https://doi.org/10.48047/AFJBS.6.11.2024.50-62

African Journal of Biological Sciences

Evaluating the Accuracy of Image Processing Based Artificial Intelligence Models

Sridevi Chitti*[1], J. Tarun Kumar[2], P. Ramchandar Rao[3], R. Archana Reddy[4]

[1,2,3]Department of Electronics and Communication Engineering, Kakatiya Institute of Technology and Science, Warangal, India

[4]Department of Humanities and Mathematics, SR University, Warangal, India

sridevireddy.arram@gmail.com, tarunjuluru@gmail.com, chandupamulaparthi@gmail.com, archanareddy.srec@gmail.com

Corresponding author: Ch. Sridevi Reddy

Corresponding mail: sridevireddy.arram@gmail.com

Article History
Volume 6, Issue 11, 2024
Received: 02 Jun 2024
Accepted: 15 Jun 2024
doi: ***10.48047/AFJBS.6.11.2024.50-62***

Abstract

When a typhoon or natural disaster occurs, a significant number of orchard fruits fall. This has a great impact on the income of farmers. In this paper, introduced an AI-based method to enhance low-quality raw images. Specifically, it focus on apple images, which are being used as AI training data. This model utilize both a basic program and an artificial intelligence model to conduct a general image process that determines the number of apples in an apple tree image. The main objective is to evaluate high and low performance based on the close proximity of the result to the actual number. The artificial intelligence models utilized in this study include the inception-V3, Convolutional Neural Network (CNN), VGG16, and Random Forest models, as well as a model utilizing traditional image processing techniques. The study found that 53 red apple fruits out of a total of 87 were identified in the apple tree image, resulting in a 62% hit rate after the general image process. The VGG16 model identified 66, corresponding to 90%, while the Random Forest model identified 37, corresponding to 85.4%. The CNN model identified 51, resulting in a 97.6% confirmation rate and inception-V3 model identified 52, resulting in a 99.2%. Therefore, selected an artificial intelligence model with outstanding performance and use a real-time object separation method employing artificial function and image processing techniques to identify orchard fruits. This application can notably enhance the income and convenience of orchard farmers.

Keywords: Machine learning, Deep learning, inception-V3, CNN models, Random forest models, VGG16 models, Image processing

1. Introduction

Recently, research has been actively conducted in the field of image processing to enable efficient management by combining object detection and tracking technology of intelligent surveillance systems with various services, products, and industries. In many fields of industry, systems that can detect objects in real time using image processing technology are used [1], but as the number of objects to be managed increases, it is difficult and expensive to detect them, so there are practical difficulties [2].

Therefore, it is necessary to apply an artificial intelligence model that can efficiently detect even if the number of objects is large at a low cost, and it is essential to conduct research to improve the performance of the existing system so that it can detect objects individually. A system that can detect and track objects using artificial intelligence is required to be used in various environments as it is used in various fields, while the more numerous and crowded the environment, the more it fails to detect individual objects in the image, so it can no longer detect and track objects, and the performance is low.

In particular, a typical object tracking algorithm can detect and track individual objects when they are separated, but in some cases, the target object is lost or mistaken for another object during the tracking [3]. One significant cause of object tracking failure is detection error, where two or more distinct objects are identified as a single entity without clear boundaries when they are in close proximity. Accurately separating these adjacent objects in real-time is essential for reducing recognition errors and improving the performance of AI detection systems. Therefore, research on object separation is actively advancing in the field of image processing. Many companies are adopting these systems due to their convenience, particularly in light of the current labor shortage in rural areas.

An example of research in this area involves a system utilizing autonomous robots for fruit detection and harvesting. This research recommends integrating robotics with computer vision to automate agricultural tasks. The study aims to develop a system that accurately detects fruit and calculates yields in orchards, providing a comprehensive overview of automated fruit detection technology and its potential applications in agriculture using various sensors and image processing technologies. It examines fruit identification technology and explores how different image processing and deep learning techniques can be employed to identify and manage fruit.

Natural disasters are currently adversely affecting farmers' incomes and agricultural productivity. Specifically, events like typhoons cause significant fruit drops in many orchards, leading to reduced yields and lower income for farmers.

Fruit crop accident insurance is designed to protect farmers from crop loss and offer additional security for fruit crops. Premiums and payouts for this insurance depend on various factors, including the type of crop, the region where it is grown, and the coverage amount. When a natural disaster occurs, a damage assessor visits the affected farmer to verify the fruit loss and determine the appropriate compensation. These methods are carried out very slowly, and the method of identifying fallen fruits is also carried out in an unscientific way, which makes it difficult to manage. In order to solve these problems, various artificial intelligence models have been applied to make important advances in the field of object detection and tracking, and they are used in various applications.

The method proposed in this paper proposes a method to accurately identify fallen fruits in orchards using various artificial intelligence models and traditional image processing methods (TIPM). The AI models used are VGG16 models, Random Forest models, and CNN models to estimate the exact number of red apple fruits open in an image of an apple tree, and to evaluate the performance of each model.

2. Related Research

Image processing is an important area for all fields related to image generation, interpretation, and recognition using computers, and was initially mainly used for diagram recognition in image formats, and there are methods such as edge detection [6] and Huff transformation [7] to recognize simple straight objects in diagrams or to extract regions of interest or denoise using morphology operations [8]. Figure 1 shows a typical image processing process.

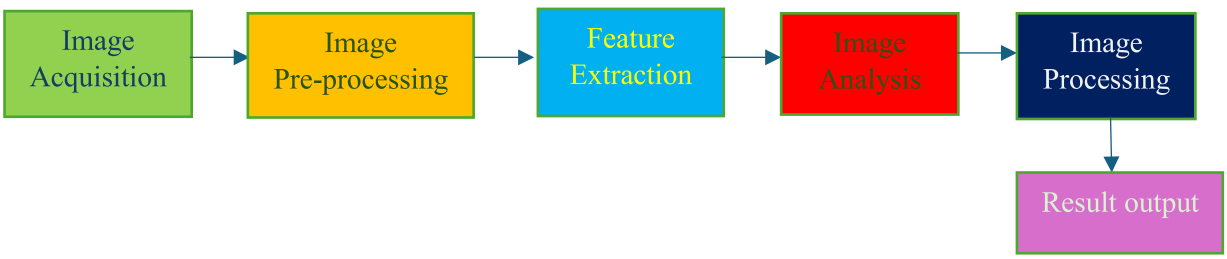

Figure 1. *General sequence of image processing*

Recently, with the advent of deep learning development tools such as Tensorflow, Keras, and PyTorch, the adoption of GPUs and the improvement of parallel computation capabilities have greatly reduced the training time of deep learning models (DLM). Due to these changes in the environment, deep learning-based research is actively being conducted in various fields, and in particular, deep learning technology based on inception-V3 is attracting great attention in the field of image recognition.

Various artificial intelligence models such as the VGG16 model, the Random Forest model, and the CNN model and inception-V3 used in this paper are used to perform image processing to detect objects in the image. These models can recognize different classes and modified objects within an image if enough training data is provided. Here, the VGG16 model is widely used for image classification and feature extraction, and it learns various features in an image, classifies an image based on it, detects a specific object, takes the image as input, and extracts the characteristics of the image by passing through various layers, which is useful in image processing tasks. The Random Forest model encompasses a set of machine learning methods employed for object recognition and image categorization. By integrating numerous decision trees, it effectively identifies objects, classifies input images, and unveils various patterns and features within them. Renowned for their superior accuracy and reliability, Random Forest models find applications across diverse image processing tasks such as object identification, image classification, and denoising. Therefore, both RandomForest and VGG16 models utilized in this study serve as indispensable tools for image processing, enabling the extraction and analysis of diverse information from images.

3. Proposed AI Model Design

In this study, we introduce a method to accurately detect fallen fruits in orchards by combining artificial intelligence models with traditional image processing methods. We utilize VGG16, RandomForest, and Inception-V3 models to estimate the exact count of red apple fruits visible in images of apple trees, enabling thorough evaluation of each model's performance. As a performance measurement method for each model, the exact number of red apple fruits open on the image of an apple tree was estimated. The exact number of red apples was followed by a process shown in Figure 2 as a method of estimating the number of fruits.

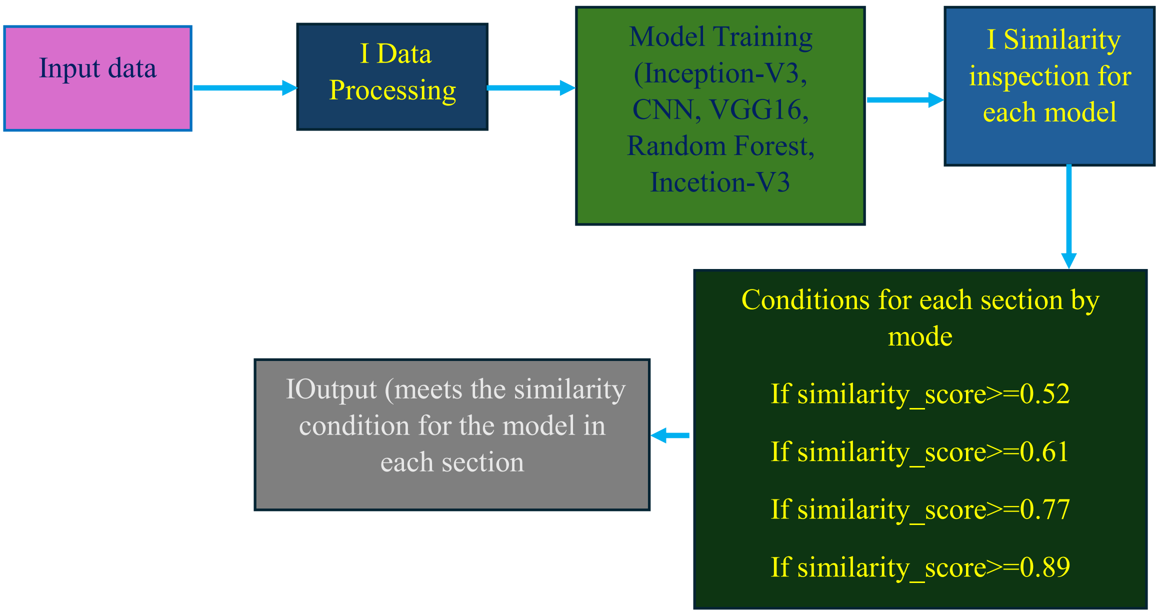

Figure 2. *Process of model training and performance evaluation*

As shown in Figure 2, the exact number of red apple fruits was estimated in 6 intervals with the condition value, and the similarity of the shape of the apple in the same shape as the learned red apple model was measured at 52%, 61%, 77%, and 99%, respectively.

The CNN model is designed as shown in Figure 3. First, the input image starts with a 224x224 pixel size and has three channels (RGB).

The first convolutional layer uses 32 3x3 filters and ReLU as the activation function. The first MaxPooling layer down samples the image to a 2x2 size. The second convolutional layer uses 64 3x3 filters and Rectified Linear Unit (ReLU) as the activation function. After the second Max Pooling layer down samples the image to a 2x2 size, the Flatten layer then converts the output from the preceding layer into a one-dimensional vector. The first Fully Connected Layer consists of 128 neurons and utilizes ReLU as its activation function.

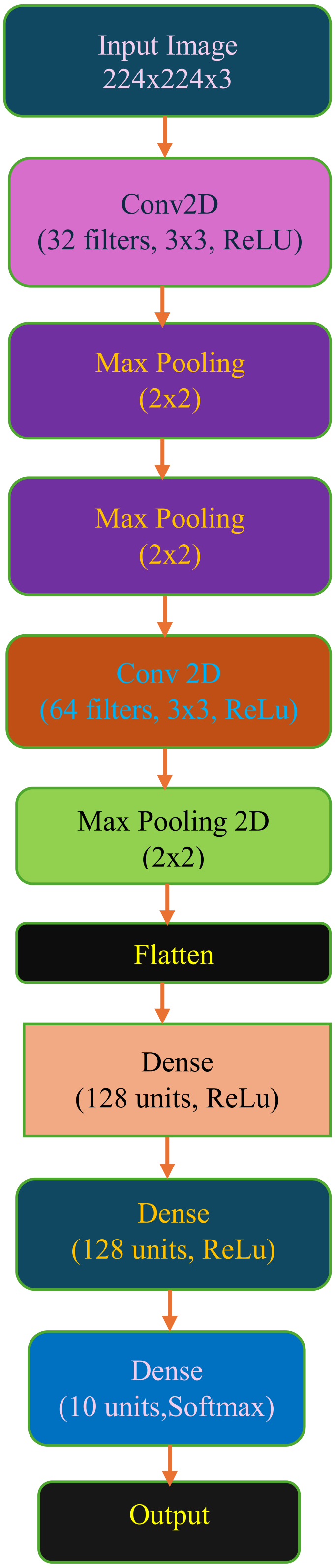

Figure 3. *Structural Design of CNN Models*

The output layer is equipped with a Sigmoid activation function and is comprised of one neuron for binary classification. These components form the foundational elements of CNN models, crucial for extracting information from images and executing classification tasks. While the Max Pooling layer down samples the image for computational efficiency, the

Convolutional layer recognizes the spatial elements in the image. The final classification is performed by the Fully Connected layer, leveraging the extracted features.

The Random Forest model is a machine learning approach based on ensembles, employing multiple decision trees to generate predictions. In the provided code snippet, we utilize the Random Forest Classifier class to instantiate and configure a Random Forest model.

 # RandomForest model creation

model = *RandomForestClassifier*(n_estimators=100, random_state=42)

The Random Forest model in the Scikit-Learn library is instantiated using the Random Forest Classifier class. Although commonly utilized for classification tasks, it's important to recognize that Random Forest can also be employed effectively for regression problems. n_estimators = 100 The number of Decision Trees that comprise the Random Forest model is specified by this parameter. We are instructed to use 100 decision trees in the provided code. More decision trees can improve the stability and performance of the model, but it also increases computational costs. random_state=42.

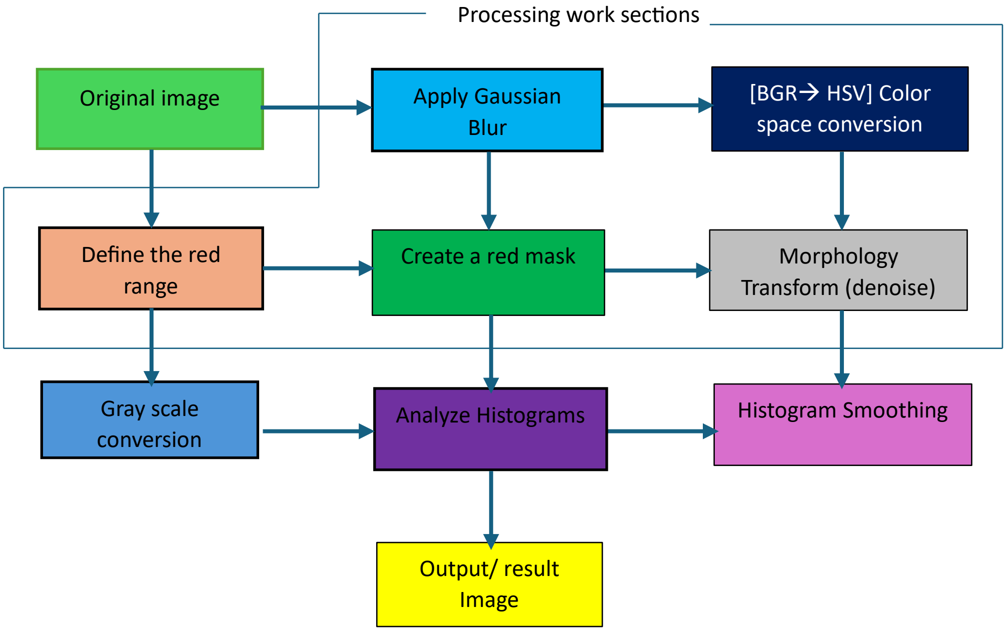

Figure 4. *Structural Design of General Image Processing*

This parameter sets the random number occurrence seed. The Random Forest model randomizes the decision tree, so each run generates a different model. Setting up the random_state ensures that the same results can be reproduced. This means that you can get the same results every time you train a model with the same data and settings.

The structure of traditional image processing techniques is designed as shown in Figure 4. Based on the code given in Figure 4, the image processing steps proceed as follows. After loading the original image from the given image file path, apply image filtering Gaussian blur to blur the image. blurred_image = cv2.GaussianBlur(image, (5, 5), 0) is the application code, where cv2. You can use the GaussianBlur() function to blur the image by applying Gaussianblur to remove noise or soften the borders. BGR to HSV conversion method hsv_image = cv2.cvtColor(blurred_image, cv2. COLOR_BGR2HSV) code, where the cv2.cvtColor() function is used to convert the image from the Blue-Green-Red (BGR) color space to the Hue-Saturation- Value (HSV) color space. The range definition method in the red area is divided by lower_red = np.array([0, 100, 100]) # the lower bound of the red range and upper_red = np.array([10, 255, 255]) # the upper bound of the red range. Here, we define a range of red in the HSV image to find the red area, which means specifyingthe Hue value of the red pixel as a range. The red mask is generated using the code red_mask = cv2.inRange(hsv_image, lower_red, upper_red), where the cv2.inRange() function is used to mask the pixels within the red range. As a morphology transformation (denoising) method, kernel = np.ones((5, 5), np.uint8),morphology_image = cv2.morphologyEx(red_mask, cv2. MORPH_CLOSE, kernel) code, where the cv2.morphologyEx() function is used to apply morphological transformations to the image to remove noise, mainly cv2. Use MORPH_CLOSE to perform a close operation. This process results in image preprocessing, which involves filtering the image, identifying red areas, and applying morphological transformations to the red mask to remove noise.

The structure of the VGG16 model shows each layer and its connection structure as shown in Figure 6. To briefly explain the structure of the VGG16 model, it first takes an input layer of 224x224 size color images, and there are 5 Convolutional Layers (Conv), each of which uses a kernel of different sizes to extract the features of the image. It uses ReLU as an activation function, and after the Conv layer, there is a Max Pooling layer, which performs the function of down sampling the image to reduce its size and increase computational efficiency.

As shown in Figure 5, the resulting image of each step is displayed as a single figure and output.

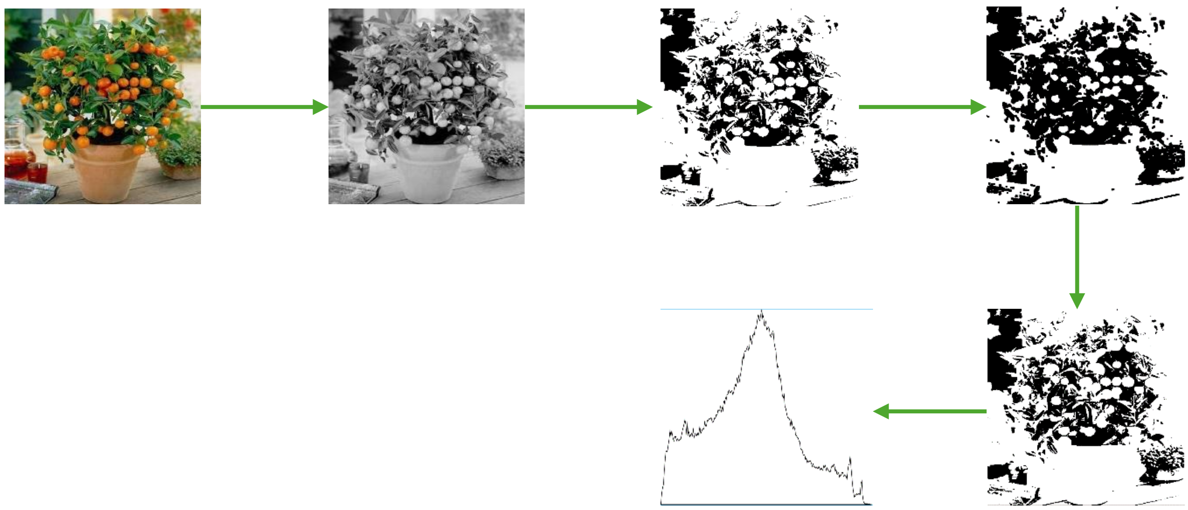

Figure 5. *The shape of the result of the image process.*

The Fully Connected Layers (FCL) produce high-dimensional feature vectors based on the features extracted by the Convolutional layer. These vectors are pivotal for image classification tasks. Finally, the Output Layer functions as the last step in the classification process, utilizing the softmax activation function to produce probabilities for each class.

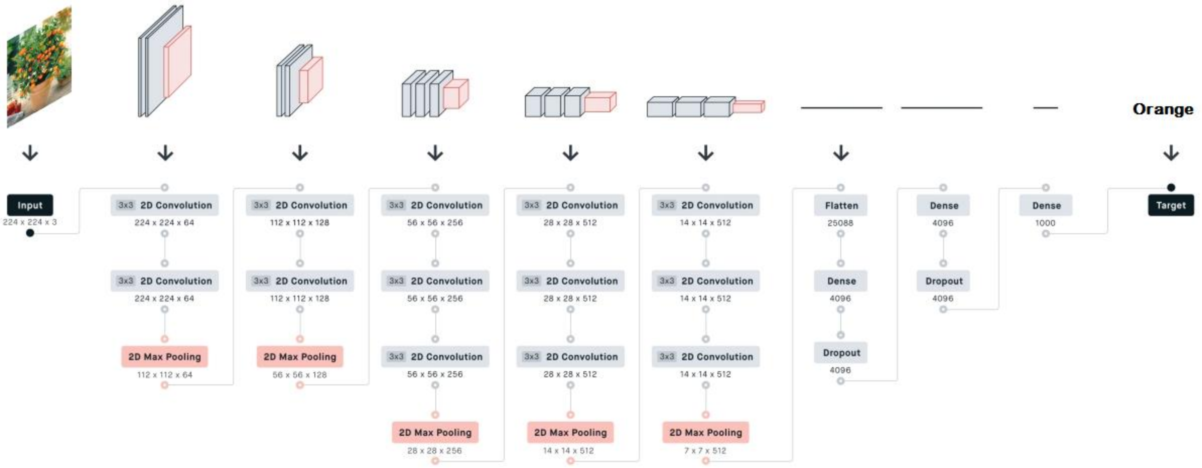

Figure 6. *Structural design of VGG16 Image Processing*

The VGG16 model is composed of 16 layers designed for image classification and feature extraction. Utilizing pre-trained weights obtained from extensive image datasets, it excels in efficient feature extraction.

4. Implementation and Results

The proposed approach integrates artificial intelligence models and conventional image processing techniques to accurately quantify fallen fruits in orchards. Utilizing models such as VGG16, Random Forest, and CNN, the method estimates the precise number of open red apples in apple tree images, assessing the performance of each model.

The objective of this study is to assess the efficiency of the proposed models in determining the precise count of red apple fruits in an image of an apple tree located in an apple orchard. In this context, the use of technical term abbreviations will be defined upon their first usage. The text adheres to a conventional structure, with clear, concise sentences and paragraphs that establish causal connections between statements for a coherent and logical flow of information. To this end, we employed artificial intelligence models, namely VGG16, Random Forest, and CNN models, along with traditional image processing methods, and compared and analyzed their performance. The paper's organization adheres to common academic sections and maintains consistent author and institution formatting, while titles are accurate and comprehensible. The language is objective, formal, and value-neutral, with passive tone and impersonal constructions. The writing uses high level, standard language and avoids biased, figurative, emotional, or ornamental expressions. Additionally, followed a consistent citation style and marked quotes to ensure proper formatting features. Finally, the writing avoids filler words, slang, and contractions, and is free from grammatical errors, spelling mistakes, and punctuation errors.

4.1 Data Set

To evaluate the AI model, we organized the dataset into apple images, apple tree leaves, and apple tree stems as shown in Table 1. Of the 352 total image data used in this paper, 292 apple images, 30 apple tree leaves, and 30 apple tree stems were used.

Table 1. Main parameters

Division	Number of apples	Number of tree leaves	Number of tree trunks	Total
Training data	210	10	10	230
Test data	42	10	10	62
Validation data	50	10	10	70
Total	**302**	**30**	**30**	**362**

In this study, the software environment for the experiment was developed using Python 3.10 version, the artificial intelligence library used the PyTorch-based MMDetection API was used.

5. Result

The method used in this paper is an artificial intelligence model, using the VGG16 model, the Random Forest model, and the CNN model, and traditional image processing methods to estimate the exact number of open red apple fruits in the image of an apple tree, and to compare and evaluate the performance of each model. Table 4 shows the apple object detection ratio by model.

Table 2. Apple detection results by model

Division	Model	Number of original objects	Correct answer	Accuracy by similarity score
1	Inception-V3	53	52	99.2%
2	CNN	53	51	97.6%
3	VGG16	53	66	90.0%
4	Random Forest	53	37	85.4%
5	Classical Image Processing	53	58	96.9%

In terms of the accuracy of each model, the Inception-V3 model first correctly detected 53 out of 52 objects, regardless of the number of original image objects and similarity scores, resulting in a 99.2% accuracy rate. the CNN model first correctly detected 53 out of 51 objects, regardless of the number of original image objects and similarity scores, resulting in a 97.6% accuracy rate. The VGG16 model achieved a 90% accuracy rate by correctly identifying 66 out of 73 objects, regardless of the number of original image objects and similarity scores. The Random Forest model attained an 85.4% accuracy rate, correctly detecting 37 out of 53 objects. In contrast, Classical Image Processing demonstrated an impressive accuracy rate of 96.9%, accurately identifying 58 out of 53 objects, irrespective of the number of original image objects and similarity scores.

These results offer a comprehensive assessment of each model's performance. The Inception-V3 and CNN models showcase impressive accuracy levels, whereas the Random Forest model demonstrates comparatively lower accuracy. Conversely, the Classical Image Processing method achieves exceptionally high precision. When choosing a model, factors such as accuracy and computational speed should be carefully considered, depending on the project's objectives.

6. Conclusion

This study employed a combination of AI models and traditional image processing techniques to improve the detection of red apple objects in orchards. By assessing their performance, the study aimed to accurately quantify the number of red apples. Key findings indicate that the Inception-V3 model achieved an outstanding accuracy of 99.2% in detecting red apple items. This highlights the model's exceptional ability to identify objects, even with low similarity scores. Moreover, the model demonstrates high reliability in item detection and conducts comprehensive analyses of each object's unique properties.

The study is poised to examine and contrast multiple models and methods for detecting red apple objects in orchards to determine the optimal choice for specific scenarios. Future investigations will involve implementing these models in authentic orchard settings to refine the development of real-time object detection systems. The anticipated outcomes include providing fruit growers and crop management with innovative insights and approaches to fruit detection and crop management.

References

[1]. Al-Hatmi, M. O., & Yousif, J. H. (2017). A review of Image Enhancement Systems and a case study of Salt &pepper noise removing. International Journal of Computation and Applied Sciences (IJOCAAS), 2(3), 171-176.

[2]. Alighaleh, P., Khosravi, H., Rohani, A., Saeidirad, M. H., & Einafshar, S. (2022). The detection of saffron adulterants using a deep neural network approach based on RGB

images taken under uncontrolled conditions. Expert Systems with Applications, 198, 116890.

[3]. Alkishri, W., Widyarto, S., Yousif, J. H., & Al-Bahri, M. (2023). Fake Face Detection Based on Colour Textual Analysis Using Deep Convolutional Neural Network. Journal of Internet Services and Information Security (JISIS), 13(3), DOI: 10.58346/JISIS.2023.I3.009.

[4]. Calderon, S., Fallas, F., Zumbado, M., Tyrrell, P. N., Stark, H., Emersic, Z., ... & Solis, M. (2018, October). Assessing the impact of the deceived non local means filter as a preprocessing stage in a convolutional neural network based approach for age estimation using digital hand x-ray images. In 2018 25th IEEE International Conference on Image Processing (ICIP) (pp. 1752-1756). IEEE.

[6]. Deepa, T., Satyavathy, G., Priya, A., & Hamrish, S. S. (2023). Deep Learning with Pytorch: Siamese Network. Tuijin Jishu/Journal of Propulsion Technology, 44(4), 4412-4421.

[7]. Fekihal, M. A., & Yousif, J. H. (2012). Self-organizing map approach for identifying mental disorders. International Journal of Computer Applications, 45(7), 25-30.

[8]. Ghandour, C., El-Shafai, W., & El-Rabaie, S. (2023). Medical image enhancement algorithms using deep learning-based convolutional neural network. Journal of Optics, 1-11.

[9]. Haji, S. H., & Abdulazeez, A. M. (2021). Comparison of optimization techniques based on gradient descent algorithm: A review. PalArch's Journal of Archaeology of Egypt/Egyptology, 18(4), 2715-2743.

[10]. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. ArXiv. /abs/1512.03385.

[11]. Hasoon, F. N., Yousif, J. H., Hasson, N. N., & Ramli, A. R. (2011). Image enhancement using nonlinear filtering based neural network. Journal of Computing, 3(5), 171-176. 261

[12]. Huang, L., Qin, J., Zhou, Y., Zhu, F., Liu, L., & Shao, L. (2023). Normalization techniques in training dnns: Methodology, analysis and application. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[13]. Khamis, Y., & Yousif, J. H. (2022). Deep learning Feedforward Neural Network in predicting model of Environmental risk factors in the Sohar region. Artificial Intelligence & Robotics Development Journal, 201-2013.

[14]. Lapid, R., & Sipper, M. (2022, July). Evolution of activation functions for deep learning-based image classification. In Proceedings of the genetic and evolutionary computation conference companion (pp. 2113-2121).

[15]. Öztürk, Ş., & Akdemir, B. (2018). Effects of histopathological image pre-processing on convolutional neural networks. Procedia computer science, 132, 396-403.

[16]. Pajankar, A., & Joshi, A. (2022). Hands-on Machine Learning with Python: Implement Neural Network Solutions with Scikitlearn and PyTorch. Apress.

[17]. Şengöz, N., Yiğit, T., Özmen, Ö., & Isik, A. H. (2022). Importance of preprocessing in histopathology image classification using deep convolutional neural network. Advances in Artificial Intelligence Research, 2(1), 1-6.

[18]. Singh, V. K., Sharma, K., & Sur, S. N. (2023). A survey on preprocessing and classification techniques for acoustic scene. Expert Systems with Applications, 120520.

[19]. Tabik, S., Peralta, D., Herrera-Poyatos, A., & Herrera Triguero, F. (2017). A snapshot of image pre-processing for convolutional neural networks: case study of MNIST.

[20]. Wang, Y., He, Y., & Zhu, Z. (2022). Study on fast speed fractional order gradient descent method and its application in neural networks. Neurocomputing, 489, 366-376.

[21]. Yousif, J. H., & Kazem, H. A. (2021). Prediction and evaluation of photovoltaic-thermal energy systems production using artificial neural network and experimental dataset. Case Studies in Thermal Engineering, 27, 101297.

[22]. Yogeshwari, M., & Thailambal, G. (2023). Automatic feature extraction and detection of plant leaf disease using GLCM features and convolutional neural networks. Materials Today: Proceedings, 81, 530-536.

[23]. Yousif, J. H., & AlRababaa, M. S. (2013). Neural technique for predicting traffic accidents in Jordan. Journal of American Science, 9(11), 528-525.

[24]. Zhou, K., Oh, S. K., Pedrycz, W., & Qiu, J. (2023). Data preprocessing strategy in constructing convolutional neural network classifier based on constrained particle swarm optimization with fuzzy penalty function. Engineering Applications of Artificial Intelligence, 117, 105580.