Harnessing AI and Computer Vision for Efficient Weed Control in Crop fields

Henry Mutegeki
Makerere University,
Department Of Computer Science
henrymutegeki117@gmail.com/henry.mutegeki@students.mak.ac.ug
Student Number: 2200701327 Registration Number: 2022/HD05/1327U

Abstract

In the realm of agricultural biodiversity, the accurate identification and classification of crop varieties and weed species is a complex task. The structural similarities among these species further complicate the task, making it a daunting challenge even for seasoned botanists. This complexity, coupled with the pressing need for efficient crop management and weed control, necessitates the development of sophisticated AI models to solve for the problem.

The goal of this research is to develop a crop field management tool capable of detecting and identifying specific crops and the presence of grass(weeds). In this paper, we present the development of AI models, specifically Gaussian Naive Bayes and Convolutional Neural Networks (CNNs), for precise crop and weed classification in agricultural settings, focusing on cassava, maize, and sugarcane. We explore the impact of different feature extraction methods, including color histogram and edge detection, on the performance of the Gaussian Naive Bayes model. Results indicate that color histogram-based features yield the highest accuracy for the Gaussian Naive Bayes model, with an accuracy 75% compared to the edge detection based model at 60%. However, the CNN model outperforms the traditional model, achieving an accuracy of over 90%. This superior performance is attributed to CNN's ability to automatically learn and extract high-level features from the input data.

The findings underscore the potential of AI and machine learning in enhancing precision and efficiency in agriculture, paving the way for more sustainable farming practices aimed at weed control.

1. Introduction

The global population has experienced exponential growth, surpassing eight billion individuals and is projected to escalate further and potentially reach a staggering 9.8 billion by the year 2050 according to United Nations(UN) report 2022[1]. The current levels of crop production are insufficient to provide enough food for the increasing population. Meeting the projected demand in order to ensure sufficient food supply poses a significant challenge for humanity. [2]. Crop fields often face challenges in weed control, which can significantly impact agricultural production efficiency[3]. Weeds compete with crops for resources such as sunlight, water, and nutrients, leading to reduced yields and economic losses. The conventional methods of weed control, such as manual labor or the use of herbicides, are time-consuming, expensive, and can have negative effects on the environment and human health[4].

Computer vision and image analysis techniques have found applications in various areas of agriculture, such as precision farming, weed detection and control, agricultural pattern analysis, and automated inspection of agricultural products[5]. Harnessing AI and computer vision technologies for efficient weed control in crop fields offers a promising approach to increase agricultural production efficiency[6]. By accurately identifying and targeting weeds, AI-powered systems can minimize the use of herbicides, reducing environmental pollution and the risk of harm to non-target species. This approach of precision agriculture enables farmers to optimize their weed control efforts, resulting in higher crop yields and improved profitability. Moreover, the integration of machine vision with actuators and computer algorithms has paved the way for automation in agriculture. This automation has greatly reduced the labor-intensive nature of weed control tasks and made them more cost-effective. Integrating computer vision technology with weed control robots allows for precise and targeted elimination of weeds[7]. These robots are capable of autonomously navigating through crop fields, thanks to improved computer vision algorithms that accurately detect and localize crops rows. With the ability to distinguish between crops and weeds, AI-powered systems can selectively apply herbicides or activate mechanical cultivators or fingers to remove weeds. Harnessing AI and computer vision for efficient weed control in crop fields has the potential to revolutionize the agricultural industry. This approach not only addresses the challenges associated with traditional weed control methods but also contributes to sustainable agricultural practices.

This research paper presents two machine learning models that aim to identify and detect weeds in gardens where specific food crops such as cassava, maize, and sugarcane are cultivated. The proposed models utilize advanced techniques from the field of machine learning to accurately identify weed species within these agricultural plots. By employing these models, farmers can effectively differentiate between unwanted weeds and their desired food crops. This technology offers a valuable tool for enhancing weed management practices by enabling timely interventions to mitigate the negative impacts of weeds on crop growth and yield.

2. Literature Review

Over the years, machine learning and deep learning have been extensively applied in image classification tasks. The methodologies employed span a wide range, including K-Nearest Neighbors (KNN), Naive Bayes and Support Vector Machines (SVM). Additionally, various deep learning models have been utilized, further expanding the scope and capabilities of image classification techniques. This section presents existing works that has been done in weed identification and detection in crop fields for smart Agriculture. The detection & identifaction methods have largely been categorized into two: traditional Machine learning Techniques & Deep Learning techniques.

2.1. Traditional Machine learning Techniques

2.1.1 Feature Extraction in Traditional ML

Feature extraction in image analysis is a critical process that involves identifying and extracting significant patterns, structures, or attributes from images. The aim is to transform raw pixel data into more sophisticated representations that encapsulate relevant information for tasks such as object recognition, classification, and image understanding. This process enables the creation of more efficient and precise algorithms for image analysis. Traditional weed detection methods based on image processing leverage the feature differences between plant leaves and weeds. This section compares the four traditional image features: texture, shape, spectrum, and color.

2.1.2 Texture Features

Texture features reflect the spatial distribution among pixels and have been widely used in image classification [8]. They can effectively distinguish crops and weeds due to the diverse vein texture and leaf surface roughness information. Texture feature methods can be categorized into statistical, structural, model-based, and transform-based methods. However, these techniques may not perform reliably in complex natural scenarios, such as high weed density, overlapping, or obscured weeds and crops. For instance, Bakhshipour et al. [9] extracted 52 texture features from wavelet multiresolution images for weed segmentation, but the technique struggled with high weed density and overlapping scenarios.

2.1.3 Shape Features

Shape features play a crucial role in weed detection image analysis. They include shape parameters, region-based descriptors, and contour-based descriptors. Shape parameters are the most intuitive, easy to implement, and unaffected by lighting. However, the shape of leaves can be distorted by disease, insects, and even human and mechanical damage. Therefore, relying solely on shape features for weed identification can be challenging, especially in field environments where overlap or occlusion of plant leaves occur. For example, Pereira et al. [10] used five shape descriptors in shape analysis to describe the contour shape of aquatic weeds, but the method struggled with distorted leaf shapes due to disease or insect damage.

2.1.4 Spectral Features

Spectral features are effective in distinguishing plants with different leaf colors. They are robust to partial occlusion and tend to decrease in calculation. However, during the growth and development stages of plants, the interaction between light and observed geometry and leaf angle distribution, as well as the variability of the spectral features of plant species, can affect hyperspectral detection. For instance, Pignatti et al. [11] distinguished corn crops and weeds using spectral indices, but the method struggled when the spectral difference between crops and weeds was unobvious or the leaf reflection was affected by moisture, plant disease, growth period, and other factors.

2.1.5 Color Features

Color-based detection highly depends on the plant being studied and its color differences. It is a common method used to segment plants from the background by using the difference in color features. However, color is the most unstable feature used for plant identification. When the color difference is unobvious, color-based methods may not be able to distinguish weeds from crops accurately. These methods can be affected by leaf disease, plant seasonal changes in color, or different lighting conditions. For example, Tang et al. [12] used the YCrCb color space to describe the green features of green crops under different illumination conditions, but the method struggled with plant seasonal changes in color and different lighting conditions.

2.1.6 Related works using Traditional ML weed detection

Chen and his colleagues [13] developed an enhanced image classification method for weed identification, leveraging the K-Nearest Neighbors (KNN) algorithm in conjunction with Gabor Wavelet (GW) and regional covariance Lie group structure. Their approach was applied to classify four types of broad-leaved weed images, achieving an impressive overall recognition accuracy of 93.13%.

Rainville presents a method for weed/crop classification that leverages computer vision and morphological analysis. This approach begins with the extraction of features from leaves, utilizing the cultivation inter-rows for weed identification. Subsequently, crop features are inferred from the resulting mixture model. These extracted features are then processed through a Naive Bayes classifier and a Gaussian mixture clustering algorithm to distinguish weeds from crops. The method demonstrated a high degree of accuracy, achieving a 94% success rate for corn and soybean plants, and an 85% success rate for weed identification [14].

Ahmed and his team [15] employed the Support Vector Machines (SVM) algorithm for the identification of six different weed types within a dataset of 224 images. The optimal combination of their feature extractor was able to achieve an accuracy rate of 97.3%.

In a separate study, Rumpf et al. [16] introduced a sequential classification method that utilized three distinct SVM models. This approach was not only capable of distinguishing between weeds and barley but also effectively differentiated between monocotyledon and dicotyledon plant weeds.

2.2. Deep Neural Networks

Deep learning methods, notably Convolutional Neural Networks (CNNs) and Fully Convolutional Networks (FCNs), have demonstrated promising results in weed detection and classification, surpassing traditional machine learning methods, particularly in complex natural environments. Deep learning's unique network feature structure allows for more effective feature extraction than manual methods. It synthesizes local features from the bottom and higher-level features from the top, which can cater to various tasks [17]. In weed detection, deep learning methods utilize spatial and semantic feature differences to improve the accuracy of weed identification and detection [18]. However, the requirement for large datasets for training underscores the challenges of deep learning methods for weed identification.

2.2.1 CNNs in Weed Detection and Identification

CNNs have been increasingly utilized in weed detection, with several studies demonstrating their effectiveness. For instance, Potena et al. used two different CNNs to process RGB and NIR images for rapid and accurate identification of crops and weeds [19]. Beeharry and Bassoo compared the performance of two weed detection algorithms based on UAV images, ANN and AlexNet, with AlexNet achieving an accuracy of over 99% [20]. These methods, which do not rely on image preprocessing and data conversion, have shown better recognition accuracy than traditional machine learning methods.

Various CNN frameworks, such as AlexNet, ResNet , VGG, U-Net, MobileNets, and DenseNet, have also been widely used in weed detection. These methods have outperformed other conventional index-based methods in various conditions [21].

2.2.2 FCNs in Weed Detection and Identification

FCNs, which learn features and implement forward and reverse processes in an end-to-end manner, have made significant strides in computer vision and remote sensing applications. For instance, Dyrmann et al. proposed a method using an FCN to automatically detect weeds in color images under severe occlusion [22]. Ma et al. proposed a SegNet semantic segmentation method based on FCNs for weed detection in rice fields, which showed significantly higher accuracy than the classic FCN model and U-Net model [23]. FCNs have been used for semantic-level image segmentation and pixel-level classification of images, advancing the problem of weed segmentation. However, this method only classifies each pixel without considering the relationship among pixels.

3. Methodology

Numerous studies highlighted in this literature review demonstrate that both traditional machine learning (ML) methods (when paired with appropriate feature extraction techniques) and deep learning models yield high accuracy in detecting and identifying weeds in gardens. The primary objective of this study is to design accurate models capable of detecting weeds(grass) and differentiating them from major food crops such as cassava, maize, and sugarcane. For the traditional ML algorithm, we explored the Naive Bayes while for deep learning, we explored CNNs.

3.1. Dataset Description

To compile the dataset for this study, a video that had been recorded in a garden with a variety of crops such as cassava, sugarcane, maize, jackfruit, bananas and weeds(grass) was then converted into a series of images using the cv2.VideoCapture function in OpenCV with the video file name and its extension as the argument.

The frame rate was set to 0.5, meaning that the function would capture a frame every 0.5 seconds, resulting in two frames (or images) per second. This process generated a sequence of approximately 1000 images.

These images were further processed using a cropping tool to focus on areas within the images that included the three primary crops (cassava, maize, and sugarcane) and grass. This resulted in a collection of 298 images for cassava, 280 images for maize, 102 images for sugarcane, and 287 images for grass, totalling 967 images of varying sizes.

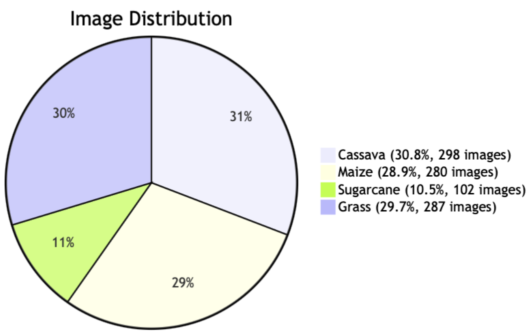

Fig 1. Crops- Weeds/Grass Image Distribution

3.2. Data Preprocessing

Prior to utilizing the dataset for model training, a data transformation/preprocessing step was conducted in order to normalize the data by resizing the images. This not only facilitates the feature extraction process but also enhances the learning capability of the model. For traditional machine learning models, the images were resized to 500 x 500 pixels, while for CNN-based deep learning models, the images were resized to 256 x 256 pixels. By standardizing the image sizes, the models can effectively process the data and learn from it in a more efficient manner.

3.3. Feature Extraction & Model Training

The process of extracting features from images is a critical component in the development of machine learning models for computer vision tasks. This process transforms the raw visual data into a more manageable and meaningful format, which aids machine learning algorithms in learning from the data more effectively.

There are numerous methods for extracting features from images, including traditional techniques such as color histograms and edge detection, as well as more complex methods like Scale-Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). Additionally, simple pixel values can also serve as features. These methods allow for the extraction of various types of features from images, including color distributions, edges or boundaries, key points, and raw pixel intensities.

For the crop and weed classification problem, we choose to use color histograms and edge detection as our primary feature extraction methods in training the Naive Bayes model. Color histograms provide a simple yet effective representation of the color distribution in an image, which can be particularly useful for tasks involving color-based object recognition or classification. Edge detection, on the other hand, helps in identifying boundaries and structural information in an image, which can be crucial for tasks like object detection and image segmentation. These two methods, when combined, provide a comprehensive set of features that capture both color and structural information in the images, making them well-suited for our task.

Data splitting was done on the image dataset with 75% of the data used for training while 25% of the same image dataset was used for testing. 20% of the training data was used for model validation for the deep learning model.

3.3.1 Traditional ML: Naive Bayes

Two different models were trained using different feature sets, that is color histograms, and egde detection. An assessment of results from all models was done using features from the test dataset to obtain one with the best performance.

1. **Applying the Color histogram feature extraction technique**

   In our approach, we utilized the OpenCV library, a powerful tool for computer vision tasks, to calculate the color histogram of our images. We began by reading the image into an array of pixel values using OpenCV's `imread()` function. To focus on the color information, we separated the image into its constituent color channels - Red, Green, and Blue - using the `split()` function. For each of these color channels, we then computed a histogram using the `calcHist()` function. This histogram provides a graphical representation of the tonal distribution in the image, effectively capturing the frequency of each color intensity level. However, raw histograms can be sensitive to changes in illumination, which could potentially affect the performance of our machine learning model. To mitigate this, we normalized the histograms using the `normalize()` function, scaling the histogram values to a specified range. This process resulted in a set of features for each color channel, representing the distribution of pixel intensities. These features, encapsulating crucial color information, were then ready to be fed into our machine learning model for further tasks such as image classification or object recognition. Fig 2. shows sample color histogram extracts.

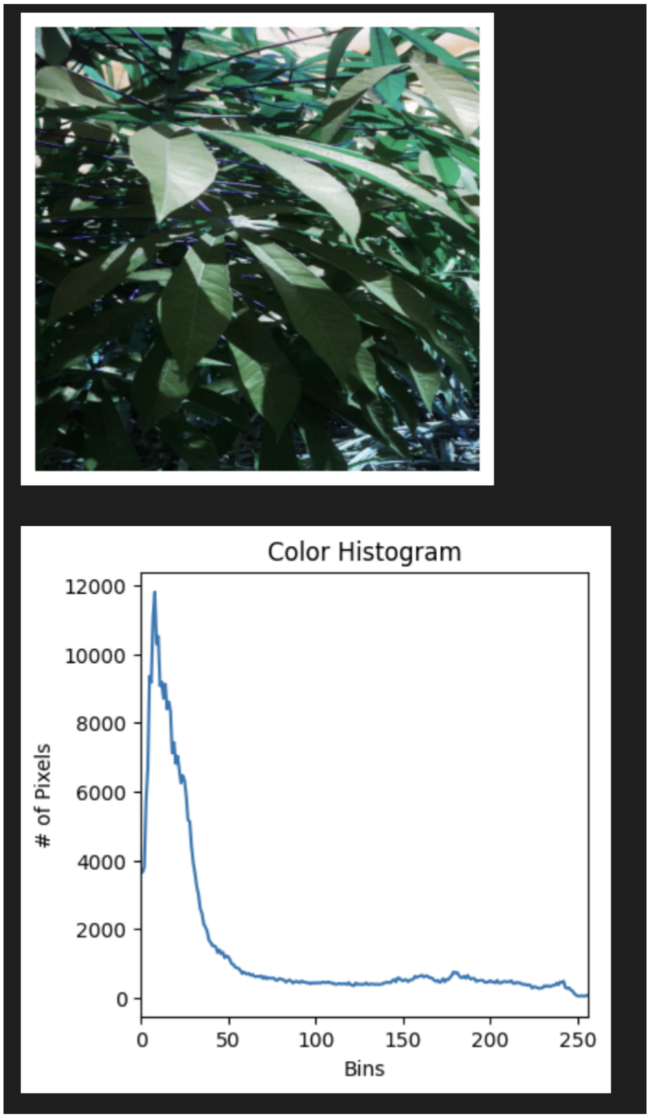

Fig 2. Color histogram for the cassava

2. **Applying the Edge detection feature extraction technique**

   In our approach, edge detection was performed using the OpenCV library, a popular tool for computer vision tasks. The process began by reading the image into an array of pixel values using OpenCV's `imread()` function. To perform edge detection, we used the Canny edge detection method, a multi-stage algorithm that is highly effective in identifying a wide range of edges in images. The `Canny()` function in OpenCV

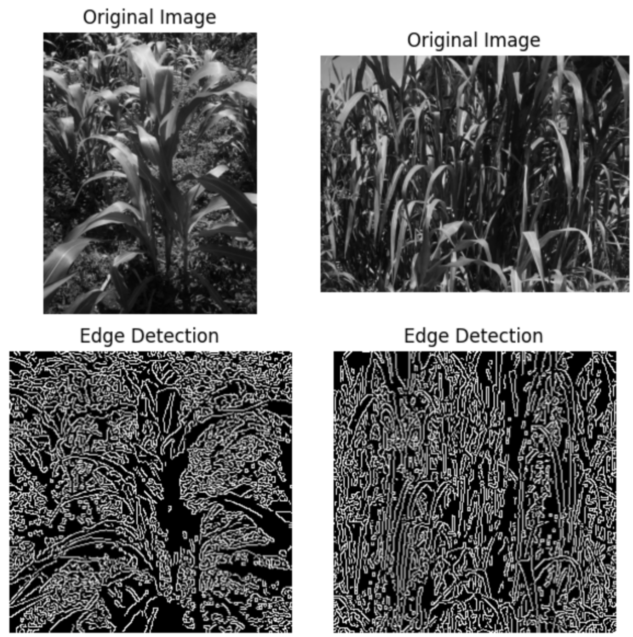

Fig 3.1 Feature Extraction with edge detection

was used for this purpose. The Canny edge detection algorithm works by first applying a Gaussian blur to the image to reduce noise and detail. It then finds the image gradient to highlight regions with rapid intensity changes, which typically correspond to edges. Non-maximum suppression is then applied to isolate the best edge pixels before using a double threshold to determine potential edges. Edge tracking by hysteresis is applied, which suppresses weak edges that are not connected to strong edges. The output of the Canny edge detection is a binary image where white pixels represent edges and black pixels represent non-edges. This edge-detected image serves as a set of features that capture the structural information in the image, which can then be used for further tasks such as image classification or object recognition. Fig 3 shows some results from applying edge detection.

3.3.2 Deep Learning: CNN

The Convolutional Neural Network (CNN) model that was trained for this task is a sequential model, meaning that the layers are stacked linearly. The model is designed using the TensorFlow's Keras API and consists of several layers, each serving a specific purpose. The first layer is a 2D convolutional layer with 32 filters, each of size 3x3. This layer uses a Rectified Linear Unit (ReLU) activation function and is designed to work with input images of size 256x256 with 3 color channels (RGB). The convolutional layer scans the input image with its filters to create feature maps that rep-

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_5 (Conv2D)           (None, 254, 254, 32)      896

 max_pooling2d_5 (MaxPooling  (None, 127, 127, 32)     0
 2D)

 conv2d_6 (Conv2D)           (None, 125, 125, 64)      18496

 max_pooling2d_6 (MaxPooling  (None, 62, 62, 64)       0
 2D)

 conv2d_7 (Conv2D)           (None, 60, 60, 128)       73856

 max_pooling2d_7 (MaxPooling  (None, 30, 30, 128)      0
 2D)

 flatten_2 (Flatten)         (None, 115200)            0

 dense_4 (Dense)             (None, 128)               14745728

 dense_5 (Dense)             (None, 4)                 516

...
Total params: 14,839,492
Trainable params: 14,839,492
Non-trainable params: 0
```

Fig 3.2 CNN model architecture

resent the input image in a more compact and abstract way. Following the first convolutional layer, a 2D max pooling layer is applied with a pool size of 2x2. This layer reduces the spatial dimensions of the input (height and width) while preserving the most important features, thereby helping to reduce computational complexity and control overfitting. The model then repeats this pattern of a convolutional layer followed by a max pooling layer two more times. The second convolutional layer has 64 filters of size 3x3, and the third convolutional layer has 128 filters of size 3x3. Both of these layers use the ReLU activation function. After the convolutional and pooling layers, the model has a flatten layer, which transforms the 2D matrix data into a 1D vector to prepare it for input into the fully connected layers. The next layer is a dense (fully connected) layer with 128 neurons, again using the ReLU activation function. This layer is designed to interpret the features extracted by the preceding layers. Finally, the model has a dense output layer with 4 neurons, corresponding to the four classes in the dataset. This layer uses a softmax activation function, which outputs a probability distribution over the four classes, making the model's output interpretable as class probabilities.

The model was trained for 10 epochs, with a batch size of 32 and then evaluated.

4. Results & Discussion

4.1. Naive Bayes Model Results

1. **Naive Bayes with Edge Detection Features:** This model applied canny edge detection for feature extraction and the features used to build a Naive Bayes model. The model registered an overal accuracy of 60% and based on the confusion matrix and classification report generated, it was able to correctly classify the crops and grass as seen in See fig.4.1 and fig. 4.2

	precision	recall	f1-score	support
0	0.55	0.62	0.58	60
1	0.64	1.00	0.78	58
2	0.62	0.41	0.49	56
3	0.00	0.00	0.00	21
accuracy			0.61	195
macro avg	0.45	0.51	0.46	195
weighted avg	0.54	0.61	0.55	195

Fig 4.1 Classification Report for Naive Bayes: Edge Detection

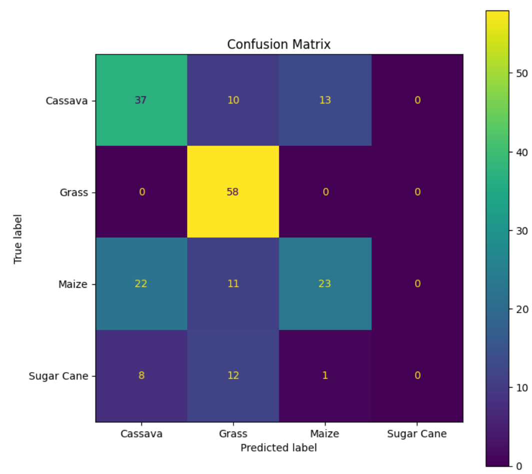

Fig 4.2. Confusion Matrix Naive Bayes: Edge Detection

2. **Naive Bayes with Color Histogram Features:** This model applied color histogram for feature extraction and the features used to build a Naive Bayes model. The model registered an overal accuracy of 75% and based on the confusion matrix and classification report generated, it was able to correctly classify the crops and grass as seen in See fig. 5.1 and fig 5.2

3. **Naive Bayes with combined Color Histogram and Edge Detection Features:**

   This model applied a combination of canny edge detection and color histogram for feature extraction and the features used to build a Naive Bayes model. The model registered an overal accuracy

	precision	recall	f1-score	support
0	0.82	0.62	0.70	60
1	0.80	0.83	0.81	58
2	0.62	0.80	0.70	56
3	0.89	0.76	0.82	21
accuracy			0.75	195
macro avg	0.78	0.75	0.76	195
weighted avg	0.77	0.75	0.75	195

Fig 5.1 Classification Report for Naive Bayes: Color Histogram

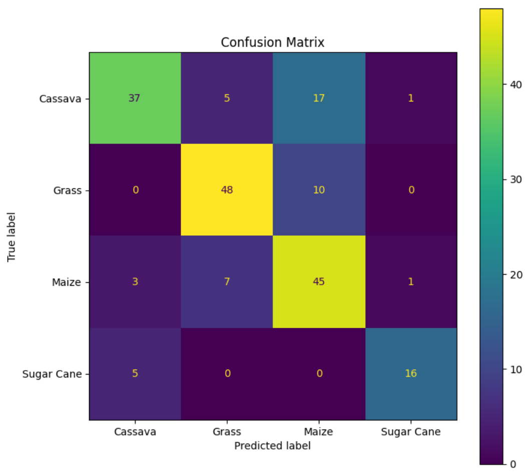

Fig 5.2 Confusion Matrix Naive Bayes: Color histogram

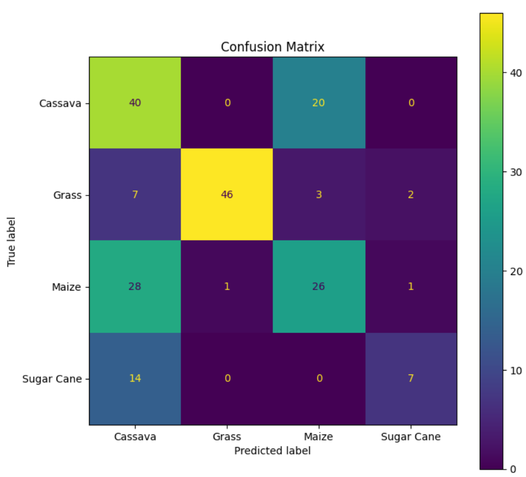

Fig 6.2 Confusion Matrix Naive Bayes: Combined Edge Detection and Color histogram

of 62% and based on the confusion matrix generated and classification report, it was able to correctly classify the crops and grass as seen in See fig 6.1 & 6.2

	precision	recall	f1-score	support
0	0.45	0.67	0.54	60
1	0.98	0.79	0.88	58
2	0.53	0.46	0.50	56
3	0.70	0.33	0.45	21
accuracy			0.61	195
macro avg	0.66	0.56	0.59	195
weighted avg	0.66	0.61	0.62	195

Fig 6.1 Classification Report for Naive Bayes: Combined Edge Detection and color histogram

4.2. CNN Model Results

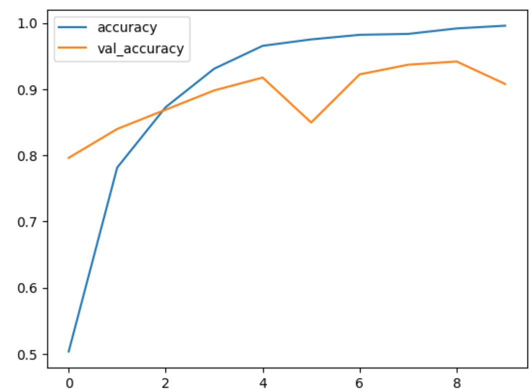

Fig 7.1 CNN learning curve

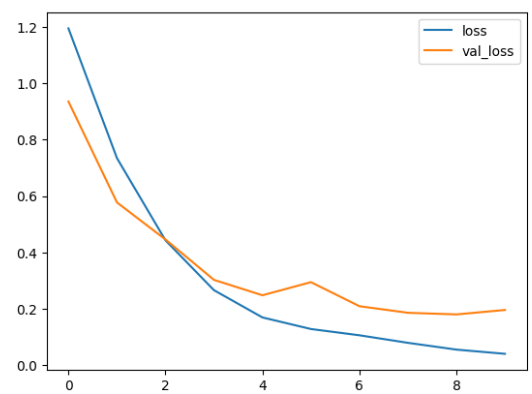

Fig 7.2 CNN loss curve

The CNN model outperfomed all the traditional ML models accross all fronts of the classification report and registered the highest accuracy of 90%. The confusion matrix also shows how it was able to correctly classify 90% of all the crops and grass which was very impressive.

	precision	recall	f1-score	support
0	0.81	0.93	0.86	82
1	0.99	0.93	0.96	72
2	0.90	0.83	0.86	77
3	0.96	0.90	0.93	29
accuracy			0.90	260
macro avg	0.91	0.90	0.90	260
weighted avg	0.90	0.90	0.90	260

Fig 7.3 CNN classification report

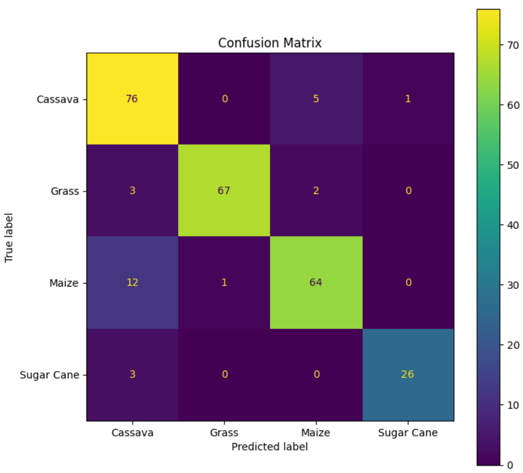

Fig 7.4 CNN Confusion Matrix

Its learning and loss curves showed the steady progress of the model as it kept learning different features of the dataset.

4.3. Results Discussion

The results of the different models used in this study demonstrate varying levels of accuracy in classifying the data. Model 1, which employed Naive Bayes with an edge detection feature extraction technique, achieved an accuracy of 61%. This accuracy level suggests that the edge detection technique alone may not be sufficient for accurately classifying the data in this particular context.

Model 2, on the other hand, utilized Naive Bayes with a color histogram feature extraction technique and achieved an accuracy of 75%. This indicates that the color histogram approach provides more discriminative information for classification compared to the edge detection technique used in Model 1. The higher accuracy suggests that the color distribution of the data is more indicative of the classes being considered.

Interestingly, when both the edge detection and color histogram techniques were combined in Model 3, the accuracy dropped to 62%. This suggests that the combination of these techniques may not be complementary or may introduce noise into the feature extraction process. Further investigation and experimentation are recommended to understand why the combined approach did not yield improved results.

Lastly, the CNN model (Model 4) achieved the highest accuracy of 90%. This indicates that the deep learning approach provided superior capabilities for feature extraction and classification in this particular task. The CNN model's ability to automatically learn discriminative features from the data likely contributed to its high accuracy.

Based on these results, it is recommended to focus on the color histogram feature extraction technique, as it showed promising results in Model 2. Further improvements could be explored by optimizing the parameters and exploring different variations of the color histogram technique. Additionally, the high accuracy achieved by the CNN model suggests that deep learning approaches should be further investigated and potentially expanded upon for this classification task.

5. Conclusion

This research has demonstrated the potential of machine learning and deep learning models in the classification of crops and grass for effecient weed control in gardens. The results indicate that the choice of feature extraction technique significantly impacts the performance of the models. While edge detection showed some promise, it was the color histogram technique that yielded better results when used with the Naive Bayes classifier. However, combining these two techniques did not enhance the performance as expected, suggesting that they may not be complementary or that the combination might introduce noise into the system.

Interestingly, the Convolutional Neural Network (CNN) model outperformed all other models, achieving the highest accuracy. This underscores the power of deep learning in automatically extracting and learning relevant features from the data, thereby enhancing the classification task's accuracy.

Despite these promising results, it is clear that there is room for improvement. Future work should focus on optimizing the parameters of the color histogram technique and exploring its different variations. Additionally, given the superior performance of the CNN model, further research should delve deeper into deep learning approaches, exploring how they can be fine-tuned and expanded for this classification task. This study serves as a stepping stone towards the development of more accurate and efficient models for crop and weed classification, which is crucial for precision agriculture.

References

[1] ”World population to reach 8 billion this year, as growth rate slows”, news.un.org, (Accessed 1 Jul. 2023).

[2] J. H. Westwood, R. Charudattan, S. O. Duke, S. A. Fennimore, P. Marrone, D. C. Slaughter, and R. Zollinger, ”Weed management in 2050: Perspectives on the future of weed science,” Weed Science, vol. 66, no. 3, pp. 275-285, 2018.

[3] M. Fontanelli, C. Frasconi, L. Martelloni, M. Pirchio, and R. Foster, ”Weed management in autumn fresh market spinach: A nonchemical alternative,” HortTechnology, vol. 25, pp. 177-184.

[4] J. H. Westwood et al., ”Weed management in 2050: Perspectives on the future of weed science,” Weed Science, vol. 66, no. 3, pp. 275-285, 2018.

[5] T. Pun, A. Neupane, R. Koech, ”Quantification Of Root-knot Nematode Infestation In Tomato Using Digital Image Analysis”, Agronomy, vol. 11, no. 12, p. 2372, 2021. https://doi.org/10.3390/agronomy11122372

[6] M. Guzel, B. Turan, İ. Kadıoğlu, B. Şin, A. Basturk, K. Ragab, ”How To Affect the Number Of Images On The Success Rate For Detection Of Weeds With Deep Learning”, Turkish JAF Sci.Tech., vol. 10, no. 8, p. 1441-1446, 2022. https://doi.org/10.24925/turjaf.v10i8.1441-1446.5183

[7] Z. Wu, Y. Chen, B. Zhao, X. Kang, Y. Ding, ”Review Of Weed Detection Methods Based On Computer Vision”, Sensors, vol. 21, no. 11, p. 3647, 2021. https://doi.org/10.3390/s21113647

[8] Y. Zhang, S. Wang, G. Ji, and P. Phillips, ”Fruit classification using computer vision and feedforward neural network,” J. Food Eng., vol. 143, pp. 167-177.

[9] A. Bakhshipour, A. Jafari, S. Nassiri, and D. Zare, ”Weed segmentation using texture features extracted from wavelet sub-images,” Biosyst. Eng., vol. 157, pp. 1-12, 2017.

[10] L. Pereira, R. Nakamura, G. Souza, D. Martins, and J. Papa, ”Aquatic weed automatic classification using machine learning techniques,” Comput. Electron. Agric., vol. 87, pp. 56-63.

[11] S. Pignatti, R. Casa, A. Harfouche, W. Huang, A. Palombo, and S. Pascucci, ”Maize crop and weeds species detection by using UAV VNIR hyper spectral data,” in Proceedings of the IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, pp. 7235-7238.

[12] J. Tang, X. Chen, R. Miao, and D. Wang, ”Weed detection using image processing under different illumination for site-specific areas spraying,” Comput. Electron. Agric., vol. 122, pp. 103-111, 2016.

[13] Y. Chen, P. Lin, Y. He, and Z. Xu, ”Classification of broadleaf weed images using Gabor wavelets and Lie group structure of region covariance on Riemannian manifolds,” Biosyst. Eng., vol. 109, pp. 220-227.

[14] F.-M. De Rainville et al., “Bayesian classification and unsupervised learning for isolating weeds in row crops,” Pattern Anal. Appl., vol. 17, no. 2, pp. 401–414, May 2014, doi: 10.1007/s10044-012-0307-5.

[15] F. Ahmed, H. Al-Mamun, A. Bari, E. Hossain, and P. Kwan, ”Classification of crops and weeds from digital images: A support vector machine approach,” Crop Prot., vol. 40, pp. 98-104.

[16] T. Rumpf, C. Römer, M. Weis, M. Sökefeld, R. Gerhards, and L. Plümer, ”Sequential support vector machine classification for small-grain weed species discrimination with special regard to Cirsium arvense and Galium aparine,” Comput. Electron. Agric., vol. 80, pp. 89-96.

[17] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, and K. Lewis, ”Sensors and systems for fruit detection and localization: A review,” Comput. Electron. Agric., vol. 116, pp. 8-19, 2015.

[18] T. Kounalakis, G. Triantafyllidis, and L. Nalpantidis, ”Deep learning-based visual recognition of rumex for robotic precision farming,” Comput. Electron. Agric., vol. 165, article 104973, 2019.

[19] C. Potena, D. Nardi, and A. Pretto, ”Fast and accurate crop and weed identification with summarized train sets for precision agriculture,” in Intelligent Autonomous Systems 14. IAS 2016, vol. 531, Advances in Intelligent Systems and Computing, pp. 105-121, 2017.

[20] Y. Beeharry and V. Bassoo, ”Performance of ANN and AlexNet for weed detection using UAV-based images,” in Proceedings of the 2020 3rd International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM), Balaclava, Mauritius, 25-27 November 2020, pp. 163-167.

[21] Ł. Chechliński, B. Siemiatkowska, and M. Majewski, ”A System for Weeds and Crops Identification—Reaching over 10 FPS on Raspberry Pi with the Usage of MobileNets, DenseNet and Custom Modifications,” Sensors, vol. 19, no. 17, article 3787, 2019.

[22] M. Dyrmann, R. Jørgensen, and H. Midtiby, ”RoboWeedSupport-Detection of weed locations in leaf occluded cereal crops using a fully convolutional neural network,” Adv. Anim. Biosci., vol. 8, pp. 842-847, 2017.

[23] K. Sudars, J. Jasko, I. Namatevs, L. Ozola, and N. Badaukis, ”Dataset of annotated food crops and weed images for robotic computer vision control,” Data Brief, vol. 31, article 105833, 2020.