model using Python programing language.
iii. evaluate our results with other existing models, to determine its performance.

Software design methodology provides a logical and systematic means of proceeding with the design process as well as a set of guidelines for decision-making. The design methodology provides a sequence of activities, and often uses a set of notations or diagrams (Khoo, 2021). The Rapid Application Development (RAD) Methodology was adopted in this approach. Rapid Application Development (RAD) is a form of agile software development methodology that prioritizes rapid prototype releases and iterations. RAD is a more adaptive approach to software development. RAD approach is based around flexibility and the ability to adapt alongside new knowledge.

The characteristics of a RAD process include:

i. Enhanced flexibility and adaptability as developers can make adjustments quickly during the development process.
ii. Quick iterations that reduce development time and speed up delivery.
iii. Encouragement of code reuse, which means less manual coding, less room for errors, and shorter testing times.
iv. Increased customer satisfaction due to high-level collaboration and coordination between stakeholders (developers, clients, and end users).
v. Better risk management as stakeholders can discuss and address code vulnerabilities while keeping development processes going.
vi. RAD includes integrations early on in the software development process

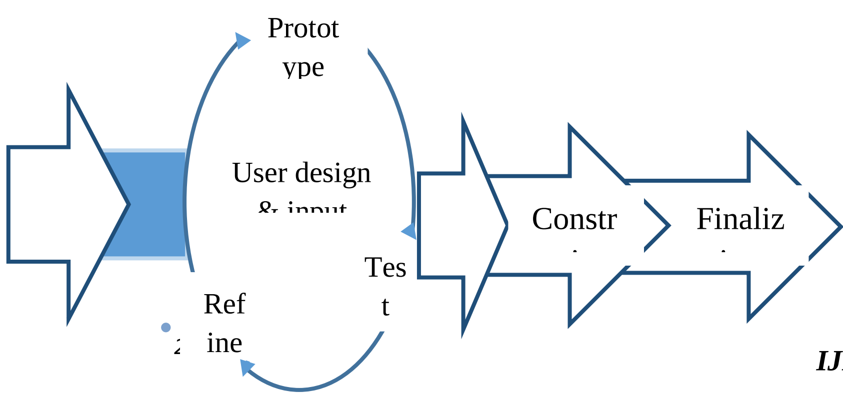


Figure 1.1. RAD Process chart (www. https://www.creatio.com/page/rapid-application-development)

REVIEW OF RELATED LITERATURE

Agriculture has been traced to the original source of sustenance for humans since the early times (Basavarajeshwari & Madhavanavar, 2017). Agricultural products are used as raw materials in so many industries, and sectors, such as medical, textile, architecture, food, etc., which underscore the importance of agriculture to human existence. Agriculture is one of the occupations that has never lost its relevance through human evolution, and advancement, rather, more resources, and ideas are continually channeled towards the continuation and expansion of this field all over the world. In countries like India, and Nigeria, agriculture is regarded as the traditional job, and it forms a major part of the economy of these countries. Very large revenues are generated each year from the agricultural sector (Kaarthik & Vivek, 2018). There are two major forms of agriculture, which include the commercial agriculture (where agriculture is practiced on a large scale for the purpose of merchandise), and subsistence farming (this form of farming is usually on a small scale for personal consumption and sustenance of the farmer). Agriculture is a wide field with so many branches, which include: crop production, animal husbandry, etc. however, our main focus in this research is on the aspect of crop production (for both commercial and subsistence farming).

Originally, agriculture was practiced using traditional implementing tools (especially in Nigeria), such as hoes, cutlasses, diggers, etc. which were used for farm practices such as: ridging, planting, cultivating, harvesting, weeding, application of manure, etc. However, due to the massive increase in population, which is directly proportional to the increase in demand for agricultural products (especial food crops, diminishments in agricultural land, environmental variation and the political instability, the agriculture industries are trying to find the new solution for enhancing the essence of the productivity and sustainability (Sridha, 2019). In order to support and satisfy these needs of the farmers and the population, precision agriculture is employed. It can help with upgrading the cultivating practices by utilizing data innovation instruments, which empowers farmers to listen, evaluate and control the farming practices, such as sufficient amount of fertilizers, pesticides and water usage. Precision agriculture involves the incorporation of advance techniques to enhance farm output and also enrich the farm inputs in profitable and environmentally sensible manner (Vibhute & Bodhe, 2012). With these techniques/tools it was now possible to reduce errors, costs to achieve ecological and economically sustainable agriculture. Farm inputs were important parameters to be controlled and if not will result in adverse effects causing reduction in yield, deteriorating plant health, etc. The use of these automated tools has helped in the simplification of most of the farm practices, and has also resulted in higher yield, variety farming, and less man hours and manpower.

Some applications of automation in farming are discussed as follows:

Remote sensing is one of the many technologies which are integrated into agriculture in other bring about automation. Navalgund et al, (2007) defined Remote Sensing as the science of identification of earth surface features and estimation of geo-biophysical properties using electromagnetic radiation. The authors presented a review of the Rs methods and its applications with optical and microwave sensors. They also stated some of the satellites launched by different countries and their uses in various field along with spatial, spectral and temporal variations of data. Analytical techniques using digital image processing, multi-source data fusion and GIS were also discussed. Applications towards agriculture providing the earth observation data which supports increased area under agriculture, increased crop intensity and productivity, etc. they opined that RS data can provide the data related to groundwater helping in irrigation, flood management. Applications like environment assessment and monitoring, disaster monitoring and mitigation, weather climate, village resource center, etc. were also discussed.

Manickavasagan, et al, (2011) discussed thermal imaging and its applications in agriculture were discussed. Thermal imaging which was a passive technique (infrared range between 3 to 14 μm) focuses on Water. Water which affects the thermal properties of plant where leaf contains different amount of water per area can be utilized as an important parameter in pre-harvesting operations. They reviewed several applications of thermal imaging such in Field nursery, irrigation scheduling, yield forecasting, green houses termite attack etc. Post harvesting operations such as maturity evaluation, bruise detection, detection of foreign substances in food etc. were also reviewed. Though the thermal imaging produces better results but cannot be accepted universally in agriculture applications as the plant physiology and climatic conditions varies from region to region.

Xavier et al, (2010) presented a study on genetic algorithm for weed extraction use combinational methods such as segmentation of vegetation and soil, crop row elimination and weed extraction. For segmentation- S1 and S2 methods, for crop row elimination- E1, E2 and E3 methods and for weed extraction- F1 and F2 methods were proposed. S1 method combines RGB to gray conversion and Gray to BW using threshold values, whereas S2 directly converts RGB to BW depending on pixel property. E1and E2 algorithms in which crop elimination was done by taking column pixels into considerations. F1 and F2 for weed extraction use the filtering and region extraction.

Then the combinations of S, E and F were processed to find out the optimum value by genetic algorithm method. Results of the methods were compared with biomass and showed accuracy up to 96% with small computational complexity.
Another researcher, Omid et al (2010) developed a hardware specially for grading raisins specially which captures the image. Image was processed with VB based algorithm for color and size of raisins. Pixel colors in RGB form were calculated and with position control, upper and lower pixels were determined. From these pixels middle position can be determined and features were extracted. Raisins with bad grade were identified as background and others in good category. From confusion matrix the classification rate obtained was higher as compared to human experts. This algorithm was also applicable to lentil and almond.
Also, in 2010, López-García presented a technique for detection of skin defects in citrus fruit PCA which was used for multivariate image analysis (MIA). They captured images using 3CCD camera were applied to MIA algorithm which unfolds the images in RGB and spatial information. Reference eigenvector formed by training with defect free citrus was used to compute T2 matrix. Threshold value decides the defect in fruit, if the value was greater, then it was considered as defect. This leads to preparation of defect map. Multi-resolution and post processing techniques were used to speed up the process with three different measures. In study of 9 defects detection average correct detection was 91.5% and classification into four damaged /sound classes was 94.2%. Author concluded with discussion of novelty detection and ability of model to identify new unpredictable defects.
Blasco et al (2009) proposed a morphological process-based image analysis of shape in real time for inspecting and sorting processed mandarin segment was developed. Images in RGB format illuminated with constant source were captured. These images were segmented in background and objects of interest. A morphological operation allows identifying the objects in complete, broken formats. The shape analysis was done by perimeter and area calculation. Once the contour was obtained, FFT was applied to discriminate low and high frequency details which were helpful in size determinations. Standard Bayesian discriminant analysis was used for classification. Mechanical system limits the speed of sorting. Model provided real time classification with enough accuracy.
Neethirajan et al (2006) performed corn variety identification on the basis of the color, shape and geometric features using discrimination analysis and neural network. To avoid illumination and manmade disturbances images were captured with flatbed scanner. Features were extracted using morphological feature analysis and color feature.

Morphological feature analysis of corn kernels was used to extract basic geometric features such as area, perimeter and derived shape features. Color analysis was used for classification. Mean and standard deviation of these color components were calculated to extract 28 color features for identification. To reduce computational burden and to enhance the performance of classification stepwise Discriminant analysis was used. Mahala Nobis distance method with back propagation neural network was used to train and classify the corns with higher accuracies. Feature selection with discriminant analysis and two stage classifier identifies the variety at rate of average 90%.
The weed can be an unwanted plant which is present in the field. They are also regraded to as plants in the wrong place. They can cause some damage to the main crop and they can take the nutrition's from them. In order to get high yield, the weed must be removed. They can be obtained all over the field and they are with different size and differ in Edge frequency, Boundary, Shape too (Kaarthik & Vivek, 2018).

Figure 2.1. Sample Images of Weed Examples (Rakhmatuiln, et al. 2021)

Weed control is a process of removing unwanted plants from the land mechanically by manual removal, chemically by using herbicides, or by other alternative means. Weeds compete with the crop for sunlight as well as for the required nutrients and resources from the soil. Effective removal of weeds improves the productivity of the crops. Identification and removal of weeds is a challenging task (Lavanya et al, 2019).
Weed detection is one of those basic agriculture tasks that are being automatized and digitized, in this case, because of toxicity related to herbicides; thereby reducing human intervention will make possible a decrease in the use of herbicides, increasing health care.

MATERIALS AND METHODS

Analysis of the Existing System

The existing system was developed by Bhongale & Gore (2017). They presented weed recognition system for crops in farms using image processing techniques and smart herbicide sprayer robot. The existing work focused on: the detection of weeds in the farms, identification of weeds in image, location of the weed plant in the field based on the weed position. Depending on position of weed spray the herbicide on the position of weed. based on erosion followed by dilation segmentation algorithm. This algorithm could detect weeds and also classify it. Currently the algorithm was tested on two types of weeds i.e., broad and narrow.

Explanation of Existing System Components

i. **Dataset:** The dataset used in the training and testing of the weed detection model was gotten from images of farm fields with crops and weeds mixed along ridges e.g. The maize field, the rice field etc. the images were carefully selected based on their resolutions and pixel qualities. Blurry images were not selected in this model. The training data contained 20 images of field lines and the testing dataset contained 10 images. The images were imported using the import function from their folder.

ii. **Preprocessing:** In the preprocessing phase the input image is converted to grayscale. The original image is gotten in RGB format. The image is first converted to grayscale and then also converted to binary format. These conversions are carried out to enhance the image and strengthen the vital information contained in the images for further processing. Another purpose of this step is to extract information from the input images and interpret their contents.

iii. **Feature Extraction:** In this step, a set of features are defined for the efficient representation of the information for classification. Recall that the input image contains three major components: the Crops, the Weeds and the Soil. However, the features of the weeds and soil are defined in order to classify them easily effectively. The features include color, shape, texture and context. Under the texture-based features we have entropy, energy, contrast etc. the SURF algorithm contained in the Computer Vision Toolbox in MATLAB was used to automatically extract features from the images while preserving the original information in the input data.

iv. **Segmentation:** This step involves the partitioning of the input image into sets of pixels also called image objects. This step is important because it defines the lines and boundaries of the preprocessed image based on the features extracted by identifying the abrupt discontinuities in the pixel value which typically indicate the edges that define the crop region. Here the crop lines are enhanced while the weeds are given are darker shade. This will notify the sprayer about the position of the normal crops in the fields so that the herbicides are not sprayed in those regions. Otsu's thresholding method was used for segmentation using the "imbinarize" function.

v. Classification: In the classification step, they built a simple classification network in MATLAB. The network was built by defining the input and the class size of the image in a matrix format.

E.g. InputSize = [20 20 2];
numClasses = 3;

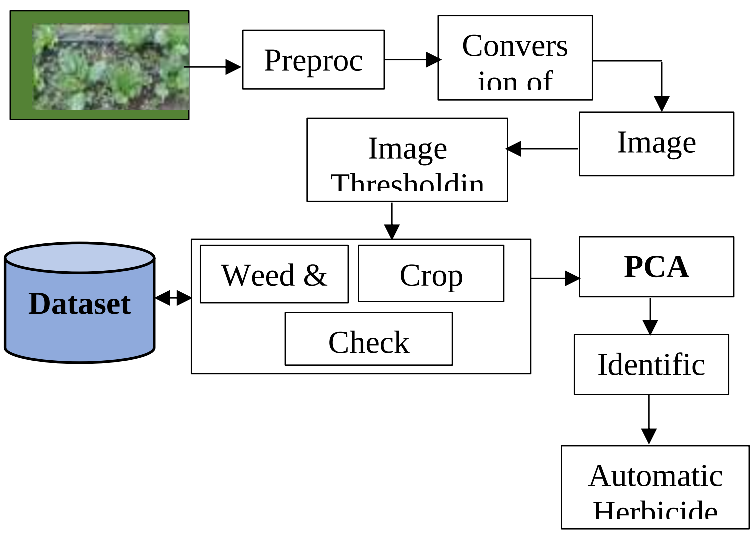


Figure 3.1.: Architecture of the Existing System (Bhongale & Gore, 2017)

All the convolutions are expanded in the Conv layers using padding size of 1 to transform the original input image size to (M + 2) x (N + 2), and then a kernel of size (3 x 3) is applied to obtain an output image of (M x N), i.e., (640 x 480). This helped the input and output matrix sizes to remain unchanged in the Conv layers. Moreover, the pooling layer, kernel, and the stride sizes are set to 2 in the Conv layers. Thus, every (640 x 480) matrix that goes past the pooling layer is converted to (640/2) x (480/2). In all of the Conv layers, the input and output sizes of the Conv and relu layers are kept the same. However, the pooling layer forces the output length and width to be 1/2 of the input.

Next, a matrix with a size of (640 x 480) is switched to (640/16) x (480/16) by the Conv layers; hence, the feature map produced by Conv layers can be associated with the original image. The feature maps are fed to the subsequent RPN and fully connected layers.

Explanation of Proposed System Components

a. **Dataset:** the v2-plant-seedling dataset was to fit the model. The datasets were split into training and test sets in a ratio of 80:20, this is because the higher the training set, the more accurate the resultant output. The datasets contain plant images of 12 different plants as shown in table 3.1 to allow for multi-class classification.

b. **Preprocessing;** the preprocessing phase is made up of four major tasks: removal of null and missing values, removal of noise and extreme lighting disorders from the image, creation of dataframe to store dataset, mapping of characters to integers, which is also called normalization, and performing the reverse action from integer to character.

c. **Training & Testing:** during training, the proposed networks are defined, and assigned certain weights such as loss functions, epochs, etc. the optimizer and model objects are also defined during training. The datasets are fit into the model to test their performance.

b) **Feature Extraction:** Feature extraction is a part of the dimensionality reduction process, in which, an initial set of the raw data is divided and reduced to more manageable groups. So, when you want to process it will be easier. Feature extraction reduces the amount of redundant data without losing any relevant data. OpenCV is used to perform the feature extraction in this study.

c) **Classification:** Pseudo feature maps are used to compute the proposal's class and, simultaneously, the final position of the detection frame is acquired by the bounding boxes. Since the network deals with PxQ input size images, they are first scaled down to a constant size of (MxN), i.e., (640x480), and passed onto the network. The convolution layers contain 11 Conv layers, 11 relu layers, and 5 pooling layers. The proposed network employs 3x3 convolution and then generates foreground or background anchors and the associated bounding box regression offsets. Then, proposals are calculated and ROI pooling is performed, which computes the feature maps and sends them to the subsequent fully connected softmax network for classification. The classification section uses the acquired property feature maps for calculating the specific category (i.e., tobacco plants and weeds) that each property belongs to via the fully-connected layer and softmax. Finally, the probability for the class is computed, and bounding box regression is once more used for obtaining the position offset for each proposal. After obtaining the 7x7 = 49 sized features, feature maps from ROI pooling, and then sending them to the succeeding network, the following two steps were performed:

i. Classification of proposals by fully-connected layer and softmax;
ii. Bounding box regression on the proposals for acquiring more accurate rectangular boxes.

Table 3.1: Dataset Distribution

Plants	Number in Dataset
Loose Silky-bent	762
Common Chickweed	713
Scentless Mayweed	607
Small-flowered Cranesbill	576
Fat Hen	538
Sugar beet	463
Tobacco	452
Cleavers	335
Black-grass	309
Shepherds Purse	274
Maize	257
Common wheat	253

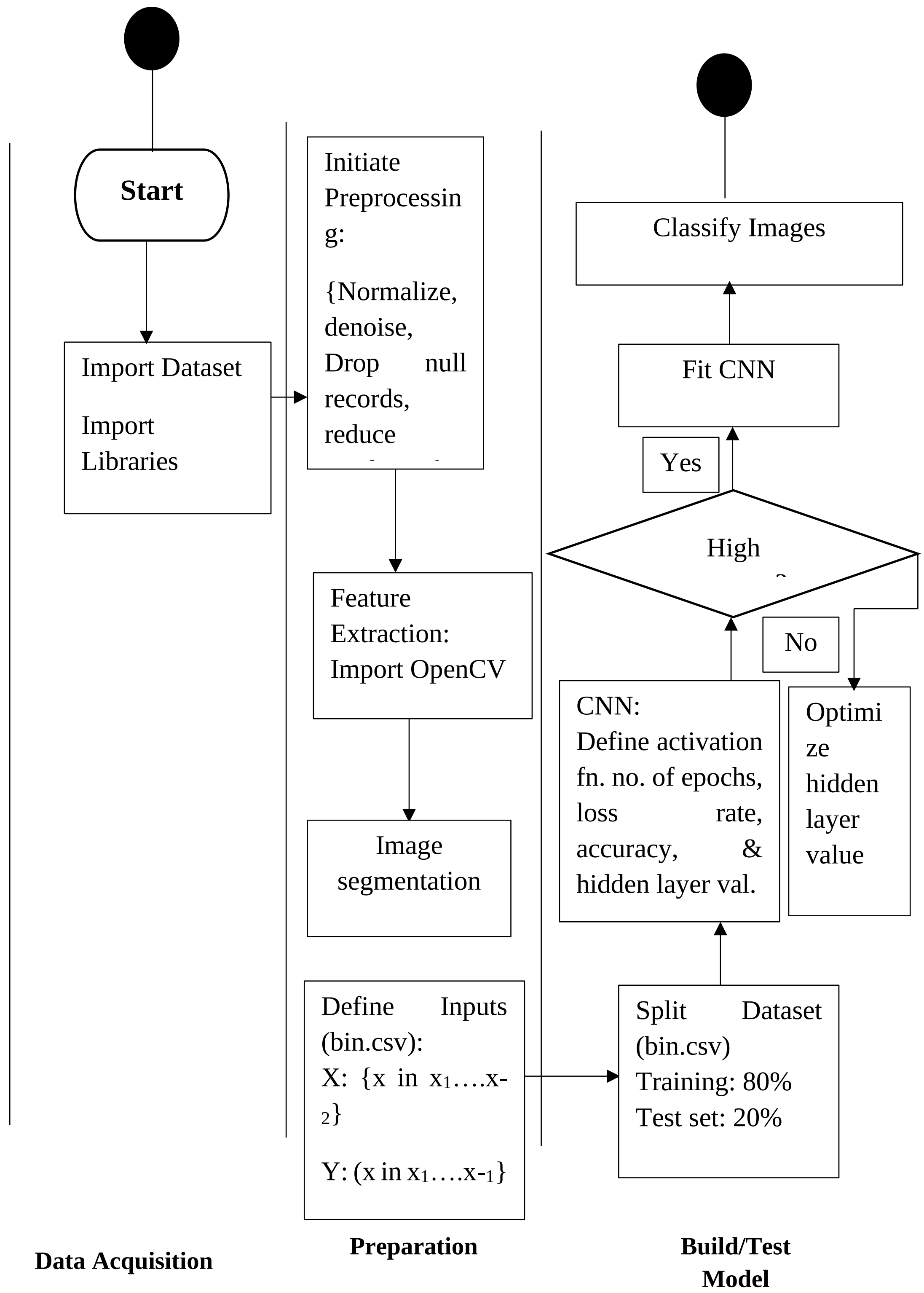
Advantages of the Proposed System

i. The proposed system makes use of deep learning algorithm i.e., CNN which is very efficient for classification of complex images.
ii. The algorithms used is very efficient and can detect and recognize even unprecise patterns from large datasets. The model can also perform multi-class classification in natural scene images.
iii. The proposed system has a good visualization model for demonstrating the results gotten from their implementation.
iv. Extreme scenarios such as blur images in both in plant dataset is efficiently handled in the proposed system without losing relevant information.

Algorithm of the Proposed System

The proposed model was developed in the following steps:

Step 1:
    Start.
Step 2:
    Import Image Dataset (RGB)
    Divide Images into Train & Test Sets
    trainSet = 20;
    testSet = 10;
Step 3:
    Convert Image to Greyscale = rgb2gray
Apply Data Augmentation
    Resize shorter size of image to (M + 2) x (N + 2),
i.e., 640x480
    Adopt random cropping
Step 4: Import OpenCV
    Initiate FE (Feature Extraction)
Step 5:
For Each convo layer:
    Kernel=3;
    Padding =0;
    Stride=2;
Step 6:
    Initiate Otsu's thresholding = imbinarize;
Step 7:
    Set Threshold Value
    IF Threshold Value = Image Resolution
Step 8:
    Initiate CNN algorithm
    Define CNN:
        Convolayers =11;
        ReluLayers = 11;
        Activation function = softmax;
        Learning_rate = 0.0002;
Epochs = 8;
    ELSE
Step 9:
    ADJUST Step 5
Step 10:
    Classify Images = Training Output
Step 11:
Input Classification Output into Automated Spraying
Machine.
Step 12:
    END.


Start
Import Dataset
Import Libraries
Initiate Preprocessing:
{Normalize, denoise, Drop null records, reduce
Feature Extraction: Import OpenCV
Image segmentation
Define Inputs (bin.csv):
X: {x in x1….x-2}
Y: (x in x1….x-1}
Classify Images
Fit CNN
Yes
High
No
CNN: Define activation fn. no. of epochs, loss rate, accuracy, & hidden layer val.
Optimize hidden layer value
Split Dataset (bin.csv)
Training: 80%
Test set: 20%
Data Acquisition
Preparation
Build/Test Model

The proposed system algorithm is an improvement of the steps followed to implement the Existing system, with enhancements such as the introduction of parameters for appraisal and one of the input variables and the training of data using the Artificial Neural Network algorithm.

Model Diagram of the Existing System

System modelling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML). Models help the analyst to understand the functionality of the system; they are used to communicate with users.
Five types of UML diagrams that are the most useful for system modeling:

i. **Activity** diagrams, which show the activities involved in a process or in data processing.
ii. **Use case** diagrams, which show the interactions between a system and its environment.
iii. **Sequence** diagrams, which show interactions between actors and the system and between system components.
iv. **Class** diagrams, which show the object classes in the system and the associations between these classes.
v. **State** diagrams, which show how the system reacts to internal and external events.

However, in this study we are going to design two UML: activity diagram and Usecase diagrams.

Usecase Diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses

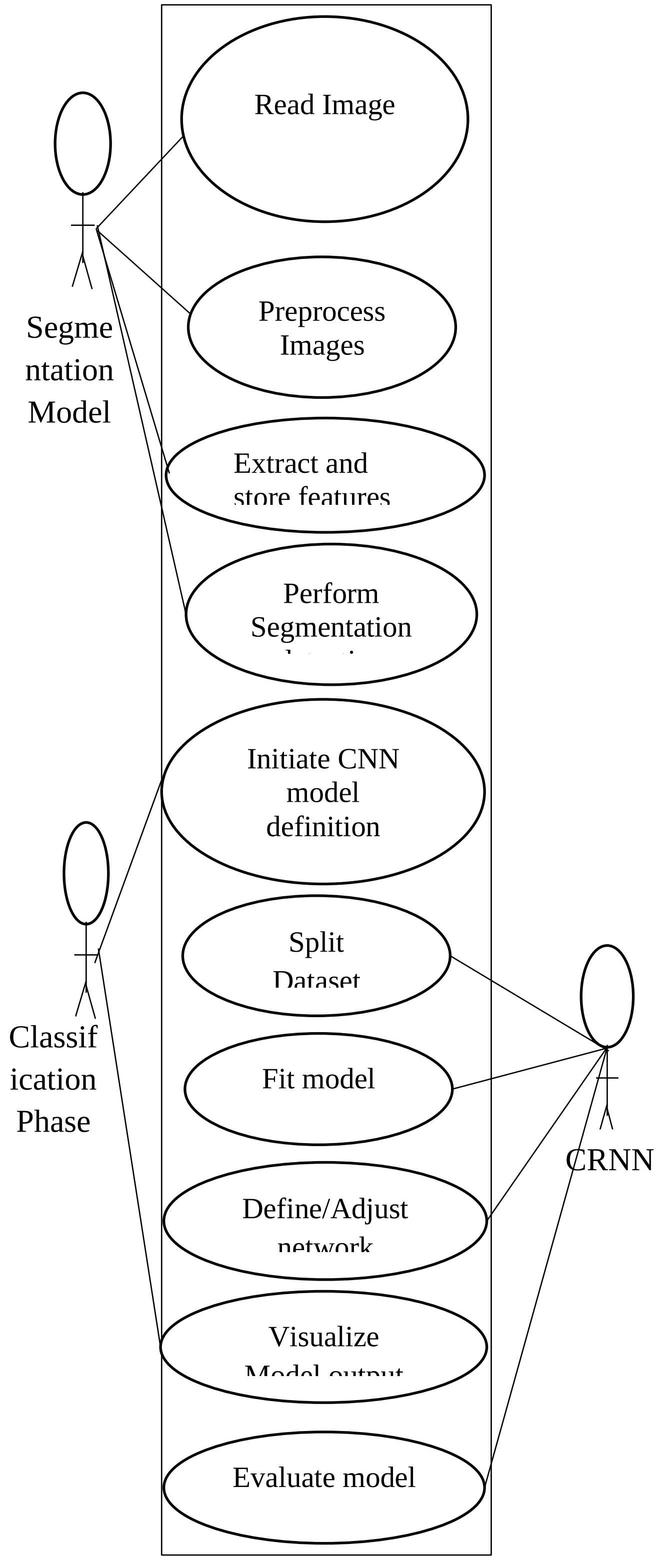


Figure 3.4: Usecase Diagram of the Proposed System

Design Specifications of the Proposed System

This section illustrates the output specification, input specification and database design of the Proposed System. The specification of the test data is also described in this section in order to provide useful insight on the type of data we are dealing with in this study. Every system has an input and output. If the desired output is gotten, it demonstrates that the system is successful and meets user's requirements, else it calls for debugging in order to arrive at the desired output as prescribed by the user or other stakeholders of the system.

Table 3.2. Output Specification

Input	Expected output
Plant Images	Extracted Features
Features	Plants/weed classification

Table 3.3. Dataset description

IMPLEMENTATION AND DISCUSSION OF RESULTS

Plants	Number in Dataset	Percentage used for Testing	Percentage used for Training
Loose Silky-bent	762	0.135379	0.138118
Common Chickweed	713	0.125451	0.129542
Scentless Mayweed	607	0.106498	0.111262
Small-flowered Cranesbill	576	0.105596	0.103363
Fat Hen	538	0.102888	0.095012
Sugar beet	463	0.081227	0.084180
Tobacco	452	0.079422	0.082148
Cleavers	335	0.062274	0.060032
Black-grass	309	0.055054	0.055969
Shepherds Purse	274	0.055054	0.051230
Maize	257	0.048736	0.044911
Common wheat	253	0.042419	0.044234

System Specification

This is a documentation that describes the behaviour of the system or an application. In this description, the hardware and software configuration of the system on with the system is to be deployed. This provides information on the requirements a system must meet in order to run a particular software or application.

A) Hardware Specification

- **i.** RAM: at least 2 GB for best performance.
- **ii.** Hard Disk: at least 1 GB.
- **iii.** Processor: 450-MHZ core (pentium) and above is recommended.
- **iv.** Battery life of 1 hour and above is highly recommended.

B) Software Specification

- **i.** Anaconda IDE (Jupyter Notebook)
- **ii.** Python3 compiler
- **iii.** Visual Studio Code
- **iv.** Operating System: Windows 7, 8 and 10.

Choice and Justification of Programming Languages Used

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Python has fewer steps when compared to Java and C. It was founded in 1991 by developer Guido Van Rossum. It is used in many organizations as it supports multiple programming paradigms. The advantage of the language over other high-level languages is that it is Interactive, Interpreted, Modular, Dynamic, Object-oriented, Portable, High level, Extensible in C++ & C.

Discussion of Results

In this section, we describe the step-by-step implementation of the proposed system. the system was using python programming language which runs on a python3 interpreter. Dependencies or requirements for the proposed model include: keras, TensorFlow, SamplePreprocessor, os, cv2, random, etc., these dependencies which are python libraries must be in place (i.e., installed) before the model can be implemented.

In order to validate and demonstrate the effectiveness of both CNN for crop/weed detection and classification, the proposed model was trained and tested on real-field images from v2-plantseedlings dataset. The dataset consists of over 5000 images; 3700 images are of different plants including tobacco and maize plants, and the rest of the images are of weeds (2068 images). Images from both classes are divided with 80 to 20 ratio into training and testing sets. The training set comprised a total of 4600 images (4000 plants and 600 weeds), whereas the testing set comprised 1068 images (900 plants and 168 weeds). In the implementation phase, the models are trained using down-sampled images (with a resolution of 640 x 480). A learning rate is initialized as 0.0002 for the training. TensorFlow is utilized for implementation purposes. Batch sizes of 1 and 8 epochs are used for training the models. Table 4.1 lists the hyper-parameters and their corresponding losses (against the epochs) for both models. It can be observed from Table 4.1 that for obtaining better results with CNN-based model, the learning rate is kept the same, whereas the other hyper-parameters did change. With an increase in the number of epochs, total loss is reduced.

After training the models with the given training set, performance evaluation of both models is conducted on the testing data from v2 plant_seedling. The accuracy results obtained by using the CNN-based vision model show its supremacy over other existing models. A total of 635 predictions were produced on unseen test images for each model. Detection results for both models are presented in Figures 4.1 to 4.6

The first step in implementation is importing libraries, numpy and pandas are used for handling datasets, glob and os are used to navigate folders, cv2 is used to open images and view them, matplotlib is used for plotting and visualization. Data processing and model development are handled using the sklearn library. Keras was used to build the CNN model.

Table 4.1.: Hyper Parameters for CNN

S/N	Learning Rate	Epoch	Loss of CNN
1	0.0002	8	0.046
2	0.0002	10	0.029
3	0.0002	20	0.028
4	0.0002	30	0.025
5	0.0002	30	0.017

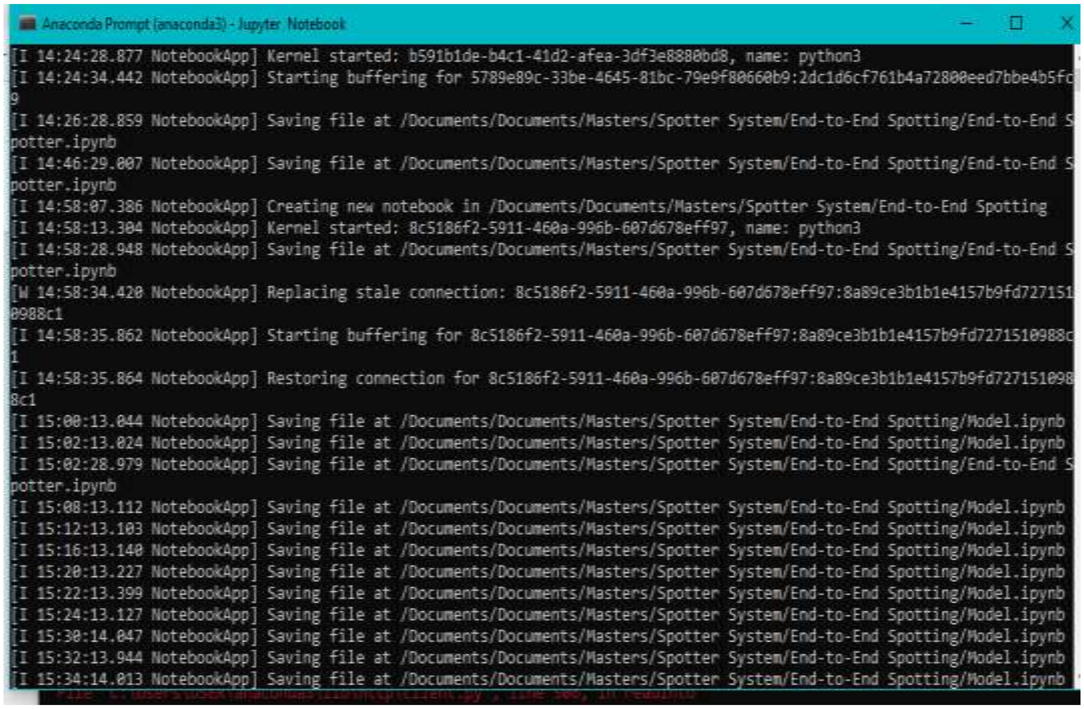


Figure 4.1: Model run history on Anaconda Prompt

The ReLU function is preferred due to the lower likelihood of a vanishing gradient (which arises when network parameters and hyperparameters are not properly set) as opposed to a sigmoid function The CNN is fed the output from the feature extraction to compute the loss value, and the mean of the loss values of the batch elements is used to train the model. The preprocessing stage includes removal of noise and reduction of light shades in the images. Noise includes everything which is not plants, e.g., rust, stones, etc.

The target is prepared using hot encoding in order to achieve multiclass classification. The shape of the images is defined into n-series: n1, n2, n3, n4, where n1 is the number of observations, n2 and n3 are the image height and width, and n4 indicates that it is a color image, so 3 planes per image. The CNN configuration is shown in table 4.2

Table 4.2.: Configuration of the CNN model

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 148, 148, 32)	896
conv2d_2 (Conv2D)	(None, 146, 146, 32)	9248

max_pooling2d_1 (MaxPooling2	(None, 73, 73, 32)	0
dropout_1 (Dropout)	(None, 73, 73, 32)	0
conv2d_3 (Conv2D)	(None, 71, 71, 64)	18496
conv2d_4 (Conv2D)	(None, 69, 69, 64)	36928
max_pooling2d_2 (MaxPooling2	(None, 34, 34, 64)	0
dropout_2 (Dropout)	(None, 34, 34, 64)	0
conv2d_5 (Conv2D)	(None, 32, 32, 128)	73856
conv2d_6 (Conv2D)	(None, 30, 30, 128)	147584
max_pooling2d_3 (MaxPooling2	(None, 15, 15, 128)	0
dropout_3 (Dropout)	(None, 15, 15, 128)	0
flatten_1 (Flatten)	(None, 28800)	0
dense_1 (Dense)	(None, 256)	7373056
dropout_4 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 12)	3084

==

Total params: 7,663,148
Trainable params: 7,663,148
Non-trainable params: 0

```
Epoch 1/8
Epoch 00000: acc improved from -inf to 0.92092, saving model to model.h5
633s - loss: 0.2382 - acc: 0.9209
Epoch 2/8
Epoch 00001: acc improved from 0.92092 to 0.93965, saving model to model.h5
686s - loss: 0.1558 - acc: 0.9396
Epoch 3/8
Epoch 00002: acc improved from 0.93965 to 0.95018, saving model to model.h5
724s - loss: 0.1275 - acc: 0.9502
Epoch 4/8
Epoch 00003: acc improved from 0.95018 to 0.95746, saving model to model.h5
728s - loss: 0.1102 - acc: 0.9575
Epoch 5/8
Epoch 00004: acc improved from 0.95746 to 0.96581, saving model to model.h5
741s - loss: 0.0906 - acc: 0.9658
Epoch 6/8
Epoch 00005: acc improved from 0.96581 to 0.97250, saving model to model.h5
745s - loss: 0.0734 - acc: 0.9725
Epoch 7/8
Epoch 00006: acc improved from 0.97250 to 0.97873, saving model to model.h5
756s - loss: 0.0562 - acc: 0.9787
Epoch 8/8
Epoch 00007: acc improved from 0.97873 to 0.98020, saving model to model.h5
767s - loss: 0.0524 - acc: 0.9802
```

Figure 4.2.: Training Sequence in 8 iterations

Summary

Automated farming is a fast-growing research field which had gained much attention in recent times. This research area has its applications in both research and industrial domains, such as the development of automated spraying robots for application of herbicides, crop production automation, etc. In this research, we have outlined various challenges associated with the existing classification and detection systems, reviewed some existing systems and their functionalities, and finally proposed a weed control approach. The major goal of this research was to develop an enhanced image processing model for weed control in automated farming. In order to achieve this goal, certain objectives such as: identify and classify weeds from plant images based on multiple parameters using convolutional neural network (CNN)., training and testing the classification using dataset of 12 different plant seedlings with 0ver 5000 plant images. etc. implementing the proposed model using python programming language, and evaluating the performance of our model using neural network evaluation parameters and compare our results with the existing model's performance. The process involved include data acquisition, preparation, feature extraction, image segmentation, model definition, Classification of plant seedling, model fitting (using our seedling datasets).

The Rapid application development methodology was adopted in this approach based on its interactive and iterative characteristics. Other design tools such as Usecase, Activity definition diagrams, database design, and algorithm definition were employed to analyze and develop his system. Python 3.9 was used to implement the model. Python libraries were used to train the neural network using the dataset, the model was tested using real-world scene images captured using a digital camera, and the model achieved high classification accuracy.

Conclusion

Image processing technique has been used to detect and differentiate between weeds and crops in a crop field to enable controlled application of the herbicides on only the weeds while preserving the crops. The study concluded on the importance of enhanced weed control for development of improved models for automated farming. The model was implemented using Python programming language and some relevant libraries such as Keras, TensorFlow, matplotlib, etc. and a deep learning algorithm called CNN. The model passed through four basic steps: preprocessing, feature extraction, segmentation and classification. The model was evaluated using parameters such as classification accuracy, classification time, precision, recall, classification error and F1. This model outperforms other existing models of weed classification. This system has addressed the challenges of the existing models such as accurate classification even with complex and noisy images, etc. The proposed system outperformed other existing system in terms of accuracy, recall, precision, F1, loss rate, etc. The overall accuracy score of the system was 95%, which outperforms the existing system with accuracy score of 92.92%.

Recommendations

The proposed weed classification model can be fully deployed into any automated farming to enhance segmentation and classification, and solve the challenges that existed before now. More research ideas should tilt towards weed recognition in order to improve the readability and save the time and resources which are currently spent using manual methods.

Contributions to Scholarship

Enhanced image processing model for weed control in automated farming has been proposed and implemented in this study, which outperforms other existing approaches. The proposed system can be utilized for extensive research in the area of Image processing, classification, detection, recognition, natural language process, and signal processing.

Future Scope

Though the proposed system has tackled most of the existing challenges, however, there is still room for more work in this field, therefore our future work will be focused on developing a full weed classification model which can provide full detection and classification of weeds according to their classes and nature of harm they cause on the plant using ensemble algorithms.

REFERENCES

Bavithra, K., & LAvanya. A.P. (2019). Weedicide Spray Robot using Image Processing." *International Research Journal of Engineering and Technology*. 6(6), 2448-2451.

Bhongale, K., & Gore, S. (2017). Weed recognition system for crops in farms using image processing techniques and smart herbicide sprayer robot. *International Journal of Current Engineering and Scientific Research (IJCESR)*, *4*(7), 10-16

Blasco, J., Aleixos, N., Cubero, S., Gómez-Sanchís, J., & Moltó, E. (2009). Automatic sorting of Satsuma (Citrus unshiu) segments using computer vision and morphological features. *Science Direct -Computers and Electronics in Agriculture*, *66*, 1-8

Desai, R. (2016). Removal of weeds using Image Processing. *International Journal of Advanced Computer Technology (*IJACT).

Esau, T. J., Zaman, Q. U., Chang, Y. K., Schumann, A.W., Percival, P.C., & Farooque, A.A. (2014). Spot-application of Fungicide for Wild Blueberry using an Automated Prototype Variable Rate Sprayer. *Precision Agriculture*, *15*(2), 147-161.

Gonzalez-de-Soto, M., Emmi, L., Perez-Ruiz, M., Aguera, J., & Gonzalez-de-Santos, P. (2016). Autonomous systems for precise spraying–Evaluation of a Robotized patch sprayer. *Bio-systems Engineering*, *146*, 165-182.

Heisel, T., Schou, T., Christensen, S., & Andreasen, C. (2001). Cutting weeds with CO2 laser. *Weed Res*. *41*, 19–29.

Karrthik, P., & Vivek, C. (2018). Weed Remover in Agricultural Field through Image Processing. *International Journal of Pure and Applied Mathematics*, *118*(8), 393-399

Lameski. P., Zdravevski, E., Trajkovik, V., & Kulakov, A. (2017). Cloud-based architecture for automated weed control.

Lavanya, N.R., Niharika, S., Deepika, C.H., Harini M., & Chetana, K.S. (2019). A Review on Weed Detection Using Image Processing. *Indian Journal of Science and Technology*, *12*(48), 1-3. https://10.17485/ijst/2019/v12i48/149410

López-García, F., Andreu-García, G., J., Aleixos, N., José-Miguel V. (2010). Automatic detection of skin defects in citrus fruits using a multivariate image analysis approach. *Science Direct Computers and Electronics in Agriculture*, *71* (2010), 189–197

Manickavasagan, A., Jayas, D.S.N., White, D.G., & Paliwal, J. (2011). Applications of Thermal Imaging in Agriculture – A Review. *The Canadian society for engineering in agriculture, food and biological systems*, 5(2), 10-16

Mustafa, M. M., Hussain, A., Ghazali, K. H. & Riyadi, S. (2007). Implementation of Image Processing Technique in Real Time Vision System for Automatic Weeding Strategy. 2007 IEEE International Symposium on Digital Signal Processing and Information Technology. 632-635.

Navalgund, R.R., Jayaraman, V., & Roy, P. S. (2007). Remote sensing applications: an overview. *current science*, *93*(12), 1747-1766.

Neethirajan, S., Karunakaran, C., Symons, S., & Jayas, S.D. (2006). Classification of virtuousness in durum wheat using soft X-rays and transmitted light images. *Science Direct- Computers and Electronics in Agriculture*, *53*, 71–78

Nosrati, M., Raoufact, M.H., & Fafigh, A. (2013). Use of Site-Specific Strategies for Weed Management to Reduce Herbicide application. *Anals of Biological Research. 4*(7), 135-144.

Omid, M. Abbasgolipour, M., Keyhani, A., & Mohtasebi, S.S. (2010). Implementation of an Efficient Image Processing Algorithm for Grading Raisins. *International Journal of Signal and Image Processing*, *1*(1), 31-34

Özlüoymak, Ö. B., Bolat, A., Bayat, A., & Güzel. E. (2019). Design, development, and evaluation of a target-oriented weed control system using machine vision. *Turkish Journal of Agriculture and Forestry*, *43*(2), 164-173.

Prema, P. (2016). A Novel approach for weed classification using curvelet transform and tamura texture feature (CTTTF) with RVM classification. *International Journal of Applied Engineering Research*.

Sonawane, P., Hassan, S., Ingale, P., & Hankare, S. (2021). Autonomous Farmbot: Weed Detection for Smart Farming Using CNN. *International Journal of Advanced Research in Science, Engineering and Technology*, *8*(6), 17516-17522.

Takhmatulin, I., & Andreasen, C. (2010). A Concept of a Compact and Inexpensive Device for Controlling Weeds with Laser Beams. *Agronomy. 10*(1616), 1-18.

Tewari, V.K., Kumar, A.A., Nare, B., Prakash, S., & Tyagi, A. (2014). Microcontroller based roller contact type herbicide applicator for weed control under row crops. *Computers and Electronics in Agriculture*, *104*, 40-45.

Thorp, K. R., & Tian, L.F. (2004). Performance Study of Variable-Rate Herbicides Application Based on Remote Sensing Imagery. *Biosystems Engineering, 88*(1), 35-47

Tian, L., Reid, J. F. & Hummel, J. W. (1999). Development of a Precision Sprayer for Site-specific Weed Management." *Transactions of the ASAE*, *42*(4), 893

Weis, M., & Gerhards, R. (2017) "Detection of Weeds using Image Processing and Clustering." *Image Analysis for Agricultural Products & Process.* 138-144.

Xiong, Y., Ge, Y., Liang, Y., & Blackmore, S. (2017). Development of a prototype robot and fast path-planning algorithm for static laser weeding. *Comput. Electron. Agric*. *142*, 494–503.

Xavier P. B., Ribeiro, A.A., & Pajares, G. (2010). Analysis of natural images processing for extraction of agriculture elements. *Elsevier- image and vision computing*, *28*, 138-149.