IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 28 December 2022, accepted 15 January 2023, date of publication 20 January 2023, date of current version 25 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3238570

RESEARCH ARTICLE

A Novel Deep Learning-Based Hybrid Method for the Determination of Productivity of Agricultural Products: Apple Case Study

FATIH BAL[1] AND FATIH KAYAALP[2]
[1]Department of Software Engineering, Kirklareli University, 39010 Kirklareli, Turkey
[2]Department of Computer Engineering, Duzce University, 81620 Duzce, Turkey
Corresponding author: Fatih Bal (fatihbal@klu.edu.tr)

ABSTRACT The production of agricultural products and the high yield in these products are of critical importance for the continuation of human life. In recent years, machine learning and deep learning technologies have been widely used in determining agricultural productivity. The purpose of this study was to estimate the yield of apple fruit by using a novel deep learning-based hybrid method. First, by using images belonging to the *golden* and *royal gala* apple varieties, a classification was made with the help of a convolutional neural network (CNN) that was designed for the study. Then, using classical machine learning algorithms and bagging and boosting algorithms, a hybrid application was performed by classifying the images whose feature extractions were done with the designed CNN. The results of the study, presented on 4 separate datasets (Datasets A, B, C, and D), were evaluated based on accuracy, precision, recall, F-measure, and Cohen kappa scores. Considering the accuracy results for Datasets B, C, and D, it was determined that the hybrid model that gave the best result was the CNN-SVM model. For Dataset A, the CNN-SVM and CNN-Gradient Boosting hybrid models gave the best and same accuracy. Dataset C was determined as the most appropriate dataset in terms of the more balanced distribution of train, test, and validation size in the datasets, the results of the proposed hybrid CNN model, and the evaluation of the results of the model. For Dataset C, it was found that the accuracy of the hybrid model was 99.70%. Precision, recall, f-measure, and Cohen kappa scores were 99%. The results of the study revealed that the hybrid models showed effective results in determining the productivity of apple fruit through images belonging to the golden and royal gala varieties.

INDEX TERMS Apple yield prediction, deep learning, ensemble methods, machine learning, smart farm.

I. INTRODUCTION

Agriculture is the process of producing food, fiber, and other products through the cultivation of certain plants. After the production process, product efficiency gains great importance, and it also plays a very important role in the economy of countries. Although there is an impact of global warming, there are decreases in fruit and vegetable production and yields due to the lack of knowledge of farmers and incorrect irrigation methods. Though the World Health Organization (WHO) says that people should consume an average of 400 grams of fruits and vegetables per day, this does not seem possible. Fruit and vegetable yields show a decrease in many countries. In order to prevent this yield decrease, important agricultural countries of the world have been developing smart agricultural systems to ensure obtaining even greater yields by preventing yield reduction in fruit and vegetable production caused by farmers, to carry out water use more consciously, and to identify diseases and wild-harmful-poisonous herbs in plants. In the yield and disease determination models developed with smart agricultural systems, image data and metadata of them are used. Inferences are made by training these data with statistical results and models. According to FAO data, apple, which makes up 10% of the

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Ayoub Khan.

world's fruit production, is the third most-produced fruit after bananas and watermelons [1]. World apple production was being estimated at 79.4 million tons in the 2019-2020 season, while this production was predicted to be 76.1 million tons in the 2020-2021 season by a decrease of about 4.1% [1].

This research was conducted to predict apple fruit yield. For the purpose of the research, a dataset consisting of images belonging to low-yielding and high-yielding *Royal Gala* and *Golden* varieties of apple fruit was used. Due to the fact that the dataset had a small number of images, GAN (Generative Adversarial Network) was used to increase this number instead of the classic data augmentation methods. Based on their weights, the images augmented by the GAN method were classified as High Yield Golden, Low Yield Golden, High Yield Royal Gala, and Low Yield Royal Gala. In the next step, four different datasets (Dataset A, Dataset B, Dataset C, and Dataset D) were created to observe the results of the classified images. First, the performance of the CNN architecture designed for this study was examined for each dataset. In the next stage, the image features were extracted by the designed CNN architecture. Finally, these features were classified by machine learning (ML), bagging, and boosting algorithms, and then a hybrid model was proposed.

This study contributes to the literature by predicting productivity in agriculture with a developed new deep-learning-based method. Different from other studies in the literature, the determination of the productivity of apples was carried out based on their weights obtained using apple fruit images. The small number of apple images was augmented with the GAN model. In addition, the designed CNN architecture was used instead of well-known pre-trained CNN models. In addition, instead of the designed hybrid models with only one ML algorithm, the most well-known ML algorithms were used to determine the hybrid model in this study. For the purpose of the study, performance metrics, cross-validation tests, and Cohen Kappa scores of the proposed hybrid model were calculated.

The rest of the paper is organized as follows. Section II presents the literature review of existing studies. Section III briefs about the dataset used, the data augmentation method, the proposed hybrid model, the CNN model, and the ML algorithms used in the architecture of the proposed hybrid model. Section III-A presents the experimental results and evaluations of the performance of the proposed hybrid model. The study is completed with the conclusion given in Section III-B.

II. LITERATURE REVIEW

Similar studies are reviewed in this part of the study. The performance evaluations of machine learning, bagging, and boosting algorithms used in conjunction with CNN networks were focused on.

In a study conducted by Jieng et al., a data set consisting of images of corn, lettuce, radish, as well as weeds growing at and around the bottom of these plants was studied for weed and crop classification in intelligent agriculture [2].

Koklu et al. classified the images, extracted with CNN, by using Graph CNN (G-CNN). In their study, the ResNet-101 architecture was used as the CNN architecture. Accuracy, precision, recall, and F1 score values were measured as 96.51%, 98.83%, 98.73%, and 97.18%, respectively. Using the image dataset of vine grape leaves, the characteristics were determined with a semi-trained MobilNetV2 CNN architecture, and these characteristics were classified by the SVM model [3].

In another study, the linear, quadratic, cubic and Gaussian kernel functions of the support vector machine algorithm were applied to the multi-class imbalanced dataset and the cubic kernel function gave the best results. Accuracy, specificity, precision, sensitivity, and F1 score values were measured as 97.60%, 99.40%, 97.62%, 97.60%, and 97.60, respectively. On images belonging to datasets including multiclass and unbalanced varieties (CIFAR-10, Fashion MNIST), classification was carried out using transfer learning. In order to improve the performance of the model, the images whose features were extracted with CNN were classified using the AdaBoost algorithm [4].

In the study conducted by Khanramaki et al., only the accuracy value of the pre-trained CNN models was examined in the detection of citrus pests. It was seen that success was achieved at a rate of 98.22%. To detect pests and fungi on citrus plant leaves, ensemble classification was proposed on AlexNet, VGG16, ResNet50, and InceptionResNetV2, which were semi-trained CNN models. The accuracy value of the proposed ensemble model was determined as 98.64% [5].

Thongsuwan et al. proposed a classification model including a CNN + XGBoost approach to solve classification problems. In the classification study carried out on many datasets belonging to the health field, the accuracy value of the proposed model was measured as 87.9% for the Annuran Calls dataset, 97.74% for Breast Cancer Wisconsin dataset, 93.7% for DrivFace dataset, and 96.4% for the Parkinsons dataset [6].

Ibrahim et al. proposed a WBA (weighted bat-inspired algorithm)-CNN model to classify poisonous and wild plant images. Accuracy, recall, precision, and f1 score values were all measured as 98% for the proposed model [7].

In another study conducted by Hao et al., the images belonging to the development process of the Gynura Bicolor plant were classified and the development stages of the plant were estimated. The images whose features were extracted using Gnet (global information) and Lnet (local information) were classified with CNN. Accuracy, precision, recall, and v2 score values for the proposed model were measured as 99.47%, 99.5%, 99.40%, and 99.5%, respectively [8].

A CSPNet (cross-stage partial network)-based model called MCF-Net based on a cross-level fusion strategy was proposed by Kong et al. for the recognition and classification of fine-grained agricultural crops in precision agriculture. In the feature extraction of the model, accuracy and f1 score values were 88.40% and 93.50% respectively when it is used with the semi-trained CNN model, while accuracy and f1 score values were measured as 90.60% and 96.20% when using CSPNet [9].

Using data on the daily water consumption of plants, Ferreira and Cunha proposed a CNN-LSTM model. They calculated the results of the proposed model for 4 separate stations. When looking at the results of the proposed model, it was found that the RMSE value for all stations was 88% [10].

Dongyao Jia et al. proposed a strong CNN-SVM model by combining the features extracted by the GLCM for the detection of cervical cancer cells with abstract attributes coming from the hidden layers of CNN. The accuracy, sensitivity, and specificity values for the proposed model were measured as 99.3%, 98.9%, and 99.4%, respectively [11].

A study was conducted by Balgetir et al. to detect MS attacks by using images belonging to the plantar pressure distribution of low-disability MS disease. The features obtained from the semi-trained CNN model were classified by ML algorithms. The accuracy values of the models were examined, and it was observed that the best-performing algorithm was the VGG19-SVM model with a success ratio of 89.23% [12].

Li and Liu proposed a CNN-RNN approach model for the analysis and classification of images of the hippocampus, a region of the brain, in Alzheimer's disease. Accuracy, sensitivity, and specificity values for the proposed method were measured as 89.1%, 89.6%, and 93.1%, respectively [13].

A TCN-DNN based power load estimation model was developed by Bian et al. using nonlinear multidimensional and time-series load data. The RMSE and MAE values of the proposed model were examined, and the results were measured as 26835.64 and 2.99, respectively [14].

In another study, by using images of corn hybrids, efficiency was estimated with machine learning algorithms (Decision Tree, Adaboost, GBM, Random Forest, Neural Network, and XGBoost). The RMSE values of the algorithms were examined [15].

As a summary of the literature, it has been reported that traditional methods such as zooming in/out, rotations to left/right have been used for image augmentation, trained or semi-trained models (GoogleNet, AlexNet etc.) have been used for training, single or ensemble machine learning or deep learning methods have been used, and various accuracy values around 93-94% have been reached in many studies.

In this current study, a hybrid model was developed to predict apple yield. Because the number of apple images was not sufficient, synthetic data was generated by using GAN. Instead of semi-trained CNN models, a new CNN architecture designed for this study was used. After these stages, the performance of the CNN model was examined first. Then, the data obtained by feature extraction with CNN were classified by the machine learning (decision tree, random forest, and support vector machine) and ensemble learning (bagging and boosting algorithms) methods. The prediction results of these methods were examined, and a hybrid model was developed for apple fruit yield prediction.

The novelty of this paper can be summarized as using a novel CNN architecture for feature extraction, using augmented images by using GAN as the dataset for experiments, experiments with popular machine learning classification methods in ensemble form and a novel deep learning based hybrid method for classification task together, and good performance results when compared to the literature with high accuracy values.

III. MATERIALS AND METHODS

This part of the study includes information about data set, data preprocessing, data augmentation method, and the proposed hybrid model.

A. DEFINITIONS OF APPLE IMAGES

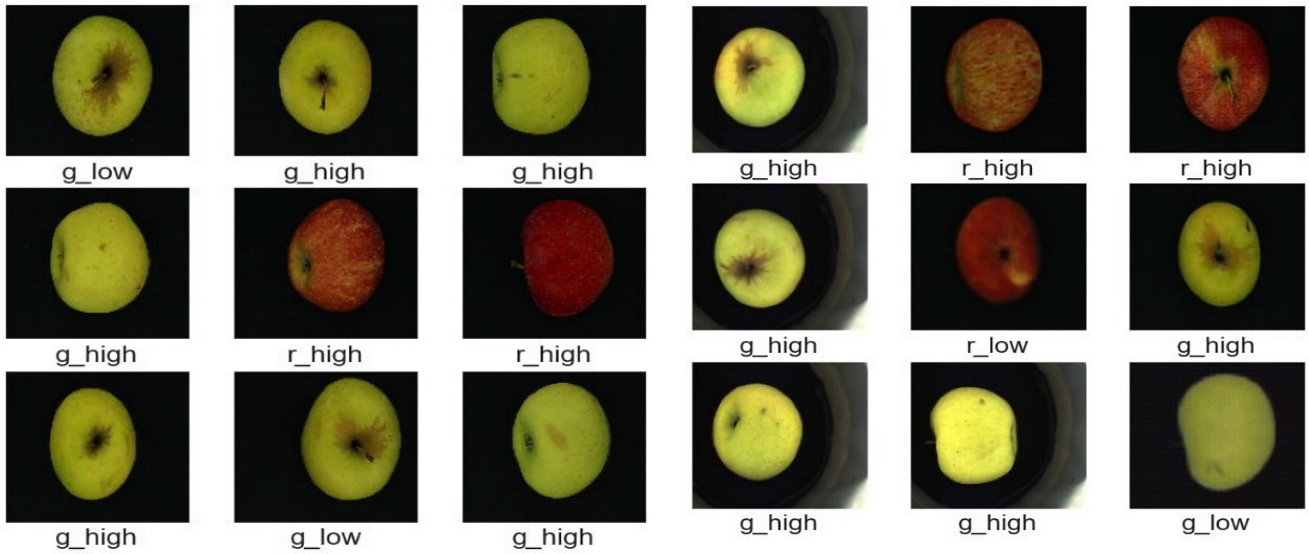
The data set of apple images used in this study was provided by COFILAB [16]. A data set was created from apple images taken at a 45-degree angle from a distance of 20 cm with a digital camera (EOS 550D, Canon Inc, Japan) in a laboratory environment. The images belonged to the *Golden* and *Royal Gala* apple varieties and consisted of 134 images brought together from multispectral and original images having a resolution of 1296 × 964.

B. PRE-PROCESSING IMAGES

134 images belonging to the apple varieties are not enough for the classification study of a model. It is clear that having a larger dataset will improve the performance of the model. Therefore, for this study, the number of images was augmented. To augment the number of images to be used in experiments, GAN (Generative Adversarial Networks) method, which is quite popular, was preferred instead of the data augmentation techniques created by classical methods. In classical methods, augmentation of the data is done by methods such as zooming in, zooming out, rotating, scaling, cutting, and Gaussian noise of the image with the help of the Tensorflow library. On the other hand, GAN models are considered as advanced data augmentation techniques. If the data desired to be augmented is given as input to the GAN model, the same type of data can be generated. For example, if apple images are given to the GAN model, apple images in the same shape are obtained.

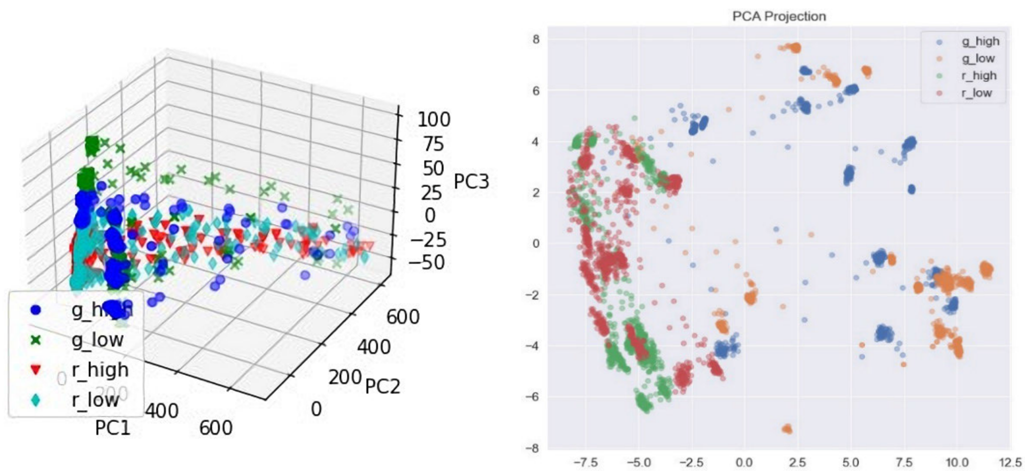
C. GENERATIVE ADVERSARIAL NETWORK (GAN) MODEL

In many studies, GAN models such as DCGAN (Deep Convolutional Generative Adversarial Networks), WGAN (Wasserstein Generative Adversarial Networks), StyleGAN (Style-based Generative Adversarial Networks), AC-GAN (Auxiliary Classifier Generative Adversarial Networks), and BigGAN (Generative Adversarial Networks) have been preferred. In this study, the Pix2Pix GAN model was used to create a synthetic image. Pix2Pix GAN is a model belonging to the CGAN (Conditional Generative Adversarial Networks) type. The reason for using Pix2Pix was to create a difference compared to other studies and to see the performance of Pix2Pix GAN in terms of synthetic data generation in smart



(a) (b)

FIGURE 1. Some apple images. (a) Real images, (b) Fake images generated with Pix-to-Pix GAN.



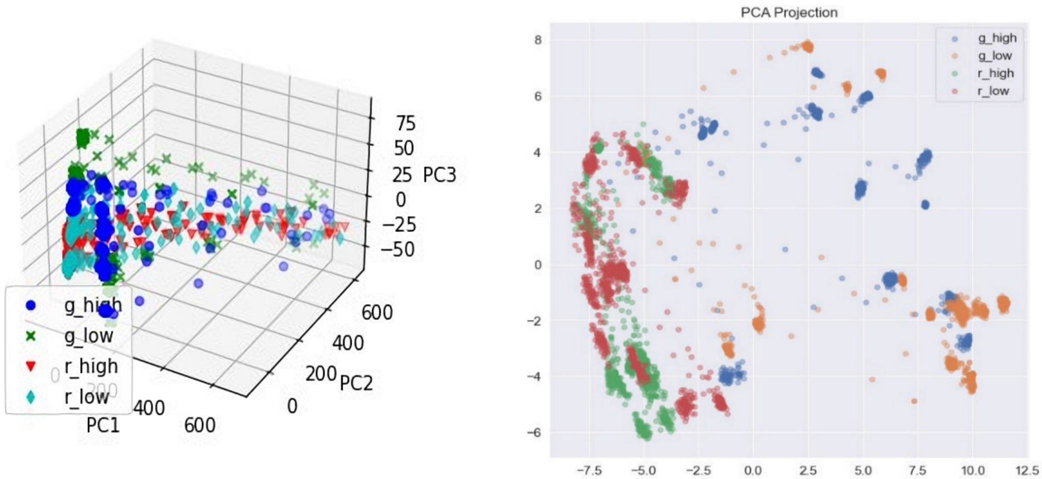
(a) (b)

FIGURE 2. Class distribution of Dataset A. (a) 3D class distribution, (b) 2D class distribution.

agriculture applications. Synthetic data were produced by the designed Pix2Pix model which combines multispectral and RGB images. Pix2Pix uses the U-Net architecture as the generator model. There is an encoder and a decoder in this architecture. The image is first compressed and encoded in a narrow pass (encoder). Then the encoded information is sent to the output layer (decoder), and it generates output based on the the characteristics of the information. Pix2Pix uses PatchGAN as the discriminator. PatchGAN checks whether the multispectral and color image is a valid conversion (Fake-Real).

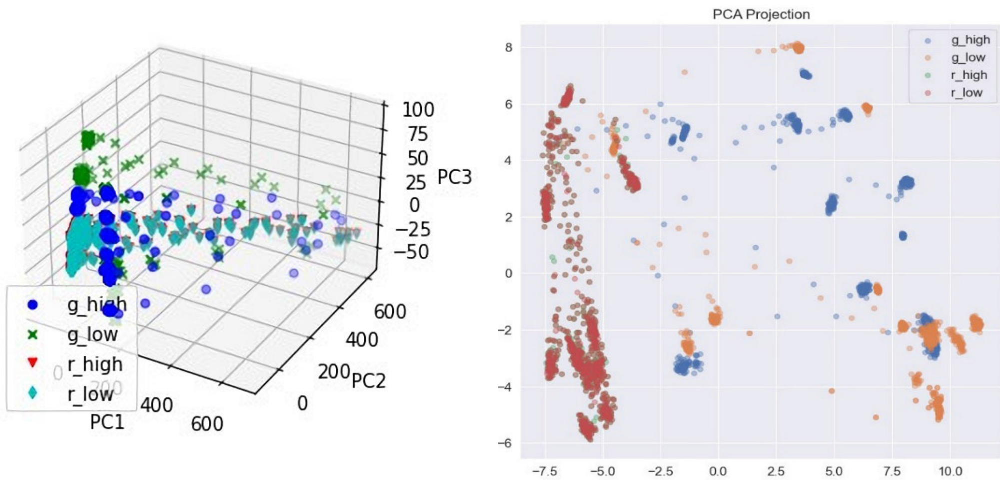
D. DATASETS

The original multispectral images and RGB images at a resolution of 1296 x 964 were resized to 256 x 256. The resized images were given as input to the Pix2Pix GAN model. The synthetic images produced with the Pix2Pix model consisted of a total of 6700 images with a resolution of 256 x 256. The image classes consisted of 4 (four) classes: high-yield golden apple-type images (*g_high*), low-yield golden apple-type images (*g_low*), high-yield royal gala apple-type images (*r_high*), and low-yield royal gala apple-type images (*r_low*). These images were divided into low and high-yield classes by taking into account the data of the company Nutritionix as a reference. For Golden Apple, the yield of 215 grams and above was considered low [17]. For Royal Gala Apple, the yield of 200 grams and above was considered low [18]. In order to compare and evaluate the results of the datasets, four different datasets were designed (Dataset A, Dataset B, Dataset C, and Dataset D). The data distribution of Dataset



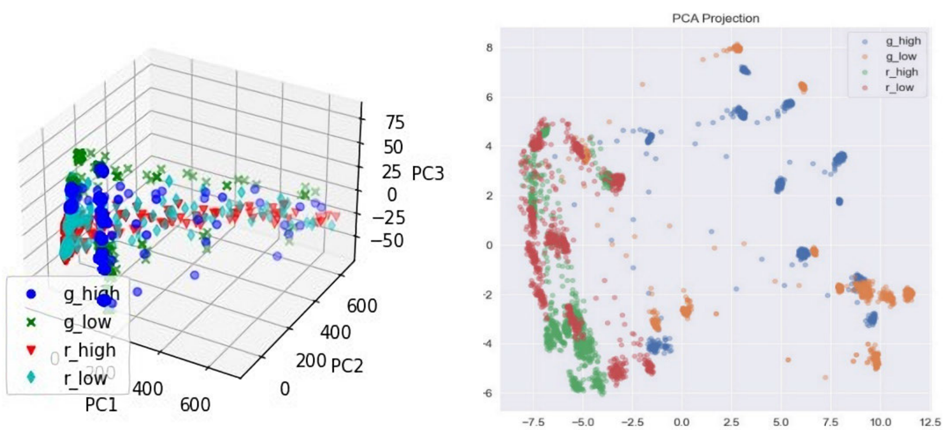
(a) (b)

FIGURE 3. Class distribution of Dataset B. (a) 3D class distribution, (b) 2D class distribution.



(a) (b)

FIGURE 4. Class distribution of Dataset C. (a) 3D class distribution, (b) 2D class distribution.



(a) (b)

FIGURE 5. Class distribution of Dataset D. (a) 3D class distribution, (b) 2D class distribution.

TABLE 1. **The numbers of images used for train, test, and validation in *Dataset A*.**

Classes of Image	Train	Validation	Test	Total
High Yield Golden Images	1024	256	320	1600
High Yield Royal Gala Images	1216	304	380	1900
Low Yield Golden Images	1024	256	320	1600
Low Yield Royal Gala Images	1024	256	320	1600
Total	**4288**	**1072**	**1340**	**6700**

TABLE 2. **The numbers of images used for train, test, and validation in *Dataset B*.**

Classes of Image	Train	Validation	Test	Total
High Yield Golden Images	960	240	400	1600
High Yield Royal Gala Images	1140	285	380	1900
Low Yield Golden Images	960	240	400	1600
Low Yield Royal Gala Images	960	240	400	1600
Total	**4020**	**1005**	**1675**	**6700**

TABLE 3. **The numbers of images used for train, test, and validation in *Dataset C*.**

Classes of Image	Train	Validation	Test	Total
High Yield Golden Images	896	224	480	1600
High Yield Royal Gala Images	1064	266	570	1900
Low Yield Golden Images	896	224	480	1600
Low Yield Royal Gala Images	896	224	480	1600
Total	**3752**	**938**	**2010**	**6700**

TABLE 4. **The numbers of images used for train, test, and validation in *Dataset D*.**

Classes of Image	Train	Validation	Test	Total
High Yield Golden Images	832	208	560	1600
High Yield Royal Gala Images	988	247	665	1900
Low Yield Golden Images	832	208	560	1600
Low Yield Royal Gala Images	832	208	560	1600
Total	**3484**	**871**	**2345**	**6700**

A was determined as 80% for the train set, 20% of train set for the validation set, and 20% for the test set, while the data distribution of Dataset B was determined as 75% for the train set, 20% of train set for the validation set, and 25% for the test set. On the other hand, while the data distribution of Dataset C was determined as 70% for the train set, 20% of train set for the validation set, and 30% for the test set, the data distribution of Dataset D was determined as 65% for the train set, 20% of train set for the validation set, and 35% for the test set.

Fig. 2 shows the 2D and 3D image data distributions, while the numbers of images used for train, test, and validation are presented in Table 1 for dataset A. Fig. 3 shows the 2D and 3D image data distributions, and the numbers of images used for train, test, and validation are presented in Table 2 for dataset B. Fig. 4 shows the 2D and 3D image data distributions, and the numbers of images used for train, test, and validation are shown in Table 3 for dataset C. Fig. 5 shows the 2D and 3D image data distributions, and the numbers of images used for train, test, and validation are given in Table 4 for dataset D.

E. THE PROPOSED HYBRID CLASSIFICATION MODEL

In this part of the study, it is aimed to demonstrate the superiority and classification ability of the proposed hybrid model over the existing models. In the proposed hybrid model, the features of the images were extracted by the designed CNN architecture for this study (Fig. 7) and they were classified using the machine learning and ensemble learning methods. The architecture designed for the proposed hybrid model is shown in Fig. 6.

In the context of the study, the data to be used in the model was collected first. A total of 134 images of Royal Gala and Golden apple fruits with a resolution of 1296 x 964 were divided into four image classes (golden apples with a high yield, golden apples with a low yield, royal gala apples with a high yield, and royal gala apples with a low yield).

In the second stage, 6700 images with a size of 256 x 256 were produced using the Pix2Pix GAN model due to the fact that a small amount of data could not give the desired result. In order to classify the generated synthetic images using the ML and ensemble methods, they were resized in a way that they would be 100 x 100 in size.

In the third stage, the resized images were first classified by ML methods. Classification operations were performed using the Support Vector Machine, Decision Tree, and Random Forest algorithms. At the stage after the results were obtained, the classification process was carried out by ensemble methods. Firstly, classification was performed with the bagging algorithm, which is one of the Ensemble methods. After that, the classification process was carried out with the AdaBoost (Adaptive Boosting), GBM (Gradient Boosting), XGBoost (Extreme Gradient Boosting), and Light GBM (Light Gradient Boosting) algorithms, which are the boosting algorithms.

In the fourth stage, the classification process of the images was carried out with the designed CNN architecture.

The fifth stage consisted of the proposed hybrid model. In the created CNN model, the information of the images, whose features were extracted from the first 5 layers with the feature extraction technique, was reshaped in order to generate the input of the algorithms and sent to the classification algorithms. Here, the extracted features were classified using the Support Vector Machine, Random Forest, and Decision Tree algorithms. After that, it was classified by Bagging algorithm, and then by AdaBoost, GBM, XGBoost,

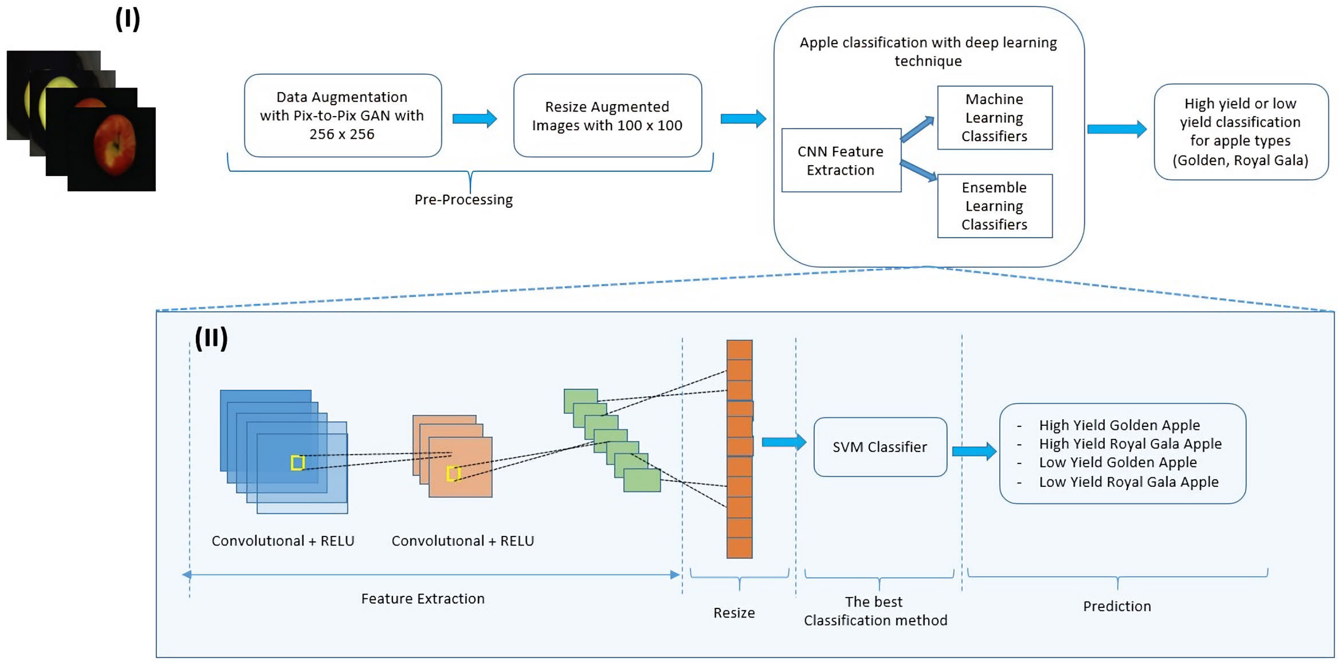


FIGURE 6. Block diagram of the proposed hybrid CNN model. (I) Review of the proposed hybrid classification model. (II) Architecture of the best hybrid classification model.

and Light GBM algorithms. The success performances of the algorithms were examined.

1) FEATURE EXTRACTION WITH CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Network (ConvNet or CNN) is the most popular deep learning algorithm. It basically performs classification by using the standard neural network. The CNN architecture is created by combining 3 layers: convolutional layer, pooling layer, and fully connected layer. The convolutional layer is the main structure of the CNN architecture. In this layer, the features of the image are extracted by applying filters to the image. Images whose features are extracted are sent to the activation layer.

In the designed CNN architecture, 5 convolutional layers and 1 fully connected layer were used. The first convolutional layer included 32 different 5×5-dimensional filters, batch normalization, and an activation layer for 100×100 image input. The second convolutional layer included 48 different 3×3-dimensional filters, a pooling layer with 2×2 filter size, and a dropout layer. There were 64 different 4×4-dimensional filters, batch normalization, and an activation layer in the third convolutional layer. There were 96 different 3×3-dimensional filters, batch normalization, activation, pooling layer, and dropout layer in the fourth convolutional layer. The last convolutional layer included 128 different 3×3-dimensional filters, batch normalization, and an activation layer. Finally, there were four neurons in the multiclass classification in the fully connected layer. 'relu' activation function was chosen for all convolution layers while 'softmax' activation was preferred in the dense layer. All datasets were fed to the CNN in packs with a batch size of 128 and the epoch number was set at 20. The hyper parameters of CNN architecture in this study are shown in Table 5.

2) SUPPORT VECTOR MACHINE

Support vector machine (SVM) is a supervised machine learning algorithm that can perform modeling for both classification and regression problems [19]. SVM draws a line to separate data placed in a plane. This line ensures that it is at the maximum distance of two classes.

3) DECISION TREE

Decision Tree (D-Tree) is a supervised machine learning algorithm that can perform modeling for both classification, regression, data mining, and statistical problems [20]. When the D-Tree algorithm performs classification, it creates decision nodes based on features and goals, and from these nodes, it also creates leaf nodes and carries out the classification process.

4) RANDOM FOREST

The Random Forest (RF) algorithm is also a supervised machine learning algorithm that can perform modeling for classification and regression problems [21]. Many individual decision trees are created in the model, and the trees are randomly gathered up. After the gathered trees form forests, forests are classified and estimation occurs.

5) BAGGING

Bagging (Bootstrap Aggregation) is an ensemble learning method aimed at improving the evaluation performance of a model [22]. BC (Bagging Classifier) is a variance reduction

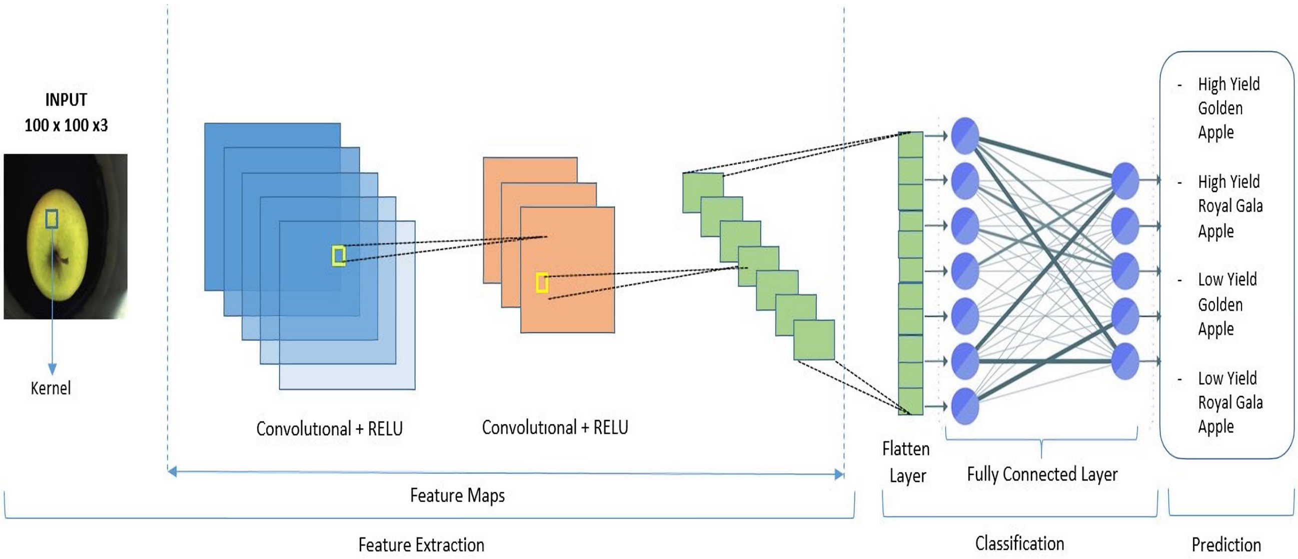


FIGURE 7. Used CNN architecture for this study.

TABLE 5. Parameters used in the CNN for this study.

Hyperparameter Name	Value
Optimizer	Adaptive Moment Estimation
Loss Function	Categorical Cross Entropy
Batch Size	128
Epoch	20
Image Size	100 x 100
Dataset Size	6700 Synthetic Apple Images
Learning Rate	1E-2
Validation Split	0.2
Average Epoch Time	1231237 milliseconds

method for certain algorithms or linear models such as decision tree [21]. In regression models, the decisions of the basic learners are calculated by taking their averages, while they are determined by voting in classification models.

6) BOOSTING

Boosting classifiers is an ensemble method and tries to get a strong learner by combining predictions coming from weak learners with voting within the framework of certain rules. Unlike bagging algorithms, when performing "voting", the learners predicted by "prediction" are sent in a sequential manner and the prediction process is carried out. Adaboost and gradient boosting algorithms, which are the most popular boosting algorithms, were used in the study. The most obvious feature of the adaboost algorithm is to distribute weak learners to their modified versions in a repeating cycle. Predictions coming from weak learners are brought together with voting and the final prediction is created [23]. In gradient boosting, first of all, after the first nodes are created, new

		ACTUAL	
		Positive	Negative
PREDICTED	Positive	TP	FP
	Negative	FN	TN

FIGURE 8. Confusion matrix.

TABLE 6. Results of used cnn model in this study for *Dataset A*.

Model	A	P	R	F	S-KFold	KFold	Cohen
CNN	0.7761	0.85	0.79	0.77	0.78 std: 0.0	0.78 std: 0.0	0.70 std: 0.0

trees are created based on prediction errors. This situation continues until no more results are obtained from the created nodes and trees.

IV. RESULTS

This part of the study includes information about the evaluation metrics and experimental results of the proposed hybrid model.

A. EVALUATION METRICS

In this part of study, performance analyses of machine learning, deep learning, bagging, boosting, and the proposed hybrid model were studied. Accuracy, precision, recall, and F-measure results of the applied models were examined.

1) CONFUSION MATRIX (CM)

CM is a table used to describe the performance of a model on a set of test data as in Fig. 8.

TABLE 7. The results of all hybrid models for *Dataset A*.

Hybrid Model	A	P	R	F	S-KFold	KFold	Cohen
CNN - Bagging	0.99552	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Random Forest	0.99701	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Decision Tree	0.99552	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Ada Boosting	0.49850	0.39	0.50	0.39	0.50 (std: 0.0)	0.50 (std: 0.0)	0.34 (std: 0.0)
***CNN - Gradient Boosting**	***0.99910***	***0.99***	***0.99***	***0.99***	***1.00 (std: 0.0)***	***1.00 (std: 0.0)***	***1.00 (std: 0.0)***
CNN - Extreme Gradient Boosting	0.99776	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Light Gradient Boosting	0.99776	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
***CNN - SVM (Linear)**	***0.99910***	***0.99***	***0.99***	***0.99***	***1.00 (std: 0.0)***	***1.00 (std: 0.0)***	***1,00 (std: 0.0)***
CNN - SVM (Radial Basis)	0.99104	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.98 (std: 0.0)
CNN - SVM (Polynomial)	0.98880	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.98 (std: 0.0)

TABLE 8. Results of used cnn model in this study for *Dataset B*.

Model	A	P	R	F	S-KFold	KFold	Cohen
CNN	0.792	0.89	0.79	0.78	0.79 std: 0.0	0.79 std: 0.0	0.72 std: 0.0

2) ACCURACY (A)

is a ratio of total prediction in the model.

3) PRECISION (P)

shows how many of the positively predicted values have turned into actual positive values.

4) RECALL (R)

shows how many of the positively predicted transactions have been predicted as positively.

5) F-MEASURE (F)

shows the harmonic average of precision and recall values.

6) COHEN's KAPPA (COHEN)

is a statistical method that measures the reliability of comparative agreement between two raters [24].

$$A = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{1}$$

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

$$F = \frac{2xPxR}{P + R} \tag{4}$$

$$Cohen = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)} \tag{5}$$

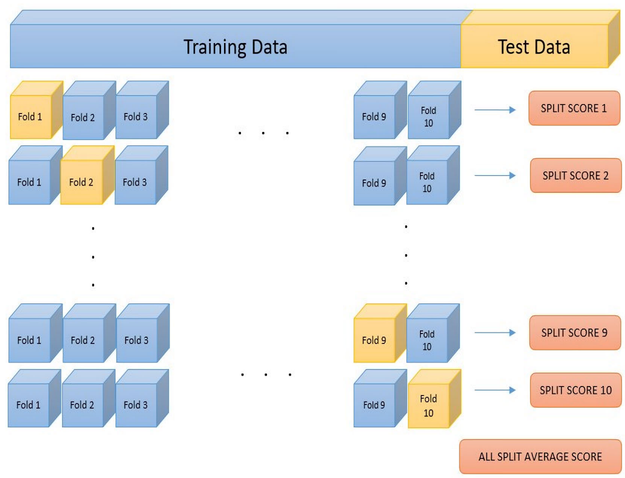


FIGURE 9. Evaluation of KFold cross validation.

B. KFold AND STRATIFIED KFold CROSS VALIDATION

KFold Cross-validation (KFold) is a statistical method applied to improve performance on data by using all the data of the applied model effectively. As shown in Figure 9, 10-KFold cross validation was used in the proposed hybrid model. In cross-validation, the train set is allocated as many as the k value determined by mixing the data set randomly. All allocated groups are used as a verification set. The evaluation scores of the model are stored for each fold and the average value is obtained. Stratified KFold (S-KFold) is a variation of KFold that return stratified folds. The folds are made by preserving the percentage of samples for each class.

C. RESULTS FOR DATASET A

When the results were examined, it was observed that in Dataset A, the accuracy result of CNN model was 77.61%. The detailed results of the performance metrics are presented in Table 6. For the hybrid model using ML methods, the

TABLE 9. The results of all hybrid models for *Dataset B*.

Hybrid Model	A	P	R	F	S-KFold	KFold	Cohen
CNN - Bagging	0.99701	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Random Forest	0.99761	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Decision Tree	0.99522	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Ada Boosting	0.76194	0.65	0.76	0.68	0.76 (std: 0.0)	0.76 (std: 0.0)	0.68 (std: 0.0)
CNN - Gradient Boosting	0.99761	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	1.00 (std: 0.0)
CNN - Extreme Gradient Boosting	0.99820	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - Light Gradient Boosting	0.99820	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
****CNN - SVM (Linear)***	***0.99882***	***0.99***	***0.99***	***0.99***	***1.00 (std: 0.0)***	***1.00 (std: 0.0)***	***0.99 (std: 0.0)***
CNN - SVM (Radial Basis)	0.99552	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - SVM (Polynomial)	0.99880	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)

TABLE 10. Results of used cnn model in this study for *Dataset C*.

Model	A	P	R	F	S-KFold	KFold	Cohen
CNN	0.902	0.93	0.90	0.91	0.90 std: 0.0	0.90 std: 0.0	0.87 std: 0.0

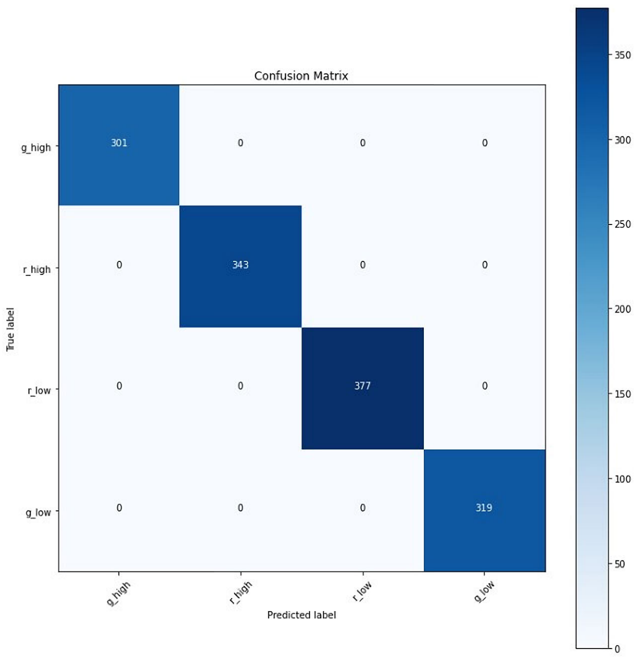


FIGURE 10. Confusion matrix of CNN-SVM (linear) hybrid model for Dataset A.

best accuracy result was determined in the CNN-SVM model (*linear*). For the hybrid model using ensemble learning methods, the best accuracy result was determined in the CNN-Gradient Boosting hybrid model. Confusion matrices for the CNN-SVM (*linear*) and CNN-Gradient Boosting are shown in Figs. 10 and 11, respectively. The accuracy result

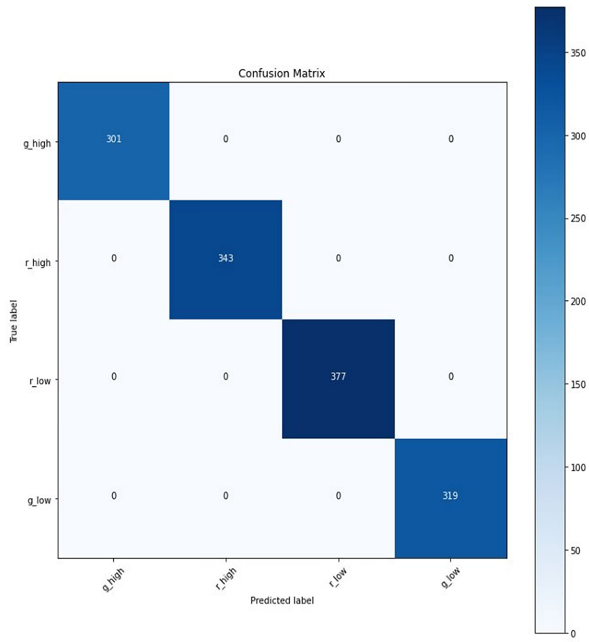


FIGURE 11. Confusion matrix of CNN-Gradient Boosting hybrid model for Dataset A.

for both hybrid models was 99.91%. The detailed results of other hybrid models and the best hybrid model are shown in Table 7.

D. RESULTS FOR DATASET B

For Dataset B, the accuracy result of the CNN model was 79.283%. The detailed results of the performance metrics are presented in Table 8. For the hybrid model using ML methods, the best accuracy result was determined in the CNN-SVM model (*linear*). Confusion matrix of CNN-SVM (*linear*) is shown in Fig. 12. On the other hand, for the hybrid model using ensemble learning methods, the best accuracy results were determined in CNN-Extreme Gradient Boost-

TABLE 11. The results of all hybrid models for *Dataset C*.

Hybrid Model	A	P	R	F	S-KFold	KFold	Cohen
CNN - Bagging	0.99303	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Random Forest	0.99601	0.99	0.99	0.99	1.00 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Decision Tree	0.99253	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Ada Boosting	0.75970	0.64	0,76	0.68	0.76 (std: 0.0)	0.76 (std: 0.0)	0.68 (std: 0.0)
CNN - Gradient Boosting	0.99402	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Extreme Gradient Boosting	0.99353	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Light Gradient Boosting	0.99601	0.99	0.99	0.99	1.00 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - SVM (Linear)	***0.99701***	***0.99***	***0.99***	***0.99***	***1.00 (std: 0.0)***	***1.00 (std: 0.0)***	***0.99 (std: 0.0)***
CNN - SVM (Radial Basis)	0.99651	0.99	0.99	0.99	1.00 (std: 0.0)	1.00 (std: 0.0)	0.99 (std: 0.0)
CNN - SVM (Polynomial)	0.99203	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)

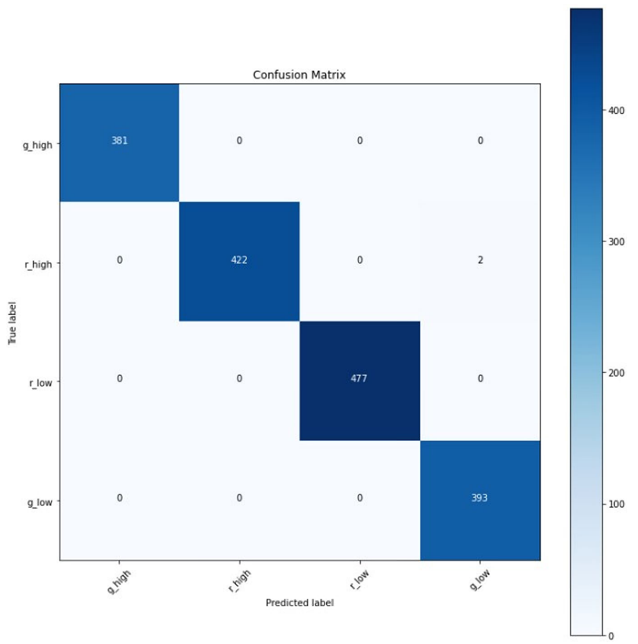


FIGURE 12. Confusion matrix of CNN-SVM (linear) hybrid model for Dataset B.

TABLE 12. Results of used cnn model in this study for *Dataset D*.

Model	A	P	R	F	S-KFold	KFold	Cohen
CNN	0.781	0.84	0.78	0.77	0.78 std: 0.0	0.78 std: 0.0	0.71 std: 0.0

ing and CNN-Light Gradient Boosting. The accuracy results were 99.82% for both ensemble hybrid models and 99.88% for ML methods. The detailed results of other hybrid models and the best hybrid models are shown in Table 9.

E. RESULTS FOR DATASET C

For Dataset C, the accuracy result was observed as 90.29% in the CNN model, and the detailed results of the performance metrics are presented in Table 10. For the hybrid model using

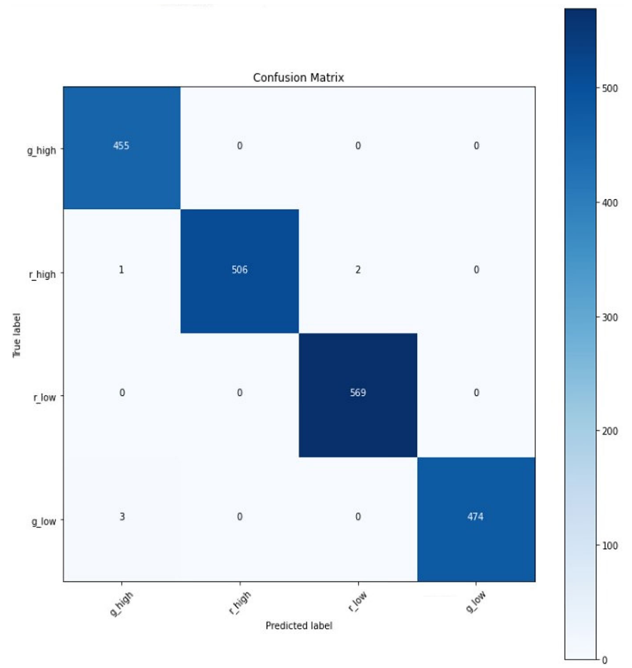


FIGURE 13. Confusion matrix of CNN-SVM (linear) hybrid model for Dataset C.

ML methods, the best accuracy result was determined in the CNN-SVM model (*linear*). On the other hand, for the hybrid model using ensemble learning methods, the best accuracy result was determined in the CNN-Light Gradient Boosting. The accuracy result for CNN-SVM (linear) hybrid model was 99.70%, while the accuracy result for CNN-Light Gradient Boosting hybrid model was 99.60%. The confusion matrix of CNN-SVM (*linear*) model is shown in Fig. 13. The detailed results of other hybrid models and the best hybrid model are shown in Table 11.

F. RESULTS FOR DATASET D

For Dataset D, in the CNN model, the accuracy result was observed as 78.16% and the detailed results of the

TABLE 13. **The results of all hybrid models for *Dataset D*.**

Hybrid Model	A	P	R	F	S-KFold	KFold	Cohen
CNN - Bagging	0.99658	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Random Forest	0.99616	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Decision Tree	0.99616	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Ada Boosting	0.73347	0.62	0.73	0.66	0.73 (std: 0.0)	0.73 (std: 0.0)	0.65 (std: 0.0)
CNN - Gradient Boosting	0.99658	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Extreme Gradient Boosting	0.99651	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - Light Gradient Boosting	0.99658	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - SVM (Linear)	***0.99744***	***0.99***	***0.99***	***0.99***	***0.99 (std: 0.0)***	***0.99 (std: 0.0)***	***0.99 (std: 0.0)***
CNN - SVM (Radial Basis)	0.99275	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)
CNN - SVM (Polynomial)	0.99147	0.99	0.99	0.99	0.99 (std: 0.0)	0.99 (std: 0.0)	0.99 (std: 0.0)

TABLE 14. **The results of similar studies and their comparison with the proposed model.**

Dataset	Method	Acc	Pre	Rec	F	Cohen
Kesar Mango [28]	The CNN performance result of a study in [26]	83.97	-	-	-	-
Kesar Mango [28]	***Proposed CNN-SVM (linear) Hybrid Model***	***94.36***	***94.53***	***94.33***	***94.37***	***88.68***
FruitNet [25]	The CNN performance result of a study in [29]	96.72	96.80	96.80	96.74	96.43
FruitNet [25]	***Proposed CNN-SVM (linear) Hybrid Model***	***98.25***	***98.24***	***98.25***	***98.23***	***98.09***
Papaya Image Dataset [27]	The CNN performance result of a study in [30]	96.50	-	-	-	-
Papaya Image Dataset [27]	***Proposed CNN-SVM (linear) Hybrid Model***	***98.88***	***98.92***	***98.88***	***98.88***	***98.32***

performance metrics are given in Table 12. For the hybrid model using ML methods, the best accuracy result was determined in the CNN-SVM model (linear). For the hybrid model using ensemble learning methods, the best accuracy result was determined in all boosting hybrid models. The accuracy result for the CNN-SVM (linear) hybrid model was 99.74%. The confusion matrix of the CNN-SVM (linear) hybrid model is shown in Fig. 14. The accuracy result for all ensemble hybrid models was 99.65%. The detailed results of other hybrid models and the best hybrid models are shown in Table 13.

When evaluating the results in general, it was seen that the CNN-SVM (linear) hybrid model gave the best results for all dataset types. On the other hand, the CNN-Gradient Boosting hybrid model gave the best results for Dataset A, while the CNN-Extreme Gradient Boosting and CNN-Light Gradient Boosting hybrid models gave the best results in the ensemble hybrid models for Dataset B, C, and D. The accuracy rate results of the proposed hybrid models were found to be higher than the CNN model. It was observed that the accuracy of the CNN model increased as the number of test data increased. However, for Dataset D, the closeness of the train, validation, and test data distributions decreased the accuracy of the CNN model.

The proposed hybrid model was applied to datasets previously studied in the literature, and the results were compared. Dataset C, the recommended dataset type *(70% of the number of images is Train Set, 30% of the number of images is Test Set, And 20% of Train Set is Validation Set*), was implemented on the FruitNet [25], Kesar Mango [26], and Papaya [27] dataset. When the performance of the hybrid model proposed

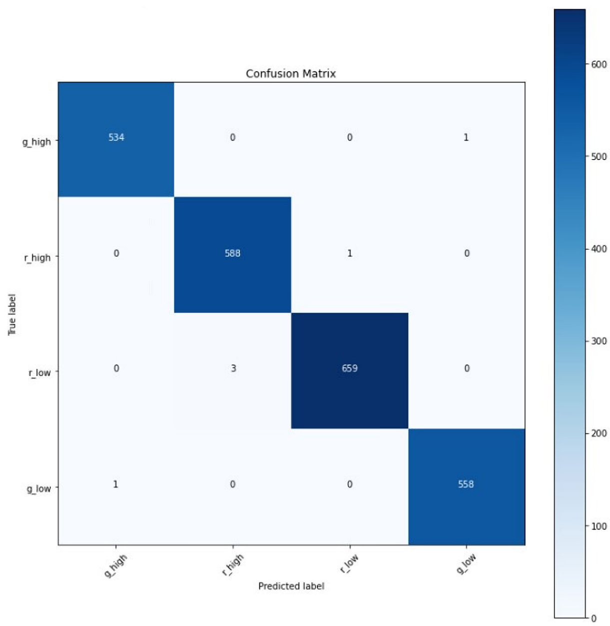


FIGURE 14. **Confusion matrix of CNN-SVM (linear) hybrid model for Dataset D.**

in this study was compared with the performances of the studies in the literature, it was observed that the performance of the proposed hybrid model was better than the other models. A performance comparison of the models is given in Table 14.

V. CONCLUSION

In the field of agriculture, the determination and classification of fruit productivity are among the important issues. The main purpose of this study was to classify the productivity of apple fruit. For the purpose of the study, Golden and Royal Gala types of apple fruit were obtained via COFILAB. The fruits were classified based on their weight by using Nutritionix data. By augmenting the small number of apple images with the Pix-to-Pix GAN model, 6700 synthetic images were produced. Depending on their characteristics, the images were divided into four different classes as high-yield *golden* images (*g_high*), low-yield *golden* images (*g_low*), high-yield *royal gala* images (*r_high*), and low-yield *royal gala* images (*r_low*). Four different datasets were designed to measure the accuracy of data and the data distribution of datasets was explained in detail in Section 3.2. Compared to other dataset types, Dataset C gave the best results. The accuracy of the proposed CNN model in this dataset was 90.29%. Images extracted with feature extractions by using the CNN model were classified by machine learning models (SVM, Decision Tree, and RF) and extreme learning models (Bagging, AdaBoost, GBM, Extreme GBM, and Light GBM). It was determined that the hybrid model giving the best results was the CNN-SVM model, and its accuracy was 99.70%. The proposed hybrid model was applied to datasets that had been used in similar studies with high accuracy values. Whereas the accuracy results were measured as 96.72, 96.50%, and 83.97% for *FruitNet*, *Papaya Images Dataset,* and *Kesar Mango* datasets in previous studies respectively, the accuracy values were determined as 98.25%, 98.88%, and 94.36% respectively in this study after applying the proposed hybrid model on the same datasets.

In conclusion, when the results were examined, it was observed that the proposed hybrid model revealed very effective results on different datasets compared to the other models.

ACKNOWLEDGMENT

This study was conducted by Ph.D. student Fatih Bal (Duzce University) under the supervision of Dr. Fatih Kayaalp (Duzce University) as part of a doctoral dissertation study.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

ETHICAL APPROVAL

This study does not require any ethics committee approval.

CODE OF PAPER

This study was carried out with python programming. You can access codes and images used in the study via Github [31].

REFERENCES

[1] FAO. (2020). *Food and Agriculture Organization of the United Nations*. [Online]. Available: https://www.fao.org/faostat/en/#search/Apples

[2] H. Jiang, C. Zhang, Y. Qiao, Z. Zhang, W. Zhang, and C. Song, "CNN feature based graph convolutional network for weed and crop recognition in smart farming," *Comput. Electron. Agricult.*, vol. 174, Jul. 2020, Art. no. 105450, doi: 10.1016/j.compag.2020.105450.

[3] M. Koklu, M. F. Unlersen, I. A. Ozkan, M. F. Aslan, and K. Sabanci, "A CNN-SVM study based on selected deep features for grapevine leaves classification," *Measurement*, vol. 188, Jan. 2022, Art. no. 110425, doi: 10.1016/j.measurement.2021.110425.

[4] A. Taherkhani, G. Cosma, and T. M. McGinnity, "AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning," *Neurocomputing*, vol. 404, pp. 351–366, Sep. 2020, doi: 10.1016/j.neucom.2020.03.064.

[5] M. Khanramaki, E. A. Asli-Ardeh, and E. Kozegar, "Citrus pests classification using an ensemble of deep learning models," *Comput. Electron. Agricult.*, vol. 186, Jul. 2021, Art. no. 106192, doi: 10.1016/j.compag.2021.106192.

[6] S. Thongsuwan, S. Jaiyen, A. Padcharoen, and P. Agarwal, "ConvXGB: A new deep learning model for classification problems based on CNN and XGBoost," *Nucl. Eng. Technol.*, vol. 53, no. 2, pp. 522–531, Feb. 2021, doi: 10.1016/j.net.2020.04.008.

[7] M. H. Ibrahim, "WBA-DNN: A hybrid weight bat algorithm with deep neural network for classification of poisonous and harmful wild plants," *Comput. Electron. Agricult.*, vol. 190, Nov. 2021, Art. no. 106478, doi: 10.1016/j.compag.2021.106478.

[8] X. Hao, J. Jia, A. Mateen Khattak, L. Zhang, X. Guo, W. Gao, and M. Wang, "Growing period classification of *Gynura bicolor* DC using GL-CNN," *Comput. Electron. Agricult.*, vol. 174, Jul. 2020, Art. no. 105497, doi: 10.1016/j.compag.2020.105497.

[9] J. Kong, H. Wang, X. Wang, X. Jin, X. Fang, and S. Lin, "Multi-stream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture," *Comput. Electron. Agricult.*, vol. 185, Jun. 2021, Art. no. 106134, doi: 10.1016/j.compag.2021.106134.

[10] L. B. Ferreira and F. F. Da Cunha, "Multi-step ahead forecasting of daily reference evapotranspiration using deep learning," *Comput. Electron. Agricult.*, vol. 178, Nov. 2020, Art. no. 105728, doi: 10.1016/j.compag.2020.105728.

[11] A. D. Jia, B. Z. Li, and C. C. Zhang, "Detection of cervical cancer cells based on strong feature CNN-SVM network," *Neurocomputing*, vol. 411, pp. 112–127, Dec. 2020, doi: 10.1016/j.neucom.2020.06.006.

[12] F. Balgetir, F. Bilek, S. Kakakus, S. Arslan-Tuncer, and C. F. Demir, "Detection of ataxia in low disability MS patients by hybrid convolutional neural networks based on images of plantar pressure distribution," *Multiple Sclerosis Rel. Disorders*, vol. 56, Nov. 2021, Art. no. 103261, doi: 10.1016/j.msard.2021.103261.

[13] F. Li and M. Liu, "A hybrid convolutional and recurrent neural network for hippocampus analysis in Alzheimer's disease," *J. Neurosci. Methods*, vol. 323, pp. 108–118, Jul. 2019, doi: 10.1016/j.jneumeth.2019.05.006.

[14] H. Bian, Q. Wang, G. Xu, and X. Zhao, "Load forecasting of hybrid deep learning model considering accumulated temperature effect," *Energy Rep.*, vol. 8, pp. 205–215, Apr. 2022, doi: 10.1016/j.egyr.2021.11.082.

[15] F. B. Sarijaloo, M. Porta, B. Taslimi, and P. M. Pardalos, "Yield performance estimation of corn hybrids using machine learning algorithms," *Artif. Intell. Agricult.*, vol. 5, pp. 82–89, Jan. 2021, doi: 10.1016/j.aiia.2021.05.001.

[16] COFILAB. (2021). *Computers and Optics in Food Inspection*. [Online]. Available: http://www.cofilab.com/downloads/

[17] Nutritionix. (2022). *Nutritionix a Syndigo Company*. [Online]. Available: https://www.nutritionix.com/i/usda/apples-raw-golden-delicious-with-skin-1-large/513fceb575b8dbbc210011f4

[18] Nutritionix. (2022). *Nutritionix a Syndigo Company*. [Online]. Available: https://www.nutritionix.com/i/usda/apples-raw-gala-with-skin-1-large/463d6237c9eb05c38196f272

[19] Rizwan-ul-Hassan, C. Li, and Y. Liu, "Online dynamic security assessment of wind integrated power system using SDAE with SVM ensemble boosting learner," *Int. J. Electr. Power Energy Syst.*, vol. 125, Feb. 2021, Art. no. 106429, doi: 10.1016/j.ijepes.2020.106429.
[20] M. Hamdi, I. Hilali-Jaghdam, B. E. Elnaim, and A. A. Elhag, "Forecasting and classification of new cases of COVID 19 before vaccination using decision trees and Gaussian mixture model," *Alexandria Eng. J.*, vol. 62, pp. 327–333, Jan. 2023, doi: 10.1016/j.aej.2022.07.011.
[21] L. Breiman, *Machine Learning*, vol. 45. Norwell, MA, USA: Kluwer Academic, 2001, pp. 5–32.
[22] R. E. Schapire, "A brief introduction to boosting," in *Proc. 16th Int. Joint Conf. Artif. Intell. (IJCAI)*, vol. 2, 1999, pp. 1401–1406.
[23] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997, doi: 10.1006/jcss.1997.1504.
[24] J. Cohen, "A coefficient of agreement for nominal scales," *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, Apr. 1960, doi: 10.1177/001316446002000104.
[25] V. Meshram and K. Patil, "FruitNet: Indian fruits image dataset with quality for machine learning applications," *Data Brief*, vol. 40, Feb. 2022, Art. no. 107686, doi: 10.1016/j.dib.2021.107686.
[26] S. Naik, "Non-destructive mango (Mangifera Indica L., cv. Kesar) grading using convolutional neural network and support vector machine," *SSRN Electron. J.*, pp. 1–9, Feb. 2019, doi: 10.2139/ssrn.3354473.
[27] S. K. Behera, A. K. Rath, and P. K. Sethy, "Maturity status classification of papaya fruits based on machine learning and transfer learning approach," *Inf. Process. Agricult.*, vol. 8, no. 2, pp. 244–250, Jun. 2021, doi: 10.1016/j.inpa.2020.05.003.
[28] S. Naik, B. Patel, and R. Pandey, "Shape, size and maturity features extraction with fuzzy classifier for non-destructive mango (Mangifera indica l., cv. Kesar) grading," in *Proc. IEEE Technol. Innov. ICT Agricult. Rural Develop. (TIAR)*, Jul. 2015, pp. 1–7, doi: 10.1109/TIAR.2015.7358522.
[29] T. B. Shahi, C. Sitaula, A. Neupane, and W. Guo, "Fruit classification using attention-based MobileNetV2 for industrial applications," *PLoS ONE*, vol. 17, no. 2, Feb. 2022, Art. no. e0264586, doi: 10.1371/journal.pone.0264586.
[30] S. Gayathri, T. U. Ujwala, C. V. Vinusha, N. R. Pauline, and D. B. Tharunika, "Detection of papaya ripeness using deep learning approach," in *Proc. 3rd Int. Conf. Inventive Res. Comput. Appl. (ICIRCA)*, Sep. 2021, pp. 1755–1758, doi: 10.1109/ICIRCA51532.2021.9544902.
[31] GITHUB. (2022). *Paper Codes and Data Sample*. [Online]. Available: https://github.com/balfatih/Determination_of_Productivity_for_Apple

FATIH BAL received the B.S. degree in software engineering from Maltepe University, in 2014, and the M.S. degree in computer engineering from Beykent University. He is currently pursuing the doctoral degree with Duzce University. Currently, he is working as an Instructor with Kirklareli University. His primary research interests include data science, data analysis, data mining, machine learning, deep learning, smart farm, and web technologies.

FATIH KAYAALP received the B.S. degree in computer science from Marmara University, in 2000, and the M.S. and Ph.D. degrees in computer science from Sakarya University, in 2005 and 2014, respectively. Currently, he is working as an Assistant Professor with Duzce University. His primary research interests include databases, web technologies, computer networks, wireless sensor networks, mobile computing, and machine learning.