MFTs-Net: A Deep Learning Approach for High Similarity Date Fruit Recognition

Abdellah El Zaar [1,*], Rachida Assawab [1], Ayoub Aoulalay [2], Nabil Benaya [1], Toufik Bakir [3], Smain Femmam [4,5], and Abderrahim El Allati [1]

[1] Laboratory of Research and Development in Engineering Sciences (RDSI), Faculty of Sciences and Techniques Al-Hoceima, Abdelmalek Essaadi University, Tetouan, Morocco; Email: Assawabrachida@gmail.com (R.A.), nabil.benaya@gmail.com (N.B.), abdou.allati@gmail.com (A.E.A.)
[2] Power Management Integrated Circuits (PMIC) Laboratory, Faculty of Sciences and Techniques Al-Hoceima, Abdelmalek Essaadi University, Tetouan, Morocco; Email: aoulalayayoub@gmail.com (A.A.)
[3] Le laboratoire Imagerie et Vision Artificielle (ImViA) Laboratory, University of Burgundy, Dijon, France; Email: toufik.bakir@u-bourgogne.fr (T.B.)
[4] Networks and Communications Department, Haute-Alsace University, Mulhouse, France; Email: smain.femmam@uha.fr (S.F.)
[5] Ecole Polytechnique Feminine (EPF) Engineering School, Paris-Cachan, France
*Correspondence: abdellah.elzaar@gmail.com (A.E.Z.)

***Abstract*—Artificial Intelligence and Deep Learning applications are well-developed as a part of human life. In the field of object recognition, Convolutional Neural Network (CNN) based methods are getting more and more important and challenging. However, existing CNN methods do not perform well on datasets that exhibit high similarities, resulting in confusion between different classes. In this study, we propose a new Deep Learning approach for recognizing date fruit categories based on the Deep Convolutional Neural Network (DCNN). The modified fine-tuning (MFTs-Net) approach can recognize with high accuracy the different date fruit categories. In order to train and to test the robustness of our proposed method, we have collected a dataset that takes into account different date fruit categories. The presented dataset is challenging as it contains classes of a unique object and presents high similarities concerning the shape, color and texture of date fruit. We show that the MFTs-Net CNN we implemented, trained and tested using the collected dataset can recognize with high accuracy the different date categories compared with state-of-the-arts works. The presented methodology works perfectly with very small datasets, which is one of the main strengths of the proposed method. Our MFTs-Net architecture performs perfectly on test data with an accuracy of 98%.**

***Keywords*—date fruit recognition, deep learning, hyper-parameter optimization, object recognition, convolutional neural network**

I. Introduction

Recently, the progress of Deep Learning algorithms in object recognition and computer vision research has been remarkable. This advancement can be attributed to the emergence of innovative techniques, which have resulted in more effective solutions to specific computer vision challenges. Machine Learning and Deep Learning techniques have been successfully deployed in diverse fields, ranging from Optical Character Recognition, Robotics, Internet of Things, Autonomous Vehicles, Medical Imaging, to Financial Time Series Forecasting, among others. Some noteworthy contributions in these domains include [1–6]. The field of object recognition and image processing using Deep Learning techniques is still facing several challenges. In real-world applications, the presence of noise and the risk of misclassifications are common, particularly when dealing with highly similar image data. As a result, the accuracy of the models decreases, which can affect their overall performance. Despite the increasing interest in Deep Learning models, such as Convolutional Neural Networks (CNNs), their robustness in handling very high similarity images requires further investigation.

In this work, we have introduced a new Deep Learning strategy based on CNN algorithm for date fruit category recognition. Date fruit category recognition faces a big challenge compared with other fruit categories recognition problems. The high similarity of the date fruit categories makes the recognition process difficult and presents a high risk of mis-classification.

Dates are the fruit of the date palm tree, which is grown in many tropical regions of the world. Dates have become quite popular in recent years. Depending on the variety, fresh dates are fairly small in size and range in color from bright red to bright yellow. Dates are chewy with a sweet flavor. They are also high in some important nutrients and have a variety of advantages and uses. For this reason, we investigated in CNN to deal with high similarity of date categories and to perform an accurate recognition system. Several works and surveys on image classification and object recognition have been recently published, Garcia *et al.* [7], Druzhkov *et al.* [8],

Manuscript received April 15, 2023; revised May 20, 2023; accepted June 30, 2023; published November 3, 2023.

doi: 10.12720/jait.14.6.1151-1158

Zhang *et al.* [9], Zaidi *et al.* [10], Kumar *et al.* [11], Hasegawa *et al.* [12]. On the other hand, prior research in this area depended on conventional Machine Learning and statistical classifiers, such as Support Vector Machines, Artificial Neural Networks, Naive Bayes, and Nearest Neighbors. For instance, Karayaneva *et al.* [13] work was based on these techniques.

Earlier methods face several problems when working with noisy and high scale databases. However, Deep Learning techniques show great performances when working with large and grid-structured data. The work done by Jiao *et al*. [14] explores the power of the new generation of Deep Learning in image processing. The CNN algorithm in [15] is one of the most successful Deep Learning algorithms. This powerful algorithm outperformed all other methods in the field of object recognition and computer vision tasks. This can be explained by the strong discriminative abilities of CNNs and their capacity to learn data dependencies. The main problem with this powerful tool is that it provides poor performance when working with small datasets. Some studies suggest working with transfer-learning strategy and data augmentation to solve this problem. The transfer-learning technique refers to transferring the knowledge of an already trained CNN model with a large sample dataset to the main model trained with a small dataset. This method gives better results compared with CNN learned from scratch strategy. Enlarging the number of sample datasets can also be a good solution to deal with small dataset problems, but it does not provide effective results.

In our case study, we utilized a small dataset consisting of date categories for the recognition process. This presents an additional challenge, alongside the high similarity in texture among the dates. To effectively solve the problem of high similarity and the problem of small datasets using CNN without using data augmentation, we design a new strategy based on a fine-tuning technique. This learning strategy finds the optimized hyper-parameters of a CNN-learned from scratch strategy and uses these parameters to train the fine-tuned CNN architecture. We call this new approach MFTs-Net. This method is low cost and provides effective results. Moreover, it improves the model's behavior without over-fitting problems. Indeed, the under-fitting and over-fitting in Deep Learning models are usually caused by the small amount of data. A training process requires a large dataset to accurately learn the dependencies between classes. When using CNN, data augmentation is a common technique to deal with the overfitting phenomenon [16]. We prove through experiments and results that this new technique can provide excellent performance without overfitting and without data enhancement in the case of all given challenges.

The rest of this paper is organized as follows: Section II presents recent works in the field of date fruit categories recognition using Deep Learning. Section III describes the proposed MFTs-Net strategy. Experiments and achieved results are discussed in Section IV. Finally, conclusion and perspectives are presented in Section V.

II. Related Work

Dates are characterized by the high similarities of shape and color, which makes the classification and recognition tasks challenging. The state-of-the-art research conducted in this area implements both Deep Learning and Machine Learning algorithms to deal with the recognition task. The majority of these artworks are based on color and texture recognition. For instance, Jana *et al.* [17] introduced a system based on image preprocessing and texture feature extraction from the gray level matrix in addition to color features from the segmented images. For the classification, they implemented the well-known Support Vector Machine algorithm. Muresan *et al*. [18] used CNN algorithm to recognize 131 fruit categories. Their approach was trained and tested using a very large dataset of 90380 images. The obtained results are satisfactory, with an accuracy of 95.25%. Chung *et al.* [19] employed CNN algorithm and Efficient Net to identify fruit categories. They have taken 17,624 images from 25 different categories. Their proposed model achieved an accuracy of 95%. Behera *et al*. [20] proposed a classification model for 40 kinds of Indian fruits by a Support Vector Machine classifier combined with CNN. They trained and tested their approach using a wide dataset of 40 categories and 23,848 images and achieved an accuracy of 97.3%.

For date fruit recognition, Magsi *et al.* [21] used different feature extraction techniques and the CNN to recognize 22 types of dates, their method shows an accuracy of 97% on training-set and 89% on test-set. Alresheedi *et al*. [22] implemented different Machine Learning and Deep Learning algorithms to recognize nine date fruit categories. They achieved an accuracy of 93% when using Deep Learning methods and 92% when using other Machine Learning based methods. Aiadi *et al.* [23] proposed a method for automatically recognizing different date varieties based on outlier detection techniques and Gaussian Mixture. They used a dataset with 11 date varieties with a recognition rate of 97.8%.

Machine Learning and Deep Learning techniques have proven to be effective for fruit recognition in general. However, traditional methods based on Machine Learning algorithms may be time-consuming. These approaches are based on three main steps: data preprocessing, feature extraction and classification. Data preprocessing is considered as an important step in the recognition process. It consists of cleaning, normalizing and splitting the data before feeding it to the model. This step is very important when working with methods based on Machine Learning techniques. Algorithms based on Machine Learning may not provide satisfactory results when working with noisy datasets [24]. In our case, and to the best of our knowledge, we implement a new method based on fine-tuning the CNN as illustrated in Fig. 1. We compared our proposed method with three other CNN strategies trained and tested using our collected dataset. Our proposed method can work with small datasets and provide excellent results without a preprocessing step.

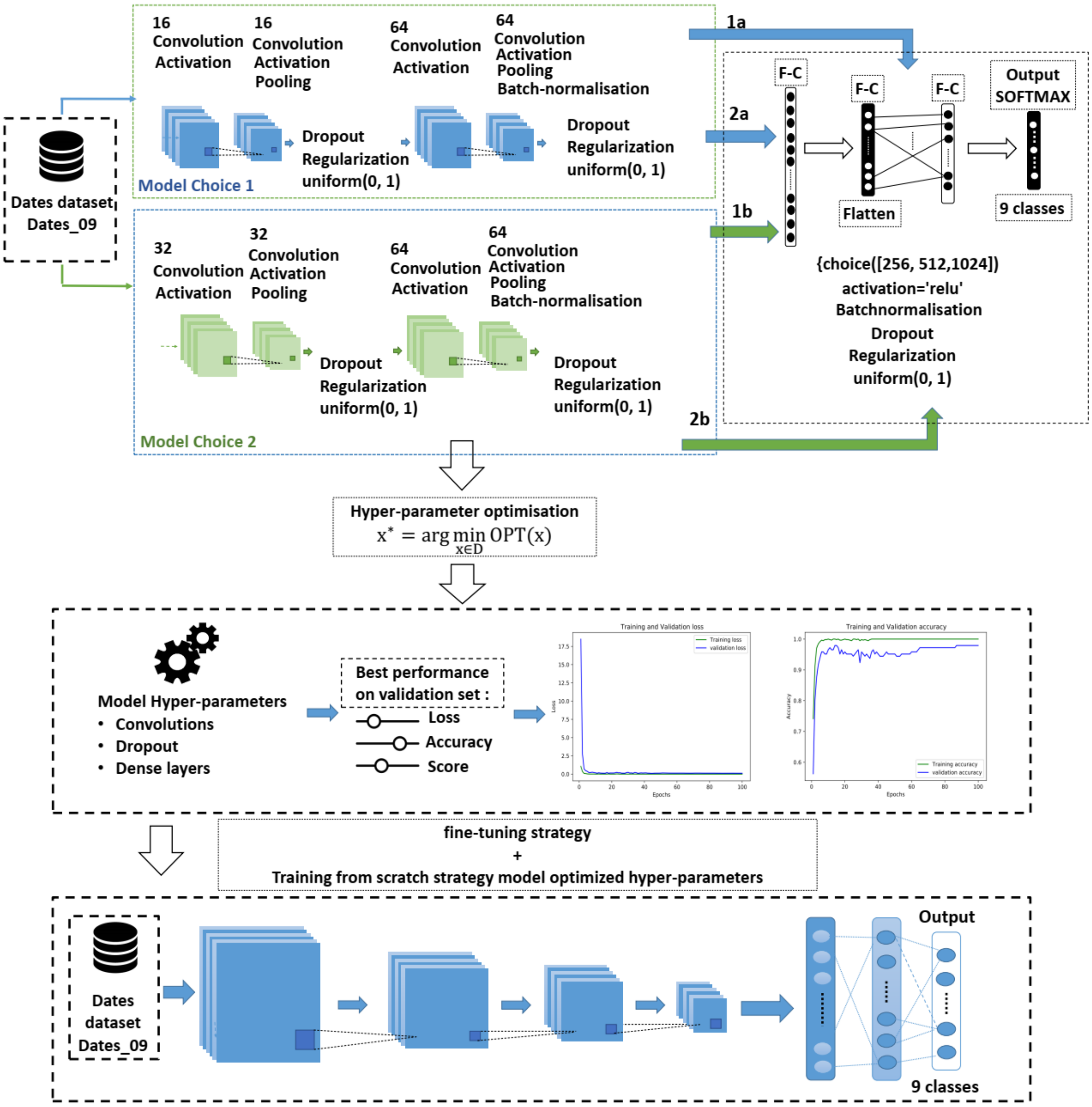


Figure 1. System overview.

III. Proposed Approach

The proposed approach is used to recognize nine categories of date, as shown schematically in Fig. 1. The collected dataset is used to train the proposed CNN model, which is implemented in three strategies. In the following, we describe in detail our proposed method MFTs-Net based on fine-tuning the CNN using the optimized hyper-parameters of the CNN trained from scratch strategy.

A. CNN Architectures

In CNN algorithm, the data is fed from the input to the output through three main layers: convolutional layer, pooling layer and fully connected layer. The convolutional operation from the n^{th} layer to the n^{th}+1 is defined as follows:

$$M_{ijz}^{n+1} = \sum_{r=1}^{K_n}\sum_{s=1}^{K_n}\sum_{k=1}^{d_n} w_{rsk}^{(z,n)}\, h_{i+r-1,j+s-1,k}^{(n)} \quad (1)$$

During the training process, the parameters are organized into sets of 3-dimensional structural units, known as filters or kernels. The dimensions of the filter in the nth layer are $K_n \times K_n \times d_n$. The depth of the resulting output of the convolutional layer depends on the number of filters and not on the dimensions of the input layer or filter. The z^{th} filter in the n^{th} layer has parameters denoted by the 3-dimensional tensor $\left[w_{r,sk}^{(z,n)}\right]$, the indices *i*, *j*, *k* indicate the positions along the height, width, and depth of the filter. The feature maps in the nth layer are represented by the 3-dimensional tensor $\left[h_{r,j,k}^{(n)}\right]$.

The difference between CNN and traditional neural network is that the latter has a fully connected architecture, and provides less performance than CNN when working with grid-structured data. In this paper, we train and test three CNN architectures: CNN learned from scratch, CNN combined with Support Vector Machine classifier and transfer learning technique. The strategy CNN-SVM combines the key characteristics of the two algorithms; CNN is used as a feature extractor and SVM is used for classification. The output of the last pooling layer of CNN is fed to the SVM classifier. Transfer learning is based on transferring the knowledge and the learned features from a source model trained on a source dataset to another target model with a target dataset and task.

In terms of accuracy, the transfer learning strategy can provide better performance than CNN learned from scratch and combined CNN-SVM strategies when working with small datasets. However, CNN learned from scratch is less complex and lower cost compared with the other strategies. The main problem with all CNN strategies is that the model can be overfitted in the case of high similarity images, especially the shape and color similarities. This problem can be explained by the similar pixel values in classes of images containing the same color, shape and texture. In this case, CNN models cannot learn perfectly dependencies and presents a high number of misclassified images. To deal with this problem when using date fruit categories, we trained and tested three CNN strategies, and we found that the new MFTs-Net architecture can learn perfectly dependencies in case of high similarities when working with date fruit category dataset.

B. MFTs-Net

Our new proposed method aims to use the optimized hyper-parameters of a CNN based on VGGnet architecture trained from scratch with the collected date fruit dataset to train a new fine-tuning model. We used VGGnet configuration to train the CNN from scratch regarding its capacity to provide better performance. This architecture design with small filters can better learn the image dependencies in our case of date fruit images. Fig. 1 presents an overview of our proposed architecture. The new MFTs-Net architecture differs from the simple fine-tuning strategy. Indeed, the main difference between these two strategies is that MFTs-Net uses the optimized hyper-parameters (number of convolutions, filters, pooling layers, dropout, and dense layers) of a CNN-trained-from-scratch with the same dataset (collected dataset) to train the new model, while the simple fine-tuning consists of taking the weights of pre-trained CNN model using a different image dataset (usually ImageNet dataset) and using them to train a new model. The simple fine-tuning strategy presents lower performances while MFTs-Net strategy works perfectly with all given challenges of the dataset (see section IV).

In the first block, illustrated in Fig. 1, the system aims to found the hyper-parameters of the CNN model that returns the best performance using the validation set. This block is trained using the collected dates dataset. The hyper-parameter optimization returns the best CNN model architecture and parameters that provides lower error rate and better accuracy. Hyper-parameter optimization can be modeled as follows:

$$h_p^* = argmin_{h \in d} f(h) \quad (2)$$

where h_p^* represents the hyper-parameters (Number of convolutions, filters, dropout, pooling and dense layers), that produce the best validation set performance, *f(h)* is the function of the error and loss to minimize. The hyper-parameters optimization block returns the number of convolutions, dropout proportion and configuration of the dense layers that provide the best model performance for the CNN trained-from-scratch strategy. The obtained configurations are then used to build a new model architecture trained and tested using the same date dataset. The MFTs-Net is built using the transferred optimized parameters of a source model trained from scratch using the same dataset to a target model fine-tuned target model. We compared our new proposed approach with the three learning strategies illustrated in Fig. 2, and it showed strong performances. To the best of our knowledge, our work is the first work that uses this new fine-tuning technique combined with CNN hyper-parameter configurations to train small and challenging datasets.

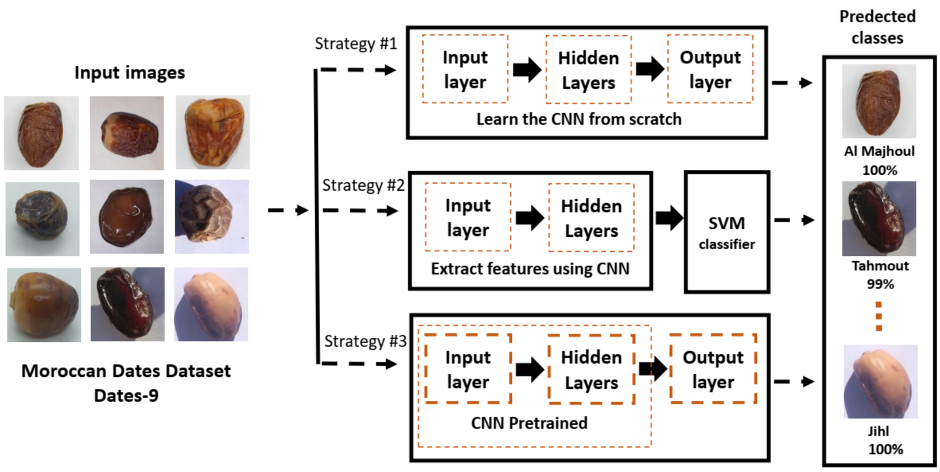


Figure 2. CNN strategies trained and tested on date fruit dataset.

C. Dates Dataset

To train and to test our proposed method, we have collected a new date fruit dataset. This dataset contains nine categories of the most popular date fruits. The collection of the dataset was manually done using different angle positions. This type of fruit is characterized by a similar shape, texture, and color, as shown in Fig. 3, which makes the recognition process challenging. The high similarity among these fruits can pose algorithm confusion and misclassification issues. To address this challenge, we employ our proposed approach to enhance the accuracy and reliability of the fruit recognition process.

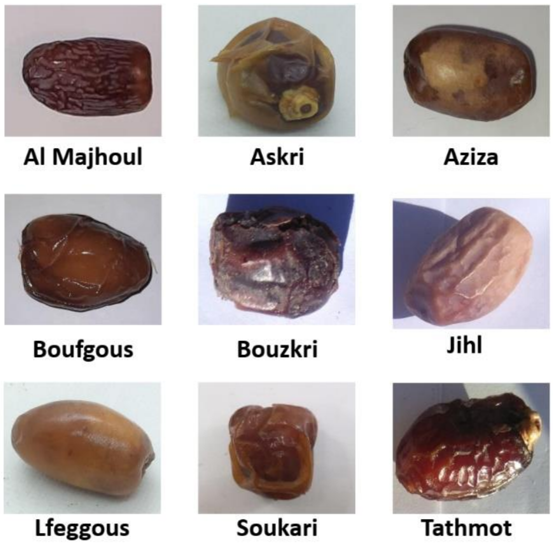


Figure 3. Date fruit categories.

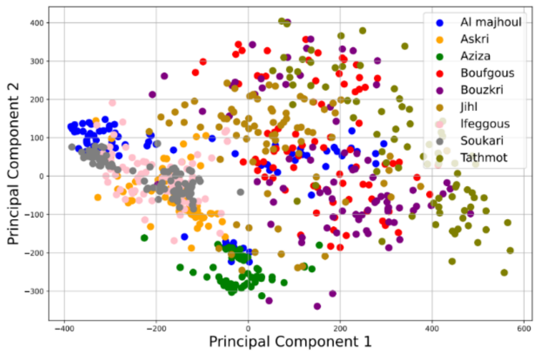
Dates fruit dataset contains 9 classes of images with 900 images, split into 2 sets: train-set with 720 images 80 images per class), test-set with 180 images (20 images per class). Fig. 4 illustrates the visualization of different categories of the collected dataset using the Principal Component Analysis (PCA) algorithm.

As it can be seen in Fig. 4, 80% of the images represents the train-set and 20% represents the test-set. This dataset division is usually made to compare model performance on train-set and test-set and to check that the final model works correctly.

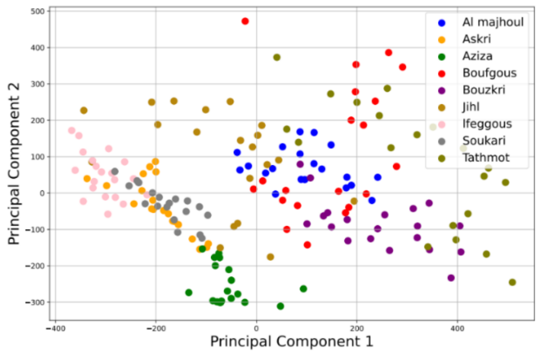
IV. Experiments and Results

In this section, we will discuss the experiments and the achieved results using the MFTs-Net method. We will also analyze the accuracy during the training process using our collected dataset and compare it with test-set accuracy. The method was coded and implemented using Python language. We have implemented several libraries for data visualization and image processing functions, such as Tensorflow, Keras models and Sklearn. We run our models on a Microsoft Azure virtual machine with a six-core processor. Working with CNN needs performing a high number of epochs and also visualizing a large amount of data that can be a time-consuming process. For this reason, we use TESLA K80 NVIDIA GPU. This accelerator gives superior performance to our model and accelerates libraries during the training process. The proposed MFTs-Net performs well without noise compared to other methods and provides strong performance during the training process. With 100 epochs, the system provides high accuracy and effective results using the collected date dataset. From Fig. 5 and Fig. 6, it can be concluded that our proposed system is less susceptible to the overfitting problem compared to CNN trained from scratch and the combined CNN-SVM strategies. In addition to that, the modified fine-tuning strategy MFTs-Net provides a less complex architecture with a training loss of 0.02%. We used the training loss as a metric to assess how our approaches fit the training data. Computationally, the training loss is calculated by taking the total number of errors for each example in the training set.

It can be seen from Fig. 5 that the trained model is overfitted when using the CNN trained-from-scratch and the combined CNN-SVM strategies. This can be explained by the high similarities in the collected data, the thing that made the CNN unable to learn dependencies in the case of date fruit data. This phenomenon is solved when using our new MFTs-Net that uses the learned weights of a CNN trained from scratch on the same data to build a new fine-tuning model.

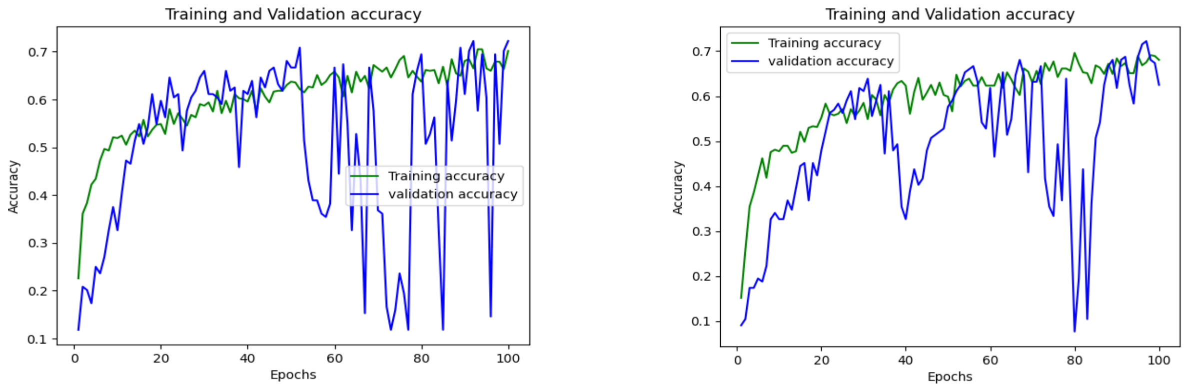


(a)



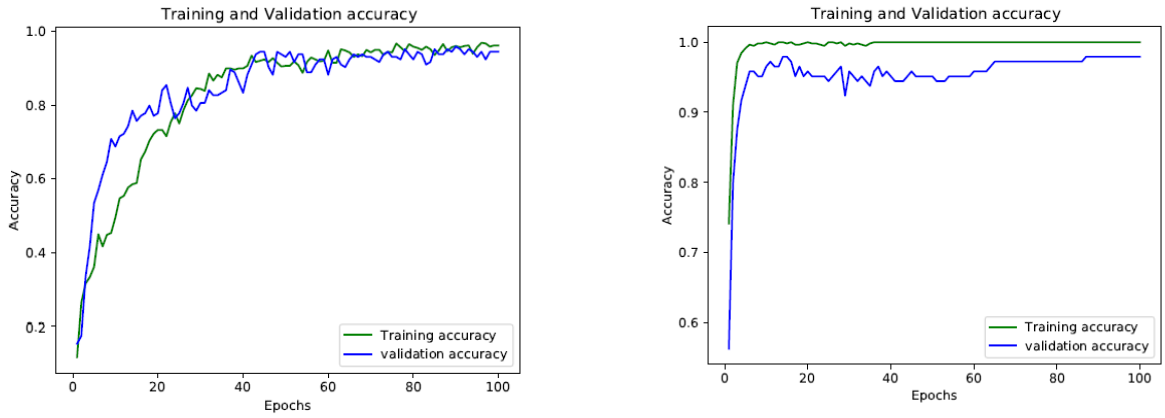
(b)

Figure 4. Dates fruits dataset: (a) train-set, (b) test-set.



(a) (b)

Figure 5. Model accuracy: (a) CNN from scratch; (b) CNN-SVM.



(a) (b)

Figure 6. MFTs-Net vs normal fine-tuning: (a) CNN normal fine-tuned; (b) MFTs-Net.

Compared with normal fine-tuning strategy, MFTs-Net shows strong performance during the training process without overfitting problems as shown in Fig. 6. MFTs-Net approach can also provide satisfactory results during the test step. This proves the high capacity of this method to deal with small datasets and to keep the same behavior with unseen data images. As presented in Table I, when using MFTs-Net we can hold an accuracy of 97% without any preprocessing step of the collected data. In other words, the CNN-SVM strategy achieves an accuracy of 90% compared with 87% for CNN learned from scratch. The preprocessing step refers to the operation of resizing and cleaning the collected dataset, when using Machine Learning algorithms, the preprocessing step can increase the model performance during the training step.

TABLE I. MODEL PERFORMANCE IN TERMS OF ACCURACY USING THE PROPOSED APPROACH

Training strategies	With preprocessing	Without preprocessing
MFTs-Net (our proposed method)	**98%**	**97%**
Normal fine-tuning	95%	93%
Hybrid CNN-SVM	92%	90%
CNN trained from scratch	89%	87%

Tables II and III show the sensitivity of the normal fine-tuning and MFTs-Net strategies using different learning rates and a variety of the number of the fully connected layers. The normal fine-tuning strategy was used as a baseline and the accuracy metrics are calculated to compare the effectiveness of the proposed approaches using training data. In general, both the normal fine-tuning strategy and MFTs-Net provides high accuracy during the training step compared with CNN from scratch and CNN-SVM. When analyzing the normal fine-tuned CNN accuracy illustrated in Table II, it can be seen that the learning strategy only provides high accuracy only for learning rate greater than or equal to 0.001 regardless of the number of fully connected layers. As indicated in Table III, MFTs-Net strategy shows high accuracy for learning rate greater than or equal to 0.01. CNNs are very sensitive to the learning rate value. Large value of learning rate may result in unstable training process as is the case when using both strategies with a learning rate equal to 0.1. Whereas too small a value can lead to a long but effective training process. Learning rate refers to the step size that the weights are updated during training. It is a configurable hyper-parameter often in the range between 0.01 and 1.0.

TABLE II. ACCURACY OF NORMAL FINE-TUNING STRATEGY ON TRAINING-DATA

FC layers	learning rate	epochs		
		100	150	200
1	0.1	0.5020	0.5031	0.5046
	0.01	0.5172	0.5215	0.5211
	0.001	**0.9484**	**0.9498**	**0.9513**
	0.0001	0.9391	0.9343	0.9359
2	0.1	0.5212	0.5215	0.5180
	0.01	0.5277	0.5278	0.5125
	0.001	**0.9434**	**0.9479**	**0.9498**
	0.0001	0.9384	0.9376	0.9359
3	0.1	0.5023	0.5040	0.5048
	0.01	0.5171	0.5215	0.5217
	0.001	0.9251	0.9380	0.9399
	0.0001	**0.9452**	**0.9510**	**0.95**

TABLE III. ACCURACY OF MFTS-NET STRATEGY ON TRAINING-DATA

FC layers	learning rate	epochs		
		100	150	200
1	0.1	0.4974	0.5047	0.5145
	0.01	0.8037	0.8362	0.8586
	0.001	**0.9674**	**0.9737**	**0.9799**
	0.0001	0.8822	0.9180	0.9391
2	0.1	0.5009	0.5131	0.5079
	0.01	0.8192	0.8589	0.8845
	0.001	**0.9772**	**0.9730**	**0.9794**
	0.0001	0.8815	0.9203	0.9318
3	0.1	0.5125	0.5093	0.5102
	0.01	0.7475	0.7487	0.7484
	0.001	**0.9645**	**0.9742**	**0.9704**
	0.0001	0.8768	0.9220	0.9271

To evaluate and test our model architectures, we used a test data that was not used for training and validation. Figs. 7 and 8 show the confusion matrices of the proposed approaches.

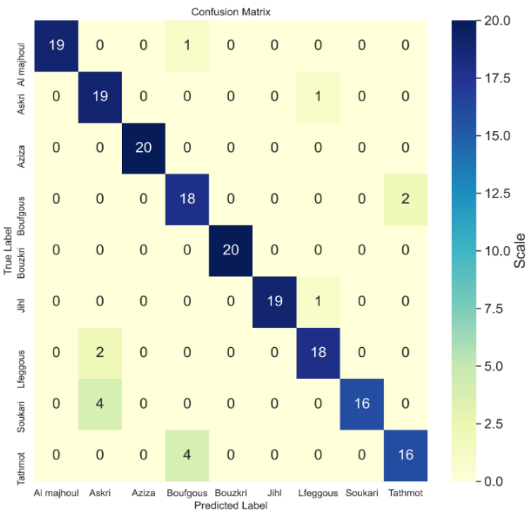
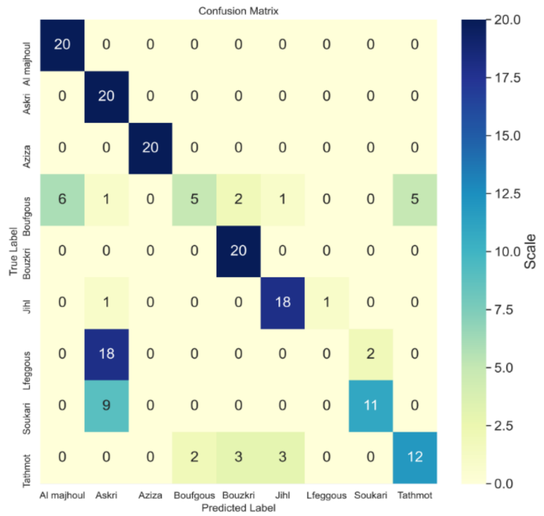
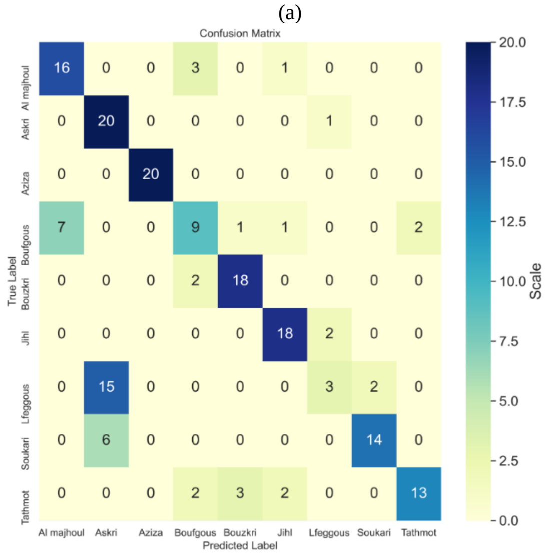


Figure 7. MFTs-Net confusion matrix.

The presented confusion matrices evaluate the performance of our model using the proposed approaches. The diagonal of each confusion matrix provides the number of well-predicted samples using the test-set. The maximum number in the confusion matrix diagonal represents the number of images in each class of the test-set. Fig. 7 illustrates the confusion matrix using MFTs-Net. It can be seen that the category (Boufgous) was confused 4 times with (Tathmot), similarly to (Askri) which was also 4 times confused with (Soukari). This can be explained by the very high similarity between these date categories. In general, the model performs perfectly using the test-set images. That is to say, the combined CNN-SVM (Fig. 8(a)) and CNN-trained-from-scratch (Fig. 8(b)) strategies provide high levels of confusion, especially between the (Askri, Soukari and lfeggous) date categories. This happens most often with CNN based methods when using small datasets. This phenomenon can also be explained by the inability of CNN to learn different dependencies in the case of high similarity of colors, shapes and textures.



(a)



(b)

Figure 8. CNN from scratch and CNN-SVM confusion matrices: (a) CNN-SVM strategy confusion matrix; (b) CNN learned from scratch strategy confusion matrix.

V. CONCLUSION

In this paper, we have introduced a new method based on a CNN algorithm to recognize date categories. Our proposed contribution gives an efficient solution to deal with small datasets using CNN. The proposed approach is based on a fine-tuning strategy and shows strong results without noise and overfitting problems. The MFTs-Net outperforms other CNN based strategies and performs well on the collected date dataset. MFTs-Net can learn easily dependencies in case of high similarities of shape, color, and texture which is proven by effective performances achieved during the training process. Compared with the normal fine-tuning strategy, MFTs-Net learning strategy demonstrates its insensitivity to different learning rate values and provides high accuracy on test-set without overfitting problems. Our proposed MFTs-Net method proves its efficiency with test images and show great results. We conclude that the MFTs-Net approach can bring an efficient solution to the problem of Deep Learning models that require small datasets. As part of our future work, we aim to explore the effectiveness of the MFTs-Net strategy for recognizing additional types of date fruits. Additionally, we plan to deploy the trained model in real-time applications, taking into account both model complexity and time consumption. We believe that these efforts will help to further enhance the practical application of our research and address any potential limitations of our current approach.

Conflict of Interest

We declare that we have no Conflict of Interest.

Author Contributions

A. El Zaar conducted the research, analyzed the data, and wrote the paper. N. Benaya, R. Assawab, and A. Aoulalay collected and preprocessed the data. T. Bakir, S. Femmam, and A. El Allati analyzed and improved the proposed approach and reviewed the paper. All authors had approved the final version.

References

[1] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, 2023.

[2] Y. Bai, B. Zhang, N. Xu, J. Zhou, J. Shi, and Z. Diao, "Vision-based navigation and guidance for agricultural autonomous vehicles and robots: A review," *Computers and Electronics in Agriculture*, vol. 205, 107584, 2023.

[3] T. W. Cenggoro, B. Pardamean *et al.*, "A systematic literature review of machine learning application in covid-19 medical image classification," *Procedia Computer Science*, vol. 216, pp. 749–756, 2023.

[4] R. P. Masini, M. C. Medeiros, and E. F. Mendes, "Machine learning advances for time series forecasting," *Journal of Economic Surveys*, vol. 37, no. 1, pp. 76–111, 2023.

[5] O. Liouane, S. Femmam, T. Bakir, and A. B. Abdelali, "Improved two hidden layers extreme learning machines for node localization in range free wireless sensor networks," *J. Commun.*, vol. 16, no. 12, pp. 528–534, 2021.

[6] O. Liouane, S. Femmam, T. Bakir, and A. B. Abdelali, "Novel DV-hop algorithm-based machines learning technics for node localization in rang-free wireless sensor networks," *International Journal of Informatics and Communication Technology (IJ-ICT)*, 2023.

[7] W. Garcia, C. Mera, L. Santana, and L. Prom "Algorithm for the recognition of a silhouette of a person from an image," *J. Image Graph*, vol. 7, no. 2, pp. 59–63, 2019.

[8] P. N. Druzhkov and V. D. Kustikova, "A survey of deep learning methods and software tools for image classification and object detection," *Pattern Recognition and Image Analysis*, vol. 26, no. 1, pp. 9–15, 2016.

[9] S. Zhang, Y. Wu, and J. Chang, "Survey of image recognition algorithms," in *Proc. 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, 2020, vol. 1, pp. 542–548.

[10] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, 103514, 2022.

[11] K. K. Kumar and H. Venkateswerareddy, "A detailed survey on deep learning techniques for real-time image classification, recognition and analysis," in *Expert Clouds and Applications*, Springer, 2022, pp. 349–360,

[12] R. Hasegawa, Y. Iwamoto, and Y.-W. Chen, "Robust Japanese road sign detection and recognition in complex scenes using convolutional neural networks," *Journal of Image and Graphics*, vol. 8, no. 3, pp. 59–66, 2020.

[13] Y. Karayaneva and D. Hintea, "Object recognition in python and MNIST dataset modification and recognition with five machine learning classifiers," *Journal of Image and Graphics*, vol. 6, no. 1, pp. 10–20, 2018.

[14] L. Jiao and J. Zhao, "A survey on the new generation of deep learning in image processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019.

[15] C. Garcia and M. Delakis, "Convolutional face finder: A neural architecture for fast and robust face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1408–1423, 2004.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[17] S. Jana, S. Basak, and R. Parekh, "Automatic fruit recognition from natural images using color and texture features," in *Proc. 2017 Devices for Integrated Circuit (DevIC)*, IEEE, 2017, pp. 620–624.

[18] H. Muresan and M. Oltean, "Fruit recognition from images using deep learning," arXiv preprint, arXiv: 1712.00580, 2017.

[19] D. T. P. Chung and D. V. Tai, "A fruits recognition system based on a modern deep learning technique," *Journal of Physics: Conference Series*, vol. 1327, 012050, 2019.

[20] S. K. Behera, A. K. Rath, and P. K. Sethy, "Fruit recognition using support vector machine based on deep features," *Karbala International Journal of Modern Science*, vol. 6, no. 2, pp. 234–245, 2020.

[21] A. Magsi, J. A. Mahar, S. H. Danwar *et al*., "Date fruit recognition using feature extraction techniques and deep convolutional neural network," *Indian Journal of Science and Technology*, vol. 12, no. 32, pp. 1–12, 2019.

[22] K. M. Alresheedi, S. Aladhadh, R. U. Khan, and A. M. Qamar, "Dates fruit recognition: From classical fusion to deep learning," *Comput. Syst. Sci. Eng.*, vol. 40, no. 1, pp. 151–166, 2022.

[23] O. Aiadi, M. L. Kherfi, and B. Khaldi, "Automatic date fruit recognition using outlier detection techniques and gaussian mixture models," *EL-CVIA: Electronic Letters on Computer Vision and Image Analysis*, vol. 18, no. 1, pp. 51–75, 2019.

[24] D. J. Hemanth and V. V. Estrela, *Deep Learning for Image Processing Applications*, *Series of Advances in Parallel Computing, Vol. 31*, IOS Press, 2017.