

## Article

# Classification of Agriculture Farm Machinery Using Machine Learning and Internet of Things

Muhammad Waleed <sup>1</sup> , Tai-Won Um <sup>2,\*</sup>, Tariq Kamal <sup>3</sup> and Syed Muhammad Usman <sup>3</sup>

<sup>1</sup> Department of Information and Communication Engineering, Chosun University, Gwangju 61452, Korea; mwkhan@chosun.kr

<sup>2</sup> College of Science and Technology, Duksung Women's University, Seoul 01369, Korea

<sup>3</sup> Department of Computer Systems Engineering, University of Engineering and Technology (UET), Peshawar 25120, Pakistan; tkamal@uetpeshawar.edu.pk (T.K.); 16pwcse1444@uetpeshawar.edu.pk (S.M.U.)

\* Correspondence: twum@duksung.ac.kr; Tel.: +82-2-901-8643

**Abstract:** In this paper, we apply the multi-class supervised machine learning techniques for classifying the agriculture farm machinery. The classification of farm machinery is important when performing the automatic authentication of field activity in a remote setup. In the absence of a sound machine recognition system, there is every possibility of a fraudulent activity taking place. To address this need, we classify the machinery using five machine learning techniques—K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF) and Gradient Boosting (GB). For training of the model, we use the vibration and tilt of machinery. The vibration and tilt of machinery are recorded using the accelerometer and gyroscope sensors, respectively. The machinery included the leveler, rotavator and cultivator. The preliminary analysis on the collected data revealed that the farm machinery (when in operation) showed big variations in vibration and tilt, but observed similar means. Additionally, the accuracies of vibration-based and tilt-based classifications of farm machinery show good accuracy when used alone (with vibration showing slightly better numbers than the tilt). However, the accuracies improve further when both (the tilt and vibration) are used together. Furthermore, all five machine learning algorithms used for classification have an accuracy of more than 82%, but random forest was the best performing. The gradient boosting and random forest show slight over-fitting (about 9%), but both algorithms produce high testing accuracy. In terms of execution time, the decision tree takes the least time to train, while the gradient boosting takes the most time.

**Keywords:** agriculture; farm machinery; remote sensing; machine learning; classification



**Citation:** Waleed, M.; Um, T.-W.; Kamal, T.; Muhammad Usman, S. Classification of Agriculture Farm Machinery Using Machine Learning and Internet of Things. *Symmetry* **2021**, *13*, 403. <https://doi.org/10.3390/sym13030403>

Academic Editor: Peng-Yeng Yin

Received: 2 February 2021

Accepted: 23 February 2021

Published: 1 March 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As the world is progressing towards the fourth Industrial Revolution [1], the use of state-of-the-art technologies, including the Internet of Things (IoT) [2], Machine Learning [3] and Cloud Computing [4], are becoming mainstream. Agriculture contributes a big chunk to the world economy [5] and modernizing it can drastically increase the production of farms, thus causing an overall growth in the world economy. Agriculture modernization requires the use of agriculture machinery to plant, cultivate and harvest the crops.

The use of the right machinery at the right time is important for getting increased productivity. Typically, the on-field machinery usage is manually supervised and verified as the usage process and remote sensing is prone to fraudulent activities. There are various ways a fraudulent activity could take place. For example, a reservation was made, but the machinery was never used or only a partial activity was performed. Similarly, a reservation was made for a specific machinery type, but a wrong machinery was used. There is also a possibility that no machinery was used at all. This is possible as the machines don't come with a remote recognition feature (or mounted devices). To mitigate this and remotely authenticate the use of agriculture machinery, a two-pronged solution is required. First,

determine the location and work area of the machinery. Second, determine the farm machinery type used for activity.

The first problem, which requires the determination of the work area, was solved using the IoT (GPS), convex hull and AI algorithms. The methods for this are already published in our previous work [6]. The second problem, which is the determination of machinery type used, is being addressed in this paper. The solution is sought using the IoT-backed remote sensing and supervised machine learning algorithms.

Machine learning, together with big data, is revolutionizing agriculture and producing better results than before. The classification is performed using neural networks, K-Nearest Neighbor (KNN) and Naïve Bayes classifier [7,8]. It is used in agriculture for machinery, crop, soil and livestock management [7]. The Support Vector Machine (SVM) algorithm has been used for soil texture classification and was found to be helpful in the choosing of crops [9]. Similarly, decision tree was used in performing the classification for predictive analysis [10]. These methods are also largely used for the classification of agriculture activities, including harvesting, bed-making, transplantation, walking and standstill. Moreover, there are studies on the classification of faults in rotating machinery using the vibration signals [11–13]. Furthermore, IoT and deep learning techniques have been used for sensing soil temperature, nutrients and humidity, controlling and analyzing water consumption [14].

To our knowledge, the machine learning algorithms have not been applied to the recognition of agricultural farm machinery. If applied, we believe that the modeling work will be useful for improving agricultural field operations and its remote monitoring. We are interested in developing a robust classification model that can be used to automatically classify the farm machinery.

Historically, vibration and tilt have been endorsed as important characteristics of machines [15,16]. This endorsement was further strengthened by our analysis, where we collected data about the vibration and tilt of machinery from leveler, rotavator, and cultivator, and observed that all three types of machinery are distinguishable in terms mean and standard deviation. We discovered that the deviation between the data is because of the different vibrations and tilt caused by the machinery during usage. More information on this analysis is given in Section 4.

We apply five supervised machine learning techniques—K-Nearest Neighbor, Support Vector Machine, Decision Tree, Random Forest and Gradient Boosting. Our experimental results reveal that all five machine learning techniques have shown good accuracy when classifying the machinery type on the vibration and tilt alone. The accuracy improves further when the vibration and tilt are used together for the training of the models. Out of the five techniques, the Random Forest showed more than 90% accuracy. The Gradient boosting and random forest show slight over-fitting (about 9%), but both algorithms produce the highest testing accuracy.

It is also important to compare the models in terms of execution time. We found the decision tree as the fastest taking the least time to train, while the gradient boosting was the slowest (taking the most time to train). Since, the models need continuous re-training to keep up with accuracy and recognize new patterns, we outline a methodology for re-training of the models.

The rest of the paper is structured as follows—Section 2 describes the related work, followed by Data Collection and Processing Framework in Section 3. Section 4 identifies the features required for classification. Section 5 discusses the classification algorithms in detail. Section 6 prepares the data for model fitting, applies the machine learning algorithms and evaluates the models and features in terms of accuracy. Section 7 outlines a mechanism for the deployment of models on the cloud, and its retraining. Discussion and concluding remarks are presented in Section 8.

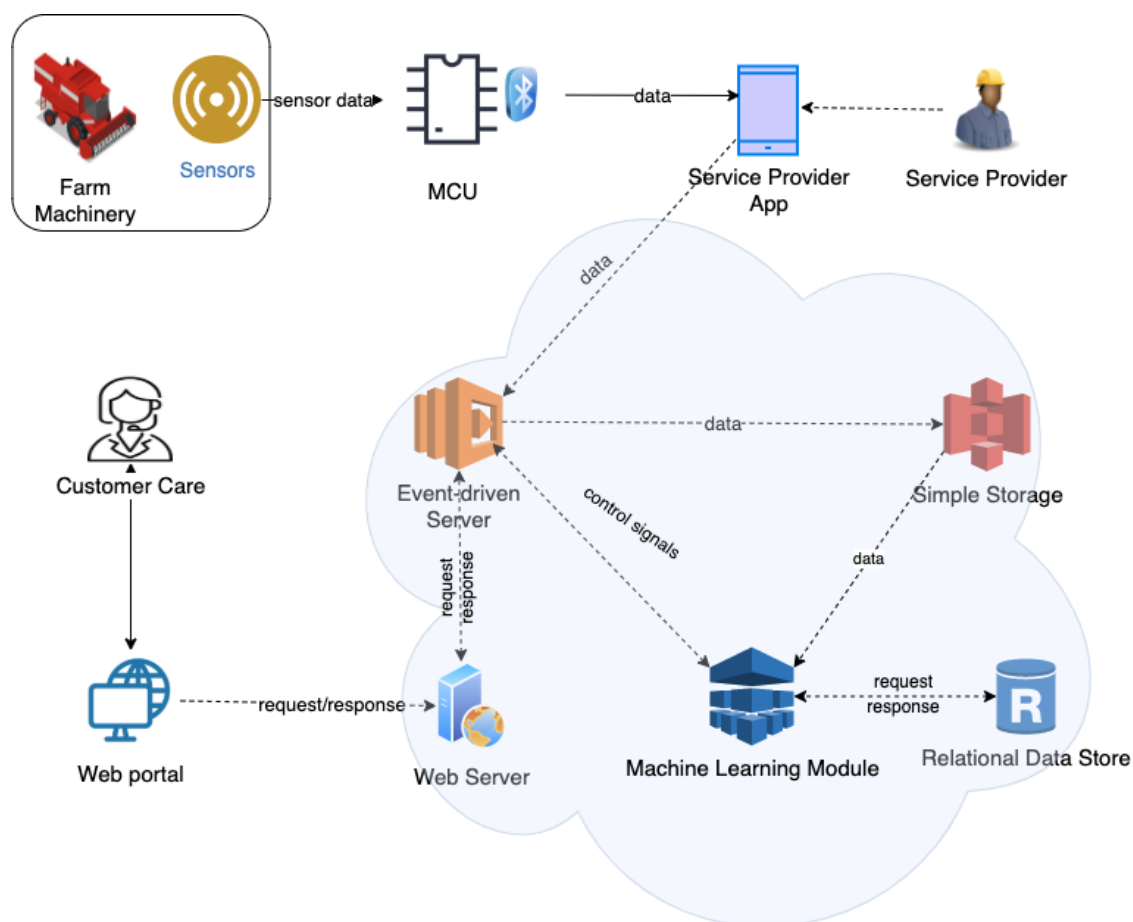
## 2. Related Work

IoT and machine learning algorithms have been extensively used in agriculture for automation and analysis [17,18]. Several recent reviews can be found on the use of classification in agriculture statistical analysis [19]. Sharma et al. used a smart phone carried by a farmer and recorded accelerometer data to perform classification of agriculture activities, including harvesting, bed-making, transplantation, walking and standstill. The classification was performed using neural networks, KNN and Naïve Bayes classifiers, where neural networks was found to be the most accurate [8]. In the same way, different machine learning methods have been deployed for soil type classification including naïve bayes, SVM and deep learning [20–22]. Alternatively, Barman et al. used the support vector machine learning algorithm for soil texture classification. The classification of soil texture could be helpful when decide to cultivate the crops [9]. The predictive analysis was performed using the decision tree [10]. Dan et al. used big data analytics to perform data analysis on agriculture machinery, saving a lot of considerable time and cost [23]. In another study, the machine learning algorithms were applied towards the classification of faults in rotating machinery using the vibration signals [11–13]. To predict the growth of plants and crops, Singh et al. have developed a classification system using differential evolution algorithms [24]. In another work, Zhao et al. used the vision-based classification approach to navigate the agriculture machinery for efficiency [25]. The deep learning models are extensively applied towards the precision agriculture on data collected using the internet of things [26]. While Eleni et al. have developed a cloud middleware approach to support precision agriculture with massive data generated from IoT [27]. The recent trends in sensors and IoT were discussed in detail in the review work of Laura and Lorena [28]. Several IoT architectures are developed for smart agriculture [29–31].

Artificial Intelligence (AI) and cloud computing together with big data is revolutionizing the agriculture sector and producing better results than before. AI in agriculture is applied on sensors data for machinery, crop, soil, and livestock management [7,32]. Bhavani et al. proposed an IoT-based agriculture system to overcome the economic losses by predicting and preventing the harmful diseases affecting the farm [10]. In another study, Waleed et al. showed the application of IoT in calculation of field activity. They found that GPS could accurately record the field location and the operation time of agriculture machinery [6]. In a similar study, a yield monitoring system was proposed to efficiently collect the GPS data and perform operational analysis of the farm machinery [33]. To analyse the agriculture machinery operational cost, Sopegno et al. have developed a smart web and mobile platform called AMACA (Agricultural Machine App Cost Analysis). The application could be used on own machinery or to hire an agriculture machinery service [34]. Gard et al. described the use of IoT and deep learning for sensing soil temperature, nutrients and humidity, controlling and analyzing water consumption [14]. Prathibha et al. used the temperature and humidity sensors to monitor the agriculture field. The data from sensors was transferred to farmers mobile for analysis and storage [35]. To fulfill the demand of high-quality agriculture machinery, Zhang et al. have devised a design of agriculture machinery service management system that is based on latest IoT and cloud technology. It consists of a mobile application and a web server for effective monitoring of the system [36].

## 3. Data Collection, Storage and Processing

In order to perform the model fitting, we need to collect data on the vibration and tilt of agricultural machinery. The platform that we have developed for the collection of data and its processing consists of an IoT module, a smartphone application and cloud modules as shown in Figure 1. The processing of data happens on the cloud supported by the event-handlers, data-storage units and machine learning modules. We use Amazon Web Services for cloud services. Details of each of the components are given below.

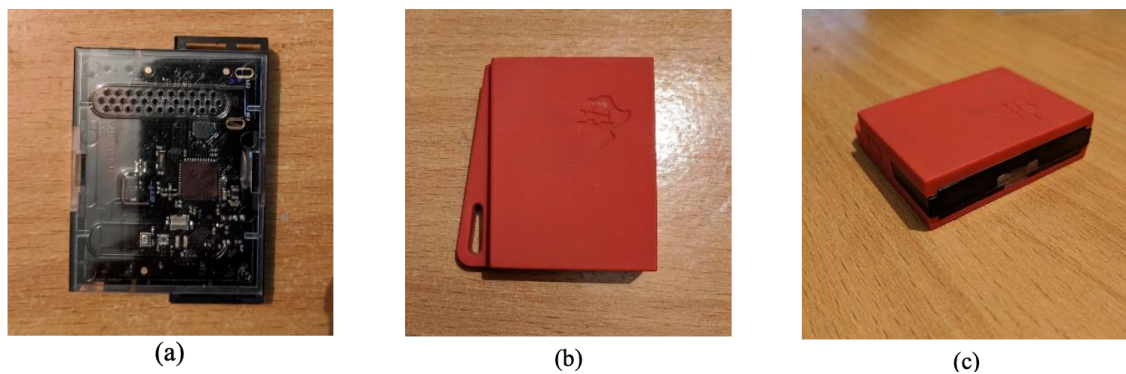


**Figure 1.** The system architecture diagram showing the Internet of Things (IoT) device, intermediate communication gateways, cloud processing and archival.

### 3.1. *Iot Module and Mobile Application*

We use the IoT Module TI CC2650 Sensor Tag developed by Texas Instruments as shown in Figure 2. The Sensor Tag has a Bluetooth low energy MCU and several sensors are mounted on it (IR Temperature, Movement 9 axis (accelerometer, gyroscope, magnetometer, and humidity, etc.). The main features of the IoT module are lower power consumption with longer battery life from the lithium battery cell, high performance ARM Cortex M3, complete development system and extreme flexibility for IoT based applications.

Data from movement sensors (accelerometer and gyroscope) from the sensor tag are used to generate data for the machine learning algorithms (responsible for machinery classification). The data consist of 16 bits or 2 bytes for each axis (total 12 bytes). The data are read using a Bluetooth connection with the service provider smartphone app. The service provider application on the driver's smart phone integrates the GPS data (longitude, latitude), date and time with sensors data coming from the IoT module. After integration, these grouped data in CSV format are sent to the cloud via 3G or 4G/LTE interface in the smartphone.



**Figure 2.** TI MCU CC2650 Sensor tag (a) Top view without cover (b) Top view with cover (c) Size view with power button visible.

### 3.2. The Cloud Data Store and Processing

For all the cloud operations (storage, processing and communication), we use the Amazon Web Services [37]. The smartphone app records the data from agriculture machinery and sends it to the cloud. The API gateway that receives the data is implemented using the lambda event-driven functions. The cloud has two storage classes—a simple storage and relational database [38]. The data from sensors are archived in non-relational storage also known as S3. The relational database (implemented using the RDS) stores the output of machine learning algorithms [39].

### 3.3. Machine Learning Modules

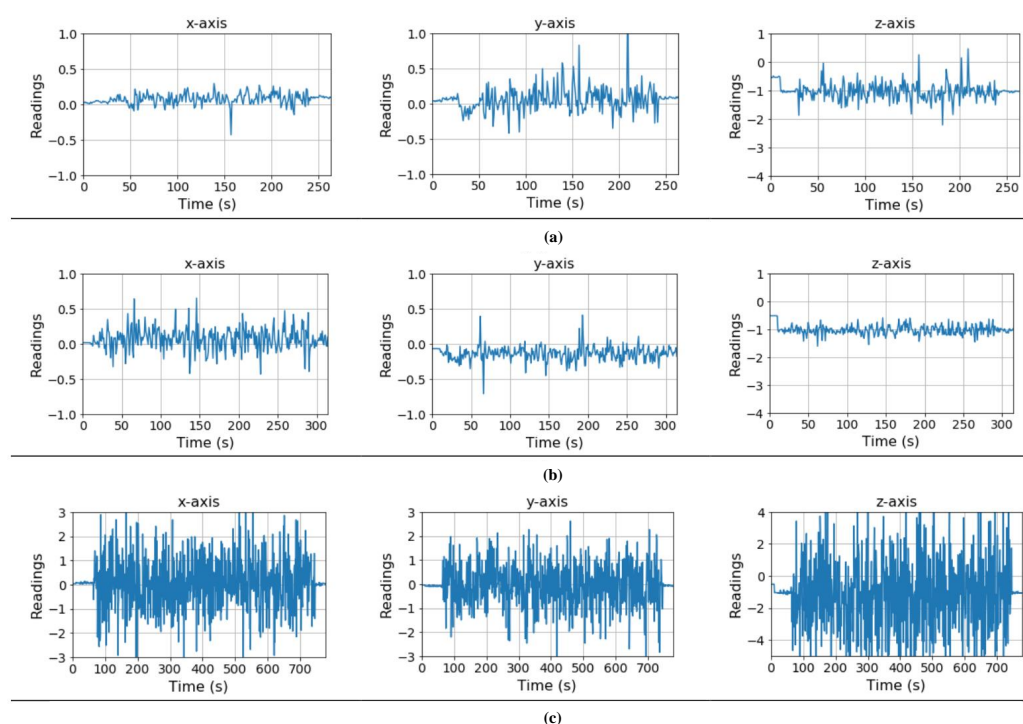
The machine learning modules are hosted on the cloud. It implements several classification models. The algorithms are run over the IoT data collected using the IoT module. The machine learning models process the received data and classify it. The classification results are outputted to the relational storage. More details about the classification models are given in Section 5.

## 4. Features Extraction

Before applying any machine learning techniques, it is important to identify the distinguishing features of the targeted machines. Our analysis shows that the tilt and vibration of machinery are distinguishing features that can help train our classification models. We record the vibration and tilt using the IoT device placed on the machinery. The IoT device contains two sensors that we will use to train our classification model, (1) 3-Axis accelerometer (2) 3-Axis gyroscope. The accelerometer is used to measure the vibrations, and the gyroscope tells us about the tilt.

We are using three types of agriculture machinery—leveler, rotavator, and cultivator. All three machines are tractor-drawn implement, with different uses and operations. The system could be scaled up and additional machinery could also be used. For now, we experiment with the above three machinery types only. We place the IoT module on these three implements and record the data. The data is recorded at the rate of one sample per second. After recording sufficient data, a subset is used to analyze and visualize the readings. The sample is visualized using the matplotlib package [40] of python.

In Figure 3, it can be observed that the data for all three types of the machinery is distinguishable in terms on vibrations in 3-axis. The cultivator and leveler have low vibrations, while rotavator is having greater vibrations. For accelerometer, the value of 1 can be represented as  $9.8 \text{ m/s}^2$  and value  $-1$  can be represented as  $-9.8 \text{ m/s}^2$  (negative sign represents opposite direction), where  $9.8 \text{ m/s}^2$  is Earth's gravity. Each reading is taken in terms of Earth's gravity. It can be observed that rotavator readings are mostly greater than 1 or less than  $-1$ , while cultivator and leveler are within the same range. The x-axis readings for the cultivator is observed to be low. However, the y-axis and z-axis readings are low for leveler compared to the cultivator.



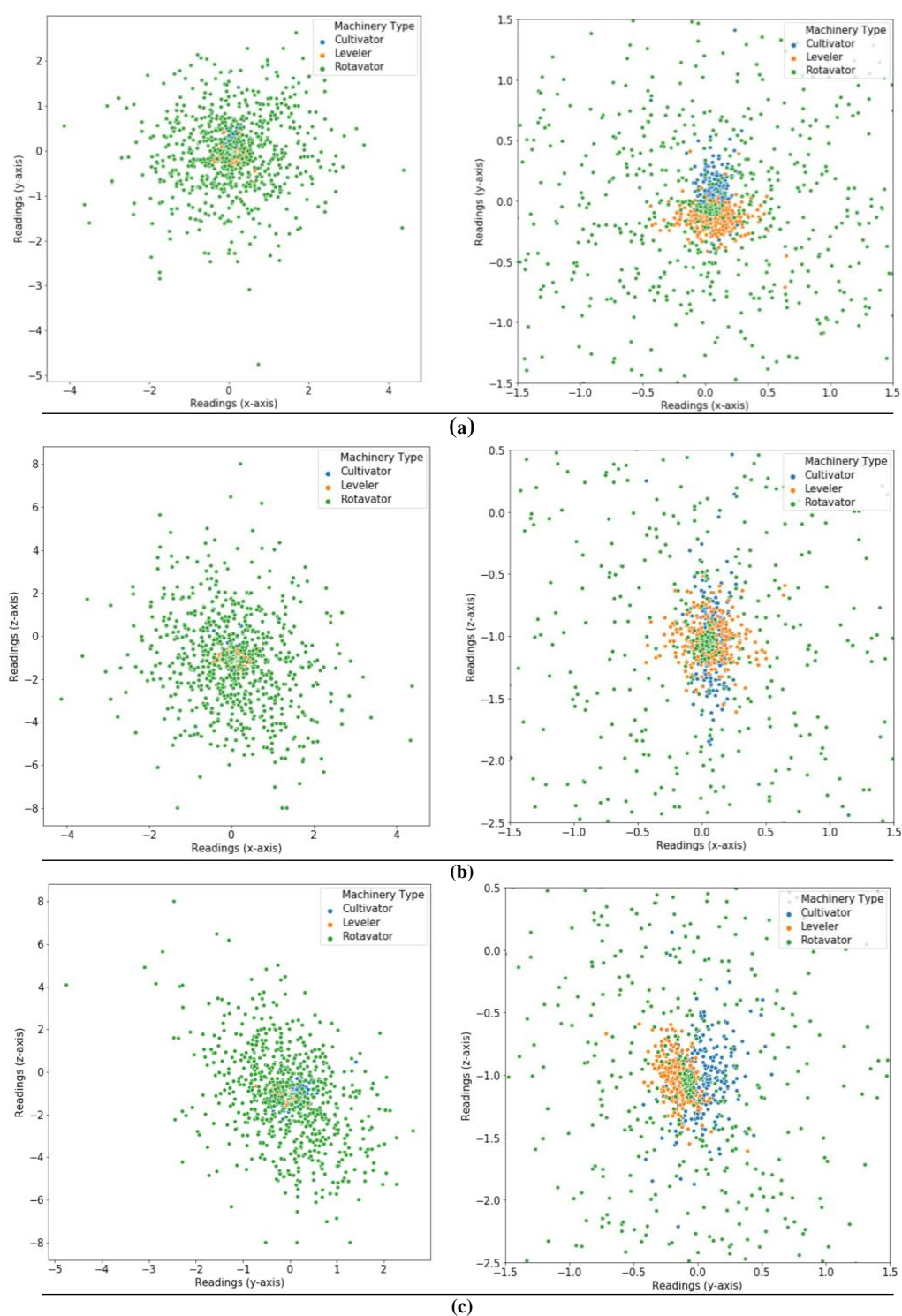
**Figure 3.** 3-axis accelerometer data visualized using the line plots. (a) Cultivator (b) Leveler, and (c) Rotavator.

We now take a look at the mean and standard deviation for the given sample of data as shown in Table 1. The mean is the average of readings, while the standard deviation tells us about the spread of data. It can be observed that there is not much of a difference in the mean for all types of machinery. However, there are variations in standard deviation. It is much higher for rotavator compared to other machines. The difference between standard deviation for cultivator and leveler are also significant. This deviation between the data is because of the different vibrations caused by the machinery. Figure 4 visualizes these data points on a scatter plot and further establishes this distinguishable relation. It can be observed that each machinery data points are distinguishable from each other. The rotavator data shows more spread, while the leveler and cultivator data points are more compact. Still, the points could be separated from each other.

The analysis shows a clear opportunity for the machine learning algorithms to classify the farm machinery based on the vibration and tilt. In Section 5, we discuss the classification algorithms that we use for the classification of agriculture farm machinery.

**Table 1.** Mean and Standard deviation values of accelerometer readings along the 3-axis.

Machinery	Readings Mean (X-axis)	Readings Standard Deviation (X-axis)	Readings Mean (Y-axis)	Readings Standard Deviation (Y-axis)	Readings Mean (Z-axis)	Readings Standard Deviation (Z-axis)
Cultivator	0.072	0.078	0.059	0.179	−1.022	0.320
Leveler	0.065	0.159	−0.139	0.104	−1.015	0.184
Rotavator	0.071	1.057	−0.070	0.851	−1.165	2.146



**Figure 4.** Three-axis accelerometer data visualized in a scatter plot for Cultivator, Rotavator, and Leveler. (a) x-axis vs y-axis (b) x-axis vs z-axis, and (c) y-axis vs z-axis.

## 5. Classification Algorithms

In this paper, we focus on five multi-class supervised classification algorithms—K-Nearest Neighbor, Support Vector Machine, Decision Tree, Random Forest and Gradient Boosting. These types of techniques can also be called algorithm adaptation techniques. Finer details about each algorithm are given as follows.

### 5.1. K-Nearest Neighbor (KNN)

K-Nearest Neighbors is a supervised machine learning algorithm [41]. The data points are trained corresponding to their class label. The point for which the class is to be predicted calculates the distance with the nearest 'K' points, where K could be any number. The distance is calculated using the Euclidean distance formula. The label for the highest number of K Nearest points is determined as the predicted class. In agriculture, the KNN algorithm was found to be very effective for the classification of different grains/grain cultivars.

### 5.2. Support Vector Machine (SVM)

Support Vector Machine is a supervised learning algorithm that classifies data based on the separator(s) [42]. The separator(s) is hyperplanes, which distinguishes the data based on the training class labels. For example, in a two-dimensional space where two data variables are present, the hyperplane could be a line dividing the plane into two parts where each class lays on either side of the line. A similar analogy is followed for higher dimensions where data points are separated by hyperplanes. Essentially, SVM is about finding hyperplanes that best separates the data classes. The predicted classes in SVM are made according to the side of the hyperplane where the data point falls. Support vector machine is a kind of structural risk minimization based learning algorithms. As a popular machine learning algorithm, SVM has been widely used in many fields, such as information retrieval and agricultural, for crop and soil classifications [22].

### 5.3. Decision Tree (DT)

Decision Tree is a supervised learning algorithm primarily used for classifications [43]. The basic intuition behind a decision tree is to map out all possible decision paths in the form of a tree. It forms a flow chart like tree structure, where each node denotes the test on attribute, and each branch represents the outcome of the test. The end-node (terminal node) holds the class label. Every time a class is to be predicted the data features pass through a certain decision path and class label is predicted at the terminal node. This technique is capable of dealing with both complete and incomplete data. And is applied into the classification problem for all kinds of agriculture datasets.

### 5.4. Random Forest (RF)

Random Forest is a supervised learning algorithm that uses ensemble learning techniques to make a strong classifier based on weak classifiers [44]. This bagging method is used to train the models, and this is responsible for an increased performance. As the name suggests random forest is a forest of decision trees, random forest builds multiple decision trees which work as a weak classifier and results of each weak classifier (decision tree) is merged together to make a strong classifier, in parallel. Random Forest Random is widely used in crop classification and has the ability to predict crop yield corresponds to the current climate and biophysical change [45].

### 5.5. Gradient Boosting (GB)

Unlike Random Forest (which is a bagging method), the Gradient Boosting is a supervised learning ensemble method that uses boosting method for training [46]. It uses weak classifiers to train in a sequential manner rather than in a parallel manner. The output of first weak classifier is given to the second weak classifier. This way, the mis-classified data points of the first are improved in the second. The practice is continued till the number of weak classifiers are made in sequence. The weak classifiers are then merged together to form a strong classifier. In gradient boosting, the loss function of the weak classifier is optimized, using a learning rate and the next classifier has a decreased loss. It is widely used in agriculture for area estimation and irrigation planning [47].

## 6. Implementation and Tuning of Models

### 6.1. Data Preparation

#### 6.1.1. Dataset

Initially, we train and test our models on a dataset of a total of 14,488 data points. The data are recorded at a frequency of one sample per second. The provided sample dataset includes 3-axis accelerometer and gyroscope data along with the label for machinery type as shown in Figure 5. Initially, we train our model for three machines only (cultivator, leveler and rotavator). However, this system could be expanded and more machines could be added.

DateAndTime	ax	ay	az	gx	gy	gz	type
27/06/2019 13:06	0.01367188	0.03320313	0.5144043	0.1373291	1.78527832	-0.0915527	rotavator
27/06/2019 13:06	0.01391602	0.03491211	0.51367188	0.22125244	1.54876709	-0.0991821	rotavator
27/06/2019 13:07	0.01660156	0.0324707	0.51074219	0.05340576	1.51062012	-0.0991821	rotavator
27/06/2019 13:07	0.01245117	0.03442383	0.50878906	0.28991699	1.5411377	0.03814697	rotavator
27/06/2019 13:07	0.01269531	0.03540039	0.51074219	0.26702881	1.62506104	0.12207031	rotavator
27/06/2019 13:07	0.01196289	0.03515625	0.50927734	0.09918213	1.52587891	-0.0076294	rotavator
27/06/2019 13:07	0.01416016	0.03222656	0.515625	0.03051758	1.54876709	-0.1831055	rotavator

**Figure 5.** Sample of data collected using IoT module showing 3-axis accelerometer (ax, ay, az) and 3-axis gyroscope (gx, gy, gz).

#### 6.1.2. Data Pre-Processing and Normalization

The data require pre-processing to make it suitable for model fitting. As the nature of the data is numerical continuous, we need to normalize each feature. This way, each feature will be equally represented (having an equal weight). The goal of normalization is to bring all features to a common scale without affecting their unique patterns [48]. The data are split into two parts—training-set and testing-set. The training-set is used for training our machine learning models, while the testing-set is used for evaluation of the trained models. As a known procedure, we split the data with a 70:30 ratio, with 70% of the data used for training, and 30% used for testing.

### 6.2. Implementation and Tuning of Models

To apply classification algorithms to our data, we used an open-source package available for python called Scikit-learn [49]. Scikit-learn is an all-in-one package for machine learning, which helps us pre-process our data, implement various machine learning algorithms and evaluate them. We implement five machine learning models and compare them in terms of accuracy and training time.

#### 6.2.1. Tuning of Models

The Hyperparameters are properties that govern the entire training process. Manually tuning hyperparameters is a time-consuming process and very hard to keep track of. It is important to remember the hyperparameters that have been tried and the one that has not been. Grid Search is an algorithm used to automate the process of finding optimal hyperparameters of certain machine learning algorithm. Grid Search is given a list of parameters, which are tested one-by-one and optimal parameters are found. Grid Search uses cross-validation technique to validate the models in terms of accuracy.

In Table 2, we have applied Grid Search using GridSearchCV function available in Sklearn. The process involves the tuning of parameters for the chosen ML algorithms. Starting with KNN, the K-nearest neighbors was tuned, and the best value was found to be 6. For SVM, the best-performing kernel RBF was selected and the hyperparameters C and Gama were tuned. Both were found to have an optimal value of 1.

For decision tree, we used the Grid Search to specify the depth of a tree. The optimal accuracy was determined at 15. For Random Forest, the number of estimators (weak

learners) were tuned, and 50 estimators showed a promising result. The gradient boosted trees had an optimal depth of 8, and the optimal learning rate was found to be 0.2.

After finding the best parameters, we evaluated each model in terms of accuracy.

**Table 2.** Tuning hyperparameters of Machine Learning models using GridSearchCV.

Algorithm	Parameters Tuned	Best Parameters
KNN	Number of Nearest Neighbors (K)	K = 6
SVM	Kernel	Kernel = Radial Basic Function
	Cost of Misclassification (C)	C = 1
	Kernel Coefficient (Gamma)	Gamma = 1
Decision Tree	Maximum Depth	Maximum Depth = 15
Random Forrest	Number of Estimators	Number of Estimators = 50
Gradient Boosting	Learning Rate	Learning Rate = 0.2
	Maximum Depth	Maximum Depth = 8

### 6.2.2. Choosing Best Classification Model

The five classification machine learning algorithms are compared in terms of accuracy and execution time when trained on the dataset mentioned in Section 6.1. Table 3 and Figure 6 provides the results, including training and testing accuracy. The figure also shows the training time for each algorithm. It can be observed that all the algorithms have a training accuracy of more than 90%. The testing accuracy of Decision Tree, SVM and KNN is less than 90%. The decision tree model is highly over fitted as the difference between the training and testing accuracy is the largest. SVM and KNN models have less over fitting compared to others, but the testing accuracy is not the best that we are getting.

The ensemble learning models, including Gradient Boost and Random Forest have a very high training accuracy. However, the results are slightly over fitted as the gap between training and testing accuracy is about 9%. Even though there is slight overfitting, both gradient boosting and random forest produce highest testing accuracy. In Table 3, the training time taken for each algorithm is also given. The Decision Tree has the lowest training time while Gradient Boosting has the highest. The reason for high training time for gradient boosting is understandable as it uses a sequential approach to train weak classifier and then make a strong classifier.

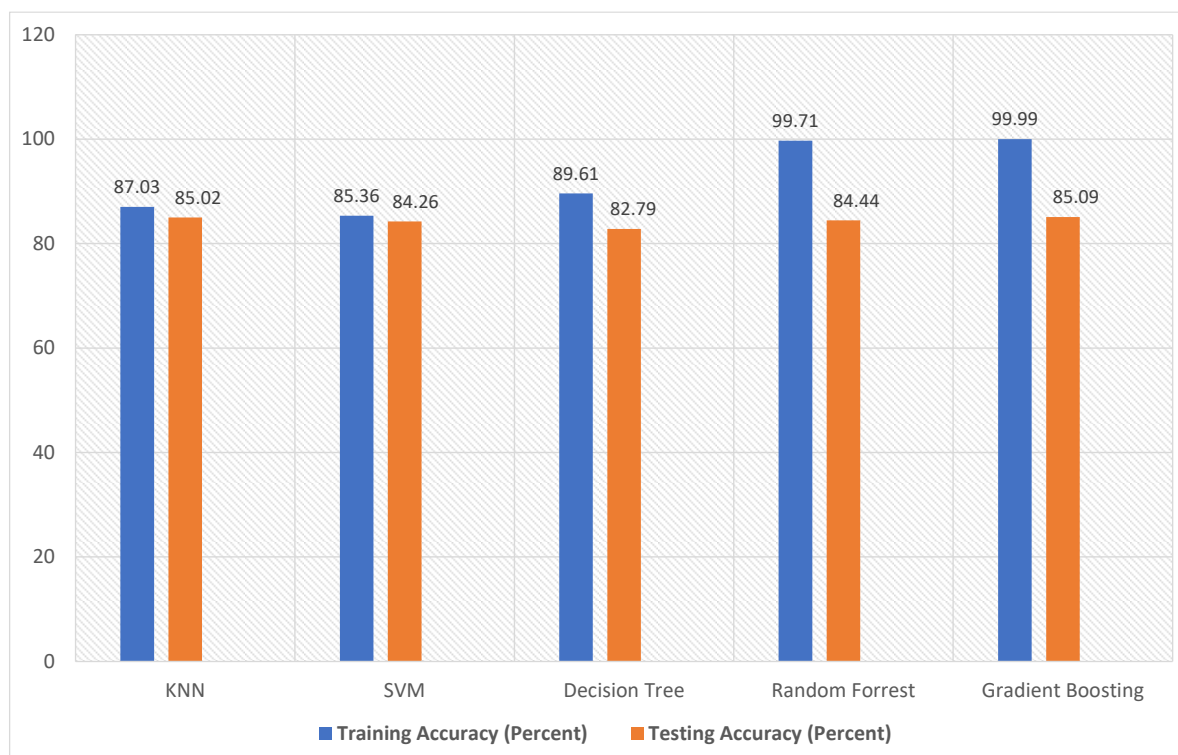
SVM has the second highest training time, and the accuracy compared to other algorithms is not very good either. KNN and Random Forest both have a reasonable training time. Keeping in view the accuracy as well as the training time, we conclude that Random Forest is the best-performing algorithm in our case. It is important to note that the current algorithm's performances are case variant. We may get different results with these algorithms trained on another dataset. Therefore, our classification model may require constant retraining with more data in hand to generalize the classification time to time.

### 6.2.3. Evaluating Important Features

The proposed research work analyzes the features of gyroscopes and accelerometers and takes a look at which feature has the most impact on the accuracy of our models. For this, we trained the algorithms on gyroscope and accelerometer data separately and then compared the results. This way, we figured out which data feature is the most important. Tables 3 and 4 provide an overview of how much accelerometer and gyroscope contribute to the machinery classification accuracy. Each sensor data is trained separately on the five algorithms. It can be observed in Figure 6, that the testing accuracy for accelerometer on all algorithms is between 82–85%. For gyroscope, shown in Figure 7, it is significantly low (between 68–72%). It is clearly visible that the 3-axis accelerometer is a more important feature for classification than the gyroscope. Though, the training time for the accelerometer is slightly higher compared to the gyroscope.

**Table 3.** Training and testing accuracy of tuned models on 3-axis accelerometer data only.

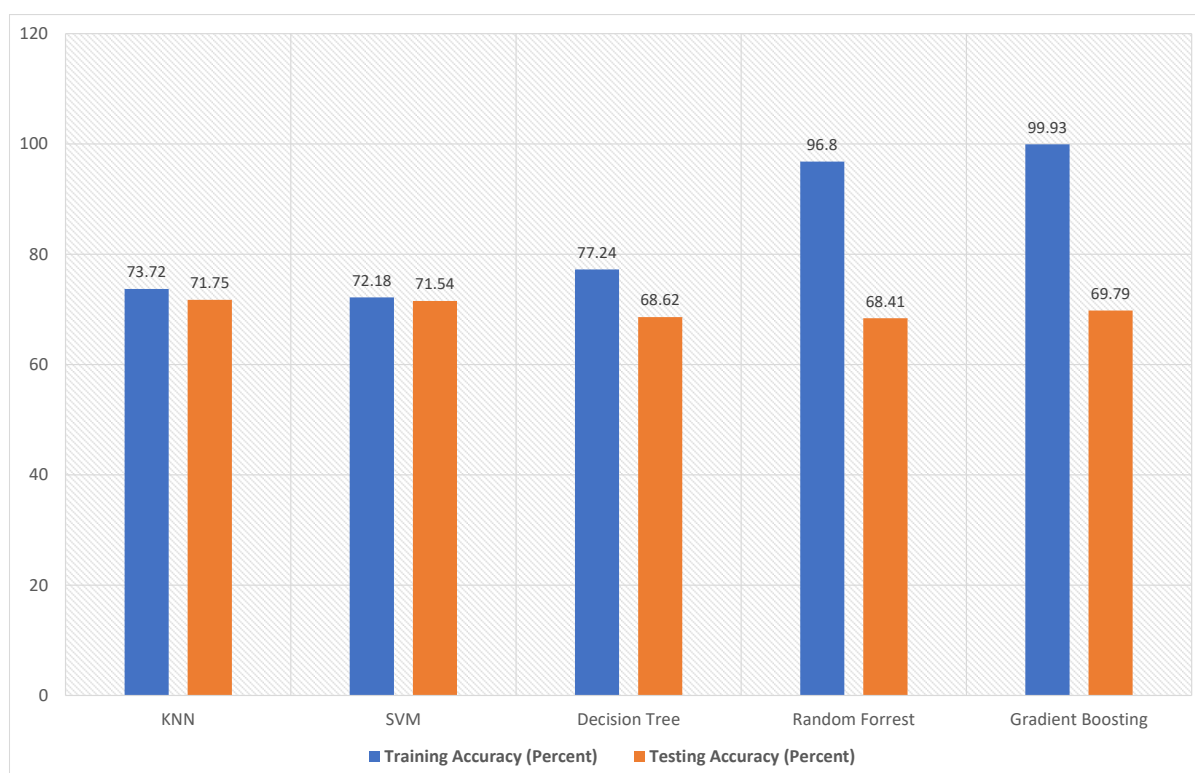
Algorithm	Training Accuracy (Percent)	Testing Accuracy (Percent)	Training Time (Seconds)
KNN	87.03	85.02	0.811
SVM	85.36	84.26	2.98
Decision Tree	89.61	82.79	0.088
Random Forrest	99.71	84.44	6.66
Gradient Boosting	99.99	85.09	0.980

**Figure 6.** Comparing algorithms in terms of training and testing accuracy trained on 3-axis accelerometer data only.

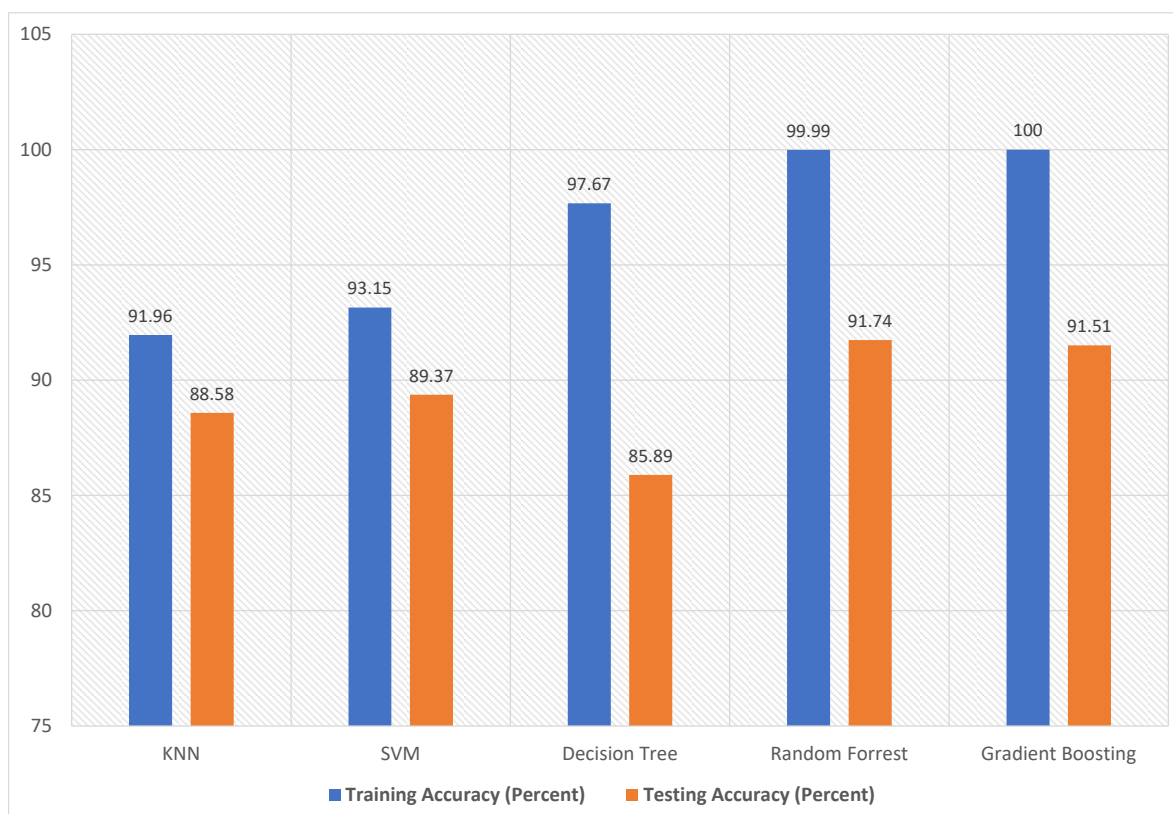
Gradient Boosting and Random Forest are both over fitted. For accelerometer, the over fitting is smaller compared to the gyroscope. The Decision Tree, SVM, and KNN do not show overfitting for both features. Table 5 and Figure 8 shows that the testing accuracy increased when gyroscope and accelerometer are used in combination (for classification). We are getting testing accuracy between 85–92%, which is very good compared to gyroscope or accelerometer alone.

**Table 4.** Training and testing accuracy of tuned models on 3-axis gyroscope data only.

Algorithm	Training Accuracy (Percent)	Testing Accuracy (Percent)	Training Time (Seconds)
KNN	73.72	71.75	0.917
SVM	72.18	71.54	5.21
Decision Tree	77.24	68.62	0.0781
Random Forrest	96.80	68.41	7.7
Gradient Boosting	99.93	69.79	1.28



**Figure 7.** Comparing algorithms in terms of training and testing accuracy trained on 3-axis gyroscope data only.



**Figure 8.** Comparing algorithms in terms of training and testing accuracy trained on 3-axis accelerometer and gyroscope data.

**Table 5.** Training and testing accuracy of tuned models on 3-axis accelerometer and 3-axis gyroscope data.

Algorithm	Training Accuracy (Percent)	Testing Accuracy (Percent)	Training Time (Seconds)
KNN	91.96	88.58	1.34
SVM	93.15	89.37	4.02
Decision Tree	97.67	85.89	0.113
Random Forrest	99.99	91.74	1.5
Gradient Boosting	100	91.51	11.7

#### 6.2.4. Evaluation on Different Data Set

To further highlight the strength of our classification model, we tested the data collected in a different agriculture field using another set of machines (leveler, rotavator and cultivator).

It can be observed in Table 6 that our classification model has made correct predictions for all six datasets. The predictions were right, regardless of the number of data-points. A low number of data-points also showed good results. However, our assumption is that a very low number of samples in excess of 100 or less may misclassify, as it might not contain enough information for the classification model. When the model starts misclassifying occasionally, we might also need to update our model by retraining it to the variations in the new datasets.

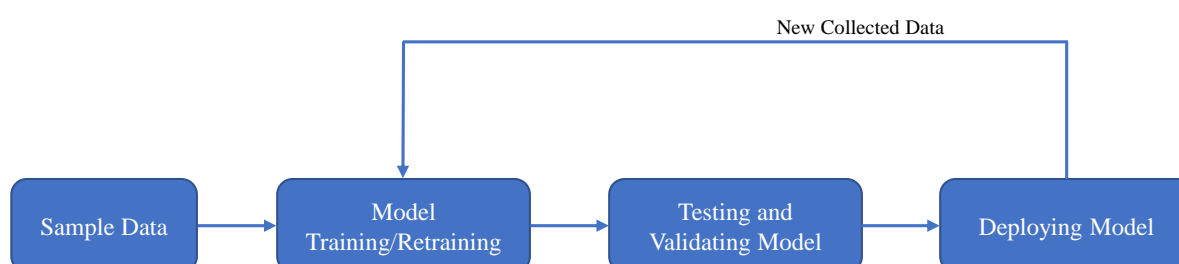
**Table 6.** Machinery classification model predictions.

Sample	Number of Data Points	Original Class	Predicted Class
Sample 1	1253	Cultivator	Cultivator
Sample 2	264	Cultivator	Cultivator
Sample 3	2076	Rotavator	Rotavator
Sample 4	779	Rotavator	Rotavator
Sample 5	1643	Leveler	Leveler
Sample 6	315	Leveler	Leveler

## 7. Deploying and Retraining Classification Model

After evaluating the classification models on a sample dataset, we figured that the Random Forest is the best performing. Now, the selected model has to be deployed on the server to make it available as a service. For this, the trained model is stored in a file. Every time a classification is required, the stored model is used, thus, saving a lot of time in training the model repeatedly. We invoke the model using an API gateway specifically implemented for this service. The gateway is backed by lambda function of Amazon Web Services (AWS).

With time, the deployed model may become outdated and its accuracy may start decreasing. This may be because it is not used to the new kind of data being given to it. For this, we need to retrain our model on some new data. The process of retraining is elaborated in Figure 9.

**Figure 9.** Model training, deployment and retraining life cycle.

## 8. Discussion and Conclusions

Many of the agriculture field operations are automated to meet the food demands of rising global population. Automated field operations are good for increasing the overall field productivity; however, it requires supervision and automated work estimation. There is a potential risk of fraudulent activities taking place in remote field supervision. It has been observed that taking advantage of remote setup or the hope of getting increased subsidy from the government or donor organization, often the wrong (or cheaper) machinery is being used. In some cases, no machinery is used at all. Given the scope of this work, the focus of discussion is on the classification of agriculture farm machinery.

The classification of farm machinery was performed using the Machine Learning algorithms. It was observed that the vibration and tilt of machinery are good indicators for recognizing a machine. The 3-axis accelerometer was used for recording the vibration and the tilt was recorded using the 3-axis gyroscope. The sensors were connected to the IoT device mounted on the farm machinery.

We applied five machine learning techniques (K-Nearest Neighbor, Support Vector Machine, Decision Tree, Random Forest, and Gradient Boosting) on the vibration and tilt of farm machinery. As was anticipated, the vibration and tilt both showed significant results in identifying the farm machinery. However, the vibration-based classification showed better accuracy (82–85%) compared to the tilt-based classification (showing an accuracy of 68–71%). Interestingly, the accuracy improves further when the vibration and tilt are used together, showing a maximum accuracy of 91%.

We found that out of the five machine learning techniques that we applied towards the model fitting, Random Forest showed the most accuracy. The Gradient boosting and Random Forest showed slight over-fitting, but both algorithms produced the highest testing accuracies. RF uses a multiple weak-classifier to build a strong classifier, and showed great promise in classification. It is widely used in crop classification and has the ability to predict crop yield corresponding to the current climate and biophysical changes. In terms of the execution time taken in training the model, we found that the Decision Tree was the quickest taking the least time to train. In contrast, the Gradient Boosting was the slowest, taking the most time to train.

In order to keep up with the accuracy and enable the new recognition of new patterns, we have developed a system for retraining. The retraining involves the inclusion of new data in redoing model-fitting.

The model is useful for the remote supervision of agricultural farm machinery and its authentication. The model alone will not be enough to avoid fraudulent activities and there is a possibility that the right machinery is used but the machine may be operational at a location that was not intended. To solve this problem, our model needs to be coupled with field recognition and area estimation algorithms.

As part of future work, we would like to apply the unsupervised machine learning techniques and deep learning and apply different implementations of it. Furthermore, it will be interesting to train our models on a larger set of machines. Additionally, exploration of other features apart from the vibration and tilt that will be worth the review. More data could be collected, and models could be retrained to further generalize the classification process.

**Author Contributions:** Conceptualization, M.W. and T.K.; methodology, M.W., T.K. and T.-W.U.; software, M.W., T.K. and S.M.U.; validation, T.K., T.-W.U. and M.W.; formal analysis, T.K. and T.-W.U.; investigation, M.W. and T.K.; resources, T.K. and T.-W.U.; data curation, M.W., S.M.U. and T.K.; writing—original draft preparation, M.W., S.M.U. and T.K.; writing—review and editing, M.W., T.K. and T.-W.U.; visualization, M.W., T.-W.U. and T.K.; project administration, T.-W.U.; and funding acquisition, T.-W.U. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00833, A study on 5G based Intelligent IoT Trust Enabler).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

- World Economic Forum. *The Future of Jobs: Employment, Skills and Workforce Strategy for the Fourth Industrial Revolution*; Global Challenge Insight Report; World Economic Forum: Geneva, Switzerland, 2016.
- Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. *Future Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [CrossRef]
- Russell, S.J.; Norvig, P. *Artificial Intelligence: A Modern Approach*; Pearson Education Limited: Malaysia, 2016.
- Hayes, B. Cloud Computing. *Commun. ACM* **2008**, *51*, 9–11. [CrossRef]
- Alston, J.M.; Pardey, P.G. Agriculture in the Global Economy. *J. Econ. Perspect.* **2014**, *28*, 121–46. [CrossRef]
- Waleed, M.; Um, T.W.; Kamal, T.; Khan, A.; Iqbal, A. Determining the Precise Work Area of Agriculture Machinery Using Internet of Things and Artificial Intelligence. *Appl. Sci.* **2020**, *10*, 3365. [CrossRef]
- Liakos, K.G.; Busato, P.; Moshou, D.; Pearson, S.; Bochtis, D. Machine Learning in Agriculture: A Review. *Sensors* **2018**, *18*, 2674. [CrossRef]
- Sharma, S.; Raval, J.; Jagyasi, B. Neural Network based Agriculture Activity Detection using Mobile Accelerometer Sensors. In Proceedings of the 2014 Annual IEEE India Conference (INDICON), Pune, India, 11–13 December 2014; IEEE: New York, NY, USA, 2014; pp. 1–5.
- Barman, U.; Choudhury, R.D. Soil Texture Classification using Multi Class Support Vector Machine. *Inf. Process. Agric.* **2020**, *7*, 318–332. [CrossRef]
- Bhavani, T.S.T.; Begum, S. Agriculture Productivity Enhancement System using IoT. *Int. J. Theor. Appl. Mech.* **2017**, *12*, 543–554.
- Paudyal, S. Classification of Rotating Machinery Fault Using Vibration Signal. Available online: <https://commons.und.edu/theses/2579/> (accessed on 12 December 2019).
- Gersch, W.; Brotherton, T.; Braun, S. Nearest Neighbor-time Series Analysis Classification of Faults in Rotating Machinery. *J. Vib. Acoust. Stress. Reliab. Des.* **1983**, *2*, 178–184. [CrossRef]
- Yu, F.; Zhou, F. Classification of Machinery Vibration Signals based on Group Sparse Representation. *J. Vibroeng.* **2016**, *18*, 1540–1554. [CrossRef]
- Garg, D.; Alam, M. Deep Learning and IoT for Agricultural Applications. In *Internet of Things (IoT)*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 273–284.
- Cutini, M.; Costa, C.; Bisaglia, C. Development of a simplified method for evaluating agricultural tractor's operator whole body vibration. *J. Terramech.* **2016**, *63*, 23–32. [CrossRef]
- Gorucu, S.; Cavallo, E.; Murphy, D. Perceptions of Tilt Angles of an Agricultural Tractor. *J. Agromed.* **2014**, *19*, 5–14. [CrossRef]
- Gómez-Chabla, R.; Real-Avilés, K.; Morán, C.; Grijalva, P.; Recalde, T. IoT Applications in Agriculture: A Systematic Literature Review. In Proceedings of the 2nd International Conference on ICTs in Agronomy and Environment, Guayaquil, Ecuador, 22–25 January 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 68–76.
- Jha, K.; Doshi, A.; Patel, P.; Shah, M. A Comprehensive Review on Automation in Agriculture using Artificial Intelligence. *Artif. Intell. Agric.* **2019**, *2*, 1–12. [CrossRef]
- Rehman, T.U.; Mahmud, M.S.; Chang, Y.K.; Jin, J.; Shin, J. Current and Future Applications of Statistical Machine Learning Algorithms for Agricultural Machine Vision Systems. *Comput. Electron. Agric.* **2018**, *156*, 585–605. [CrossRef]
- Bhargavi, P.; Jyothi, S. Applying Naive Bayes Data Mining Technique for Classification of Agricultural Land Soils. *Int. J. Comput. Sci. Netw. Secur.* **2009**, *9*, 117–122.
- Heung, B.; Ho, H.C.; Zhang, J.; Knudby, A.; Bulmer, C.E.; Schmidt, M.G. An Overview and Comparison of Machine-learning Techniques for Classification Purposes in Digital Soil Mapping. *Geoderma* **2016**, *265*, 62–77. [CrossRef]
- Shi, L.; Duan, Q.; Ma, X.; Weng, M. The Research of Support Vector Machine in Agricultural Data Classification. In *IFIP Advances in Information and Communication Technology, Proceedings of the International Conference on Computer and Computing Technologies in Agriculture, Beijing, China, 29–31 October 2011*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 370, pp. 265–269. [CrossRef]
- Li, D.; Zheng, Y.; Zhao, W. Fault Analysis System for Agricultural Machinery based on Big Data. *IEEE Access* **2019**, *7*, 99136–99151. [CrossRef]
- Singh, D.; Leavline, E.J.; Priyanka, V.; Swathi, V. Agriculture Classification System using Differential Evolution Algorithm. *Int. Adv. Res. J. Sci. Eng. Technol.* **2016**, *3*, 24–28. [CrossRef]
- Enrong, Z.B.W.M.M.; Zhenghe, Z.X.S. Recognition and Classification for Vision Navigation Application Environment of Agricultural Vehicle. Available online: [https://en.cnki.com.cn/Article\\_en/CJFDTOTAL-NYJX200907035.htm](https://en.cnki.com.cn/Article_en/CJFDTOTAL-NYJX200907035.htm) (accessed on 8 December 2020).
- Jin, X.b.; Yu, X.H.; Wang, X.Y.; Bai, Y.T.; Su, T.L.; Kong, J.L. Deep Learning Predictor for Sustainable Precision Agriculture Based on Internet of Things System. *Sustainability* **2020**, *12*, 1433. [CrossRef]

27. Symeonaki, E.; Arvanitis, K.; Piromalis, D. A Context-Aware Middleware Cloud Approach for Integrating Precision Farming Facilities into the IoT toward Agriculture 4.0. *Appl. Sci.* **2020**, *10*, 813. [\[CrossRef\]](#)
28. García, L.; Parra, L.; Jimenez, J.M.; Lloret, J.; Lorenz, P. IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture. *Sensors* **2020**, *20*, 1042. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Shoaib, F.; Umer, R.A.A.; Yousaf, Z. Role of IoT Technology in Agriculture: A Systematic Literature Review. *Electronics* **2020**, *9*, 319. [\[CrossRef\]](#)
30. Lloret, J.; Tomas, J.; Canovas, A.; Parra, L. An Integrated IoT Architecture for Smart Metering. *IEEE Commun. Mag.* **2016**, *54*, 50–57. [\[CrossRef\]](#)
31. Shah, S.H.; Yaqoob, I. A survey: Internet of Things (IOT) technologies, applications and challenges. In Proceedings of the IEEE Smart Energy Grid Engineering (SEGE), Oshawa, ON, Canada, 21–24 August 2016; pp. 381–385. [\[CrossRef\]](#)
32. Ruiz Real, J.; Uribe-Toril, J.; Antonio, J.; Torres, J.; De Pablo, J. A Look at the Past, Present and Future Research Trends of Artificial Intelligence in Agriculture. *Agronomy* **2020**, *10*, 1839. [\[CrossRef\]](#)
33. Shamshiri, R.; Ismail, W.I.W. Exploring GPS Data for Operational Analysis of Farm Machinery. *Res. J. Appl. Sci. Eng. Technol.* **2013**, *5*, 3281–3286. [\[CrossRef\]](#)
34. Sopegno, A.; Calvo, A.; Berruto, R.; Busato, P.; Bocthis, D. A Web Mobile Application for Agricultural Machinery Cost Analysis. *Comput. Electron. Agric.* **2016**, *130*, 158–168. [\[CrossRef\]](#)
35. Prathibha, S.; Hongal, A.; Jyothi, M. IoT based Monitoring System in Smart Agriculture. In Proceedings of the 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT), Bangalore, India, 16–17 March 2017; IEEE: New York, NY, USA, 2017; pp. 81–84. [\[CrossRef\]](#)
36. Zhang, R.; Hao, F.; Sun, X. The Design of Agricultural Machinery Service Management System based on Internet of Things. *Procedia Comput. Sci.* **2017**, *107*, 53–57. [\[CrossRef\]](#)
37. Cloud Computing with AWS. Available online: <https://aws.amazon.com/> (accessed on 20 July 2020).
38. Brantner, M.; Florescu, D.; Graf, D.; Kossmann, D.; Kraska, T. Building a Database on S3. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008; ACM: New York, NY, USA, 2008; pp. 251–264. [\[CrossRef\]](#)
39. Sivasubramanian, S. Amazon dynamoDB: A Seamlessly Scalable Non-relational Database Service. In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, Scottsdale, AZ, USA, 20–24 May 2012; ACM: New York, NY, USA, 2012; pp. 729–730. [\[CrossRef\]](#)
40. Hunter, J.D. Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [\[CrossRef\]](#)
41. Guo, G.; Wang, H.; Bell, D.; Bi, Y.; Greer, K. KNN Model-based Approach in Classification. In *OTM Confederated International Conferences "on the Move to Meaningful Internet Systems"*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 986–996. [\[CrossRef\]](#)
42. Hsu, C.W.; Chang, C.C.; Lin, C.J. A Practical Guide to Support Vector Classification. Available online: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.224.4115> (accessed on 22 December 2020).
43. Dai, Q.Y.; Zhang, C.P.; Wu, H. Research of Decision Tree Classification Algorithm in Data Mining. *Int. J. Database Theory Appl.* **2016**, *9*, 1–8. [\[CrossRef\]](#)
44. Ali, J.; Khan, R.; Ahmad, N.; Maqsood, I. Random Forests and Decision Trees. *Int. J. Comput. Sci. Issues (IJCSI)* **2012**, *9*, 272.
45. Ok, A.; Akar, O.; Gungor, O. Evaluation of random forest method for agricultural crop classification. *Eur. J. Remote Sens.* **2012**, *45*, 421–432. [\[CrossRef\]](#)
46. Amari, S. The Handbook of Brain Theory and Neural Networks. Available online: <https://mitpress.mit.edu/books/handbook-brain-theory-and-neural-networks> (accessed on 25 November 2020).
47. Ponraj, A.; Vigneswaran, T. Daily evapotranspiration prediction using gradient boost regression model for irrigation planning. *J. Super Comput.* **2020**, *76*. [\[CrossRef\]](#)
48. Kotsiantis, S.; Kanellopoulos, D.; Pintelas, P. Data Preprocessing for Supervised Learning. *Int. J. Comput. Sci.* **2006**, *1*, 111–117.
49. Varoquaux, G.; Buitinck, L.; Louppe, G.; Grisel, O.; Pedregosa, F.; Mueller, A. Scikit-learn: Machine Learning without Learning the Machinery. *GetMobile: Mob. Comput. Commun.* **2015**, *19*, 29–33. [\[CrossRef\]](#)