



Article

Transfer Learning Approach to Seed Taxonomy: A Wild Plant Case Study

Nehad M. Ibrahim ¹, Dalia G. Gabr ², Atta Rahman ^{1,*}, Dhiaa Musleh ¹, Dania AlKhulaifi ¹
and Mariam AlKharraa ¹

¹ Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam 31441, Saudi Arabia; damusleh@iau.edu.sa (D.M.)

² Botany and Microbiology Department, Faculty of Science (Girls Branch), Al Azhar University, Cairo 11651, Egypt

* Correspondence: aaurrahman@iau.edu.sa

Abstract: Plant taxonomy is the scientific study of the classification and naming of various plant species. It is a branch of biology that aims to categorize and organize the diverse variety of plant life on earth. Traditionally, plant taxonomy has been performed using morphological and anatomical characteristics, such as leaf shape, flower structure, and seed and fruit characters. Artificial intelligence (AI), machine learning, and especially deep learning can also play an instrumental role in plant taxonomy by automating the process of categorizing plant species based on the available features. This study investigated transfer learning techniques to analyze images of plants and extract features that can be used to cluster the species hierarchically using the k-means clustering algorithm. Several pretrained deep learning models were employed and evaluated. In this regard, two separate datasets were used in the study comprising of seed images of wild plants collected from Egypt. Extensive experiments using the transfer learning method (DenseNet201) demonstrated that the proposed methods achieved superior accuracy compared to traditional methods with the highest accuracy of 93% and F1-score and area under the curve (AUC) of 95%, respectively. That is considerable in contrast to the state-of-the-art approaches in the literature.

Keywords: clustering; deep learning; plants taxonomy; AI; transfer learning; DenseNet201



Citation: Ibrahim, N.M.; Gabr, D.G.; Rahman, A.; Musleh, D.; AlKhulaifi, D.; AlKharraa, M. Transfer Learning Approach to Seed Taxonomy: A Wild Plant Case Study. *Big Data Cogn. Comput.* **2023**, *7*, 128. <https://doi.org/10.3390/bdcc7030128>

Academic Editors: Min Chen, Salvador García López, Christine Dewi and Rung-Ching Chen

Received: 29 April 2023
Revised: 22 June 2023
Accepted: 28 June 2023
Published: 4 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Plant taxonomy has been performed using morphological and anatomical characteristics, such as leaf shape, flower structure, and seed and fruit characters. Additionally, with the advent of molecular biology and genetics, it is now possible to use DNA analysis to aid in the classification of plant species [1]. The morphological characteristics of the seed play an important role in the identification and classification of the plant. These characteristics are collected and modified as a data matrix and used with a statistical program such as PRIMER to help the scientists to illustrate the relationship as a dendrogram. Recently, clustering algorithms have been used to discover underlying patterns in the image data and to form categories that can be used for image taxonomy. Image clustering is the process of grouping similar images together to simplify and organize large image datasets. Clustering has been extensively studied in the field of machine learning and computer vision. Deep learning has revolutionized the field of image clustering, enabling the more accurate and efficient clustering of large-scale image datasets. This research review will summarize the recent advancements in image clustering using deep learning. AI deep learning algorithms complement and enhance the traditional methods of plant taxonomy by providing a more efficient and objective approach to categorizing plant species. This can help in the preservation and protection of biodiversity and gaining a better understanding of the relationships between different plant species. Image clustering can be performed

using various algorithms such as k -means, hierarchical clustering, and spectral clustering. However, these methods have limitations in terms of accuracy and scalability. Deep learning-based approaches have shown promising results in overcoming these limitations. Deep learning algorithms are capable of automatically learning features from images, which can then be used for clustering. Considering that datasets are an essential part of model building and evaluation, and although the study sample is not representative of all wild plants and cannot generalize it to cultivated plants, they may need to apply to different species.

In the current study, two separate datasets consist of images of wild plant seeds collected from Egypt. After image preprocessing, such as image denoising, filtering, and resizing, convolutional neural networks (CNNs) are used to extract features from the images and clustering is consequently performed. The extracted features are based on the shape of the seed, the color of the seed, and other characteristics of the seed. The proposed clustering algorithm groups the plants based on their similarities in these features. Making and constructing a CNN model from scratch is a tedious job compared to utilizing the existing pretrained deep learning models. These models are pretrained on relatively larger datasets and better fine-tuned. As a result, depending on the application, multiple models may be used or retrained, a notion known as transfer learning. Fine-tuning includes adjusting the number of convolutional layers, the number of filters, the stride window length, the filter size, the max-pooling, and the inclusion of dropout between layers [2]. A variety of pretrained CNN models are available for prediction, feature extraction, and fine-tuning. The performance of CNN models is heavily dependent on their architecture. For instance, AlexNet [3] is one of the best-known CNN architectures that performs very well in image classification, followed by VGG-16 [4], ResNet [5], DenseNet [6], and MobileNet [7], respectively. The current study picks up the best models among them based on a comprehensive literature review and authors' experience.

The rest of the article is sectioned as follows: Section 2 presents a review of the related literature. Section 3 is dedicated to presenting the proposed transfer learning approach. The results and discussion are provided in Section 4, while Section 5 concludes the paper.

2. Review of the Literature

Deep learning is a subfield of machine learning that is focused on developing artificial neural networks with multiple layers to model complex data. Over the past few decades, deep learning has made significant advances in various domains, such as computer vision, natural language processing, speech recognition, and robotics.

2.1. Classification Using Deep Learning

CNNs are a class of neural networks that are particularly well-suited for image and video recognition tasks [8]. In recent years, researchers have developed numerous variations of CNNs, such as residual networks (ResNets), inception networks, and attention mechanisms. These networks have achieved state-of-the-art results in image classification, object detection, and semantic segmentation tasks. Another technique used a deep learning approach to learn discriminative features from leaf images with classifiers for plant identification [9]. The authors in [10] introduced a new convolutional-neural-network-architecture-based model for classifying plant images.

The new method used in image classification is the transfer learning technique; three models are visible in [11], in which the AyurLeaf CNN model is assessed and compared to the AlexNet, Leaf, and fine-tuned AlexNet versions, with an accuracy of 96.76%. The study of plant taxonomy examines the classification of various plant species. The aforementioned study discovered that transfer learning enhances the performance of deep learning models, particularly those that employ deep features and use fine-tuning to produce better performance.

The authors in [12] presented an extension work to [13] with an adaptive algorithm that relies on deep adaptive CNNs, which are a class of neural networks that are particu-

larly well-suited for image and video recognition tasks. In recent years, researchers have developed numerous variations of CNNs, such as residual networks (ResNets), inception networks, and attention mechanisms. These networks have achieved state-of-the-art results in image classification, object detection, and semantic segmentation tasks. In [14], D-Leaf, a cutting-edge CNN-based strategy, was presented. The three pretrained CNN models—pretrained AlexNet, fine-tuned AlexNet, and D-Leaf—were applied, respectively. With respect to three publicly accessible datasets—the MalayaKew, Flavia, and Swedish Leaf Datasets—these techniques have an accuracy rate of 90–98%, enhancing the performance of classifying plant species.

Similarly, the authors in [15] presented Inception v3, ResNet50, and DenseNet201, used to further increase a dataset's diversity; they used a variety of augmentation operations on the dataset, which contained 256,288 samples, and a noisy set, with 1,432,162 samples. Currently, a pretrained AlexNet that has been improved is ranked fourth [16]. To provide thorough empirical guidance indicating that residual networks are easier to refine and can attain precision by significantly increasing depth with relatively lower complexity as evident in the ECG classification study [17].

An ensemble of these residual nets managed a 3.57% error rate on the ImageNet test range. An inception convolutional neural network (CNN) model-based network was used to pretrain the model with ImageNet and refine it with the PlantCLEF database. They combined the results of five CNNs after they had been tuned using randomly chosen database segments. On the other hand, the optimization of the hyperparameters was not finished [18]. The authors in [19] presented a design of a multi-input CNN for large-scale flower grading and achieved 89.6% accuracy by using the augmentation technique. The study in [20] proposed a method of reliably matching between various views of an object or scene using distinctive invariant features that can be extracted from images. An overview of techniques for identifying plant species and extracting features from leaf images was given in [21].

A similar study, [22], presented a proposed model which performed better when using validation data when compared to other well-known transfer learning techniques. In [23], the authors created a data article for a dataset containing examples of pictures of fruit and leaves from healthy citrus trees. Transfer learning is a machine learning technique in which knowledge learned from one task is applied to a related but different task [24]. Ibrahim et al. in [25] proposed a novel deep learning approach to fruit identification and its family classification based on a fruit image dataset. In this regard, two different datasets were used individually as well in an augmented form. Several deep learning models were investigated, and it was concluded that the proposed CNN model outperformed the other models with the highest accuracy of 99.82%. In transfer learning, the knowledge gained from a pretrained model is used as a starting point for training a new model, rather than starting from scratch [26–28].

2.2. Clustering Using Deep Learning

Clustering is a fundamental unsupervised machine learning technique that involves grouping similar data points together. However, evaluating the quality of clustering results can be challenging due to the absence of ground truth labels. Several clustering evaluation methods have been proposed in the literature to assess the effectiveness of clustering algorithms. One of the commonly used clustering evaluation methods is the silhouette score [29]. This method measures the degree of similarity between data points within clusters and dissimilarity between data points in different clusters. A higher silhouette score indicates that clustering is more effective.

The Rand index and adjusted mutual information [30] are two evaluation methods that compare the clustering results to a known ground truth clustering. The Rand index measures the similarity between the clustering results and the ground truth, while AMI adjusts for chance agreement. A higher Rand index or AMI indicates better agreement between the clustering results and the ground truth. In conclusion, there are several

clustering evaluation methods that can be used to assess the effectiveness of clustering algorithms. However, it is essential to keep in mind that each method has its strengths and weaknesses, and using multiple methods is often necessary to gain a comprehensive understanding of the clustering results. This approach proves advantageous in situations where there is a scarcity of data for the specific task at hand or when the task closely resembles an existing task for which ample data and computational resources are already accessible.

Several deep learning algorithms have been proposed for image detection, classification, and clustering, including CNNs, autoencoders, and generative adversarial networks (GANs) [31,32]. CNNs have been widely used for image clustering due to their ability to automatically learn hierarchical features from images. Deep embedded clustering (DEC) is a popular clustering algorithm based on CNNs that uses a two-stage process of unsupervised pretraining followed by clustering. Other CNN-based clustering algorithms include convolutional autoencoder clustering (CAE-C) and deep convolutional autoencoder clustering (DCAE-C) [33].

Autoencoders are neural networks that learn a compressed representation of the input data. They have been used for image clustering by training an autoencoder to reconstruct input images and using the learned encoder to generate feature vectors for clustering. Clustering using deep autoencoders (CDAs) is an example of an autoencoder-based clustering algorithm [34].

Based on the literature review, the following can be concluded:

1. Deep learning is among the potential successful candidates in image feature extraction.
2. In botanical studies, there is a dire need to investigate transfer learning algorithms for seed taxonomy.
3. Hierarchical clustering algorithms can be investigated for automated clustering seeds that can set potential applications for classification in the future.
4. The following are the most commonly used deep learning models for images: DenseNet121, DenseNet201, ResNet50V2, EfficientNetB6, EfficientNetB1, EfficientNetB0, MobileNetV2, EfficientNetB3, VGG16, VGG19, EfficientNetB5, EfficientNetB7, EfficientNetB2, and EfficientNetB4.

3. Proposed Transfer Learning Approach

Transfer learning allows a model to learn from a pretrained model, which reduces the amount of data and computational resources required to train the model from scratch. It can help improve the performance of a model by leveraging the knowledge gained from a pretrained model. Moreover, it can improve the generalization performance of a model, and it allows the model to learn from a larger and more diverse set of data [26–28]. In the context of plants, taxonomy involves classifying plants into different groups based on their morphological, physiological, and molecular characteristics.

This study proposes applying deep learning methods to extract characteristics from seed images, such as (Seed Color, Seed Texture, Seed Shape, Seed Margin, Hilum Position, Hilum Shape, Hilum Level, Coma Color, Coma Duration, and Coma Position). The system architecture consists of seven phases, as shown in Figure 1.

The diagram provides the steps involved in the proposed approach as well as the traditional approach. Data were collected from various sources comprising of images of various plant species. In the proposed approach, a dataset is preprocessed by image denoising, filtering, and resizing so it can become suitable for the following feature extraction block where various pretrained deep learning models have been employed. After that, clustering is performed as illustrated by the dendrogram. Finally, the evaluation takes place by means of contrasting with the traditional approach.

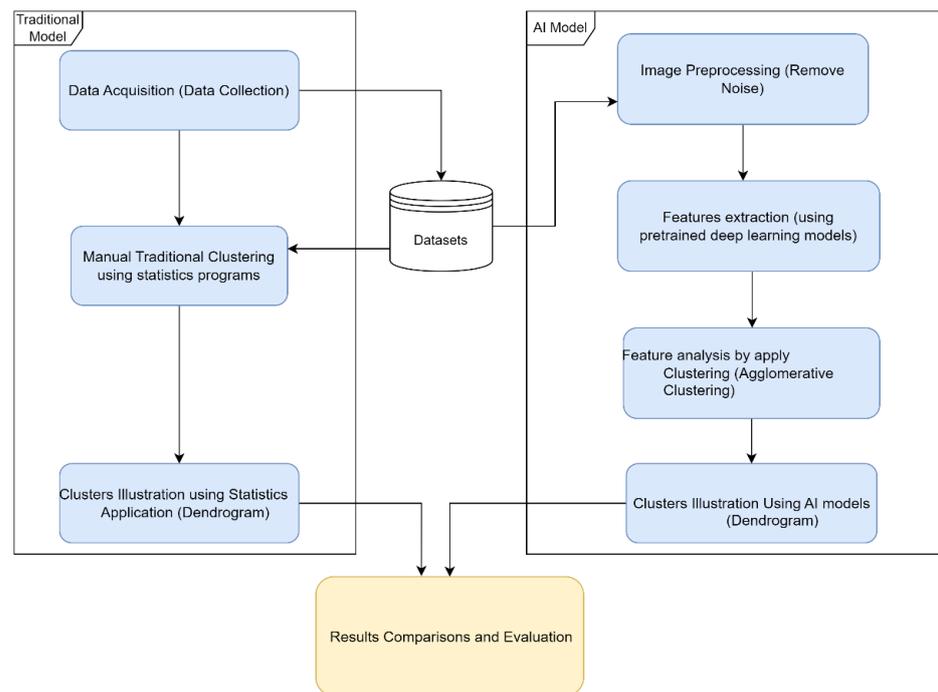


Figure 1. Proposed system architecture.

1. Common Processes:
 - a. Data acquisition.
2. Traditional Statistical Processes:
 - a. Feature extraction;
 - b. Clustering method.
3. Deep Learning:
 - a. Image preprocessing;
 - b. Feature extraction;
 - c. Deep learning clustering.

The proposed system shown in Figure 1 presents an AI model that utilizes a pretrained deep learning model to extract image features for the purpose of image clustering. The model aims to leverage the power of pretrained models, which have been trained on large-scale datasets, to extract high-level and discriminative features from images. These extracted features will then be used to group similar images together in a clustering algorithm; the following sections present these steps one by one.

3.1. Data Acquisition

The dataset of this research was seeds of three families (Brassicaceae, Apocynaceae, and Asclepiadaceae); the first family of samples include 22 seed images, and the second family includes 14. All these samples were collected from areas of wild plant families from the various deserts of Egypt, and the current study involved the first machine learning and AI experiments applied on this dataset. We aimed to take advantage of modern technology such as deep learning techniques to extract the features of these image samples without effort, since before, great effort was required to extract the features which are the core of the classification or clustering process.

Figure 2 shows the first family with 22 species images and Figure 3 shows the second family with 14 species. These species images were used in the current study.

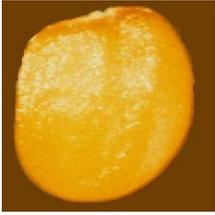
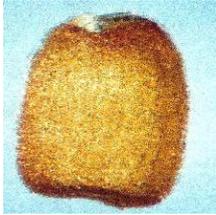
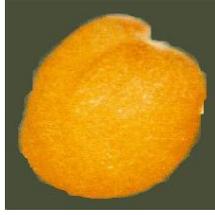
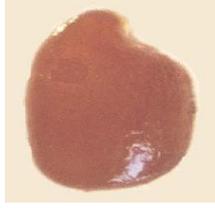
			
1— <i>Sisymbrium irio</i>	2— <i>Sisymbrium orientale</i>	3— <i>Eremobium aegyptiacum</i> var. <i>lineare</i>	4— <i>Matthiola longipetala</i> subsp. <i>livida</i>
			
5— <i>Matthiola incana</i>	6— <i>Farsetia aegyptia</i>	7— <i>Lobularia maritima</i>	8— <i>Diplotaxis harra</i> subsp.
			
9— <i>Brassica tournefortii</i>	10— <i>Brassica juncea</i>	11— <i>Brassica nigra</i>	12— <i>Eruca sativa</i>
			
13— <i>Erucaria hispanica</i>	14— <i>Cakile maritima</i>	15— <i>Cakile arabica</i>	16— <i>Zilla spinosa</i> subsp. <i>spinosa</i>
			
17— <i>Enarthrocarpus lyratus</i>	18— <i>Raphanus raphanistrum</i> subsp.	19— <i>Raphanus sativus</i>	20— <i>Moricandia nitens</i>
			
21— <i>Coronopus didymus</i>	22— <i>Capsella bursa-pastoris</i>		

Figure 2. First family with 22 species.

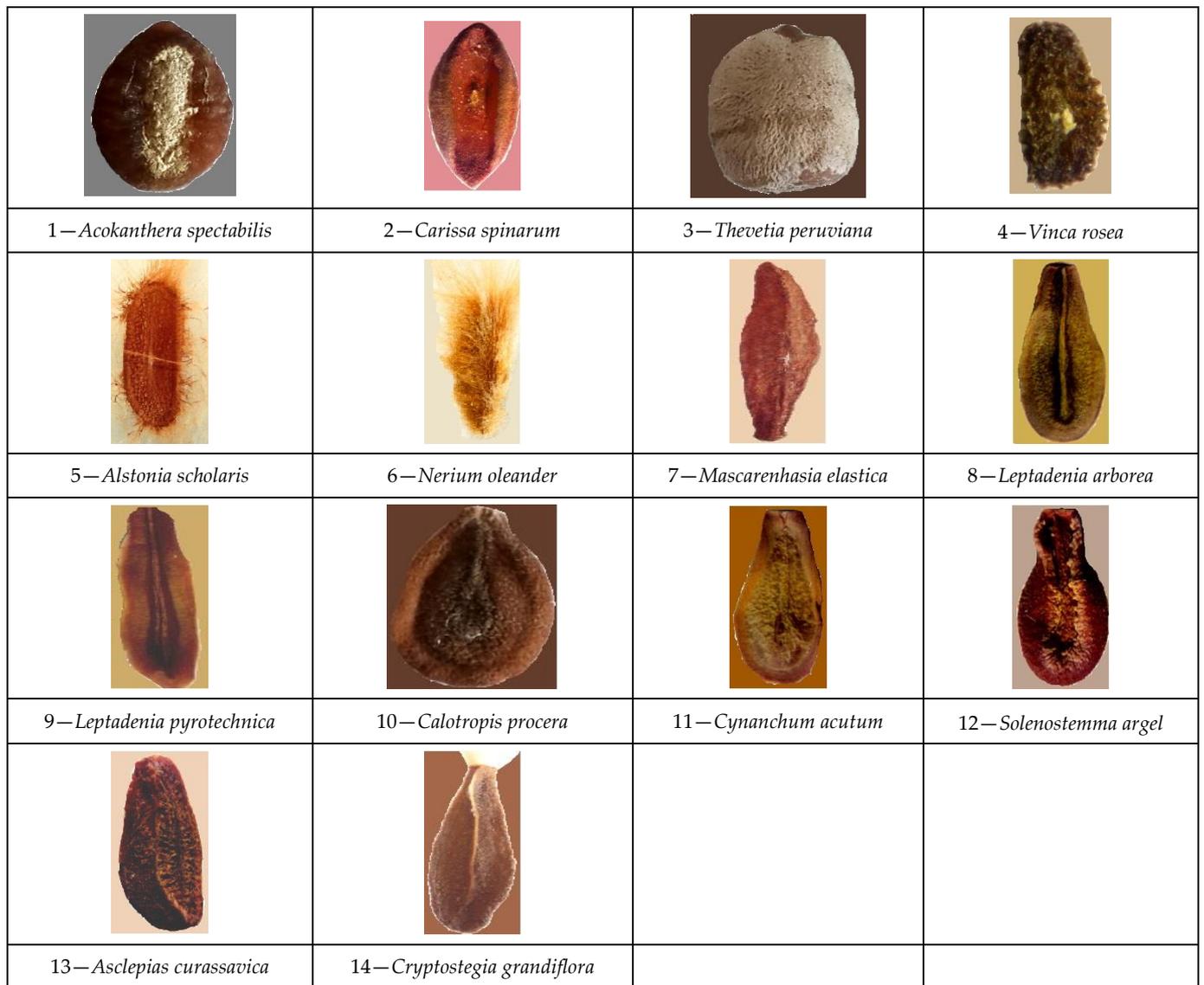


Figure 3. The second family with 14 species.

3.2. Traditional Statistical Clustering Approach

In the traditional statistical clustering method, the feature list descriptions for the first dataset are specified in Table 1. As already mentioned, the first dataset contained 22 species of Brassicaceae, as given in Figure 1. There were five features of each seed, as enlisted in Table 1, namely: color (with four possible values, each encoded as 1 to 4), texture (with possible four values from 1 to 4), shape (with five possible values ranging from 1 to 5), margin (with two possible values, 1–2, as seeds with a wing or no wing), and hilum position (with two possible values of terminal or subterminal). In this step, the features' values are encoded in decimals, depending on the possible outcomes (types).

Table 1. Feature list for species of Brassicaceae.

Feature	Values
Color	Brown [1]/yellowish-brown [2]/yellow [3]/dark brown [4].
Texture	Smooth [1]/reticulate [2]/tuberculate [3]/rough [4].
Shape	Oblong [1]/globose [2]/broad ovate–subglobose [3]/D-shaped [4]/kidney [5].
Margin	Seed winged [1]/not winged [2].
Hilum position	Subterminal [1]/terminal [2].

The features are listed in Table 2 with their explained encoding, which was used as an input to the statistical application to measure the distance between the studied species and subsequently demonstrate the result as dendrograms. The table presents the data matrix of the Brassicaceae seed morphological characters.

Table 2. Data matrix of Brassicaceae seed morphological characters listed in Figure 1.

#	<i>Sisymbrium irio</i>	<i>Sisymbrium orientale</i>	<i>Eremobium aegyptiacum</i> var. <i>lineare</i>	<i>Matthiola longipetala</i> subsp. <i>livida</i>	<i>Matthiola incana</i>	<i>Farsetia aegyptia</i>	<i>Lobularia maritima</i>	<i>Diplotaxis harra</i> subsp. <i>harra</i>	<i>Brassica tournefortii</i>	<i>Brassica juncea</i>	<i>Brassica nigra</i>	<i>Eruca sativa</i>	<i>Erucaria hispanica</i>	<i>Cakile maritima</i>	<i>Cakile arabica</i>	<i>Zilla spinosa</i> subsp. <i>spinosa</i>	<i>Enarthrocarpus lyratus</i>	<i>Raphanus raphanistrum</i> subsp. <i>raphanistrum</i>	<i>Raphanus sativus</i>	<i>Moricandia nitens</i>	<i>Coronopus didymus</i>	<i>Capsella bursa-pastoris</i>
1	3	2	3	1	1	1	1	3	1	4	1	2	2	1	1	1	4	1	1	1	3	1
2	1	1	3	1	4	1	1	3	2	2	2	2	1	2	1	1	2	2	2	1	1	1
3	1	1	1	1	3	3	3	1	2	2	2	3	1	4	1	1	1	3	3	1	5	1
4	2	2	1	2	2	1	1	2	2	2	2	2	2	2	1	2	2	2	2	2	2	2
5	1	1	1	1	1	2	1	1	2	2	2	2	1	1	1	1	2	1	2	1	2	2

The columns’ labels in Table 2 present the 22 species’ names presented in Figure 1, while the table entries correspond to the data values presented in Table 1 against each feature presented in terms of five rows of color, texture, shape, margin, and hilum position (as expressed in Table 1).

The feature list description for the second dataset that is the species of Apocynaceae and Asclepiadaceae seeds is presented in Table 3. As mentioned already, the second dataset contained 14 species of Apocynaceae and Asclepiadaceae. The feature list provided in Table 3 shows the three main types of seed, hilum, and coma; the seed features are color, texture, shape, and margin, with corresponding three, five, two, and four possible values, as given in the third column. Similarly, hilum exhibits three features of position (with two values), shape (with five possible values), and level (with four possible values). Likewise, the third type, known as coma, exhibits three features of color, duration, and position with three, three, and four possible values, respectively.

Table 3. Feature list for species of Apocynaceae and Asclepiadaceae.

Type	Feature	Values
Seed	Color	Brown [1]/off-white [2]/black [3].
	Texture	Warty [1]/tuberculate [2]/irregular striated [3]/hairy [4]/smooth [5].
	Shape	Globose [1]/flattened [2].
	Margin	Wingless [1]/winged [2]/folded [3]/hairy [4].
Hilum	Position	Terminal [1]/central [2].
	Shape	Oblong [1]/elliptic [2]/conical [3]/linear [4]/oblong-ovate [5].
	Level	Depressed [1]/semidepressed [2]/elevated [3]/superficial [4].
Coma	Color	Brown [1]/off-white [2]/absent [3].
	Duration	Deciduous [1]/persistence [2]/absent [3].
	Position	Terminal [1]/peripheral [2]/covering the whole seed’s surface [3]/absent [4].

The features are listed in Table 4 with their explained encoding, which was used as an input to the statistical application to measure the distance between the studied species and subsequently demonstrate the result as dendrograms for the second dataset. The table presents the data matrix of the Apocynaceae and Asclepiadaceae seed morphological characteristics. The columns' labels in Table 4 present the 14 species' names presented in Figure 2, while the table entries correspond to the data values presented in Table 3 against three types of corresponding features, presented in terms of ten rows (4 + 3 + 3). This results in the second dataset with 14 species.

Table 4. Data matrix of Apocynaceae and Asclepiadaceae seed morphological characteristics.

#	<i>Acokanthera spectabilis</i>	<i>Carissa spinarum</i>	<i>Thevetia peruviana</i>	<i>Vinca rosea</i>	<i>Alstonia scholaris</i>	<i>Nerium oleander</i>	<i>Mascarenhusia elastica</i>	<i>Leptadenia arborea</i>	<i>Leptadenia pyrotechnica</i>	<i>Calotropis procera</i>	<i>Cynanchum acutum</i>	<i>Solenostemma argel.</i>	<i>Asclepias curassavica</i>	<i>Cryptostegia grandiflora</i>
1	1	1	2	3	1	1	1	1	1	1	1	1	1	1
2	1	2	3	2	2	4	2	2	5	4	2	2	1	2
3	1	2	1	1	2	2	2	2	2	2	2	1	2	2
4	1	3	1	1	4	1	1	2	2	2	2	3	1	1
5	2	2	1	2	2	2	2	2	2	2	2	2	2	2
6	1	2	3	2	2	2	1	4	4	5	4	4	4	4
7	3	1	4	3	2	3	2	4	2	2	4	1	4	4
8	3	3	3	3	1	1	1	2	2	2	2	2	2	2
9	3	3	3	3	2	2	1	1	1	1	1	1	1	1
10	4	4	4	4	2	3	1	1	1	1	1	1	1	1

3.3. Clustering Using Statistical Applications

The associations between the investigated species dataset are shown as dendrograms in Figures 4 and 5, respectively. These figures were produced using the application named PRIMER 6, version 6.1.6 for an analysis using the aggregation of the schedule measure of Euclidean distance and complete linkage between groups.

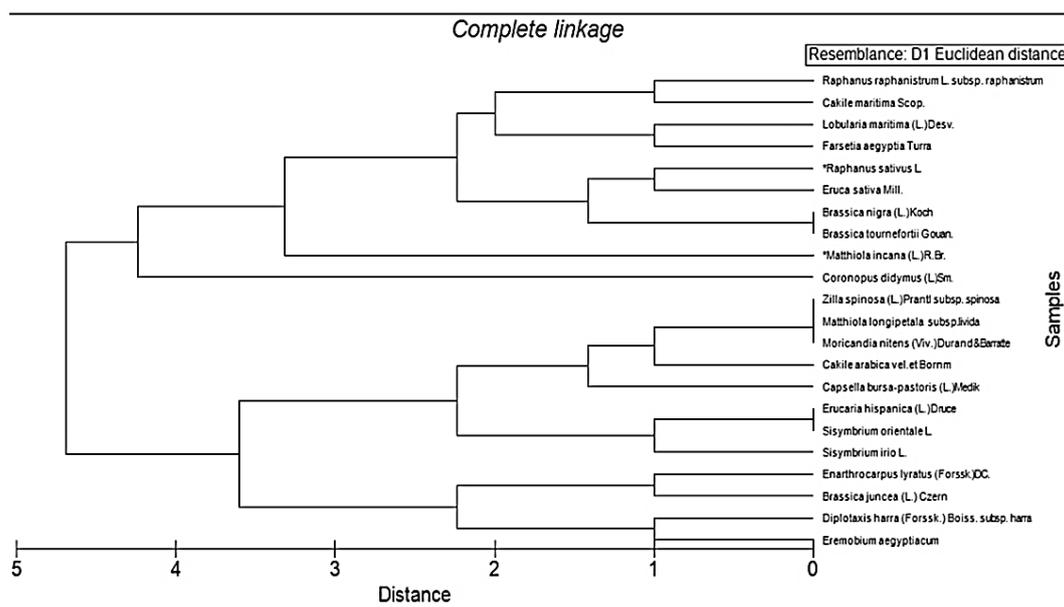


Figure 4. Dendrograms showing the interrelationships between 22 species of Brassicaceae based on seed characters.

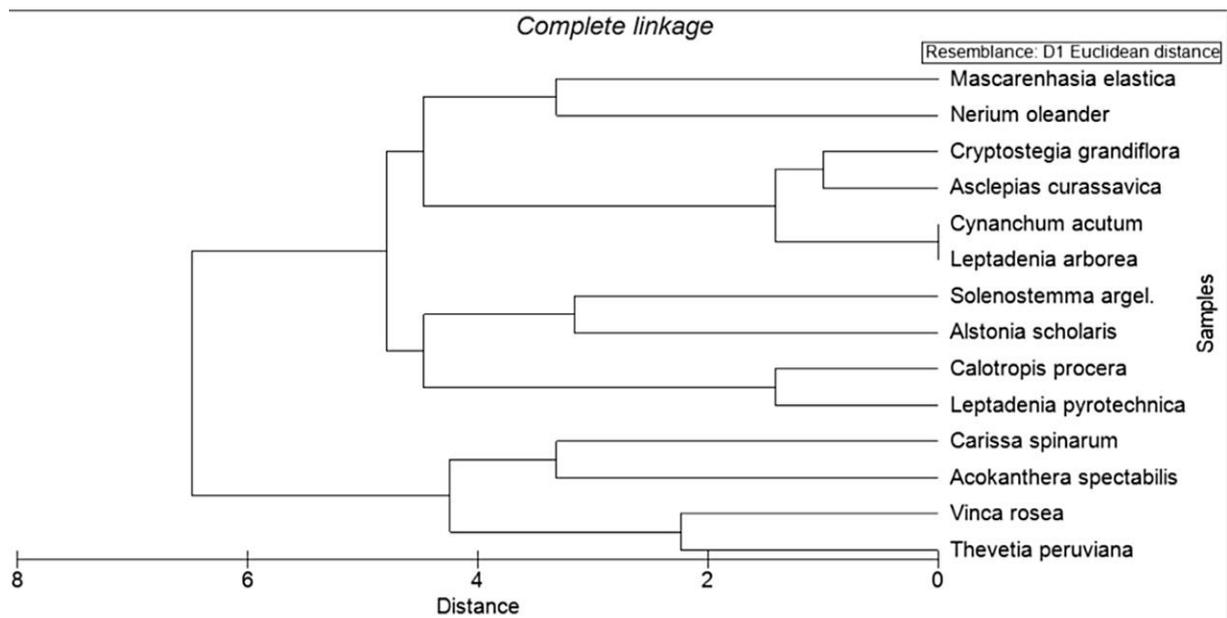


Figure 5. Dendrograms showing the interrelationships between 14 species of Apocynaceae and Asclepiadaceae based on seed characters.

In a dendrogram, species are represented by branches, and the length of the branches represents the distance or dissimilarity between them; the shorter branches are more similar.

3.4. Deep Learning Model

3.4.1. Image Preprocessing

Preprocessing is a technique for removing undesirable noise and improving images by employing image processing techniques such as smoothing and sharpening [35]. If dataset images are high-quality, the enhancement process will be very small. This study used image enhancement methods to improve image quality such as sharpening and removing noise in the image, and a segmentation process using image segmentation techniques. Open Source Computer Vision (OpenCV) package was used to apply these techniques; the results of this phase are shown in Figure 6.

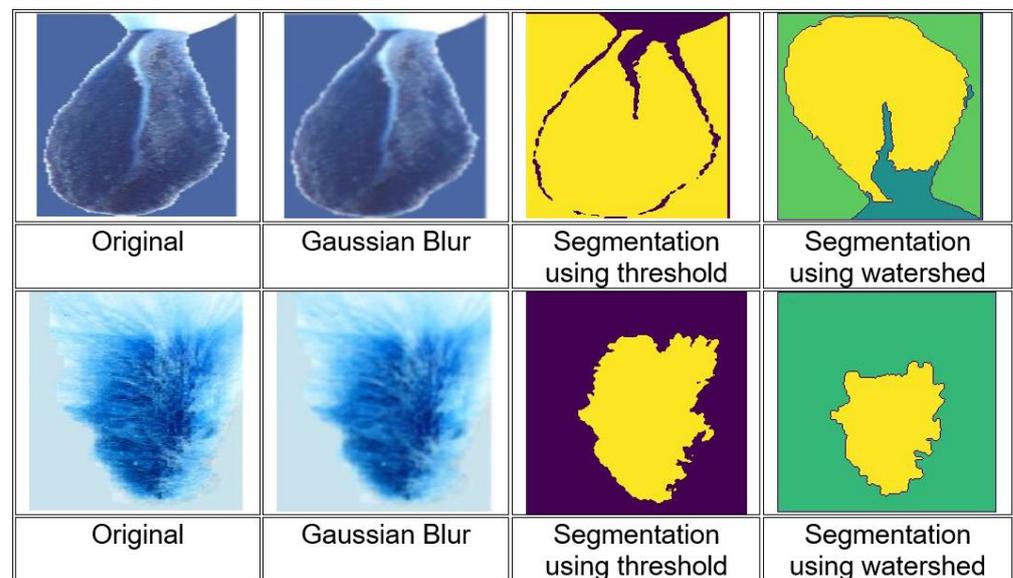


Figure 6. Image preprocessing samples.

3.4.2. Feature Extraction

Extracting the features of interest from the segmented seed might involve computing morphological features, such as shape, texture, and margin, as well as color features, such as the color of the seed and the coma; extracting features related to the hilum can extract the position, shape, level, and color. One of the most widely used global descriptors investigated in the object detection group is the histogram of oriented gradients (HOG) descriptor. We created a feature extraction process using the CNN model for the current study [36]. The current study implemented the proposed method using keras applications by importing (densenet, mobilenet_v2, mobilenet_v3, resnet_v2, . . . , etc.) packages to use the deep learning models for feature extraction; also, the ReliefF algorithm is commonly used in machine learning applications to improve the accuracy of models by reducing the number of irrelevant or redundant features [37]. The formula for ReliefF is as follows:

$$\text{ReliefF}(S, i) = \text{sum}(w(S, i, x, y) * \text{diff}(x, y, i)) / k \quad (1)$$

where S is the dataset, i is the index of the feature being evaluated, $w(S, i, x, y)$ is the weight assigned to the feature i for the samples x and y , $\text{diff}(x, y, i)$ is the difference between the values of feature i for samples x and y , and k is the number of nearest neighbors used in the algorithm. The algorithm works by iterating over each feature in the dataset and computing the relevance score for that feature. Features with high relevance scores are important and are retained in the final feature set, while those with low scores are discarded.

- *Time Complexity:*

The time complexity of extracting features from each image depends on the complexity of the pretrained model and the size/resolution of the images. For using DenseNet, the time complexity is typically proportional to the number of layers and the number of operations required for each layer. This complexity is usually in the order of $O(N)$, where N is the number of images.

- *Space complexity:*

The space complexity of feature extraction depends on the memory requirements of the pretrained model and the size of the images being processed. The memory required is typically proportional to the model size and the number of layers. The space complexity is generally constant in the order of $O(1)$ per image, since only one image is processed at a time.

3.4.3. Deep Learning Clustering

Agglomerative clustering begins with N clusters, each containing one data point. Following that, a series of merging operations are performed, which finally force all objects into the same group. Using previously formed clusters, hierarchical algorithms find succeeding clusters [38]. These algorithms can be agglomerative (“bottom-up”) or divisive (“top-down”). Agglomerative algorithms start with each element as a single cluster and gradually combine them into bigger clusters. Divisive algorithms start with the entire set and then, consequently, split it into smaller and smaller groups [38]. The relationships between the studied species are demonstrated as dendrograms in Figures 7 and 8 by using the proposed AI model after applying preprocessing; the DenseNet201 deep learning model for feature extraction; and the agglomeration of schedule measure of Euclidean distance, using complete linkage between groups.

The dendrogram in Figure 7 represents the clustering of 14 samples of wild plant seeds, and it appears in this hierarchy that it is close to the hierarchy which was carried out manually and using statistical programs; the error rate between them was calculated at 5%, and this percentage is considered excellent, given that the method of artificial intelligence does not involve any effort. Only the seed images were entered into the model, which extracted the features from the images and performed automatic clustering; this was the main objective of this article.

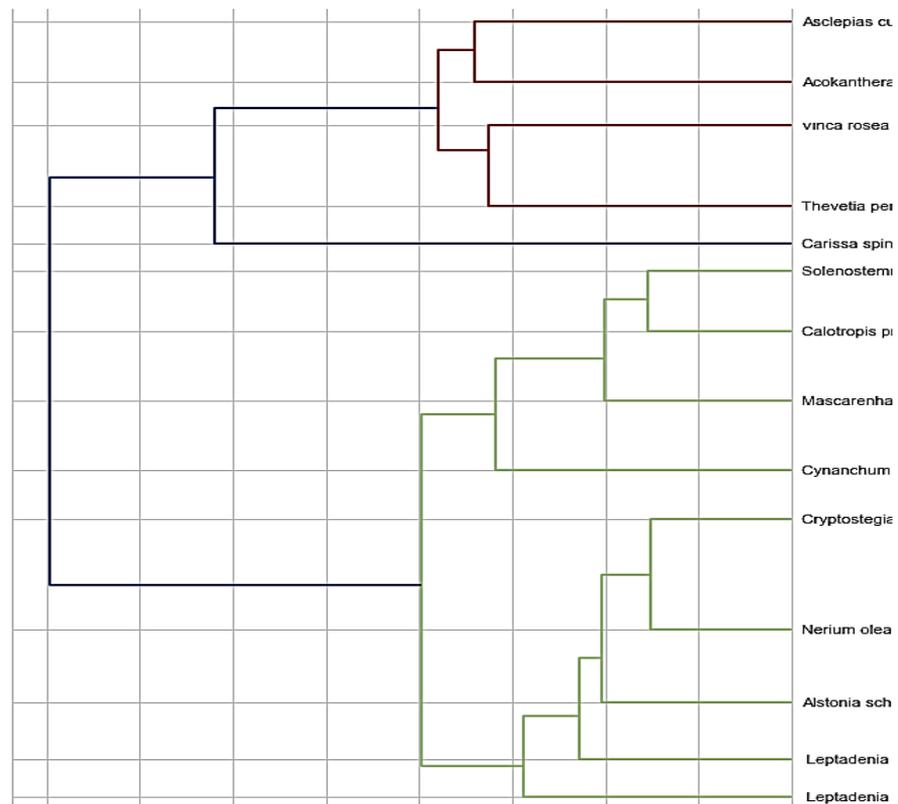


Figure 7. AI dendrogram for Dataset-2.

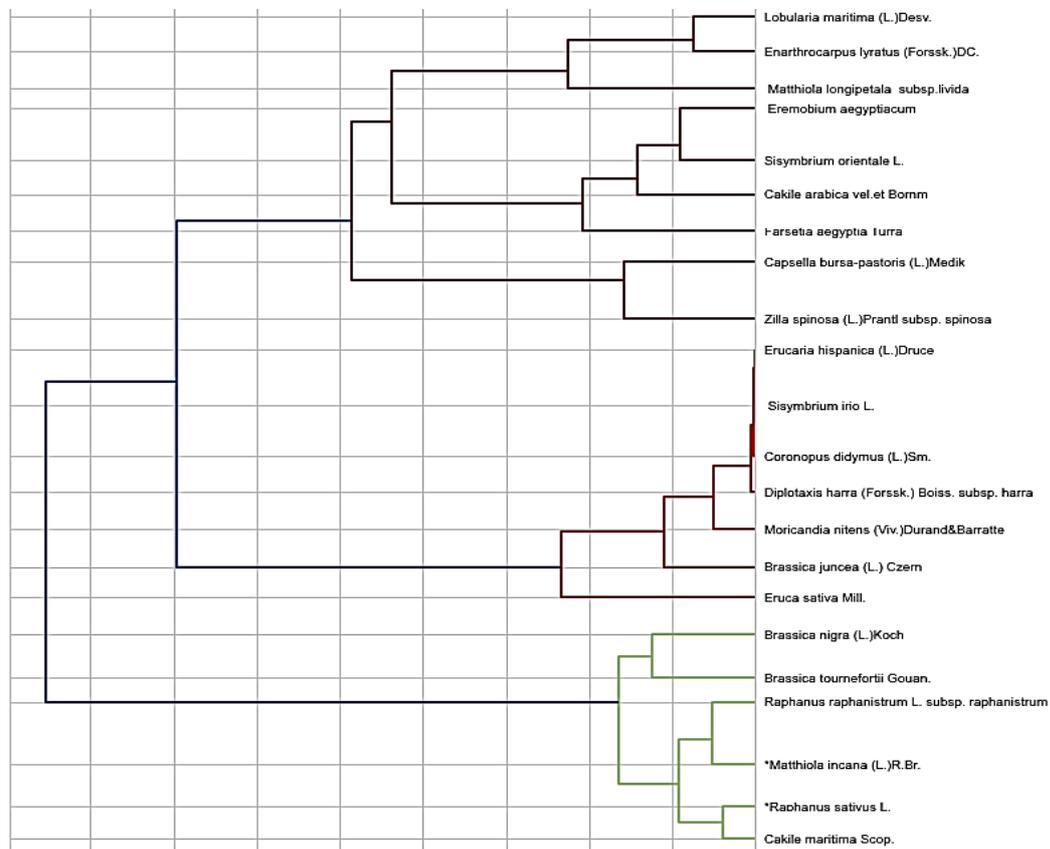


Figure 8. AI dendrogram for Dataset-1.

The dendrogram in Figure 8 represents the clustering of 22 samples of wild plant seeds, and it appears in this hierarchy that it is close to the hierarchy which was carried out manually and using statistical applications. The error rate between them was calculated as 18%, and this reference percentage is considered good. Once the deep learning models were trained, the seed images were entered into the model, which extracted the features from the images and performed automatic clustering; this was the main objective of this article. A dendrogram typically starts with each species represented as a separate cluster and then merges them into larger clusters as the algorithm progresses gradually. The result is a tree-like structure where the species are grouped into clusters at different levels of the tree.

The time and space complexity of the proposed AI image clustering method can vary depending on the specific algorithm and implementation used:

- *Time Complexity:*

The time complexity of the agglomerative hierarchical clustering algorithm is typically $O(N^3)$, where N is the number of data points. This is because, at each step, the algorithm needs to calculate the distance between all pairs of clusters, resulting in a total of $N/2$ steps until a single cluster is formed. Calculating the distance between clusters can be computationally expensive and requires $O(N)$ operations.

- *Space complexity:*

The space complexity of agglomerative hierarchical clustering is typically $O(N^2)$, where N is the number of data points. This is because the algorithm requires storing the pairwise distance matrix or a similarity matrix, which has a size of $N \times N$. The matrix stores the distance or similarity values between each pair of data points.

4. Results and Discussion

Evaluation Metrics

Accuracy and error are commonly used to evaluate the performance of deep learning models. They show the relationship between the predicted and actual values of the model. To assess the performance of the proposed model on the given datasets, four measures are used: accuracy, F-score, recall, and precision [39–42].

- Accuracy: The result of dividing the number of true classified outcomes by the whole of classified instances. The accuracy is computed by the equation:

$$\text{Accuracy} = \frac{\text{TruePositive} + \text{TrueNegative}}{\text{TruePositive} + \text{TrueNegative} + \text{FalsePositive} + \text{FalseNegative}} \quad (2)$$

- Recall: The percentage of positive tweets that are properly determined by the model in the dataset. The recall calculated by:

$$\text{Recall} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}} \quad (3)$$

- Precision: The proportion of true positive tweets among all forecasted positive tweets. The equation of precision measure calculated by:

$$\text{Precision} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}} \quad (4)$$

- F-score: A harmonic mean of precision and recall. The F-score measure equation is:

$$\text{F - measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

The study investigated sixteen deep learning models to automatically extract the plant image features in the feature extraction phase, and then used these features in the clustering method. The sixteen models used in this study (DenseNet121, DenseNet201, ResNet50V2, EfficientNetB6, EfficientNetB1, EfficientNetB0, MobileNetV2, EfficientNetB3, VGG16, VGG19, EfficientNetB5, EfficientNetB7, EfficientNetB2, and EfficientNetB4) were

evaluated and the results are shown in Tables 5 and 6, for both datasets, respectively. Based on Tables 5 and 6, the best dendrograms of the best deep learning model (DenseNet201) are shown in Figures 9 and 10 for both Dataset-1 and Dataset-2, respectively. Comparing these results with other published work [43], for the transfer learning from the deep learning DenseNet201 model, the accuracy of the classification was 49.29%, while the MobileNet accuracy in the same work was 94.1%. The project [44] worked on determining seven skin diseases by using the oversampling and data augmentation technique; the result of the accuracy was 94.4%.

Table 5. Clustering evaluation result (Dataset-1).

#	Method	AUC Score	F1-Score	Accuracy Score	Random Score	Mean Square Error
1	EfficientNetB4	0.25	0	0.14	0.35	0.86
2	EfficientNetB7	0.4	0.4	0.36	0.03	0.64
3	VGG19	0.43	0.17	0.29	0.03	0.71
4	DenseNet121	0.43	0.17	0.29	0.03	0.71
5	MobileNetV2	0.5	0.59	0.5	0	0.5
6	VGG16	0.55	0.18	0.36	0.04	0.64
7	EfficientNetB2	0.55	0.67	0.57	0.01	0.43
8	EfficientNetB6	0.55	0.18	0.36	0.04	0.64
9	EfficientNetB0	0.6	0.3	0.43	0.09	0.57
10	EfficientNetB1	0.6	0.3	0.43	0.09	0.57
11	EfficientNetB5	0.6	0.33	0.43	0.09	0.57
12	NASNet	0.6	0.33	0.43	0.09	0.57
13	EfficientNetB3	0.75	0.67	0.64	0.26	0.36
14	ResNet50V2	0.78	0.84	0.79	0.22	0.21
15	InceptionV3	0.78	0.84	0.79	0.22	0.21
16	DenseNet201	0.95	0.95	0.93	0.7	0.07

Table 6. Clustering evaluation result (Dataset-2).

#	Method	AUC Score	F1-Score	Accuracy Score	Random Score	Mean Square Error
1	MobileNetV2	0.28	0.33	0.27	0.15	0.73
2	EfficientNetB1	0.33	0.35	0.32	0.09	0.7
3	DenseNet121	0.38	0	0.41	0.13	0.59
4	ResNet50V2	0.43	0.14	0.45	0.03	0.55
5	EfficientNetB5	0.43	0.25	0.45	0.02	0.55
6	EfficientNetB6	0.47	0.5	0.45	0.003	0.55
7	EfficientNetB7	0.51	0.52	0.5	0	0.5
8	InceptionV3	0.5	0.48	0.5	0	0.5
9	VGG19	0.57	0.62	0.55	0.02	0.45
10	EfficientNetB2	0.55	0.55	0.55	0.01	0.45
11	NASNet	0.57	0.62	0.55	0.02	0.45
12	EfficientNetB0	0.6	0.6	0.59	0.03	0.4
13	EfficientNetB4	0.6	0.6	0.63	0.17	0.27
14	EfficientNetB3	0.64	0.64	0.64	0.06	0.36
15	VGG16	0.67	0.59	0.68	0.09	0.32
16	DenseNet201	0.8	0.75	0.82	0.4	0.18

The entries in Table 5 are presented in almost ascending order with respect to AUC and accuracy values obtained against the sixteen deep learning models. So, in this regard, DenseNet201, with a minimum means square error (MSE) of 7%, achieved the best metrics for Dataset-1 that were 93% accuracy and an F1-score and AUC value of 95% each. This was followed by InceptionV3 and ResNet50V2, with identical performance in terms of accuracy, F1-score, and AUC (79%, 84%, and 78%, respectively). EfficientNetB4 exhibited poor performance in every aspect of Dataset-1. Figure 9 visualizes the entries of Table 5.

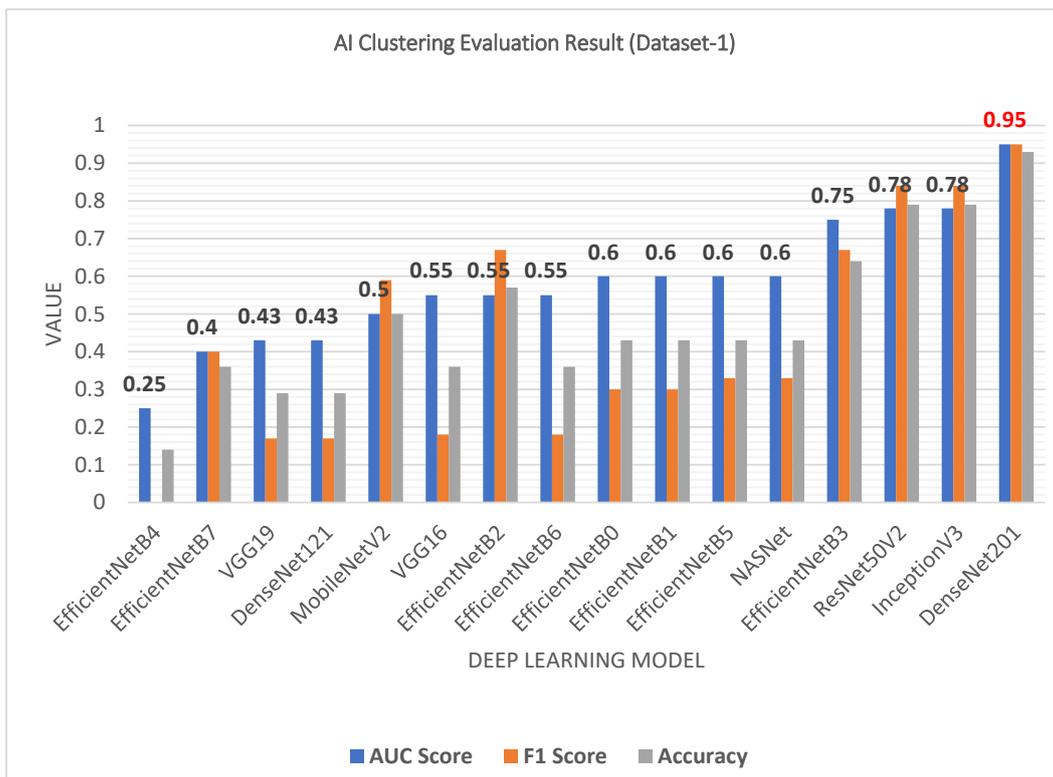


Figure 9. AI clustering evaluation result (Dataset-1).

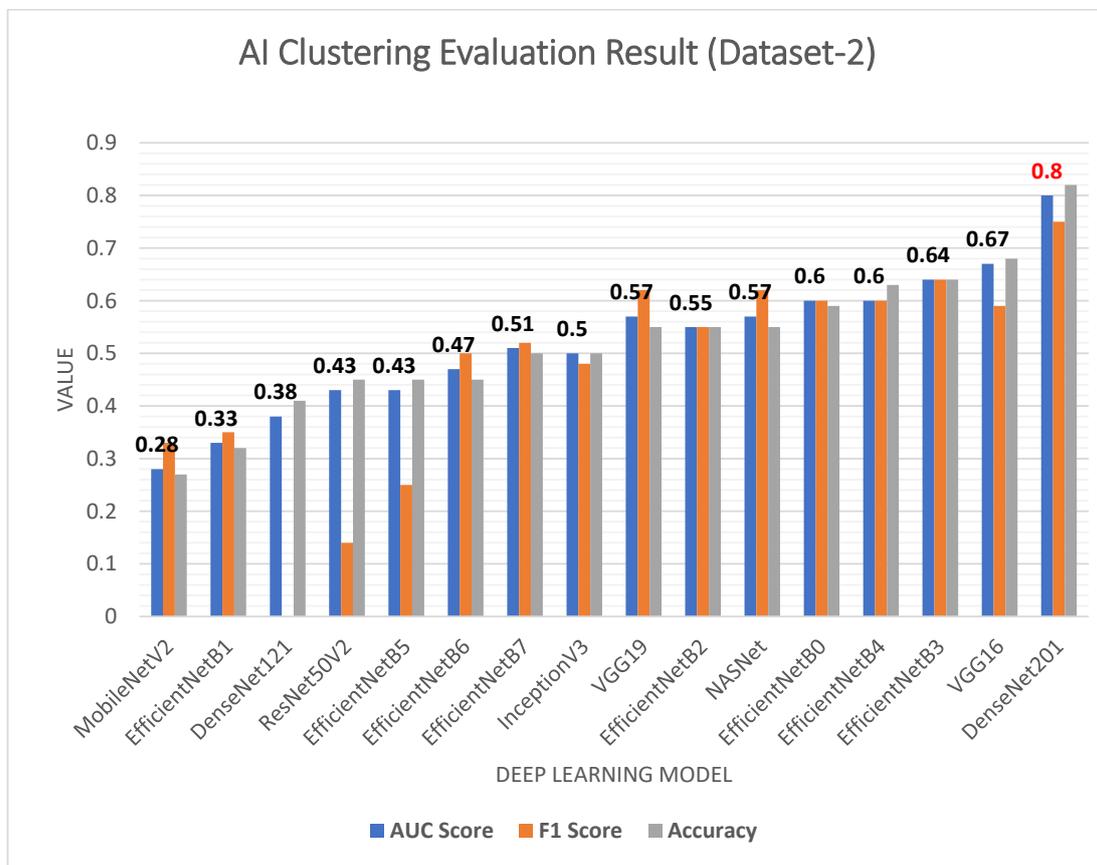


Figure 10. Clustering evaluation result (Dataset-2).

Similarly, the entries in Table 6 are presented in almost ascending order with respect to the AUC and accuracy values obtained against the sixteen deep learning models for the second dataset. As expressed in the table, DensNet201 achieved the best metrics for Dataset-2, with 82% accuracy and F1-score and AUC values of 75% and 80%, respectively. This was followed by VGG13, with lower performance in terms of accuracy, F1-score, and AUC (68%, 59%, and 67%, respectively). Furthermore, MobileNetV2 and EfficientNetB1 exhibited the poorest performance in every aspect for Dataset-2 with respective values of accuracy, F1-score, and AUC. Figure 10 visualizes the entries of Table 6.

The elbow technique is used to determine the optimal number of clusters in each dataset. The method involves plotting the explained variation as a function of the number of clusters and selecting the number of clusters at the “elbow” of the curve. This is the point of diminishing returns, where the addition of another cluster does not significantly improve the fit of the model, as shown in Figures 11 and 12 for Dataset-1 and Dataset-2, respectively.

As a summary of the experiments, the deep-learning-based image clustering showed promising results in clustering large-scale image datasets. In this regard, several algorithms were investigated for image clustering and DensNet201 was proven to be the best algorithm for both datasets. Although the accuracy for the first dataset was more than the accuracy for the second dataset, it outperformed the other fifteen algorithms in terms of all the evaluation metrics. Various evaluation metrics such as clustering accuracy, MSE, AUC, and F1-score were used to evaluate the performance of image clustering algorithms. DenseNet201 emerged as the best-performing deep learning model for both datasets, showcasing high accuracy, F1-score, and AUC values. The study provides a comprehensive evaluation of deep learning models and their performance in extracting plant image features for clustering purposes.

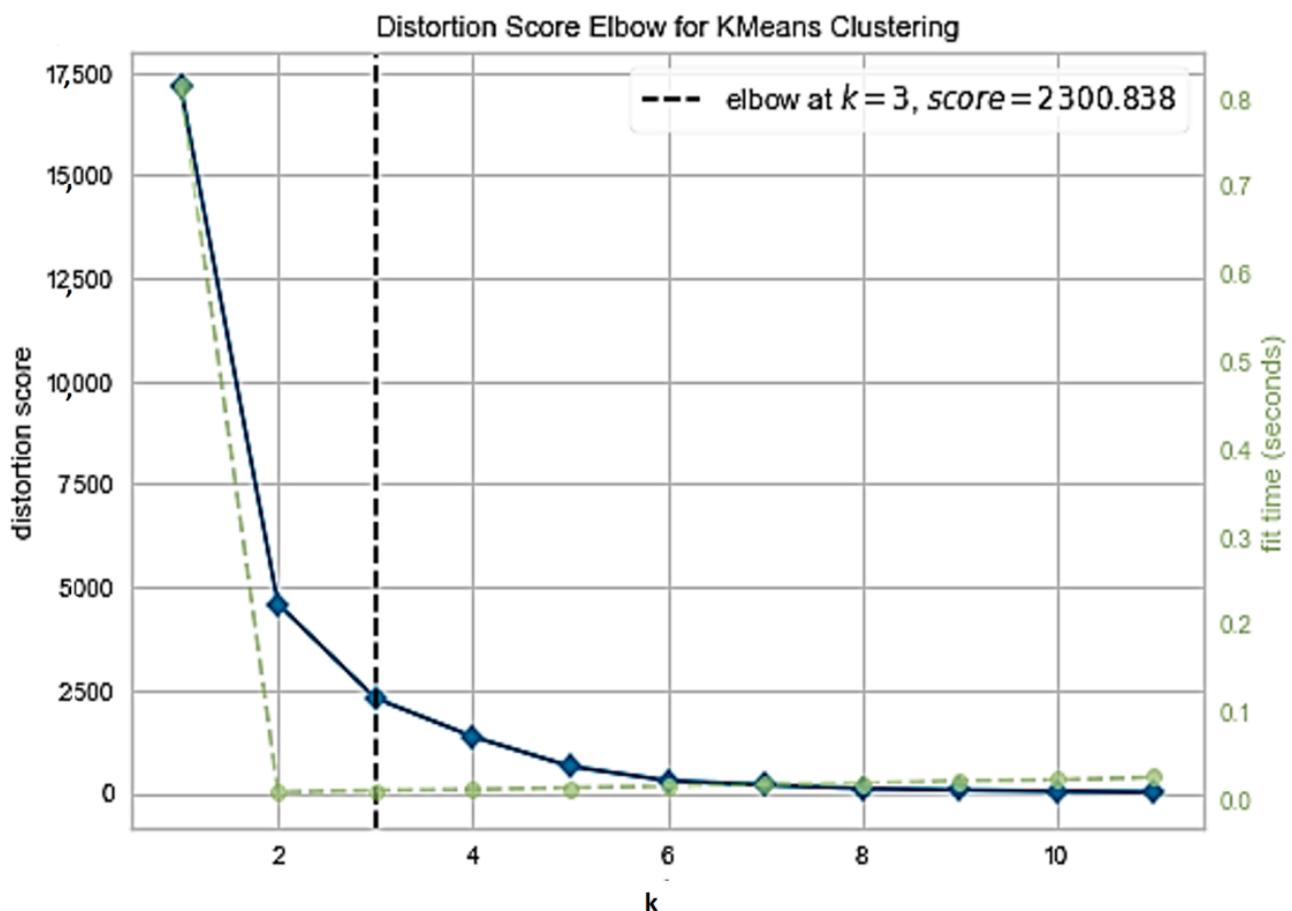


Figure 11. Distortion score for elbow k-means clustering of Dataset-1.

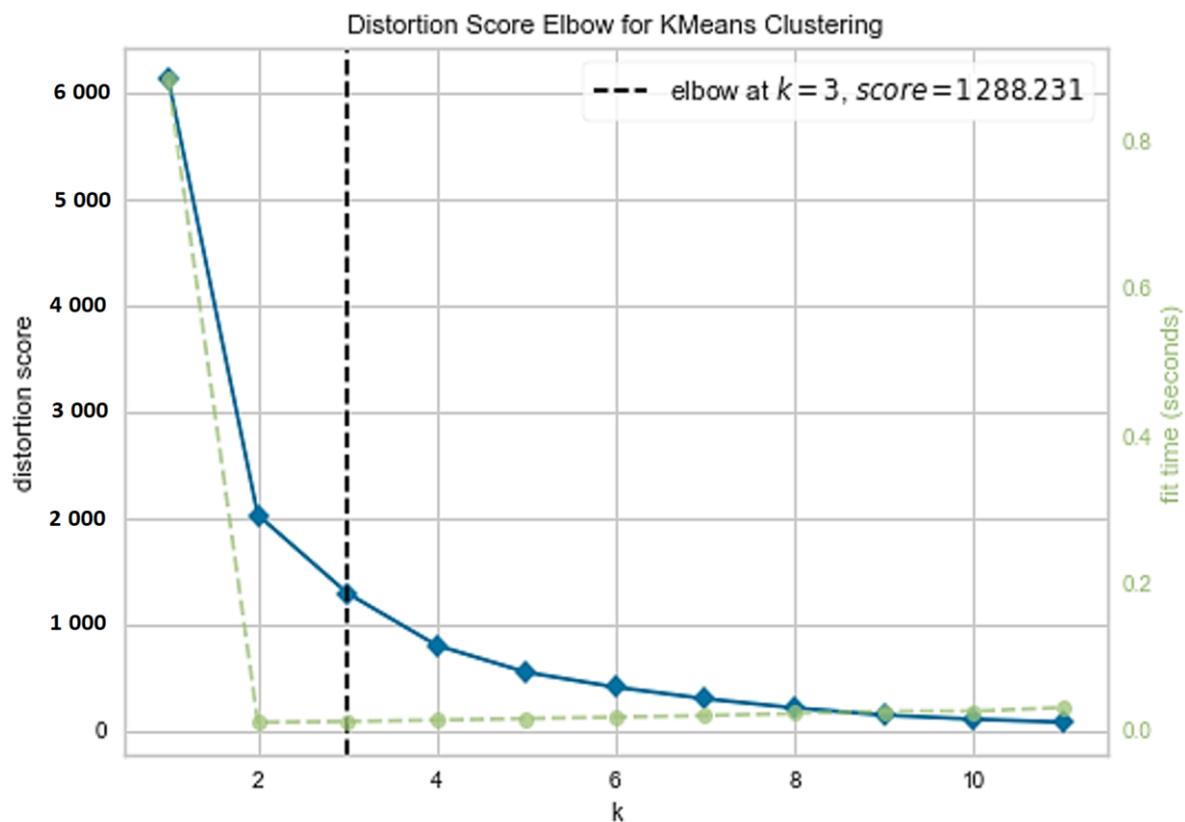


Figure 12. Distortion score for elbow k-means clustering of Dataset-2.

5. Conclusions

The current study's objective was to use deep learning to support plant taxonomy. The dataset for the untamed plant was gathered in Egypt, where it is native. The study applied two approaches: The first was the traditional method of extracting the features manually and applying clustering using statistics application; we considered this method as a reference for the second approach. The second approach was to model and train deep learning models using transfer learning concepts including image preprocessing and feature reduction techniques. Subsequently, all results were compared with the first method. Amongst the sixteen investigated deep learning models, DensNet201 achieved the highest accuracy of 95% in Dataset-1, which included 22 species, and achieved the highest accuracy of 82% in Dataset-2, which included 14 species. Future work can investigate different fine-tuning strategies, including variations in learning rates, layer freezing, or the inclusion of additional custom layers, and can investigate the effectiveness of ensemble methods in image taxonomy using transfer learning [45] Additionally, it can focus on developing more efficient and scalable deep learning algorithms for image clustering and using plant DNA images for clustering to assist biological researchers in plant taxonomy.

Author Contributions: Conceptualization, N.M.I. and D.G.G.; formal analysis, D.A.; funding acquisition, A.R., D.A. and M.A.; investigation, D.G.G., A.R. and M.A.; methodology, D.G.G. and A.R.; project administration, A.R.; software, N.M.I.; supervision, D.G.G.; validation, A.R. and D.M.; visualization, N.M.I.; writing—original draft, N.M.I.; writing—review and editing, A.R., D.M., D.A. and M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data can be provided upon request from the corresponding author.

Acknowledgments: This work acknowledges botanical studies and research.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Takamitsu, Y.; Orita, Y. Effect of glomerular change on the electrolyte reabsorption of the renal tubule in glomerulonephritis (author's transl). *Jpn. J. Nephrol.* **1978**, *20*, 1221–1227.
2. Wani, J.A.; Sharma, S.; Muzamil, M.; Ahmed, S.; Sharma, S.; Singh, S. *Machine Learning and Deep Learning Based Computational Techniques in Automatic Agricultural Diseases Detection: Methodologies, Applications, and Challenges*; Springer: Dordrecht, The Netherlands, 2021; Volume 29.
3. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
4. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015; pp. 1–14.
5. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]
6. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [[CrossRef](#)]
7. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
8. Lee, J.W.; Yoon, Y.C. Fine-Grained Plant Identification using wide and deep learning model 1. In Proceedings of the 2019 International Conference on Platform Technology and Service (PlatCon), Jeju, Republic of Korea, 28–30 January 2019; pp. 1–5. [[CrossRef](#)]
9. Lee, S.H.; Chan, C.S.; Wilkin, P.; Remagnino, P. Deep-plant: Plant identification with convolutional neural networks. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 452–456. [[CrossRef](#)]
10. Gyires-Tóth, B.P.; Osváth, M.; Papp, D.; Szucs, G. Deep learning for plant classification and content-based image retrieval. *Cybern. Inf. Technol.* **2019**, *19*, 88–100. [[CrossRef](#)]
11. Dileep, M.R.; Pournami, P.N. AyurLeaf: A Deep Learning Approach for Classification of Medicinal Plants. In Proceedings of the TENCON 2019—2019 IEEE Region 10 Conference (TENCON), Kochi, India, 17–20 October 2019; pp. 321–325. [[CrossRef](#)]
12. Picon, A.; Alvarez-Gila, A.; Seitz, M.; Ortiz-Barredo, A.; Echazarra, J.; Johannes, A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Comput. Electron. Agric.* **2019**, *161*, 280–290. [[CrossRef](#)]
13. Johannes, A.; Picon, A.; Alvarez-Gila, A.; Echazarra, J.; Rodriguez-Vaamonde, S.; Díez Navajas, A.; Ortiz-Barredo, A. Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. *Comput. Electron. Agric.* **2017**, *138*, 200–209. [[CrossRef](#)]
14. Tan, J.W.; Chang, S.W.; Abdul-Kareem, S.; Yap, H.J.; Yong, K.T. Deep Learning for Plant Species Classification Using Leaf Vein Morphometric. *IEEE/ACM Trans. Comput. Biol. Bioinforma.* **2020**, *17*, 82–90. [[CrossRef](#)]
15. Haupt, J.; Kahl, S.; Koworko, D.; Eibl, M. Large-scale plant classification using deep convolutional neural networks. *CEUR Workshop Proc.* **2018**, *2125*, 1–7.
16. Reyes, A.K.; Caicedo, J.C.; Camargo, J.E.; Nari, U.A. Fine-tuning Deep Convolutional Networks for Plant Recognition. *CLEF* **2015**, *1391*, 467–475.
17. Gajendran, M.K.; Khan, M.Z.; Khattak, M.A.K. ECG Classification using Deep Transfer Learning. In Proceedings of the 2021 4th International Conference on Information and Computer Technologies (ICICT), Kahului, HI, USA, 11–14 March 2021; pp. 1–5. [[CrossRef](#)]
18. Choi, S. Plant identification with deep convolutional neural network: SNUMedinfo at LifeCLEF plant identification task 2015. *CEUR Workshop Proc.* **2015**, *1391*, 2–5.
19. Sun, Y.; Zhu, L.; Wang, G.; Zhao, F. Multi-Input Convolutional Neural Network for Flower Grading. *J. Electr. Comput. Eng.* **2017**, *2017*, 9240407. [[CrossRef](#)]
20. Low, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. Available online: <https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf> (accessed on 6 January 2023). [[CrossRef](#)]
21. Xiao, J.; Wang, J.; Cao, S.; Li, B. Application of a Novel and Improved VGG-19 Network in the Detection of Workers Wearing Masks Application of a Novel and Improved VGG-19 Network in the Detection of Workers Wearing Masks. *J. Phys. Conf. Ser.* **2020**, *1518*, 012041. [[CrossRef](#)] [[PubMed](#)]
22. Zhang, S.; Huang, W.; Huang, Y.; Zhang, C. Neurocomputing Plant species recognition methods using leaf image: Overview. *Neurocomputing* **2020**, *408*, 246–272. [[CrossRef](#)]
23. Ullah, M.I.; Attique, M.; Sharif, M.; Ahmad, S.; Bukhari, C. Data in brief A citrus fruits and leaves dataset for detection and classification of citrus diseases through machine learning. *Data Brief* **2019**, *26*, 104340. [[CrossRef](#)]
24. Wang, X.; Yang, Y.; Guo, Z.; Zhou, Z.; Liu, Y.; Pang, Q.; Du, S. Real-World Image Super Resolution via Unsupervised Bi-directional Cycle Domain Transfer Learning based Generative Adversarial Network. *arXiv* **2022**, arXiv:2211.10563.
25. Ibrahim, N.M.; Gabr, D.G.I.; Rahman, A.-U.; Dash, S.; Nayyar, A. A deep learning approach to intelligent fruit identification and family classification. *Multimed. Tools Appl.* **2022**, *81*, 27783–27798. [[CrossRef](#)]

26. Khan, T.A.; Fatima, A.; Shahzad, T.; Rahman, A.U.; Alissa, K.; Ghazal, T.M.; Al-Sakhnini, M.M.; Abbas, S.; Khan, M.A.; Ahmed, A. Secure IoMT for Disease Prediction Empowered with Transfer Learning in Healthcare 5.0, the Concept and Case Study. *IEEE Access* **2023**, *11*, 39418–39430. [[CrossRef](#)]
27. Asif, R.N.; Abbas, S.; Khan, M.A.; Rahman, A.U.; Sultan, K.; Mahmud, M.; Mosavi, A. Development and Validation of Embedded Device for Electrocardiogram Arrhythmia Empowered with Transfer Learning. *Comput. Intell. Neurosci.* **2022**, *2022*, 5054641. [[CrossRef](#)] [[PubMed](#)]
28. Nasir, M.U.; Zubair, M.; Ghazal, T.M.; Khan, M.F.; Ahmad, M.; Rahman, A.-u.; Hamadi, H.A.; Khan, M.A.; Mansoor, W. Kidney Cancer Prediction Empowered with Blockchain Security Using Transfer Learning. *Sensors* **2022**, *22*, 7483. [[CrossRef](#)]
29. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [[CrossRef](#)]
30. Vinh, N.X.; Epps, J.; Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.* **2010**, *11*, 2837–2854.
31. Xie, J.; Girshick, R.; Farhadi, A. Unsupervised deep embedding for clustering analysis. In Proceedings of the 33rd International Conference on International Conference on Machine Learning ICML, New York, NY, USA, 19–24 June 2016; Volume 1, pp. 740–749.
32. Ahmed, M.I.B.; Zaghdoud, R.; Ahmed, M.S.; Sendi, R.; Alsharif, S.; Alabdulkarim, J.; Saad, B.A.A.; Alsabt, R.; Rahman, A.; Krishnasamy, G. A Real-Time Computer Vision Based Approach to Detection and Classification of Traffic Incidents. *Big Data Cogn. Comput.* **2023**, *7*, 22. [[CrossRef](#)]
33. Guo, X.; Liu, X.; Zhu, E.; Yin, J. Deep clustering with convolutional autoencoders. In Proceedings of the 25th ACM International Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 582–590.
34. Yang, J.; Parikh, D.; Batra, D. Joint unsupervised learning of deep representations and image clusters. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5147–5156.
35. Pathak, A.K.; Parsai, M. A Study of Various Image Fusion Techniques. *Int. J. Eng. Trends Technol.* **2014**, *15*, 59–62. [[CrossRef](#)]
36. Gogul, I.; Kumar, V.S. Flower species recognition system using convolution neural networks and transfer learning. In Proceedings of the 2017 4th International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, India, 16–18 March 2017; pp. 1–6. [[CrossRef](#)]
37. Robnik-Šikonja, M.; Kononenko, I. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Mach. Learn.* **2003**, *53*, 23–69. Available online: <http://lkm.fri.uni-lj.si/xaigor/slo/clanki/MLJ2003-FinalPaper.pdf> (accessed on 10 January 2023). [[CrossRef](#)]
38. Sembiring, R.W.; Zain, J.M.; Embong, A. A Comparative Agglomerative Hierarchical Clustering Method to Cluster Implemented Course. *arXiv* **2011**, arXiv:1101.4270.
39. Alghamdi, A.S.; Rahman, A. Data Mining Approach to Predict Success of Secondary School Students: A Saudi Arabian Case Study. *Educ. Sci.* **2023**, *13*, 293. [[CrossRef](#)]
40. Alqarni, A.; Rahman, A. Arabic Tweets-Based Sentiment Analysis to Investigate the Impact of COVID-19 in KSA: A Deep Learning Approach. *Big Data Cogn. Comput.* **2023**, *7*, 16. [[CrossRef](#)]
41. Vallabhajosyula, S.; Sistla, V.; Kolli, V.K.K. Transfer learning-based deep ensemble neural network for plant leaf disease detection. *J. Plant Dis. Prot.* **2022**, *129*, 545–558. [[CrossRef](#)]
42. Aydin, K.; Ali Seydi, K.; Cagatay, C.; Hamdi, Y.Y.; Huseyin, T.; Bedir, T. Analysis of transfer learning for deep neural network based plant classification models. *Comput. Electron. Agric.* **2019**, *158*, 20–29. [[CrossRef](#)]
43. Velasco, J.S.; Catipon, J.V.; Monilar, E.G.; Amon, V.M.; Virrey, G.C. Classification of Skin Disease Using Transfer Learning in Convolutional Neural Networks. *arXiv* **2023**, arXiv:2304.02852. [[CrossRef](#)] [[PubMed](#)]
44. Velasco, J.; Pascion, C.; Alberio, J.W.; Apuang, J.; Cruz, J.S.; Gomez, M.A.; Jorda, R.A., Jr. Smartphone-Based Skin Disease Classification Using MobileNet CNN International Journal of Advanced Trends in Computer Science and Engineering. *Int. J. Adv. Trends Comput. Sci. Eng.* **2019**, *8*, 2–8. Available online: <http://www.warse.org/IJATCSE/static/pdf/file/ijatcse116852019.pdf> (accessed on 23 January 2023).
45. Narvekar, C.; Rao, M. Flower classification using CNN and transfer learning in CNN- Agriculture Perspective. In Proceedings of the 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 3–5 December 2020; pp. 660–664. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.