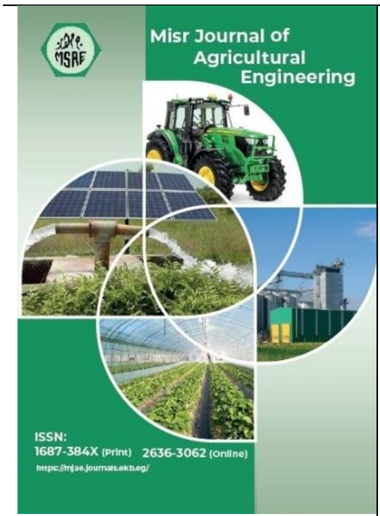
MISR JOURNAL OF AGRICULTURAL ENGINEERING ISSN-Print: 1687-384X
https://mjae.journals.ekb.eg/ ISSN-Online: 2636-3062

Misr J. Ag. Eng., 41 (---): ….. - ….. **DOI:** 10.21608/mjae.2024.286079.1137

CLASSIFICATION OF DATES QUALITY USING DEEP LEARNING TECHNOLOGY BASED ON CAPTURED IMAGES

Waleed Kamel ElHelew[1], Dalia Abo Bakr[2], Sahar Zayan [3] and Muhammad Ahmad Mayhoub[4]*

[1] Assoc. Prof., Ag. Eng. Dept., Fac. of Ag., Ain Shams U., Al Qalyubiyah, Egypt.
[2] Grad. Stud., Ag. Eng. Dept., Fac. of Ag., Ain Shams U., Al Qalyubiyah, Egypt.
[3] Head of Res. at the Plant Pathology Research Institute., Ag. Res. Center (A. R. C)., Giza, Egypt.
[4] Assist. Prof., Ag. Eng. Dept., Fac. of Ag., Ain Shams U., Al Qalyubiyah, Egypt.
* E-mail: mohamedmayhoub@agr.asu.edu.eg, ORCID: 0000-0002-1422-289X



Keywords:
AI Techniques; Convolutional Neural Network (CNN); Transfer Learning; Date Fruits Handling; Rejected Fruits.

ABSTRACT

Dates are a common fruit in many Middle Eastern and African nations and have religious and cultural value. one of the key elements in judging the quality of dates is sorting according to their health state. Combining rejected dates with accepted ones causes significant economic losses in both storage and exportation. Despite being a crucial stage for obtaining high-quality dates and reducing losses, this sorting process is still conducted using traditional methods. Thus, this study aims to classify date fruit quality (accepted or rejected) with machine learning technology to reduce cost, time, and improve the quality of final product. In this study, several Convolutional Neural Network architectures (Inception-v3, Inception-ResNet-v2, VGG19) were used to classify three varieties of date fruit (Mejdool, Saiedi, El-Wadi). These varieties were classified into accepted and rejected samples to build the dataset im-ages. An Arduino Automatic mobile camera shutter controller captured the dataset images. In addition to the Kaggle dataset which was added to the accepted images. The total dataset consisted of 5,945 images, comprising 3,142 accepted images and 2,803 rejected images. By comparing the results of different architectures, Inception-ResNet-v2 demonstrated the best performance, achieving an accuracy of 98.99% and a loss of 0.0344. Therefore, it can be concluded that the Inception-ResNet-v2 model could be utilized to develop a suitable computer vision system, thereby enhancing the date sorting process and facilitating the packaging of high-quality dates.

INTRODUCTION

There are more than 200 different varieties of date fruit on the globe. It's fascinating to observe that each sort differs from the others due to a few distinctive qualities. Pests, insects, mites, and mechanical equipment harm some of the products during the growing, ripening, and harvesting stages of dates, and these flaws result in large financial losses for the storage and exporting of date fruit (Sarraf et al., 2021). The Egyptian Dates Sector Development

Strategy encourages modern technology to reduce post-harvest losses (FAO, 2018). Egypt is the largest date producer country in the world with 1747714 tons of production and 32557 tons of exported quantities in 2021 (FAO, 2022), there was a gap between production and exportation quantity.

There are several ways to identify faulty fruits and vegetables, including conventional methods (by sight, or at the labors), imaging techniques (such as hyperspectral imaging systems, X-ray imaging, magnetic resonance imaging (MRI), and thermal imaging), among others (Adedeji et al., 2020; Nturambirwe and Opara, 2020). Classifying dates in Egypt is still done traditionally like sorting by sight.

To overcome the date sorting challenges using modern technology like deep learning will lead to improved sorting processes. Recently machine learning and artificial intelligence have been widely used in the agriculture sector, especially using convolutional neural networks as it is a branch of deep learning widely utilized in classification. Convolutional Neural Network (CNN) is the most famous and commonly employed algorithm. The main benefit of CNN compared to its predecessors is that it automatically identifies the relevant features without any human supervision (Alzubaidi et al., 2021) . CNNs have been applied in image classification for a long time. Compared with other methods CNNs can achieve better classification accuracy on large-scale datasets, due to their capability of joint feature and classifier learning (Gu et al., 2018). As used for large-scale datasets, there was a lack of open-sourced targeted datasets for this research.

CNNs have drawn a lot of interest since they are a cutting-edge and promising type of architecture, and numerous real-world uses in industries like agriculture have been made (Wang et al., 2021). Researchers (Yu et al., 2023) utilized transfer learning and the following CNN architectures (AlexNet and VGG-19) (ResNet-18, ResNet-50, and ResNet-101) to detect and classify 13 classes of apples using two datasets. The results demonstrate that the dataset configuration significantly affected the classification outcomes, with all models achieving above 96.1 percent accuracy on dataset A (training-to-testing = 2.4:1.0) as opposed to 89.4-93.9 percent accuracy on dataset B (training-to-testing = 1.0:3.7). On dataset A and 93.9 percent on dataset B, the VGG-19 model had the best accuracy. Also, researchers in (Azadnia et al., 2023) categorized the hawthorn photos using the suggested CNN, Inception-V3, ResNet-50, and architectures for each of its three phases of maturation, which achieved test accuracies of 100%, 99.63%, and 99.63%, respectively.

This study used dates at the Tamr stage for the Mejdool, Saeidi, and El-Wadi varieties. The main objective of this research is to use some of the different CNN models (Inception-v3, Inception-ResNet-v2, and VGG 19) to evaluate the classification process for the quality of dates and to determine the best model for a high-quality classification process.

MATERIALS AND METHODS

This section will discuss the methodology of implementing classifying models for sorting accepted dates from rejected ones. Different convolutional neural networks (CNNs) were used to compare the dataset. Figure. 1 shows the methodology steps.

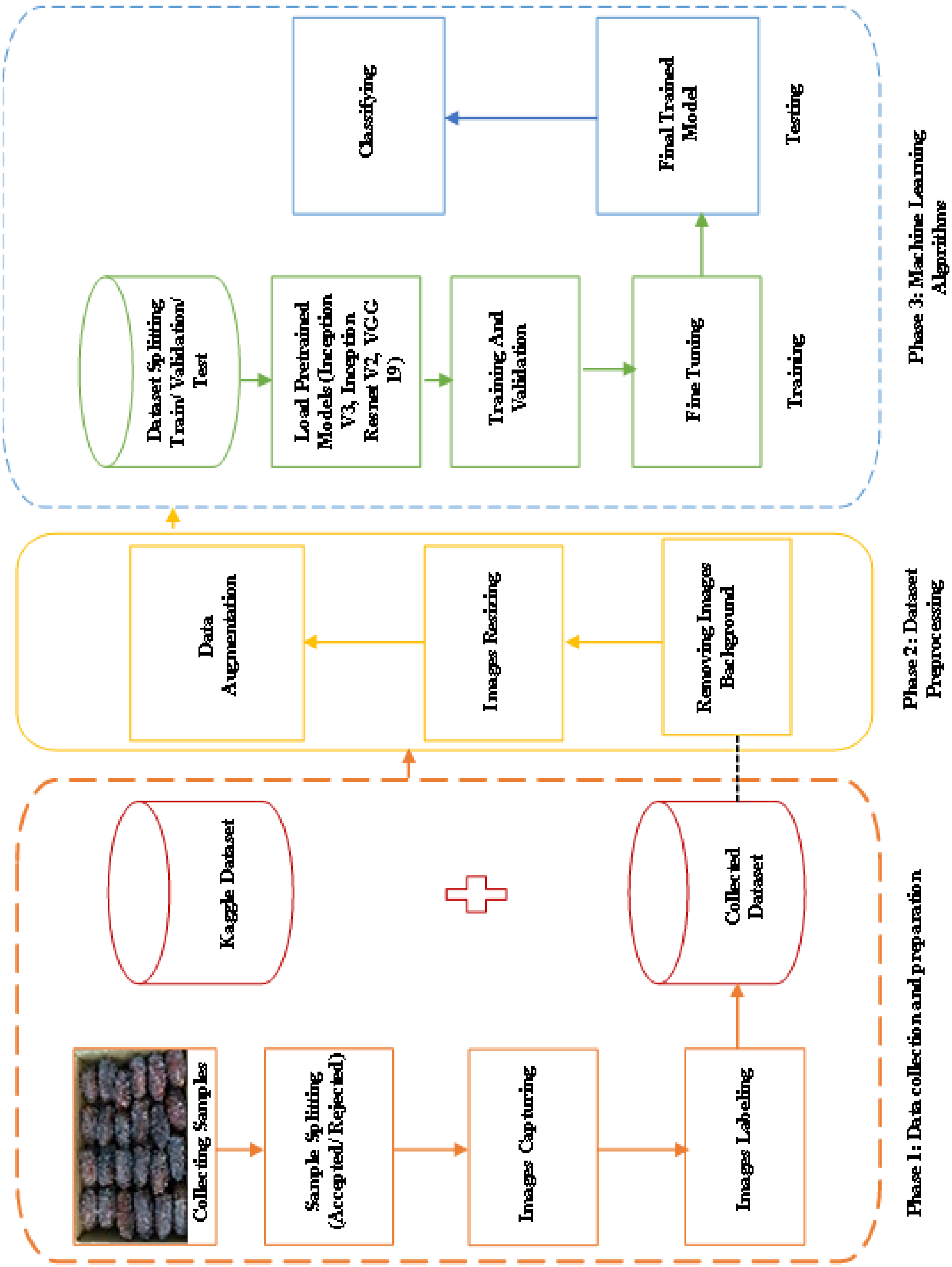


Fig. (1): Flow chart of date classification quality methodology.

2.1 Data Collection and Preparation

2.1.1 Collecting and splitting samples.

Samples

The date varieties (Mejdool, Saeidi, El-Wadi) were gathered from different regions around Egypt. Tamr maturity stage dates are utilized. The rejected dates were from samples of factory-rejected fruits according to Codex Standard 143-1985 (Codex, 1985). All dates had been divided into two groups (accepted and rejected) for each variety. The accepted samples were samples that had no defects or any other infected dates. The rejected samples were samples that had insects, mechanical flaws, bacterial diseases, physiological flaws, and disordered fruits. Mejdool samples for accepted fruits were 93 fruits, while those for rejected fruits were 126 fruits, and there were 330 Saeidi samples for fruit that was accepted, and 576 fruit that was rejected. Lastly, El-Wadi samples for accepted fruits were 50 fruits and for the rejected fruits were 186 fruits. Each sample was given a label and placed in its case.

Datasets

The collected dataset and the Kaggle dataset were two categories for the utilized dataset. Three date varieties of Mejdool, Saeidi, and El-Wadi were included in the collected dataset, and each variety included an average of three images for both accepted and rejected samples as shown in Figure. 2. They were then divided and given new labels in separate files for each group. In addition, the Kaggle data collection is used to reduce the imbalance between the accepted and rejected datasets as the accepted are fewer than the rejected ones, it was constructed by (Alhamdan and Howe, 2021) using dataset information gathered from a total of 1658 high-quality images. Image dimensions were set at 120 x 120. The Kaggle dataset was included in an accepted dataset that was collected to enlarge the trained dataset to get better performance for our study. From Table 1 total dataset images were 5945 images, the accepted images were 52.85% of the total, and the rejected images were 47.15% of the total images.

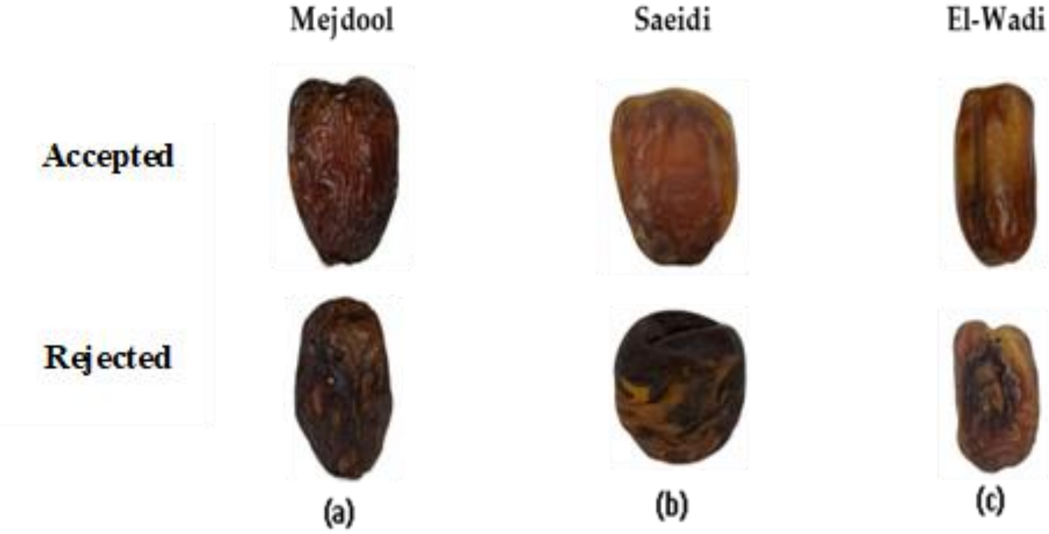


Fig. (2): Samples of collected dataset images for the accepted and rejected three date varieties (a) Mejdool, (b) Saeidi, (c) El-Wadi

Table 1: Number of date samples and captured images for each date variety for the accepted and rejected dataset.

Classes	Accepted		Rejected	
	Samples No.	Images No.	Samples No.	Images No.
Mejdool	93	377	126	508
Saeidi	330	987	576	1734
El-Wadi	50	120	186	561
Total	473	1484	888	2803

2.1.2 Images capturing mechanism.

Building dataset for the accepted and rejected dates by capturing images with the self-made mechanism. It is described as a turntable with an automatic mobile camera shutter controller. Samsung Galaxy A71 phone camera used for capturing the collected with the following specifications: 1- Camera resolution 48 megapixels, 2- Dimensions 2604x4624, 3- Exposure time 1/50 sec, and 4- Focal length 5mm. The software is programmed by Arduino software to take approximately 3-4 images for each fruit, and when the button is pressed, the mechanism powers up and the motor rotates continuously at 120 degrees for each shot as shown in Figure 3. Once all the photos have been taken with no zoom and no flash, the turn table stops so that the fruit may be switched to another one. To improve illumination around the fruit, all of the components were placed in a handmade light box, as shown in Figures 3 and 4. The phone is held on a ring light stand, and the cork plate that the fruit was fastened to is placed on top of the servo motor with a needle.

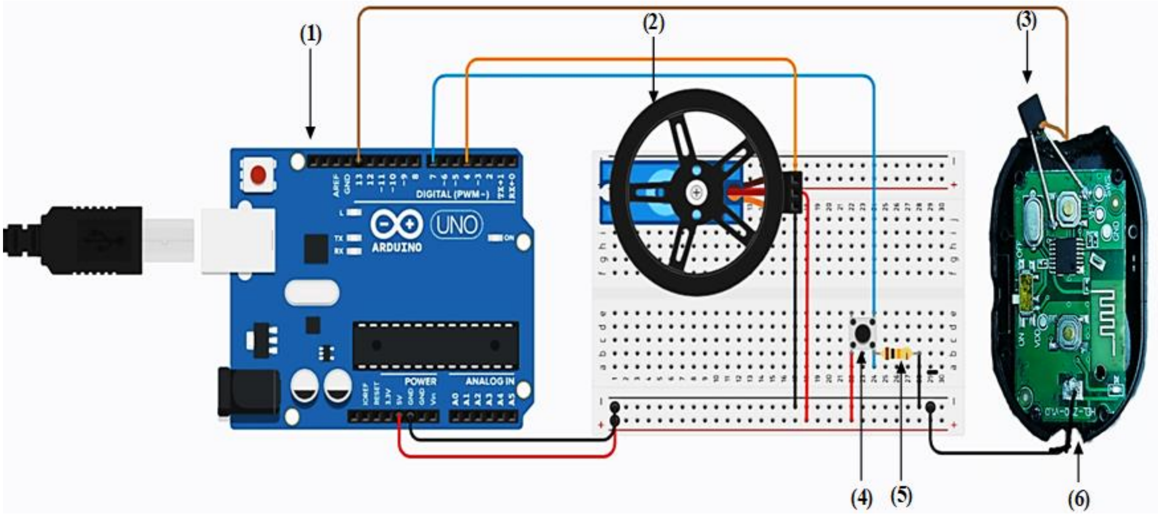


Fig. (3): Automatic mobile camera shutter controller for date fruit photography Arduino circuit components: 1- Arduino Uno, 2- A 360 degrees servo motor, 3- NPN transistors, 4- Bush button 5- 10k Ohm resistor, 6- Bluetooth camera remote for shutter controlling

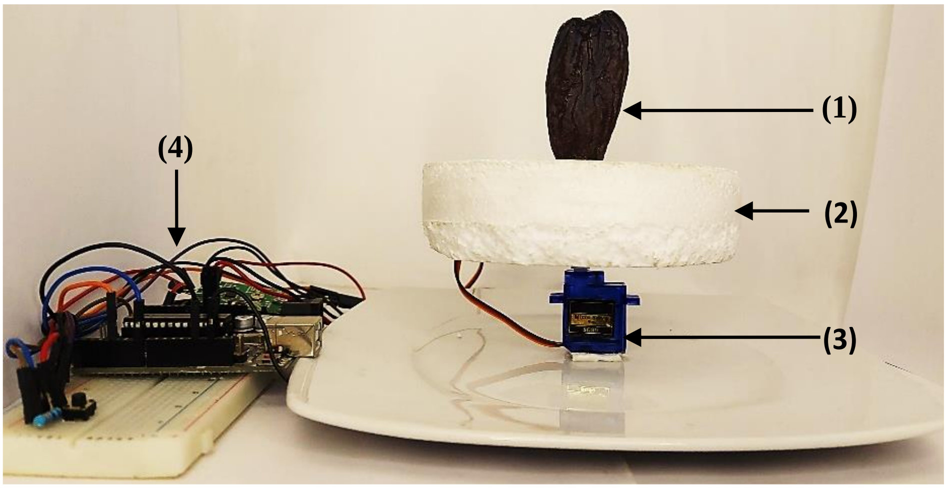
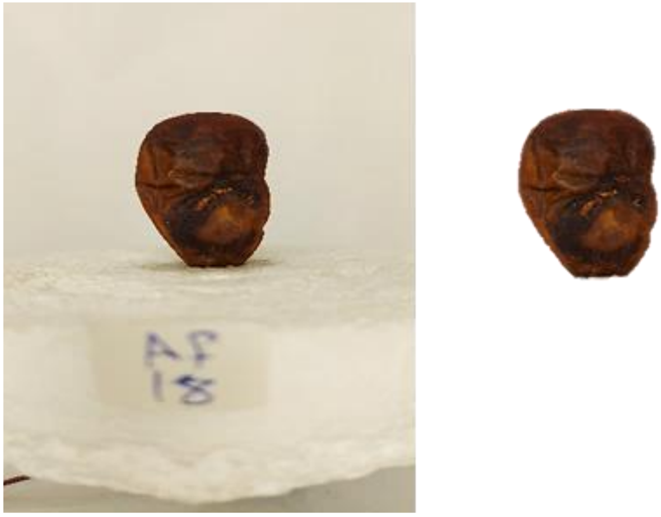


Fig. (4): Automatic mobile camera shutter controller mechanism components: 1- Date sample, 2- cork plate, 3- servo motor, 4- Arduino circuit

2.2. Images preprocessing

2.2.1 Removing background.

By eliminating distracting backdrop elements from the image and concentrating on the characteristics collected from the fruits alone, removing the background from an image improves the identification of the differences between the two classified date categories. A Python library called REMBG is used to remove background from photos in bulk as shown in Figure 5. It relies on a deep network architecture utilizing U2-Net to identify salient objects (SOD) (Qin et al., 2020).



(a) (b)

Fig. (5): Sample of images: (a) before removing image background, and (b) after removing image background preprocessing.

2.2.2 Images resizing and data augmentation.

After removing the background images' extensions were changed from png to jpg extension to reduce their storage capacity. Also, all images were resized and cropped with an aspect ratio of 1:1 after background removal using the Fast Stone Image Viewer software program. Another method for resizing while training the dataset is to fit the image size with the selected CNN architecture. Each image size for the architecture was as follows: Inception-v3 299 x 299 pixels, Inception-ResNet-v2 299 x 299 pixels, and VGG 19 244 x 244 pixels.

The usage of data augmentation techniques is to enlarge our small dataset, as it is one potential answer if the objective is to expand the amount of accessible data and prevent the overfitting problem. The used techniques in this study were as follows: 1- Flip: Switching the horizontal axis occurs far more frequently than flipping the vertical axis, horizontal flipping was used in this study. 2- Rotation: The picture is rotated to the right or left along an axis between 1° and 359° to perform rotation augmentations. The rotation degree parameter has a significant impact on the safety of rotational augmentations. Our dataset was rotated using a random rotation of 0.2 according to (Shorten and Khoshgoftaar, 2019) slight rotations such as between 1 and 20 or − 1 to − 20 could be useful for recognition. 3- Zoom: zooming in and out values were used as (0.5, 0.2) (Alzubaidi et al., 2021; Shorten and Khoshgoftaar, 2019). The three augmentation techniques are all done on each image together with random scales.

2.3 Machine Learning Algorithms

2.3.1 Experimental setup

The models were implemented on jupyter notebook with Python 3.9 software, and they were trained and tested on PC with the following specifications: Intel core i5-11400f @2.6GHz, memory 16 GB, operating system Windows 10 pro 64 bits, and with graphics GeForce RTX 3060 Ti.

2.3.2 CNN architectures for dataset classification

Data splitting

This section reported the training of the dataset, from a total of 5945 images, the dataset is randomly split into train get 80% of the images, verification 10%, and finally tested images 10% of the total input images. This ratio provided a compromise between increasing the volume of training data and reducing volatility in performance testing. This split ratio resembles others that are employed elsewhere (Mohanty et al., 2016; Nasiri et al., 2019; Verma et al., 2020). The images were rescaled to fit each of the chosen pre-trained models' input size requirements.

Transfer learning (pre-trained models)

The dataset was trained with three different pre-trained models by using transfer learning as shown in Figure. 6. As our data is small transfer learning is better for getting high performance than the full-scale scratched models (Alsirhani et al., 2023). The selected pre-trained models were Inception-v3, Inception-ResNet-v2, and VGG 19, they were chosen according to their high accuracy in classification (Alzubaidi et al., 2021).

By mentioning the pre-trained architectures, there are explanations for each of them as follows: Inception-v3 architecture as its fundamental building block of Inception is made up of four parallel branches: a 1x1 kernel convolution, two 3x3 convolutions, a 1x1 convolution, a 3x3 convolution, a pooling, and lastly a 1x1 convolution. There are ten modules in Inception, albeit the number of modules decreases as the depth of the net increases. To lower the computational cost, two convolutions, a 1x7 followed by a 7x1 are used in place of the n x n convolutions in five of the modules. The last two convolutions are replaced by the last two modules, which are a 3x3 of the first branch with two convolutions each, and a 1x3 followed by a 3x1 this time in parallel. Inception-v3 contains a total of 42 layers, and with an input image size shape (299x299). (Pérez-Pérez et al., 2021; Szegedy et al., 2015a)

Inception-ResNet-v2 architecture is a sub that lowers the top-5 error rate to under 7% and was created utilizing the notion of the Inception module established by (Szegedy et al., 2015b). Additionally, they suggested some basic ideas and enhancements to scale up the convolution network utilizing an Inception type network for computer vision (Szegedy et al., 2015a).The Inception module makes use of some different hyperparameters, such as multiple feature maps and different kernel sizes, such as 1x1, 3x3, and 5x5, which enables the detection of patterns utilizing a range of different scales of layers, strides, and paddings. The layers utilize the same padding and a stride of 1 with the Inception module, and with an input image size shape (299x299) (Tran et al., 2019).

Visual geometry group (VGG 19) architecture in the examination of how the convolutional network's depth affects accuracy in a context where a lot of images need to be recognized. Its primary contribution is a detailed analysis of networks with increasing depth utilizing an architecture with extremely tiny (3x3) convolution filters, which demonstrates that extending the depth to 16–19 weight layers may significantly outperform existing designs, with an input image size shape (224x224) (Simonyan and Zisserman, 2015). As shown in the following Figure 6 the differences between Inception-v3, Inception-ResNet-v2, and VGG 19, the input image size for each model, and the number of layers that differ from one model to another.

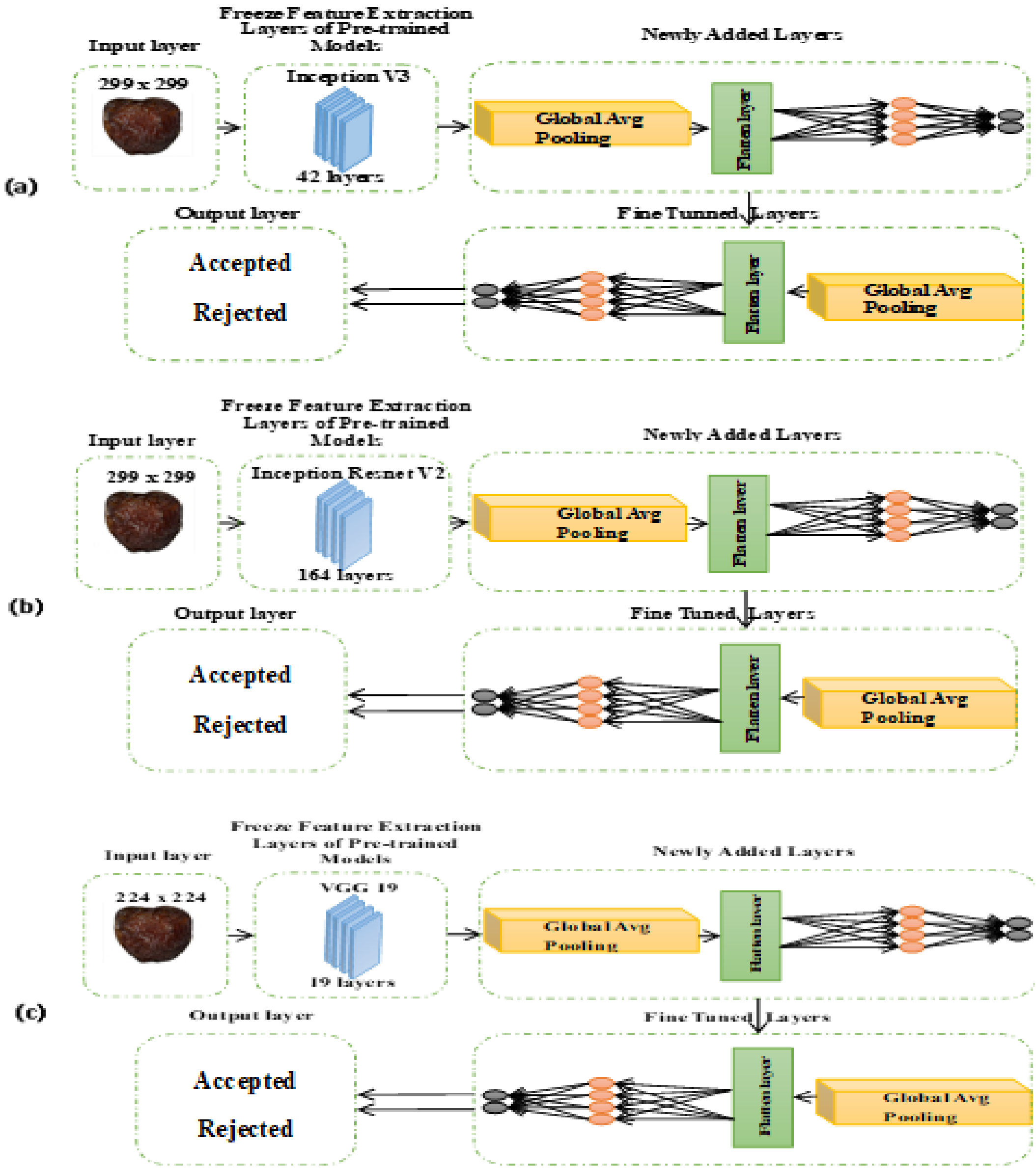


Fig. (6): Three CNN architecture layers used for classifying dates quality: (A) Inception-v3 pre-trained model, (B) Inception-ResNet-v2 pre-trained model, (C) VGG 19 pre-trained models.

For classification purposes, these pre-trained models were developed using the ImageNet dataset. 22,000 categories and 15 million images may be found in ImageNet (Krizhevsky et al., 2017). These models' learning weights may be used to recognize images in a variety of fields thanks to the vast dataset that was used to train them. The three models mentioned above are utilized as the base models. As a result, the feature extraction layers (convolutional and pooling layer pairs) are frozen to maintain their weights that have been tuned for ImageNet, preventing information loss and increasing feature extraction capabilities for training the next tasks.(et al., 2022). After trimming the original fully connected layers of the pre-trained models, additional layers were added to differentiate between the accepted and rejected dates.

An average pooling layer, it down samples the feature map by averaging the values of all the pixels in each batch. The output size of the pooling layer is calculated as follows:

$$Op= [\frac{WI-F}{S}+1] \quad (1)$$

Where WI represents the input size of the pooling operation, F is the size of the pooling filter, and S is the stride size.

A flattened layer to create a one-dimensional array out of the down-sampled feature map.

A fully connected layer by utilizing 64 filters and a Rectified Linear Unit (ReLU) activation to link every neuron in the layers before and after. The issue of disappearing gradients is helped by ReLU. (Yang et al., 2022) The equation below is used to calculate it:

$$f(x) = max\ (0, x) \ (2)$$

A dropout layer due to the smaller differences in our dataset a dropout of 0.3 was used (Hassan et al., 2021; Li et al., 2022), the dropout ratio is to reduce issues with model overfitting.

An output layer using the sigmoid activation function, it is possible to determine if a date is accepted or rejected by marking it as positive or negative. The binary parameters were chosen for comparing features between the two groups. Probabilities of binary classification are predicted using the input of the activation function as a real integer, and the allowed output range of the activation function is 0 to 1, the following mathematical formula (Alzubaidi et al., 2021).

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (3)$$

The loss function is selected as a binary cross entropy to define the differences between the actual output layer y and the predicted output layer $\hat{y}$. According to its definition, it serves as a loss function for binary classification jobs. If each classification can be reduced to a binary decision, the binary cross entropy is a highly valuable tool for training a model to answer several classification problems at once. (Usha Ruby, 2020)

$$L_{cross\_entropy\ (\hat{y},y)} = -y\ (log(\hat{y}) + (1-y)\ log(1-\hat{y})) \ (4)$$

Batch normalization layer to speed up the training of deep neural networks, batch normalization attempts to minimize internal covariate shift. This is done by fixing the means and variances of the input layer layers during a normalization stage (Saeed et al., 2023). The following equations describe the calculation for BN operation:

1) Determine the mean of mini-batch ($\mathcal{B}$)

$$\mu_{\mathcal{B}} = \frac{1}{m}\sum_{i=1}^{m} x_i \quad (5)$$

2) calculate the variance of the mini-batch ($\mathcal{B}$)

$$\sigma_{\mathcal{B}}^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \ (6)$$

3) normalize the mini batch ($\mathcal{B}$)

$$\hat{x}_i = \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \varepsilon}} \quad (7)$$

4) scale and shift the mini-batch ($\mathcal{B}$)

$$y_i = \gamma \hat{x}_i + \beta = \mathcal{B}N_{\gamma,\beta}(x_i) \quad (8)$$

Where γ and β are learnable parameters and values of x over mini-batch $\mathcal{B}$, m is the size of a mini-batch.

The optimization function is used to reduce the loss function, the selected optimizer for this study is Adam, it is a technique for effective stochastic optimization that just needs first-order gradients and uses little memory for learning weights (Kingma and Ba, 2017).

2.3.3 Training and fine-tuning parameters

The updated models' newly added layers were developed using the dates dataset. The models are tested against a verification set after every training epoch with batch size 32 images and dropout 0.3. The number of neurons at the output layer was adjusted to two for this research to study a binary classification problem for the accepted and rejected classes. The dataset was trained with hypermeters as an initial epoch 15 and a learning rate of 0.001. In addition to improving the feature extraction for our dataset after training, there were additional fine-tuned layers. Fine tuning is a method to retrain some layers with different hypermeters for more feature extractions, to enhance the evaluating matrices results for the selected pre-trained models. Models were fine-tuned with final epochs 25 and a learning rate of 0.0001.

2.3.4 Evaluation matrices

To get the optimal classifier, evaluation metrics used in Deep Learning tasks are essential (M and M.N, 2015). They were used in two key stages of training and testing of a typical data categorization process. It is used in the training phase to improve the classification algorithm. This indicates that the evaluation measure is applied to identify and choose the optimal option (Alzubaidi et al., 2021). The following evaluation metrics were used to evaluate our models.

Confusion matrix: A visual illustration of an algorithm's performance in statistical classification or machine learning is the confusion matrix or error matrix. The columns serve as ground truths or real examples, whereas the rows serve as projected cases (Darwish, 2020).

Accuracy: The percentage of accurate predictions to all occurrences assessed is what the accuracy metric calculates.

$$Accuracy = \frac{Number\ of\ correct\ prediction}{Total\ number\ of\ prediction} = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

Precision: The percentage of accurately predicted positive patterns in a positive class is measured by precision.

$$Precision = \frac{particular\ class\ predicted\ correctly}{all\ class\ predictions} = \frac{TP}{TP + FP} \quad (10)$$

Recall: The percentage of positive patterns that are correctly categorized is measured by recall.

$$Recall = \frac{Correctly\ Predicted\ Class}{All\ Real\ Classes} = \frac{TP}{TP + FN} \quad (11)$$

The symbols refer to (TP) as True Positives or samples that are correctly classified in the class. (FP) refers to the False Positive or samples that are incorrectly classified in the class. (TN) determined as the True Negative or samples that are correctly rejected from the class. Finally (FN) is defined as a False Negative or samples that are incorrectly rejected from the class.

RESULTS AND DISCUSSION

3.1 Classification and learning curve results.

Training and verification of our dataset with the three architectures for period 15 epochs. Fig. (7) showed that results accuracy did not exceed (90%, 92%, and 85%) for Inception-v3, Inception-ResNet-v2, and VGG 19 respectively, and the loss results ranged from 20-50% for select models. to improve the models' performance, increase the number of epochs while fine-tuning with different hypermeters, and it will make a significant difference in the accuracy and the loss results due to the iteration.

As noticed from Table (2) the pre-trained models were evaluated with the accuracy, loss, recall, precision, and confusion matrix. The test accuracy% was (96.96%, 98.99%, and 95.13%) for Inception-v3, Inception-ResNet-v2, and VGG 19, respectively. Besides, loss values were (0.0665, 0.0344, and 0.2115) for Inception-v3, Inception-ResNet-v2 and VGG 19, respectively. On the other hand, the precision% results were (99.62%, 95.28%, and 100%) for Inception-v3, Inception-ResNet-v2 and VGG 19, respectively. Finally, recall% results were (93.95%, 98.5%, and 89.67%) for Inception-v3, Inception-ResNet-v2 and VGG 19, respectively.

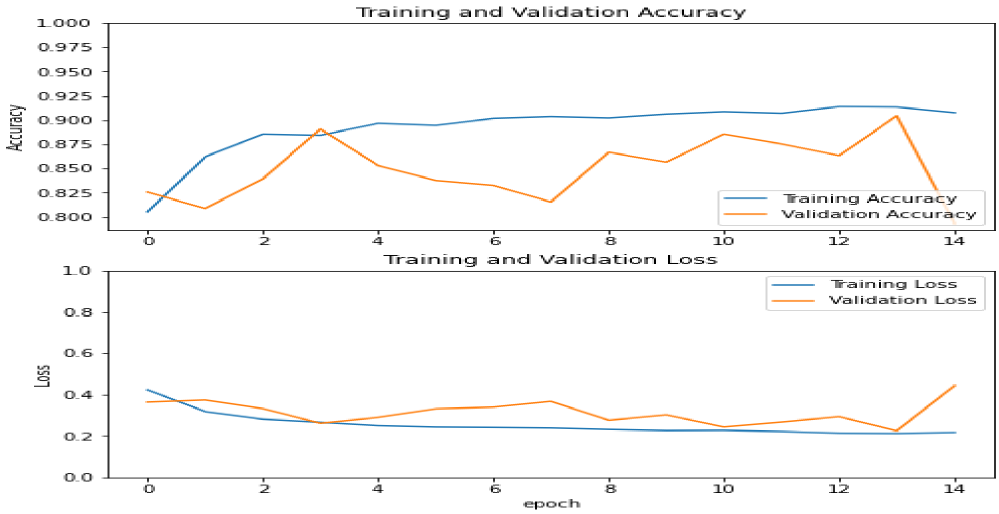
It was observed from Table (2) and Fig. (8) that the best performance was for Inception-ResNet-v2, by using the selected hypermeters. Also, as noticed from Fig. (8) after fine-tuning our data, the performance of models was improved according to the iteration of the learning steps. The graph showed that the gap between train and verification was best fitted in Inception-ResNet-v2, in addition to the stability of train and learning features it was found that the best performance also is for the Inception-ResNet-v2.

3.2 Confusion matrix results

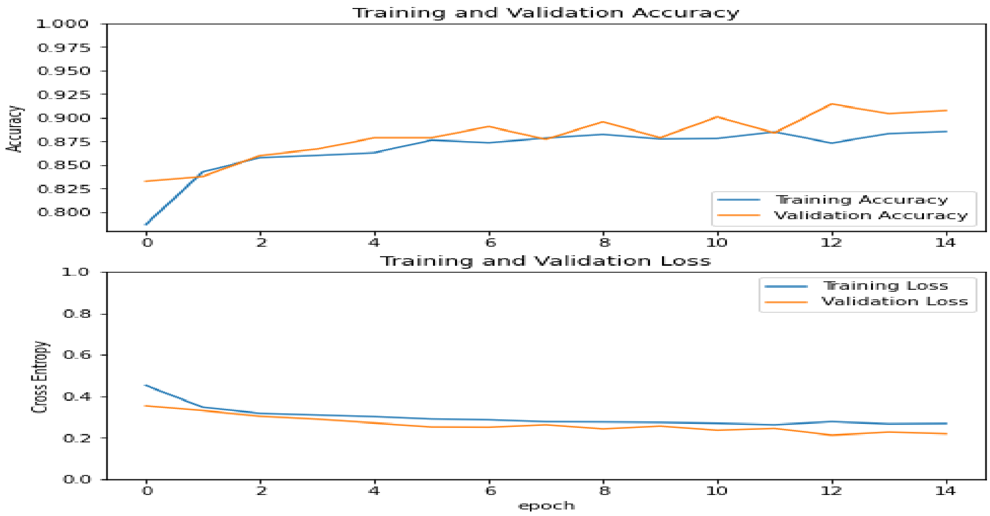
On the other hand, by evaluating our test dataset with a confusion matrix for 569 images, we found that Inception-v3 architecture determined 314 images as accepted and 264 as rejected with misclassified 18 images in their corrected category. But in the determination of the test dataset, Inception-ResNet-v2 classified (313, 277) correctly for accepted and rejected respectively, and only 6 images misclassified. Finally, VGG 19 architecture was the highest misclassification for the dataset with 29 images that didn't recognize it and (315, 252) classified correctly for accepted and rejected respectively. As shown in Figure 9.

3.3 Performance analysis

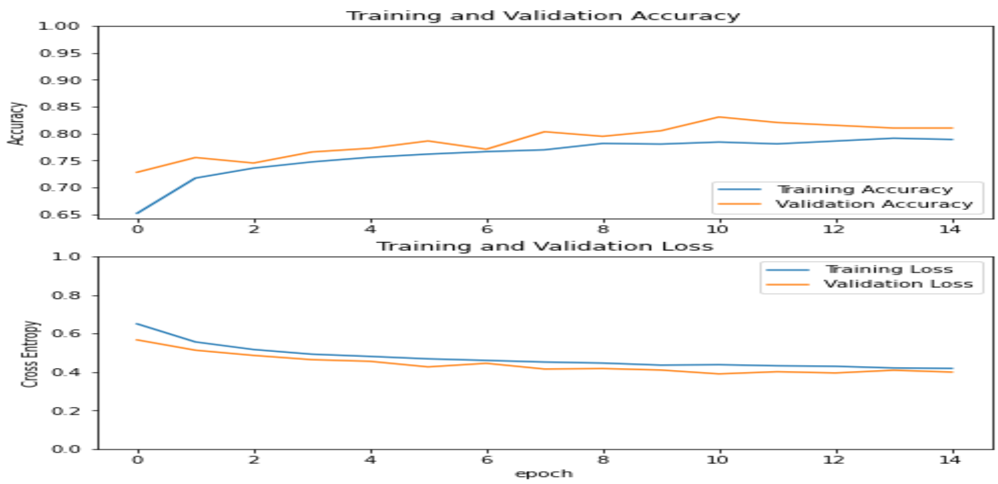
In line with this study, different studies used several CNN architectures to classify dates. Various deep-learning architectures have been used to sort and grade dates according to different features like ripping stage, size, color, health state, etc. Researchers (Pérez-Pérez et al., 2021) used CNN architectures to classify the Mejdool dates according to their ripping stage the evaluated architectures were VGG-16, VGG-19, ResNet-50, ResNet-101, ResNet-152, AlexNet, Inception-v3, and CNN from scratch. The best performance is obtained from VGG-19 architecture with an accuracy of 99.32%.



(a) Inception-v3

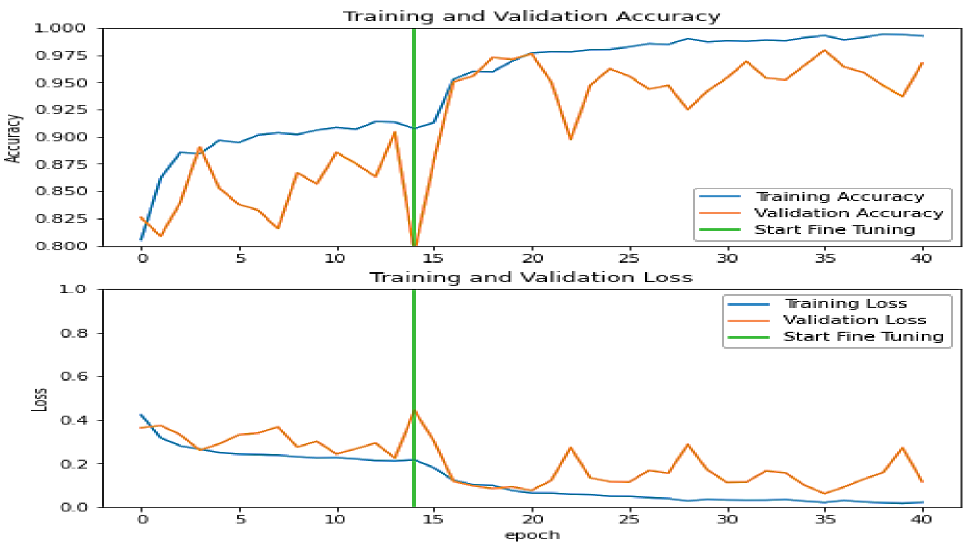


(b) Inception-ResNet-v2

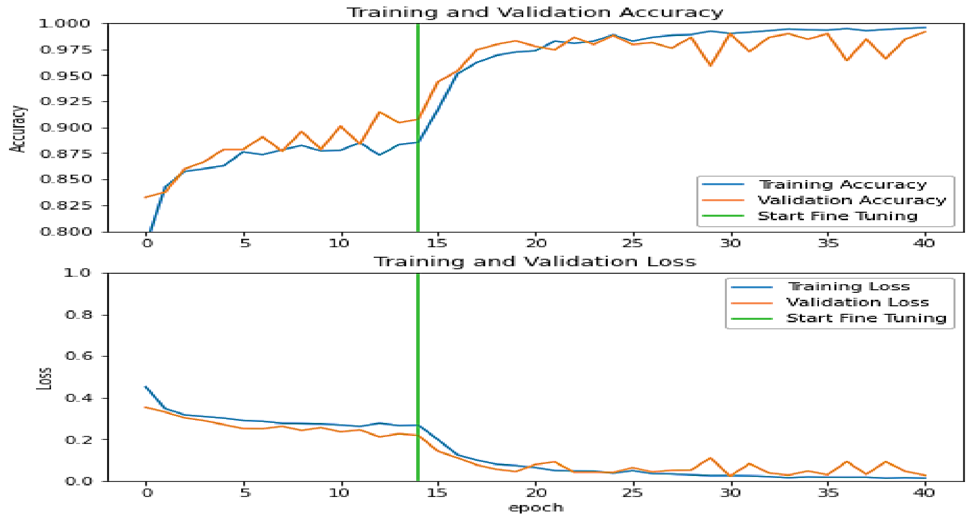


(C) VGG 19

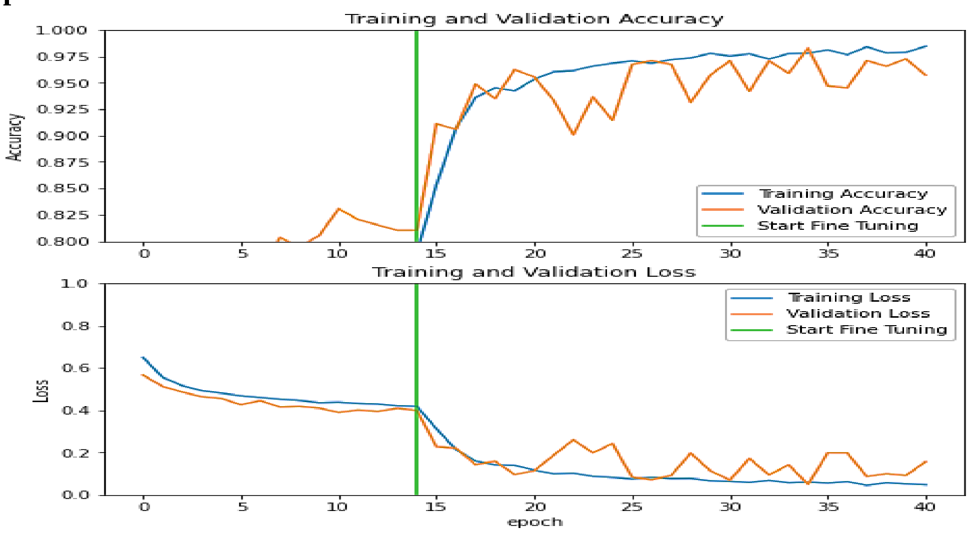
Fig. (7): The accuracy and loss results plot comparison between (a) Inception-v3, (b) Inception-ResNet-v2, and (c) VGG 19 for training and verification of the dataset before fine tuning layers.



(a) Inception-v3



(b) Inception-ResNet-v2



(c) VGG 19

Fig. (8): The accuracy and loss results plot comparison between (a) Inception-v3, (b) Inception-ResNet-v2, and (c) VGG 19 for training and verification of the dataset before and after fine tuning layers

Table 2. Results for the test evaluation of the pre-trained models for the date dataset classification

Evaluation Matrices	Machine learning models		
	Inception-v3	Inception-ResNet-v2	VGG19
Loss	0.0665	0.0344	0.2115
Accuracy%	96.96	98.99	95.13
Precision%	99.62	95.28	100
Recall%	93.95	98.5	89.67

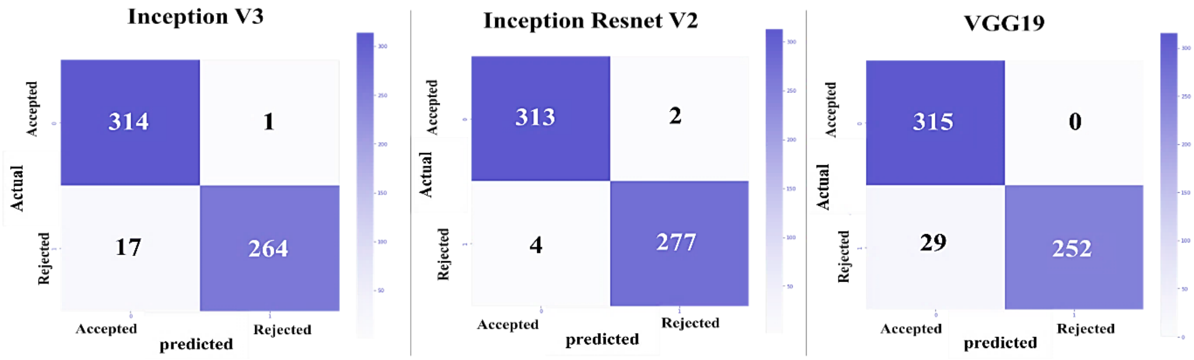


Fig. (9): Confusion matrix comparison for the pre-trained models (a) Inception-v3, (b) Inception-ResNet-v2, and (c) VGG 19 for the date dataset quality classification

Researchers (Nasiri et al., 2019) designed a CNN model that is constructed from VGG-16 architecture to classify dates according to maturity stage and if it is rejected, the CNN model was able to achieve an overall classification accuracy of 96.98%. Researchers (khayer et al., 2021) compare the following pre-trained models to classify dates according to the variety MobileNet-v1, MobileNet-v2, Inception-v1, Inception-v2, Inception-v3, Inception-ResNet-v2, and ResNet-v1. The highest performance is for Mobilenet-v1 with an accuracy of 82.67%. In this study, three CNN pre-trained models were selected to compare the accepted and rejected dates. Our purpose of this classification is to improve the sorting process to avoid having a rejected fruit in the final product package, which will ruin all the packages and cause losses. After reviewing other methods for classifying dates, it was found that there was a gap in this field compared to our study. Will summarize some of them: 1- Few studies classified dates according to their health state, as it is still done in traditional ways so there is a need to take a step in this field, to improve the quality of the final product, and decrease losses during the handling process. 2- Usage of the large dataset will be a good indicator for getting a higher performance, building our dataset because of the lack of rejected open-source datasets to make a balance between accepted and rejected datasets.

CONCLUSIONS

This study applied transfer learning as pre-trained models to recognize the different features between accepted dates and rejected dates, as it is one of the handling date stages. This classifying process is for improving sorting dates, so it will lead to getting high-quality products to decrease the gap between production and exportation. The date samples were collected from different regions in Egypt. The rejected dates collected from the factory rejected samples according to Codex Standard 143-1985.

Three pre-trained models (Inception-ResNet-v2, Inception-v3, and VGG 19) were chosen to train three date varieties (Mejdool, Saeidi, and El-Wadi) datasets. To enhance the performance of models a dataset was built in a controlled environment by an Arduino Automatic mobile camera shutter controller. The total number of images was 1484 accepted and 2803 images rejected. In addition to an open source (Kaggle) dataset, 1658 images were added to be accepted to balance the data. Among the results, it showed that the high-performance result for classifying dates is the Inception-ResNet-v2 model with the highest accuracy of 98.99% and lowest loss of 0.0344 at the test stage.

Acknowledgments: The authors wish to express their appreciation to Rasha Ahmed Zinhoum, Associate Professor at Plant Protection Research Institute, Agriculture Research Center, Giza, Egypt for her efforts in classifying date samples for the practical experiment.

Conflicts of Interest: The authors declare no conflicts of interest and no competing interests.

REFERENCES

Adedeji, A.A., Ekramirad, N., Rady, A., Hamidisepehr, A., Donohue, K.D., Villanueva, R.T., Parrish, C.A.,and Li, M., 2020. Non-Destructive Technologies for Detecting Insect Infestation in Fruits and Vegetables under Postharvest Conditions: A Critical Review. Foods 9, 927. https://doi.org/10.3390/foods9070927.

Alhamdan, W.S.N., and Howe, J.M., 2021. Classification of Date Fruits in a Controlled Environment Using Convolutional Neural Networks, in: Hassanien, A.-E., Chang, K.-C., Mincong, T. (Eds.), Advanced Machine Learning Technologies and Applications, Advances in Intelligent Systems and Computing. Springer International Publishing, Cham, pp. 154–163. https://doi.org/10.1007/978-3-030-69717-4_16.

Alsirhani, A., Siddiqi, M.H., Mostafa, A.M., Ezz, M., and Mahmoud, A.A., 2023. A Novel Classification Model of Date Fruit Dataset Using Deep Transfer Learning. Electronics 12, 665. https://doi.org/10.3390/electronics12030665.

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., and Farhan, L., 2021. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53. https://doi.org/10.1186/s40537-021-00444-8.

Azadnia, R., Fouladi, S., and Jahanbakhshi, A., 2023. Intelligent detection and waste control of hawthorn fruit based on ripening level using machine vision system and deep learning techniques. Results in Engineering 17, 100891. https://doi.org/10.1016/j.rineng.2023.100891.

Codex, 1985. CODEX STANDARD FOR DATES. FOOD AND AGRICULTURAL ORGANIZATION OF THE UNITED NATIONS, Codex Alimentarius 5.

Darwish, M., 2020. FRUIT CLASSIFICATION USING COVOLUTIONAL NEURAL NETWORK 67.

Dr.A, .Usha Ruby, 2020. Binary cross entropy with deep learning technique for Image classification. IJATCSE 9, 5393–5397. https://doi.org/10.30534/ijatcse/2020/175942020.

FAO, 2022. FAO Statistics. FOOD AND AGRICULTURAL ORGANIZATION OF THE UNITED NATIONS.

FAO, 2018. Strategy for developing the palm and palm trees sector in Egypt. FOOD AND AGRICULTURAL ORGANIZATION OF THE UNITED NATIONS 86.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T., 2018. Recent advances in convolutional neural networks. Pattern Recognition 77, 354–377. https://doi.org/10.1016/j.patcog.2017.10.013.

Hassan, S.M., Maji, A.K., Jasiński, M., Leonowicz, Z., and Jasińska, E., 2021. Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach. Electronics 10, 1388. https://doi.org/10.3390/electronics10121388.

khayer, Md.A., Hasan, Md.S., and Sattar, A., 2021. Arabian Date Classification using CNN Algorithm with Various Pre-Trained Models, in: 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV). Presented at the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), IEEE, Tirunelveli, India, pp. 1431–1436. https://doi.org/10.1109/ICICV50876.2021.9388413.

Kingma, D.P., and Ba, J., 2017. Adam: A Method for Stochastic Optimization.

Krizhevsky, A., Sutskever, I., and Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM 60, 84–90. https://doi.org/10.1145/3065386.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J., 2022. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Trans. Neural Netw. Learning Syst. 33, 6999–7019. https://doi.org/10.1109/TNNLS.2021.3084827.

M, H., M.N, S., 2015. A Review on Evaluation Metrics for Data Classification Evaluations. IJDKP 5, 01–11. https://doi.org/10.5121/ijdkp.2015.5201.

Mohanty, S.P., Hughes, D.P., and Salathé, M., 2016. Using Deep Learning for Image-Based Plant Disease Detection. Front. Plant Sci. 7, 1419

. https://doi.org/10.3389/fpls.2016.01419.

Nasiri, A., Taheri-Garavand, A., and Zhang, Y.-D., 2019. Image-based deep learning automated sorting of date fruit. Postharvest Biology and Technology 153, 133–141. https://doi.org/10.1016/j.postharvbio.2019.04.003.

Nturambirwe, J.F.I., and Opara, U.L., 2020. Machine learning applications to non-destructive defect detection in horticultural products. Biosystems Engineering 189, 60–83. https://doi.org/10.1016/j.biosystemseng.2019.11.011.

Pérez-Pérez, B.D., García Vázquez, J.P., and Salomón-Torres, R., 2021. Evaluation of Convolutional Neural Networks' Hyperparameters with Transfer Learning to Determine Sorting of Ripe Medjool Dates. Agriculture 11, 115. https://doi.org/10.3390/agriculture11020115.

Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O.R., and Jagersand, M., 2020. U^2-Net: Going Deeper with Nested U-Structure for Salient Object Detection. Pattern Recognition 106, 107404. https://doi.org/10.1016/j.patcog.2020.107404.

Saeed, A., Abdel-Aziz, A.A., Mossad, A., Abdelhamid, M.A., Alkhaled, A.Y., and Mayhoub, M., 2023. Smart Detection of Tomato Leaf Diseases Using Transfer Learning-Based Convolutional Neural Networks. Agriculture 13, 139. https://doi.org/10.3390/agriculture13010139.

Sarraf, M., Jemni, M., Kahramanoğlu, I., Artés, F., Shahkoomahally, S., Namsi, A., Ihtisham, M., Brestic, M., Mohammadi, M., and Rastogi, A., 2021. Commercial techniques for preserving date palm (Phoenix dactylifera) fruit quality and safety: A review. Saudi Journal of Biological Sciences 28, 4408–4420. https://doi.org/10.1016/j.sjbs.2021.04.035.

Shorten, C., and Khoshgoftaar, T.M., 2019. A survey on Image Data Augmentation for Deep Learning. J Big Data 6, 60. https://doi.org/10.1186/s40537-019-0197-0.

Simonyan, K., and Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z., 2015a. Rethinking the Inception Architecture for Computer Vision.

Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., 2015b. Going deeper with convolutions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Boston, MA, USA, pp. 1–9. https://doi.org/10.1109/CVPR.2015.7298594.

Tran, T.-T., Choi, J.-W., Le, T.-T., and Kim, J.-W., 2019. A Comparative Study of Deep CNN in Forecasting and Classifying the Macronutrient Deficiencies on Development of Tomato Plant. Applied Sciences 9, 1601. https://doi.org/10.3390/app9081601.

Verma, S., Chug, A., and Singh, A.P., 2020. Application of convolutional neural networks for evaluation of disease severity in tomato plant. Journal of Discrete Mathematical Sciences and Cryptography 23, 273–282. https://doi.org/10.1080/09720529.2020.1721890.

Wang, C., Liu, B., Liu, L., Zhu, Y., Hou, J., Liu, P., and Li, X., 2021. A review of deep learning used in the hyperspectral image analysis for agriculture. Artif Intell Rev 54, 5205–5253. https://doi.org/10.1007/s10462-021-10018-y.

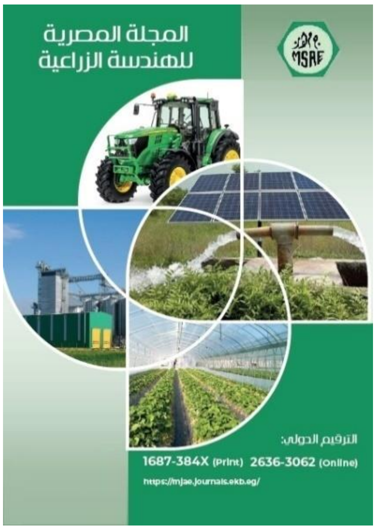
Yang, L., Wang, S.-H., and Zhang, Y.-D., 2022. EDNC: Ensemble Deep Neural Network for COVID-19 Recognition. Tomography 8, 869–890. https://doi.org/10.3390/tomography8020071.

Yu, F., Lu, T., and Xue, C., 2023. Deep Learning-Based Intelligent Apple Variety Classification System and Model Interpretability Analysis. Foods 12, 885. https://doi.org/10.3390/foods12040885.

تصنيف جودة التمور باستخدام تقنية التعلم العميق المبنية على الصور الملتقطة

وليد كامل محمد سالم الحلو[١]، داليا أبو بكر [٢]، سحر عبده زيان [٣] محمد احمد ميهوب[٤]

[١]أستاذ مساعد - قسم الهندسة الزراعية - كلية الزراعة - جامعة عين شمس - القليوبية – مصر.
[٢]طالب دراسات عليا - قسم الهندسة الزراعية - كلية الزراعة - جامعة عين شمس - القليوبية – مصر.
[٣]رئيس بحوث بمعهد بحوث امراض النبات - مركز البحوث الزراعية - الجيزة – مصر.
[٤]مدرس – قسم الهندسة الزراعية – كلية الزراعة – جامعة عين شمس – القليوبية – مصر.



الكلمات المفتاحية:
تقنيات الذكاء الاصطناعي، الشبكات العصبية الاصطناعية الالتفافية، نقل التعلم، تداول ثمار التمر، الثمار المرفوضة.

الملخص العربي

الهدف الرئيسي لهذه الدراسة هو استخدام أسلوب التعلم العميق من خلال الشبكات العصبية CNN للحصول على أعلى أداء نموذجي لتصنيف حالة التمور من ناحية قبوله من رفضه، وتحقيق منتج نهائي عالي الجودة، وذلك للحد من الفجوة بين الكمية المصدرة مقارنة بالإنتاج.

ومن ثم استخدمت بعض نماذج الشبكات العصبية التلافيفية CNN مثل "Inception-v3"، "Inception-ResNet-v2"، "VGG19" للوصول للنموذج الأمثل بعد تدريبه على بيانات التمور المقبولة من المرفوضة التي تم الحصول عليها من مصدرين هما: ١- عن طريق عمل آلية تصوير تعتمد على الأردوينو وربطها بموبايل للحصول على الصور بشكل تلقائي، ٢- قاعدة بيانات مفتوحة المصدر (Kaggle dataset) مضافة إلى الصور السليمة من البيانات الملتقطة وذلك لتقليل التباين بين قاعدة البيانات للصور المقبولة والمرفوضة. بإجمالي عدد ٥٩٤٥ صورة منهم ٣١٤٢ صورة مقبولة، و٢٨٠٣ صورة مرفوضة. واستخدمت هذه النماذج السابق ذكرها لمعرفة ايهم أكثر دقة لعملية تصنيف التمور للوصول لمنتج نهائي يرضي ذوق العملاء.

وأوضحت النتائج ان أفضل أداء تم تسجيله هو Inception-ResNet-v2 بدقة Accuracy مقدارها ٩٨,٩٩٪ وفقد Loss مقداره ٠,٠٣٤٤. ونتيجة لذلك، يمكن استنتاج أن استخدام نموذج Inception-ResNet-v2 لتطوير نظام رؤية حاسوبي مناسب يساعد في تعزيز عملية فرز التمور ويؤدي إلى منتج من التمور ذو جودة عالية.