MDPI

Article

Predicting Operational Events in Mechanized Weed Control Operations by Offline Multi-Modal Data and Machine Learning Provides Highly Accurate Classification in Time Domain

Stelian Alexandru Borz [1] and **Andrea Rosario Proto** [2,*]

1   Department of Forest Engineering, Forest Management Planning and Terrestrial Measurements, Faculty of Silviculture and Forest Engineering, Transilvania University of Brasov, Șirul Beethoven 1, 500123 Brasov, Romania; stelian.borz@unitbv.ro
2   Department of AGRARIA, University of Reggio Calabria, Feo di Vito snc, 89122 Reggio Calabria, Italy
*   Correspondence: andrea.proto@unirc.it

Abstract: Monitoring of operations has become a critical activity in forestry, aiming to provide the data required by planning and production management. Conventional methods, on the other hand, come at a high expense of resources. A neural network was trained, validated, and tested in this study based on multi-modal data to classify relevant operational events in mechanized weed control operations. The architecture of a neural network was tuned in terms of the number of hidden layers and neurons, and the regularization term was set at various values to obtain optimally tuned models for three data modalities: triaxial acceleration data coupled with speed extracted from GNSS signals (AS), triaxial acceleration (A), and speed alone (S). In the training and validation phase, the models based on AS and A achieved a very high classification accuracy, accounting for 92 to 93% when considering four relevant events. In the testing phase, which was run on unseen data, the classification accuracy reached figures of 91 to 92%, indicating a good generalization ability of the models. The results point out that multimodal data are able to provide the features for distinguishing events and add spatial context to the monitored operations, standing as a suitable solution for offline, partly automated monitoring. Future studies are required to see how the capabilities of online, real-time technologies such as deep learning coupled with computer vision can add more context and improve classification performance.

Keywords: forest operations; tasks; clear-cutting; poplar; rotary tiller; performance

1. Introduction

The management of forests for timber production requires a series of operations aimed at establishing, tending, and harvesting the wood [1,2]. Poplar forests are highly valued worldwide, mainly due to their shorter production cycles, fast growing capacity, and the high-quality wood they provide [3,4], whereas poplar wood has several well-established and emergent uses in industries such as plywood and panel production, replacement of plastic as a material, outdoor construction applications, agri-food, medical, and packaging industries [5]. However, the management of poplar forests typically requires a higher number of operations, compared to conventional forestry [6]. Young poplar forests, for instance, require weed control operations to enhance the ability of trees to compete with the surrounding vegetation [3]. Typically, such operations are implemented several times in the youth stages of the poplar forest stands.

Modern weed control operations are supported by a full mechanization, although some plot-level smaller areas may require additional manual work [7]. For mechanical weed control, commonly used equipment comprises a farm tractor fitted with active devices such as plows or rotary tillers [8]. Of these, rotary tillers have been increasingly used in Romania, due to their capability to comminute and mix the weeds with the soil.

On the other hand, forest managers charged with weed control operations often lack a decision support system that can estimate the time needed for operations, provide the spatial context for planning, allocate resources such as money and fuel, and provide updates on the level of operational completeness. As a consequence, the planning and monitoring of operations is usually performed using conventional methods and protocols, which require manual inputs of data, incur a low level of data processing automation, and necessitate extensive paperwork.

The developments in machine learning for forest operations activity recognition has provided new ways to collect, process, and interpret the data, by removing the human effort and expertise required to a large extent [9,10]. These advancements are welcome as forest management is increasingly focusing its resources on production and activities that directly contribute to income. However, this brings a lack of well-trained personnel that need to make an effort to develop statistics and models to support the planning and implementation of the operations. Moreover, the improvements in operational equipment, as well as the implementation of different technologies in new areas [11,12], are progressing at an unprecedented rate. This makes it very difficult to keep the pace with studies that evaluate the operational performance, whereas the knowledge provided by the existing studies has only a limited applicability in new contexts [13]. Commonly used algorithms such as neural networks, random forests, and support vector machines are now available for implementation in several freeware software platforms [14–16]. Such algorithms may be trained to work with multi-modal data [17], and significant progress has been made in terms of using these data in recognizing activities with high accuracy and adding temporal and spatial context to classification problems, which are essential for operational planning, implementation, and monitoring [18,19].

Since farm tractors enable multifunctionality, and they are a common choice due to cost affordability [20–22]; developing and using systems able to collect, pre-process, and transfer the information for decision making is less likely to happen for these machines. Developing such systems could provide contextualized operational classifications in the time and space domains but will also increase the machine cost. In other words, algorithms that may correctly detect the tasks forming an operation with a given active device such as a plow could be less effective, for instance, when using a rotary tiller, mainly due to the mechanical behavior of the two. In addition, farm tractors are typically used for a long time with the aim of recovering the investments in machines, and many owners prefer to keep them in use at the expense of higher maintenance and repair costs, which means that an important share of the fleet is still dominated by old machines. Commonly, these machines are not equipped with devices able to collect useful data on their state such as CAN-BUS [23], and among the possible solutions for activity recognition and monitoring are those making use of external sensors.

On the other hand, the task of collecting and classifying time study data is typically transferred to the science of forest operations, since there is a lack of qualified personnel to undertake such jobs. Performing it conventionally, this activity is prone to error [24] and uses up important resources [25]. As a consequence, at least in the short term, a system able to collect and process data will be limited to offline applications able to prove concepts and, eventually, to produce data and statistics that support a full-scale, online deployment. To this end, conventional machine leaning models can be used in an offline approach to make accurate and contextualized predictions on relevant operational events based on multi-modal data such as that combining GNSS, which is able to contextualize location and movement parameters, and accelerometer data, which are able to document, by magnitude, specific events showing contrasting patterns in the time domain [26–28].

The goal of this study was to check whether the combined use of GNSS and triaxial accelerometer signals can provide a suitable input for an accurate neural network classification of relevant operational events in the time domain for a rotary tiller-based weed control operation. Two objectives were set for this study: (i) to run tests by trial-and-error, aiming to see which set of signals (modality) works best in producing an overall highly

accurate classification, which was performed by iteratively training and validating the neural network on various sets of input signals while varying its architecture and tuning its parameters; and (ii) to test the best performing neural network models on unseen data, so as to check their generalization ability, which was achieved by saving the best performing model of each set of input signals and feeding it with unseen data.

2. Materials and Methods

2.1. Data Sourcing and Processing

The data required by this study were collected in Romania in June of 2018. In this country, most of the poplar forests are concentrated along the Danube River [29], and their management involves operations such as preparing the soil, planting, cultivation and weed management, thinning, and final harvesting by clearcutting [30,31]. The field study was carried out in the forests managed by Poiana Mare Forest District, which is under the administration of Dolj Forest Directorate of the National Forest Administration–RNP Romsilva (Figure 1). According to the Romanian management of poplar forests, final harvesting is performed at an age of 25–30 years by clear-cutting implemented in areas that do not exceed 5 hectares. Once harvested, these areas require stump removal, soil preparation, planting, which currently is achieved partly mechanized [32], and, at given time intervals, weed control operations, which can be implemented manually, mechanically, or as a combination of both [7,30]. The decision on the degree of mechanization in weed control operations depends on the size of the plots, availability of machines, and manual labor requirements. In Romania, the poplar stands are harvested by clear cutting at the end of rotation, and the harvested plots have a limited size of up to 5 hectares.
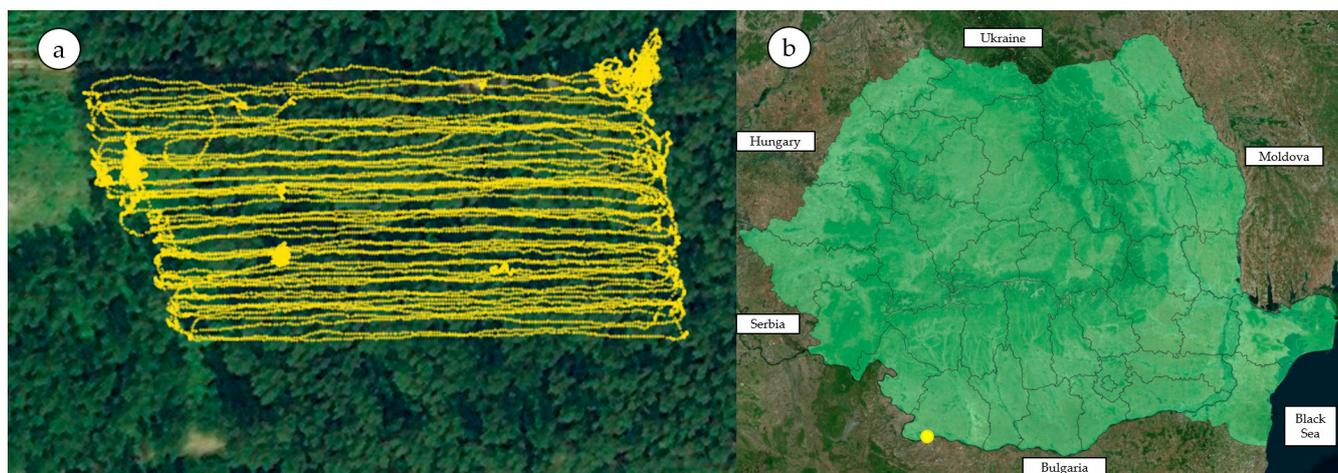
Figure 1. Study location: (**a**) the plot taken into study showing the GPS locations (yellow) collected during the field observation; (**b**) location of the study area at national level (yellow dot). Note: the maps were created in open source QGis software (QGis 3.4.12) by using freely available vector datasets of the country and the open access Bing ® aerial data as accessible at the study time.

A farm tractor equipped with a rotary tiller was taken into study, as shown in Figure 2, since this equipment configuration is being increasingly used in the country, mainly due to the operational performance and quality of the work. The tractor was a Romanian-made U650 model (UTB—Uzina Tractorul Brașov, Brașov, Romania) manufactured in 2004, equipped with a diesel engine and featuring an engine output of 47.8 kW. A Maschio Gaspardo W 145 (Maschio Gaspardo S.p.A., Campodarsego, Italy) rotary tiller was attached to the tractor using the power take-off. It was manufactured in 2016, and designed to operate on strips with a width of 145 cm at a depth of 18 cm. Its operation is sustained by a power range of 18–30 kW.

Figure 2. Configuration of the equipment taken into study: (**a**) the farm tractor equipped with the rotary tiller showing the location of the handheld GNSS receiver (1); (**b**) the rotary tiller observed in the study, (**c**) placement of the triaxial accelerometer (2) on the rotary tiller.

The data collectors used were a GNSS handheld unit which was placed on the tractor cab and set to collect data at a rate of one second, a triaxial accelerometer which was placed on the rotary tiller and set to collect data at a rate of one second, and a miniaturized camera that was placed in the cab with the field of view oriented towards the rotary tiller so as to also see the activity of the driver, and set to collect video data continuously. The technical features of the equipment used to collect the data, as well as the protocols used to collect, pair, and process data are similar to those described in [26,27,32].

At the office, GNSS and accelerometer data were paired in a Microsoft Excel sheet based on the time labels collected by both devices. Since both devices provided samples at a rate of one second along with their corresponding time labels standing for the date, hour, minute, and second of sampling, these data were used to pair the samples collected by the two devices. Then, video data were carefully analyzed to document the database by the main operational events. The observed work, which is typical to weed control operations, includes the advancement of the farm tractor equipped with the tiller on the space between two tree rows. After arriving at the plot's headland, maneuvers are taken to exit a given operated strip and to enter on a new one. Besides these types of events, other maneuvers can be taken in the plot, as well as full stops with the engine of the tractor running without power transmission to the tiller, or full stops with the tractor's engine off.

Based on the video data visualization, the work was divided into relevant operational events. These included (Table 1): (i) the equipment completely stopped with the engine turned off, which is typical to various delays, equipment preparation for work, meal time and other breaks, (ii) the tractor with the engine on, but with no movement of the equipment, which is typical to short breaks taken in the plot or at the headland, (iii) maneuvering, which is required at the headlands to exit an operated strip and to enter on a new one, or in the plot to pass over various obstacles and to change the row of operation, as well as (iv) active work, during which the tractor moves while powering the tiller which operates the soil. As observed in the field and in the video files, the maneuvers were taken with

disengagement of the power provided to the tiller while the latter was lifted from the ground when maneuvering.

Table 1. Events observed and documented based on the analysis of video data.

Event	Abbreviation Used for Annotation	Description
Active work	WORK	The tractor moves and powers the tiller which is used to remove the weed and till the soil. Conventionally, tiller disengagement was included in this category.
Maneuvering	MAN	Includes the exit and entering maneuvers, maneuvers taken in the plot, and very short stops (couple of seconds) in the plot. Tiller was disengaged during these events.
No movement, engine on	STOP	The tractor is stopped with the engine running for longer periods of time.
No movement, engine off	OFF	The tractor is stopped with the engine turned off, typically at the headland and at the beginning and end of the working day.

According to the time consumption classification in forest operations [19], the events described in Table 1 stand, by cumulation in a given day, for the gross time consumption. This time category along with those operated during that time may provide important figures on the gross productivity. Taken together, active work and maneuvering provide a good approximation of the net time consumption which, used along with the operated area, may provide estimates on the net productivity. The time spent by the equipment in events with no movement can be used as a good approximation of other categories of time consumption which are relevant to understanding the operational performance, such as repair and maintenance time, meal and other breaks time, and delays, respectively. However, to plan the operations, science and practice are typically interested in the net time, since it is used to develop piece-rate systems used to evaluate and pay for the work being performed [33], whereas accurately documenting different kind of delays typically requires extensive field data [34].

In total, the collected dataset had 15,688 records taken at a rate of one second. Before data annotation, and based on the analysis of video files, 1092 records were removed since they covered the time used to prepare the study. Although this time could be documented separately by conventional studies [18], it is rather irrelevant when monitoring the operations. The annotated dataset used for machine learning contained a number of 14,595 records.

2.2. Prediction of Operational Events by Machine Learning

The annotated dataset was arbitrarily divided in two parts, using the first 10,003 records to support the first objective of this study (hereafter called dataset 1–DS1), and using the last 4592 records to support the second objective of this study (hereafter called dataset 2–DS2). The division of the initial dataset was conducted whilst having in mind that each dataset should contain data for all of the documented events, as well as the fact that a given sequence of records belonging to a given event in the time frame should not be split and allocated to different datasets, this way preserving the original pattern in data.

GNSS data provide several attributes such as the location, sampling rate, heading, speed, and sampling date and time. These were extracted from GPX files with the use of Garmin BaseCamp (Version 4.7.0) software, in a way which was similar to [27,28]. For activity recognition and monitoring, besides the location, movement speed is important to distinguish between given operational events. As such, this feature was used as an input signal for machine learning, and was called S (km/h). Triaxial accelerometer data provide readings on the three axes (x, y and z), as well as an aggregated measure called the vector magnitude, which is computed from the triaxial measurements, as generically described in Equation (1). An important feature of vector magnitude is that it accommodates the effects of a sensor's orientation; therefore, it virtually removes the inconvenience related to the

location at which the sensor is placed on different machines, or on the same machine on different days. Vector magnitude was used as an input signal for machine learning, and it was called A (g). In total, three approaches were possible to support the first objective of the study: using the A and S data as features taken together (hereafter called modality AS), using solely A data (hereafter called modality A), or using S data (hereafter called modality S).

$$A = \sqrt[2]{x^2 + y^2 + z^2} \tag{1}$$

where A is the vector magnitude, and x, y, and z are magnitudes measured on the x, y, and z axes, respectively.

Three main steps were considered in this study, which were adapted to the common language used in machine learning. Accordingly, training and validation are typically the steps used to check the performance of models before feeding them with unseen data, whereas later, testing is conducted to check for their generalization ability. As a first step, the three modalities were checked for accuracy by training and validation on DS1. For that, the functionalities of the Orange Visual Programming [14] (Version 3.31.1) software were used. The workflow used to test and validate the best modalities in terms of classification performance is described in Figure 3. The neural network functionalities supported by the used software can be configured in terms of architecture and tuned for hyperparameters. For each modality, the architecture was configured so as to have 1, 5, and 10 hidden layers. For each option in terms of the number of hidden layers, the number of neurons was set at 10, 100, and 1000, respectively. The Rectified Linear Unit (ReLU) as well as the stochastic gradient descent optimizer (ADAM) were used for all these options as a standard activation function and solver, respectively, due to their performance and popularity in machine learning [35–37]. The number of iterations was set to the maximum enabled by the software (1,000,000) and, for each option the regularization term was tuned at α = 0.0001, 0.001, 0.01, 0.1, 1, and 10. By the number of modalities, number of hidden layers, number of neurons per layer and regularization term values, a total of 162 models were trained and validated (54 models for each modality).
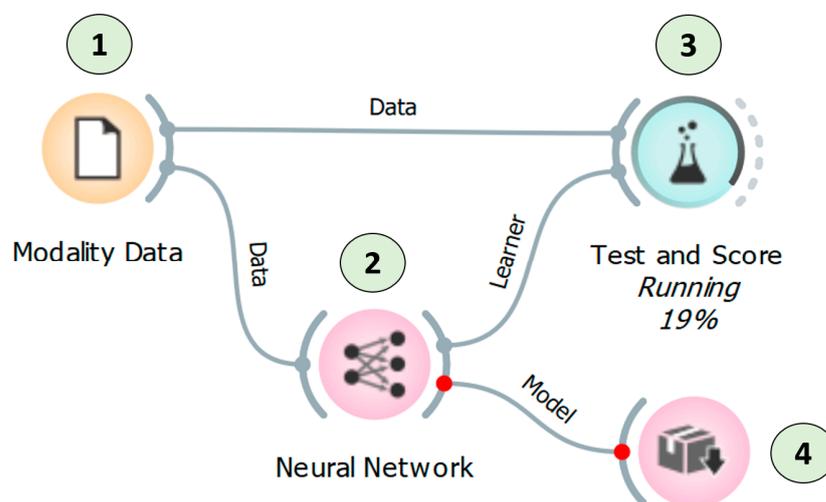
Figure 3. The workflow used to train, validate the models, and to save the best performing model in Orange Visual Programming. Legend: 1—the "Data" widget enables loading data and setting the feature and target variables, 2—the "Neural Network" widget enables the configuration of the network architecture and hyperparameter tuning each time a model is trained and validated, 3—the "Test and Score" widget enables the selection of the training and validation options as well as the extraction of classification performance metrics once the process is complete, and 4—the "Save Model" widget enables model saving. Note: the software enables the communication of various kinds of information between the widgets by connecting them.

To select the best model, the performance of classification was evaluated based on the overall classification accuracy (CA). Using the "Test and Score" widget, the software enables the calculation of the commonly used metrics for classification performance, most of which are described, for instance, in [38,39]. They include the classification accuracy (CA), precision (PREC), recall (REC), and cross-entropy (LOSS). In addition, the software enables the computation of training and validation time. Classification accuracy was used as the main metric to characterize the classification performance because it indicates the share of correctly predicted instances of the total dataset [39]. In the steps of testing and validation, the overall classification accuracy was used as a criterion to select the best performing models.

To supplementarily document the overall classification performance, training (TT) and validation (VT) time, the area under the ROC curve (AUC), classification accuracy (CA), F1 score (F1), precision (PREC), recall (REC), specificity (SPEC), and cross-entropy (LOSS) were computed and reported for each model as well. The steps of training and validation were performed by a stratified cross-validation using 10 folds. Cross-validation is a technique used to prevent overfitting by dividing the dataset into a number of folds, of which one is retained for validating the results trained on the rest; the procedure is repeated multiple times by using a different fold for validation each time, and then the results are averaged to produce more robust figures on a model's performance [40]. Training models by cross-validation is also a good strategy to deal with datasets that are limited in size [41].

Procedurally, all 162 models were first trained and validated. Based on the figures on classification accuracy (CA), a final model was retained and saved for each modality. Then, these three models were used to make predictions on unseen, although annotated, data. The workflow used for making predictions on unseen data is described in Figure 4." The workflow interconnects the data loaded by a "Data" widget with a given model loaded by the "Load Model widget and with a "Predictions" widget, which pairs the true classes of unseen data with those predicted by the model and returns the main metrics of classification performance. In addition, the workflow supports the computation of a confusion matrix, which may be used to map the absolute or relative frequency of actual instances against the predicted ones.
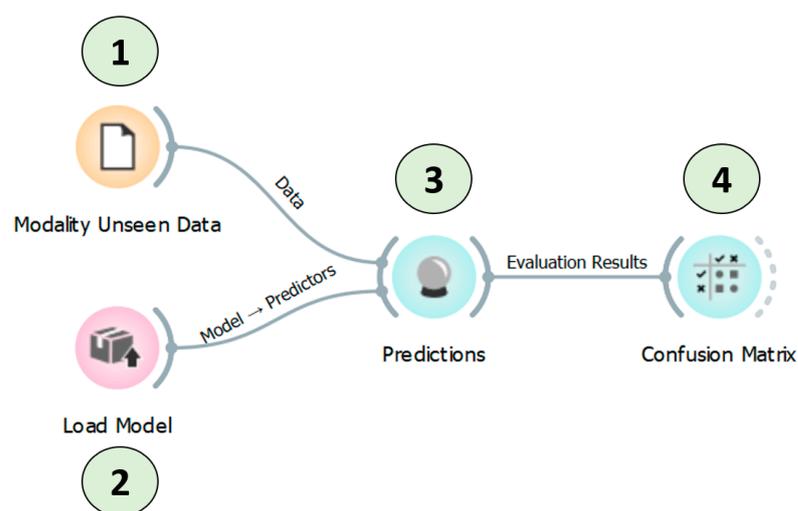
Figure 4. The workflow used to make predictions on unseen data by the saved models. Legend: 1—a "Data" widget used to load data and set the feature and target variables for unseen data, 2—the "Load Model" widget used to successively load the best performing models saved for AS, A and S modalities, 3—the "Predictions" widget used to make predictions on classification performance for unseen data, and 4—the "Confusion Matrix" widget used to show misclassifications. Note: the software enables the communication of various kinds of information between the widgets by connecting them.

2.3. Computer Architecture and Data Visualization

All the tasks involving the training and validation of the models were carried out on a Dell Alienware 17 R3 machine equipped with an Intel © Core ™ i7–6700HQ CPU, featuring 2.60 GHz, 4 cores, and 8 Logical Processors, with an installed physical memory of 16 GB, under a Microsoft Windows 10 Home edition operating system (Microsoft, Redmond, WA, USA). Effects brought on by the modality, network architecture, and hyperparameters were visually assessed using the "Scatter Plot" widget connected to a "Data" widget which was used to load the necessary information from a Microsoft Excel file. The steps of pairing the GNSS and accelerometer data, as well as coding of data according to the events seen in the video files, were supported by Microsoft Excel.

3. Results

3.1. Training and Validation

The main results characterizing the classification performance during the training and validation of the models are included for the three modalities in Table 2, where the optimal models are described in terms of architecture and hyperparameters. Using AS and A modalities to train and validate the models provided similar classification performances. For these two modalities, the overall classification accuracy (CA, OVERALL, Table 2) reached values of 92.8 (modality A) to 93% (modality AS), indicating the benefits of using the acceleration as data input for classification. When using both acceleration and GNSS speed, the classification accuracy improved for maneuvering (MAN, 98.2%), stop (STOP, 94.2%), and off (OFF, 97%) events, emphasizing the effect of class imbalance in DS1, where "Work" events accounted for 7261 observations (73% of DS1), as well as the increased power of the A signal to discriminate the events. In the same dataset (DS1), maneuvering (MAN, 2046, 20%), stop (STOP, 299, 3%), and off (OFF, 397, 4%) events accounted for less than 25%. Overall, the classification accuracy for modality A was lower, accounting for 92.8%. However, "WORK" and "STOP" events were better classified, accounting for classification accuracies of 97.8 and 97%, respectively. For these models, the multi-class cross-entropy (LOSS, Table 2) had values of 0.19 and 0.23, respectively, pointing out that working with AS modality may provide lower errors during generalization.

Table 2. Summary of the best models in terms of classification performance.

Modality and Dataset	Optimal Architecture and Hyperparameters	Event Class	AUC	CA	F1	PREC	REC	SPEC	LOSS
AS, DS1	N = 1, n = 1000, α = 0.001, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.986	0.930	0.917	0.906	0.930	0.964	0.191
		WORK	0.986	0.930	0.917	0.906	0.930	0.964	0.191
		MAN	0.998	0.982	0.987	0.987	0.987	0.966	0.048
		STOP	0.984	0.942	0.868	0.816	0.927	0.946	0.131
		OFF	0.939	0.970	0.000	0.000	0.000	1.000	0.084
A, DS1	N = 1, n = 100, α = 0.001, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.972	0.928	0.914	0.902	0.928	0.952	0.227
		WORK	0.985	0.978	0.985	0.981	0.989	0.950	0.080
		MAN	0.972	0.942	0.867	0.819	0.921	0.948	0.154
		STOP	0.924	0.970	0.000	0.000	0.000	1.000	0.088
		OFF	0.948	0.965	0.558	0.566	0.549	0.983	0.086
S, DS1	N = 5, n = 1000, α = 1, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.886	0.848	0.819	0.792	0.848	0.800	0.463
		WORK	0.918	0.911	0.941	0.913	0.971	0.755	0.268
		MAN	0.853	0.855	0.665	0.630	0.703	0.894	0.343
		STOP	0.885	0.970	0.000	0.000	0.000	1.000	0.099
		OFF	0.878	0.960	0.000	0.000	0.000	1.000	0.120

Note: N—number of hidden layers, n—number of neurons per hidden layer, α—regularization parameter, AUC—area under the receiver operating curve, CA—classification accuracy, F1—F1 score, PREC—precision, REC—recall, SPEC—specificity, LOSS—cross entropy.

Using the S data alone led to an important drop in classification accuracy (OVERALL, CA = 84.8%, Table 2), as well as to a high increment in multi-class cross-entropy (OVERALL,

LOSS = 0.463, Table 2), indicating a limited ability of the GNSS speed alone to accurately classify the events taken into study. At the event class level, the classification accuracy was better for events that did not involve a machine's movement such as "STOP" (CA = 97%, Table 2) and "OFF" (CA = 96%, Table 2), indicating that lower ranges of GNSS speed may accurately predict such events. "WORK" events were still highly accurately classified (CA = 91.1%, Table 2), maybe due to a differentiation in the magnitude of speed, but the overall result in classification accuracy may be the consequence of "MAN" events being misclassified based on speed, since this event class had a classification accuracy of 85.5%.

There were differences in the classification performance of the trained and validated models, as shown in Tables A1–A3. Training and validation time also increased in direct relation to the complexity of the neural network's architecture (Figure A2) and the value used for the regularization term. Training and validating all the models presented in Tables A1–A3 took about 92 h, a time which was quite equally shared between the models using AS data (approximately 30 h), A (approximately 29 h), and S (approximately 33 h) modalities. Of these, working with complex architectures (five or ten layers containing 1000 neurons each) took more time; it accounted for 7.6 (A), 8.1 (AS) and 9.7 (S) hours when working with a neural network of five layers holding 1000 neurons each, and for 19.1 (A), 20.4 (AS), and 21.6 (S) hours when working with a neural network of ten layers holding 1000 neurons each. These architectures did not improve the classification performance and, excepting the models based on the S modality, regularization terms set at 1 and 10 returned poorer classification performances.

3.2. Testing on Unseen Data

The main results characterizing the models' performance on unseen data are included for the three modalities in Table 3. Classification performance on unseen data is important to understanding how these models will work in real operational conditions. The testing dataset (DS2) contained a number of 4592 observations covering three classes of events: "WORK" (3235, about 71% of the data), "MAN" (1243, about 27% of the data), and "OFF" (114, about 2% of the data). As shown, the order of classification performance changed, pointing out that the model based on the A modality would perform better.

Table 3. Summary of the models' performance on unseen data.

Modality and Dataset	Optimal Architecture and Hyperparameters	Event Class	AUC	CA	F1	PREC	REC	SPEC	LOSS
AS, DS2	N = 1, n = 1000, α = 0.001, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.984	0.906	0.922	0.946	0.906	0.985	0.187
		WORK	0.998	0.906	0.988	0.998	0.977	0.995	0.187
		MAN	0.976	0.906	0.814	0.882	0.755	0.962	0.187
		OFF	0.927	0.906	0.250	0.165	0.518	0.933	0.187
A, DS2	N = 1, n = 100, α = 0.001, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.981	0.922	0.931	0.943	0.922	0.976	0.188
		WORK	0.995	0.922	0.984	0.993	0.976	0.983	0.188
		MAN	0.974	0.922	0.852	0.881	0.825	0.958	0.188
		OFF	0.921	0.922	0.283	0.206	0.447	0.956	0.188
S, DS2	N = 5, n = 1000, α = 1, Activation function = ReLU, solver = ADAM, number of iterations = 1,000,000, stratified cross-validation by 10 folds.	OVERALL	0.902	0.901	0.887	0.876	0.901	0.836	0.352
		WORK	0.916	0.901	0.948	0.917	0.981	0.787	0.352
		MAN	0.892	0.901	0.811	0.852	0.774	0.950	0.352
		OFF	0.868	0.901	0.000	0.000	0.000	1.000	0.352

Note: N—number of hidden layers, n—number of neurons per hidden layer, α—regularization parameter, AUC—area under the receiver operating curve, CA—classification accuracy, F1—F1 score, PREC—precision, REC—recall, SPEC—specificity, LOSS—cross entropy.

The classification performance would not be particularly different when using models based on AS and S data, respectively. These returned a classification accuracy of 90% on unseen data, while the model based on modality A returned a classification accuracy of 92% (Table 3). However, the error levels were preserved at similar levels and in a similar order of magnitude among the AS, A, and S models, indicating a higher error in the case of the S model working on unseen data (LOSS = 0.352, Table 3). Overall, the classification accuracy of AS and A models on unseen data was close to that of the training and validation phase (Tables 2 and 3), with differences that accounted for 2.4 and 0.6%, respectively.

Figure 5 shows the confusion matrices of the AS, A, and S models when working with unseen data. For instance, the AS model tested on unseen data confused "WORK" events with "MAN" events for a number of 73 observations, which may be due to similar movement speeds. In some other cases, it confused "MAN" with "OFF" events, which may also be due to a lower magnitude in acceleration and similar magnitudes in speed. As such, irrespective of the model used, there was some degree of inter-class similarity in the data, leading to confusion from the models.

	Predicted			
	MAN	**OFF**	**WORK**	**Σ**
MAN	939	299	5	1243
OFF	53	59	2	114
WORK	73	0	3162	3235
Σ	1065	358	3169	4592

(**a**)

	Predicted			
	MAN	**OFF**	**WORK**	**Σ**
MAN	1026	196	21	1243
OFF	61	51	2	114
WORK	78	0	3157	3235
Σ	1165	247	3180	4592

(**b**)

	Predicted			
	MAN	**OFF**	**WORK**	**Σ**
MAN	962	0	281	1243
OFF	106	0	8	114
WORK	61	0	3174	3235
Σ	1129	0	3463	4592

(**c**)

Figure 5. Confusion matrices of predictions made on unseen data: (**a**) predictions made by the best model for AS modality; (**b**) predictions made by the best model for A modality; (**c**) predictions made by the best model for S modality.

In all cases, "WORK" events were not confused as "OFF" events (Figure 5a–c). Interestingly, the model based on S data was better at setting apart the data from "OFF" events since, in contrast to AS and A models, it did not confuse "MAN" with "OFF" events.

4. Discussion

Monitoring forest operations by advanced techniques is important and increasingly required in practice and science to remove the subjectivity of assessments and the errors that may occur due to data collection over long periods of time, as well as to control the costs incurred by collecting, processing, and analyzing the data. It may also serve as a better method to accommodate safety concerns when working nearby dangerous machines.

The results of this study indicate that the type of data sourcing modality is important in the performance of a neural network designed for event-based classification tasks. This is supported by the classification performance outcomes when using the vector magnitudes of triaxial acceleration coupled with GNSS speed as data inputs, for which the overall classification accuracy in the training and validation phase was of 93%. Using this combination of input signals also provides the context required for the localization of operations since the GNSS data are documented with features such as geographical coordinates, altitude, and heading. As such, the acceleration data are more likely to provide the distinctive features needed to identify the relevant events, whereas the GNSS speed helps in improving the accuracy of classification, particularly in events that involve movement. However, by comparing the classification accuracies of the best performing neural network models on DS1, it seems that the GNSS speed makes only a marginal contribution to the improvement in classification accuracy in the training and validation steps. For instance, the classification accuracies of the AS, A, and S modalities were 93, 92.8, and 84.8%, respectively, indicating that using the GNSS speed alone comes at a high expense in event-based classification accuracy. Also, using the vector magnitude of the acceleration signal alone provides a classification accuracy close to that provided when combined with the GNSS speed but, by using this approach, the spatial context of the operations is lost.

In terms of the neural network's architecture and hyperparameter tuning, the AS and A modalities required a single hidden layer to achieve the best classification performance, whereas the S modality required a network with five hidden layers to show its best performance (Tables 2 and A1–A3). This hints at how the neural networks managed the complexity stored in the signals, indicating that accurately learning from the S modality with a single hidden layer was much more difficult to achieve, requiring more hidden layers but with fewer neurons. Also, finer regularization terms (α = 0.001 and 0.0001) were

the best option to increase the classification accuracy based on the AS and A modalities, respectively, whereas a coarser regularization term ($\alpha = 1$) was more suitable for the S modality. Nevertheless, very coarse regularization terms ($\alpha = 10$) generally led to a dilution in classification accuracy, irrespective of the modality, depth, and width of the networks used (Tables A1–A3). Neural networks using the AS and A modalities required a higher number of neurons per layer to give their best classification performance. In this regard, classification accuracy based on AS data worked the best with 1000 neurons, whereas classification accuracy based on A data worked the best with 100 neurons. In comparison, the neural network using S data required only 10 neurons to show its best performance. This may be another hint at how data complexity stored in the modalities' signals was handled, since it seems that arriving at the best classification accuracy depended on a decreasing number of weights in the order AS, A, and S, respectively. It is worth mentioning that training and validating the neural networks using 5 and 10 hidden layers of 1000 neurons each did not provide additional benefits in terms of classification accuracy but used up important computational resources, irrespective of the modality used as data input.

The performance of the models on unseen data is essential to understanding how they will behave in real world applications. Although there was a minor dilution in classification accuracy, the tuned models of the AS, A, and S modalities showed a high classification accuracy in predicting the events on unseen data. While this confirms the generalization ability of the trained and validated models, the most frequently confused events were "WORK" and "MAN", pointing out the inter-class similarity of the data. As a potential solution, these two classes could be merged to improve the classification performance. However, this approach would have led to missing an important work element for time study classification, and it would have inflated the time spent as main work time [18,19]. What is clear is that the "OFF" events were never misclassified as "WORK" or "MAN" events, which is promising considering that the time spent in such events stays outside the productive time [42],. From this point of view, separating the productive time from other time consumption categories would probably lead to an improved classification accuracy since working with binary classification problems based on triaxial acceleration data typically returns close to 100% in classification accuracy [26,43].

Poplar forests, on the other hand, are located on sites showing a wide variability in terms of soil condition [44], weed characteristics, and spacing of the trees [7,44], all of which may at least affect the distribution of events on classes, as well as the magnitude of the acceleration signals recorded. To these factors, one can add the variations in magnitude of acceleration due to the type of rotary tiller used, and particularly due to its size class. To some extent, these variations can be accommodated by the way in which the neural network models are built, since they use a standard data preprocessing flow that runs a standardization before training [45]. Data standardization provides input datasets that are scaled so as to have a mean value of zero and a standard deviation of 1, a procedure that is robust to outliers [46]. This is helpful to balance the effect of high magnitude observations, which adds to the benefits of using the vector magnitude of acceleration data to remove the effects of datalogger orientation [9,26]. Then, the GNSS signal in terms of the magnitude of the speed could be affected only by the operational conditions, for which sustaining much higher speeds than those observed in this study is less likely. As a fact, the moving speeds of machines used in short rotation forests are typically low and in the range of those observed in this study [47–51].

The main limitation of this study is related to the type of the dataset used. Ideally, robust models would consider variations that may be brought on by the type of machine used, the experience of the operators, and practice in various areas. For the time being, accounting for all these factors when developing the machine learning models is challenging. This comes mainly from the fact that there is no available data annotated for the operational monitoring of operations such as in this study, which is a common challenge in machine learning applications [17]. In addition to a resource-intensive process and the expertise required to annotate the data, large datasets may require an increased computational power

to be able to train the models. In this study, training and testing took from a couple of minutes to a couple of hours, and there was a trend in the magnitude of computational resources depending on the architecture used for the neural networks. Future studies could accommodate this limitation and check whether there will be a significant difference in the classification accuracy of the models trained on newcoming data. Due to the way in which deep learning models are built, they are likely to perform with higher classification performances, an option that should be kept in mind in future studies. For instance, valorizing the available knowledge and models that shape the science of deep learning and computer vision holds a lot of potential for better understanding the events and making better and more contextualized predictions based on a combination of 1D and 2D channel data.

5. Conclusions

The classification performance of mechanized weed control operational events by conventional neural network machine learning was found to be high based on triaxial acceleration and speed data extracted from GNSS signals in both training–validation and testing phases. This option enables forest managers to obtain accurate time- and space-contextualized estimations of the machine performance and will help in removing many of the less efficient processes required to collect, process, and analyze time study data. Provided that an offline option will be sufficient, the methods described herein can solve the problem of operational monitoring. For more advanced tools, multi-modal online solutions could be researched, as well as how well deep learning and computer vision perform would in operational classification.

Appendix A

Table A1. Classification performance of the AS modality in the training and validation phase.

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
	1	10	0.0001	113.013	0.018	0.985	0.926	0.911	0.901	0.926	0.966	0.198
	1	10	0.001	108.273	0.015	0.985	0.925	0.910	0.900	0.925	0.966	0.199
AS	1	10	0.01	108.526	0.017	0.985	0.920	0.899	0.895	0.920	0.964	0.204
	1	10	0.1	60.654	0.016	0.980	0.913	0.885	0.865	0.913	0.962	0.234
	1	10	1	32.598	0.018	0.955	0.907	0.878	0.856	0.907	0.938	0.292
	1	10	10	16.659	0.016	0.935	0.859	0.827	0.797	0.859	0.794	0.455

Table A1. *Cont.*

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
AS	5	10	0.0001	57.992	0.021	0.986	0.928	0.915	0.904	0.928	0.965	0.191
	5	10	0.001	64.708	0.032	0.986	0.928	0.916	0.905	0.928	0.965	0.191
	5	10	0.01	61.388	0.023	0.985	0.928	0.915	0.905	0.928	0.965	0.193
	5	10	0.1	66.964	0.020	0.985	0.928	0.915	0.904	0.928	0.964	0.193
	5	10	1	80.314	0.033	0.978	0.914	0.885	0.864	0.914	0.957	0.239
	5	10	10	45.376	0.022	0.936	0.864	0.832	0.803	0.864	0.810	0.442
AS	10	10	0.0001	58.585	0.029	0.983	0.918	0.908	0.900	0.918	0.967	0.208
	10	10	0.001	59.474	0.030	0.984	0.924	0.911	0.900	0.924	0.963	0.201
	10	10	0.01	57.133	0.028	0.984	0.925	0.911	0.900	0.925	0.963	0.198
	10	10	0.1	82.339	0.034	0.984	0.924	0.910	0.899	0.924	0.962	0.201
	10	10	1	88.714	0.036	0.981	0.913	0.884	0.864	0.913	0.959	0.232
	10	10	10	77.527	0.031	0.500	0.726	0.611	0.527	0.726	0.274	0.790
AS	1	100	0.0001	114.736	0.028	0.986	0.929	0.915	0.904	0.929	0.964	0.192
	1	100	0.001	112.851	0.040	0.986	0.927	0.913	0.902	0.927	0.964	0.193
	1	100	0.01	82.035	0.029	0.982	0.913	0.885	0.865	0.913	0.961	0.223
	1	100	0.1	86.150	0.029	0.982	0.913	0.885	0.865	0.913	0.961	0.223
	1	100	1	29.645	0.030	0.956	0.908	0.880	0.857	0.908	0.942	0.289
	1	100	10	13.096	0.028	0.935	0.858	0.825	0.795	0.858	0.790	0.451
AS	5	100	0.0001	110.955	0.091	0.981	0.914	0.903	0.893	0.914	0.964	0.216
	5	100	0.001	101.378	0.072	0.981	0.917	0.902	0.891	0.917	0.954	0.215
	5	100	0.01	100.513	0.087	0.978	0.910	0.897	0.885	0.910	0.957	0.235
	5	100	0.1	151.335	0.080	0.983	0.920	0.908	0.897	0.920	0.966	0.207
	5	100	1	222.884	0.087	0.978	0.912	0.884	0.863	0.912	0.954	0.241
	5	100	10	158.203	0.084	0.935	0.866	0.835	0.806	0.866	0.820	0.429
AS	10	100	0.0001	194.814	0.156	0.980	0.918	0.908	0.899	0.918	0.960	0.218
	10	100	0.001	186.888	0.168	0.982	0.915	0.905	0.896	0.915	0.964	0.213
	10	100	0.01	189.664	0.170	0.979	0.917	0.900	0.889	0.917	0.961	0.221
	10	100	0.1	275.760	0.153	0.981	0.921	0.906	0.895	0.921	0.960	0.214
	10	100	1	334.031	0.145	0.976	0.910	0.882	0.862	0.910	0.956	0.244
	10	100	10	434.342	0.158	0.500	0.726	0.611	0.527	0.726	0.274	0.791
AS	1	1000	0.0001	339.500	0.141	0.986	0.929	0.917	0.906	0.929	0.964	0.191
	1	**1000**	**0.001**	**326.498**	**0.158**	**0.986**	**0.930**	**0.917**	**0.906**	**0.930**	**0.964**	**0.191**
	1	1000	0.01	329.788	0.152	0.982	0.914	0.886	0.885	0.914	0.962	0.221
	1	1000	0.1	325.097	0.163	0.982	0.914	0.886	0.885	0.914	0.962	0.221
	1	1000	1	122.386	0.158	0.956	0.908	0.879	0.857	0.908	0.941	0.288
	1	1000	10	62.033	0.153	0.935	0.861	0.829	0.800	0.861	0.802	0.446
AS	5	1000	0.0001	3444.300	1.977	0.982	0.920	0.905	0.895	0.920	0.964	0.212
	5	1000	0.001	3353.729	2.016	0.983	0.921	0.911	0.903	0.921	0.965	0.206
	5	1000	0.01	5764.162	2.073	0.981	0.917	0.905	0.895	0.917	0.966	0.220
	5	1000	0.1	5371.765	1.968	0.981	0.917	0.904	0.893	0.917	0.961	0.225
	5	1000	1	5936.627	2.029	0.970	0.906	0.877	0.854	0.906	0.935	0.265
	5	1000	10	5273.351	1.935	0.918	0.788	0.731	0.718	0.788	0.511	0.573
AS	10	1000	0.0001	11,822.498	4.064	0.971	0.916	0.895	0.887	0.916	0.955	0.265
	10	1000	0.001	10,310.822	5.091	0.973	0.914	0.892	0.884	0.914	0.947	0.254
	10	1000	0.01	11,545.202	6.115	0.975	0.897	0.892	0.890	0.897	0.965	0.255
	10	1000	0.1	11,281.575	4.423	0.967	0.892	0.875	0.861	0.892	0.926	0.282
	10	1000	1	18,172.055	5.781	0.969	0.876	0.854	0.849	0.876	0.955	0.311
	10	1000	10	10,284.011	5.245	0.500	0.726	0.611	0.527	0.726	0.274	0.792

Note: N—number of hidden layers, n—number of neurons per hidden layer, α—regularization parameter, TT—training time (seconds), VT—validation time (seconds), AUC—area under the receiver operating curve, CA—classification accuracy, F1—F1 score, PREC—precision, REC—recall, SPEC—specificity, LOSS—cross entropy. The best performing model is highlighted in bold.

Table A2. Classification performance of the A modality in the training and validation phase.

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
A	1	10	0.0001	131.311	0.012	0.972	0.924	0.910	0.898	0.924	0.951	0.232
	1	10	0.001	131.927	0.015	0.972	0.924	0.909	0.897	0.924	0.951	0.232
	1	10	0.01	84.366	0.014	0.971	0.915	0.893	0.886	0.915	0.947	0.249
	1	10	0.1	41.572	0.017	0.971	0.911	0.883	0.860	0.911	0.947	0.274
	1	10	1	33.968	0.014	0.940	0.911	0.882	0.861	0.911	0.949	0.322
	1	10	10	20.710	0.016	0.941	0.820	0.776	0.750	0.820	0.630	0.530
A	5	10	0.0001	35.899	0.015	0.969	0.927	0.915	0.904	0.927	0.953	0.226
	5	10	0.001	37.753	0.019	0.970	0.926	0.914	0.904	0.926	0.953	0.226
	5	10	0.01	36.095	0.018	0.970	0.925	0.913	0.902	0.925	0.952	0.228
	5	10	0.1	50.689	0.019	0.971	0.927	0.916	0.905	0.927	0.952	0.225
	5	10	1	68.317	0.020	0.964	0.911	0.883	0.860	0.911	0.947	0.272
	5	10	10	53.425	0.019	0.879	0.855	0.820	0.790	0.855	0.737	0.548
A	10	10	0.0001	42.265	0.028	0.972	0.926	0.912	0.900	0.926	0.950	0.224
	10	10	0.001	38.729	0.021	0.968	0.926	0.913	0.900	0.926	0.949	0.228
	10	10	0.01	43.539	0.027	0.969	0.927	0.914	0.902	0.927	0.949	0.226
	10	10	0.1	47.266	0.030	0.967	0.927	0.915	0.903	0.927	0.951	0.225
	10	10	1	46.889	0.024	0.967	0.927	0.915	0.903	0.927	0.951	0.225
	10	10	10	132.933	0.025	0.500	0.726	0.611	0.527	0.726	0.274	0.790
A	**1**	**100**	**0.0001**	**177.114**	**0.033**	**0.972**	**0.928**	**0.914**	**0.902**	**0.928**	**0.952**	**0.227**
	1	**100**	**0.001**	**174.069**	**0.032**	**0.972**	**0.928**	**0.914**	**0.902**	**0.928**	**0.952**	**0.227**
	1	100	0.01	166.468	0.030	0.972	0.925	0.911	0.899	0.925	0.951	0.231
	1	100	0.1	103.447	0.033	0.969	0.911	0.883	0.860	0.911	0.946	0.265
	1	100	1	68.159	0.032	0.939	0.911	0.882	0.861	0.911	0.948	0.319
	1	100	10	17.931	0.028	0.943	0.796	0.744	0.721	0.796	0.564	0.524
A	5	100	0.0001	107.925	0.090	0.969	0.923	0.913	0.904	0.923	0.954	0.235
	5	100	0.001	117.054	0.084	0.969	0.926	0.914	0.903	0.926	0.952	0.235
	5	100	0.01	113.997	0.094	0.970	0.927	0.915	0.905	0.927	0.953	0.230
	5	100	0.1	187.261	0.090	0.972	0.925	0.914	0.903	0.925	0.954	0.232
	5	100	1	286.817	0.087	0.966	0.912	0.883	0.860	0.912	0.946	0.269
	5	100	10	123.166	0.084	0.938	0.855	0.820	0.791	0.855	0.740	0.477
A	10	100	0.0001	172.370	0.153	0.969	0.918	0.904	0.892	0.918	0.946	0.246
	10	100	0.001	162.744	0.165	0.968	0.923	0.911	0.899	0.923	0.949	0.241
	10	100	0.01	190.324	0.159	0.970	0.919	0.902	0.890	0.919	0.944	0.239
	10	100	0.1	190.054	0.088	0.972	0.925	0.914	0.903	0.925	0.954	0.232
	10	100	1	294.704	0.085	0.966	0.912	0.883	0.860	0.912	0.946	0.269
	10	100	10	132.107	0.099	0.938	0.855	0.820	0.791	0.855	0.740	0.477
A	1	1000	0.0001	376.896	0.153	0.971	0.927	0.913	0.901	0.927	0.952	0.227
	1	**1000**	**0.001**	377.893	5.242	0.971	0.926	0.913	0.901	0.926	0.952	0.227
	1	1000	0.01	381.633	0.165	0.971	0.927	0.913	0.901	0.927	0.952	0.228
	1	1000	0.1	393.555	0.153	0.970	0.912	0.883	0.861	0.912	0.948	0.253
	1	1000	1	172.661	0.142	0.940	0.911	0.883	0.861	0.911	0.949	0.319
	1	1000	10	80.511	0.150	0.942	0.829	0.789	0.760	0.829	0.673	0.523
A	5	1000	0.0001	3523.763	2.005	0.967	0.908	0.899	0.891	0.908	0.94806	0.256
	5	1000	0.001	4215.993	2.090	0.968	0.914	0.901	0.889	0.914	0.94573	0.253
	5	1000	0.01	4812.080	2.035	0.967	0.912	0.900	0.889	0.912	0.94658	0.249
	5	1000	0.1	6389.862	2.044	0.967	0.914	0.896	0.883	0.914	0.94518	0.243
	5	1000	1	7175.113	2.049	0.957	0.911	0.882	0.860	0.911	0.94395	0.280
	5	1000	10	4760.121	2.123	0.925	0.782	0.723	0.718	0.782	0.47803	0.567

Table A2. *Cont.*

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
	10	1000	0.0001	8451.184	4.251	0.967	0.913	0.895	0.884	0.913	0.945	0.260
	10	1000	0.001	7971.356	4.504	0.960	0.906	0.890	0.876	0.906	0.915	0.289
A	10	1000	0.01	15,169.579	4.909	0.960	0.908	0.895	0.883	0.908	0.940	0.281
	10	1000	0.1	10,456.199	4.993	0.960	0.892	0.885	0.880	0.892	0.941	0.297
	10	1000	1	15,724.293	4.441	0.961	0.912	0.883	0.860	0.912	0.943	0.277
	10	1000	10	10,937.570	4.479	0.500	0.726	0.611	0.527	0.726	0.274	0.794

Note: N—number of hidden layers, n—number of neurons per hidden layer, α—regularization parameter, TT—training time (seconds), VT—validation time (seconds), AUC—area under the receiver operating curve, CA—classification accuracy, F1—F1 score, PREC—precision, REC—recall, SPEC—specificity, LOSS—cross entropy. The best performing models are highlighted in bold and the model with the lowest computational requirements was retained for testing.

Table A3. Classification performance of the S modality in the training and validation phase.

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
	1	10	0.0001	22.730	0.014	0.900	0.847	0.818	0.791	0.847	0.800	0.454
	1	10	0.001	22.928	0.015	0.899	0.847	0.818	0.791	0.847	0.800	0.454
S	1	10	0.01	22.376	0.014	0.899	0.847	0.818	0.791	0.847	0.801	0.454
	1	10	0.1	24.338	0.012	0.900	0.846	0.816	0.788	0.846	0.793	0.455
	1	10	1	29.579	0.016	0.884	0.844	0.812	0.783	0.844	0.769	0.473
	1	10	10	25.196	0.015	0.870	0.816	0.775	0.747	0.816	0.641	0.565
	5	10	0.0001	31.968	0.021	0.899	0.848	0.820	0.794	0.848	0.814	0.450
	5	10	0.001	37.113	0.018	0.897	0.848	0.820	0.794	0.848	0.815	0.451
S	5	10	0.01	34.499	0.023	0.898	0.848	0.820	0.794	0.848	0.816	0.451
	5	10	0.1	43.378	0.018	0.898	0.848	0.820	0.794	0.848	0.816	0.451
	5	10	1	61.593	0.020	0.884	0.848	0.819	0.792	0.848	0.805	0.462
	5	10	10	59.856	0.023	0.500	0.726	0.611	0.527	0.726	0.274	0.790
	10	10	0.0001	34.280	0.026	0.899	0.848	0.820	0.794	0.848	0.814	0.451
	10	10	0.001	33.086	0.024	0.898	0.848	0.819	0.793	0.848	0.810	0.451
S	10	10	0.01	37.279	0.022	0.898	0.847	0.820	0.794	0.847	0.814	0.452
	10	10	0.1	62.776	0.030	0.898	0.848	0.820	0.795	0.848	0.816	0.451
	10	10	1	113.624	0.030	0.877	0.848	0.819	0.792	0.848	0.804	0.462
	10	10	10	123.910	0.023	0.500	0.726	0.611	0.527	0.726	0.274	0.790
	1	100	0.0001	28.583	0.028	0.900	0.847	0.820	0.794	0.847	0.816	0.451
	1	100	0.001	27.898	0.027	0.899	0.847	0.820	0.794	0.847	0.816	0.451
S	1	100	0.01	25.951	0.028	0.900	0.847	0.820	0.794	0.847	0.814	0.451
	1	100	0.1	31.891	0.026	0.899	0.848	0.819	0.793	0.848	0.806	0.452
	1	100	1	49.729	0.031	0.889	0.844	0.812	0.783	0.844	0.771	0.470
	1	100	10	25.909	0.025	0.870	0.816	0.775	0.747	0.816	0.641	0.561
	5	100	0.0001	81.134	0.088	0.898	0.847	0.820	0.795	0.847	0.818	0.452
	5	100	0.001	90.476	0.089	0.898	0.847	0.819	0.794	0.847	0.815	0.452
S	5	100	0.01	132.277	0.090	0.899	0.847	0.819	0.792	0.847	0.807	0.451
	5	100	0.1	222.879	0.088	0.898	0.847	0.818	0.792	0.847	0.808	0.454
	5	100	1	212.696	0.082	0.896	0.848	0.819	0.792	0.848	0.806	0.457
	5	100	10	133.832	0.081	0.863	0.796	0.748	0.725	0.796	0.563	0.614
	10	100	0.0001	199.142	0.165	0.899	0.847	0.818	0.792	0.847	0.807	0.453
	10	100	0.001	208.465	0.158	0.899	0.847	0.819	0.793	0.847	0.810	0.452
S	10	100	0.01	328.109	0.171	0.896	0.847	0.820	0.795	0.847	0.817	0.455
	10	100	0.1	498.491	0.152	0.896	0.846	0.818	0.793	0.846	0.815	0.458
	10	100	1	413.117	0.176	0.888	0.847	0.819	0.793	0.847	0.811	0.462
	10	100	10	604.866	0.166	0.500	0.726	0.611	0.527	0.726	0.274	0.791

Table A3. *Cont.*

Modality	N	n	α	TT	VT	AUC	CA	F1	PREC	REC	SPEC	LOSS
	1	1000	0.0001	123.832	0.168	0.898	0.848	0.820	0.793	0.848	0.810	0.453
	1	1000	0.001	117.616	0.144	0.898	0.848	0.820	0.793	0.848	0.810	0.453
S	1	1000	0.01	122.734	0.162	0.898	0.848	0.820	0.793	0.848	0.810	0.453
	1	1000	0.1	135.329	0.155	0.899	0.848	0.819	0.792	0.848	0.805	0.453
	1	1000	1	125.781	0.159	0.889	0.844	0.813	0.785	0.844	0.776	0.471
	1	1000	10	141.933	0.141	0.871	0.819	0.779	0.750	0.819	0.655	0.556
	5	1000	0.0001	3967.578	2.070	0.897	0.847	0.819	0.793	0.847	0.814	0.456
	5	1000	0.001	6176.544	2.029	0.899	0.848	0.820	0.794	0.848	0.814	0.452
S	5	1000	0.01	6545.242	2.077	0.900	0.848	0.819	0.793	0.848	0.810	0.452
	5	1000	0.1	5265.886	2.064	0.893	0.847	0.820	0.795	0.847	0.818	0.461
	5	**1000**	**1**	**8161.303**	**2.186**	**0.886**	**0.848**	**0.819**	**0.792**	**0.848**	**0.800**	**0.463**
	5	1000	10	4835.667	2.274	0.811	0.759	0.686	0.688	0.759	0.409	0.681
	10	1000	0.0001	9320.816	4.993	0.897	0.846	0.818	0.792	0.846	0.806	0.458
	10	1000	0.001	17,128.945	4.442	0.897	0.848	0.820	0.794	0.848	0.814	0.455
S	10	1000	0.01	14,130.691	3.850	0.894	0.847	0.819	0.793	0.847	0.813	0.456
	10	1000	0.1	13,607.806	4.461	0.898	0.847	0.819	0.793	0.847	0.812	0.454
	10	1000	1	12,868.969	3.838	0.884	0.844	0.817	0.793	0.844	0.817	0.488
	10	1000	10	11,687.663	4.818	0.500	0.726	0.611	0.527	0.726	0.274	0.794

Note: N—number of hidden layers, n—number of neurons per hidden layer, α—regularization parameter, TT—training time (seconds), VT—validation time (seconds), AUC—area under the receiver operating curve, CA—classification accuracy, F1—F1 score, PREC—precision, REC—recall, SPEC—specificity, LOSS—cross entropy. The best performing model is highlighted in bold.
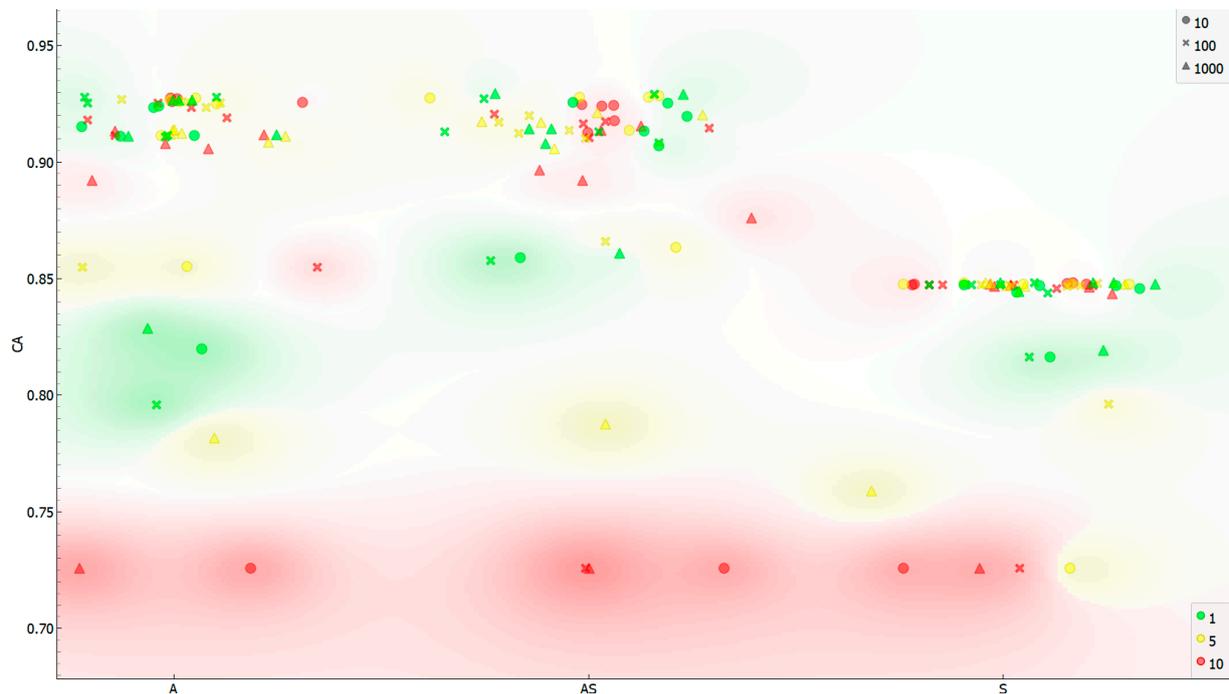
Figure A1. Classification accuracy (CA) by the modality used (AS, A, S) and the architecture of the neural network. Legend: green stands for a single hidden layer, yellow stands for 5 hidden layers, red stands for 10 hidden layers, circles stand for 10 neurons, crosses stand for 100 neurons and triangles stand for 1000 neurons.
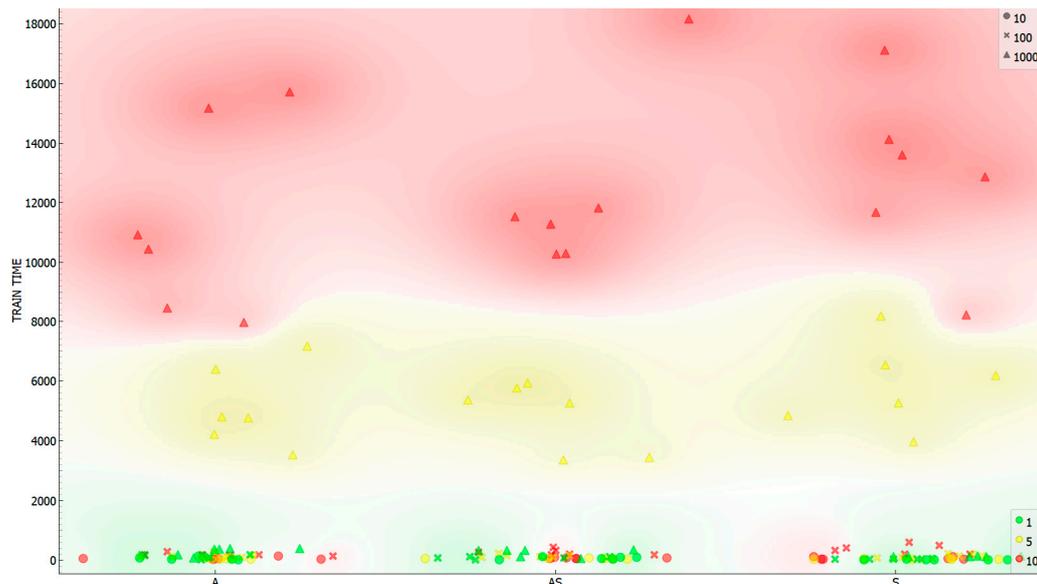
Figure A2. Training time (seconds) by the modality used (AS, A, S) and the architecture of the neural network. Legend: green stands for a single hidden layer, yellow stands for 5 hidden layers, red stands for 10 hidden layers, circles stand for 10 neurons, crosses stand for 100 neurons and triangles stand for 1000 neurons.

References

1. Mologni, O.; Dyson, P.; Amishev, D.; Proto, A.R.; Zimbalatti, G.; Cavalli, R.; Grigolato, S. Tensile Force Monitoring on Large Winch-Assist Forwarders Operating in British Columbia. *Croat. J. For. Eng.* **2018**, *39*, 193–204.
2. Stoilov, S.; Proto, A.R.; Angelov, G.; Papandrea, S.F.; Borz, S.A. Evaluation of salvage logging productivity and costs in the sensitive forests of Bulgaria. *Forests* **2021**, *12*, 309. [CrossRef]
3. Casado, M.; Acuña, L.; Basterra, L.A.; Ramón-Cueto, G.; Vecilla, D. Grading of structural timber of *Populus×euramericana* clone I-214. *Holzforschung* **2012**, *66*, 633–638. [CrossRef]
4. Papandrea, S.F.; Cataldo, M.F.; Bernardi, B.; Zimbalatti, G.; Proto, A.R. The Predictive Accuracy of Modulus of Elasticity (MOE) in the Wood of Standing Trees and Logs. *Forests* **2022**, *13*, 1273. [CrossRef]
5. Industrial Applications of Poplar Wood. Available online: https://propopulus.eu/en/industrial-applications-of-poplar-wood/ (accessed on 1 October 2024).
6. Dickmann, D.I. Silviculture and biology of short-rotation woody crops in temperate regions: Then and now. *Biomass Bioenerg.* **2006**, *30*, 696–705. [CrossRef]
7. Marogel-Popa, T.; Cheța, M.; Marcu, M.V.; Duță, I.C.; Ioraș, F.; Borz, S.A. Manual cultivation operations in poplar stands: A characterization of job difficulty and risks of health impairment. *Int. J. Environ. Res. Public Health* **2019**, *16*, 1911. [CrossRef]
8. Zhai, S.; Shi, Y.; Zhou, J.; Liu, J.; Huang, D.; Zou, A.; Jiang, P. Simulation Optimization and Experimental Study of the Working Performance of a Vertical Rotary Tiller Based on the Discrete Element Method. *Actuators* **2022**, *11*, 342. [CrossRef]
9. Keefe, R.F.; Zimbelman, E.G.; Wempe, A.M. Use of smartphone sensors to quantify the productive cycle elements of hand fallers on industrial cable logging operations. *Int. J. For. Eng.* **2019**, *30*, 132–143. [CrossRef]
10. Becker, R.M.; Keefe, R.F. A novel smartphone-based activity recognition modelling method for tracked equipment in forest operations. *PLoS ONE* **2022**, *17*, e0266568. [CrossRef]
11. Wang, G.; Zhang, W.; Chen, M.; Ji, M.; Diao, X.; Miao, H. Enhancing Sustainable Afforestation through Innovative Earth Auger Design: A Simulation Study in Hilly Regions. *Sustainability* **2024**, *16*, 5402. [CrossRef]
12. Tampekis, S.; Kantartzis, A.; Arabatzis, G.; Sakellariou, S.; Kolkos, G.; Malesios, C. Conceptualizing Forest Operations Planning and Management Using Principles of Functional Complex Systems Science to Increase the Forest's Ability to Withstand Climate Change. *Land* **2024**, *13*, 217. [CrossRef]
13. Hiesl, P.; Benjamin, J.G. Applicability of international harvesting equipment productivity studies in Maine, U.S.A.: A literature review. *Forests* **2013**, *4*, 898–921. [CrossRef]
14. Demsar, J.; Curk, T.; Erjavec, A.; Gorup, C.; Hocevar, T.; Milutinovic, M.; Mozina, M.; Polajnar, M.; Toplak, M.; Staric, A.; et al. Orange: Data Mining Toolbox in Python. *J. Mach. Learn. Res.* **2013**, *14*, 2349–2353.
15. Building A Neural Net from Scratch Using R—Part 1. Available online: https://rviews.rstudio.com/2020/07/20/shallow-neural-net-from-scratch-using-r-part-1/ (accessed on 23 July 2024).
16. Python AI: How to Build a Neural Network & Make Predictions. Available online: https://realpython.com/python-ai-neural-network/ (accessed on 23 July 2024).

17. Chen, K.; Zhang, D.; Yao, L.; Guo, B.; Yu, Z.; Liu, Y. Deep learning for sensor-based human activity recognition: Overview, challenges and opportunities. *J. ACM* **2021**, *77*, 1–40. [CrossRef]

18. Acuna, M.; Bigot, M.; Guerra, S.; Hartsough, B.; Kanzian, C.; Kärhä, K.; Lindroos, O.; Magagnotti, N.; Roux, S.; Spinelli, R.; et al. *Good Practice Guidelines for Biomass Production Studies*; CNR IVALSA Sesto Fiorentino (National Research Council of Italy—Trees and Timber Institute): Sesto Fiorentino, Italy, 2012; 52p.

19. Björheden, R.; Apel, K.; Shiba, M.; Thompson, M. *IUFRO Forest Work Study Nomenclature*; Swedish University of Agricultural Science, Department of Operational Efficiency: Grapenberg, Sweden, 1995; 16p.

20. Ruiz-Garcia, L.; Sanchez-Guerrero, P. A Decision Support Tool for Buying Farm Tractors, Based on Predictive Analytics. *Agriculture* **2022**, *12*, 331. [CrossRef]

21. Lanças, K.P.; Marques Filho, A.C.; Santana, L.S.; Ferraz, G.A.e.S.; Faria, R.O.; Martins, M.B. Agricultural Tractor Test: A Bibliometric Review. *AgriEngineering* **2024**, *6*, 2229–2248. [CrossRef]

22. Puška, A.; Nedeljković, M.; Šarkoćević, Ž.; Golubović, Z.; Ristić, V.; Stojanović, I. Evaluation of Agricultural Machinery Using Multi-Criteria Analysis Methods. *Sustainability* **2022**, *14*, 8675. [CrossRef]

23. Bacescu, N.M.; Cadei, A.; Moskalik, T.; Wisniewski, M.; Talbot, B.; Grigolato, S. Efficiency assessment of fully mechanized harvesting system through the use of fleet management system. *Sustainability* **2022**, *14*, 16751. [CrossRef]

24. Spinelli, R.; Laina-Relano, R.; Magagnotti, N.; Tolosana, E. Determining observer and method effects on the accuracy of elemental time studies in forest operations. *Balt. For.* **2013**, *19*, 301–306.

25. Musat, E.C.; Apafaian, A.I.; Ignea, G.; Ciobanu, V.D.; Iordache, E.; Derczeni, R.A.; Spârchez, G.; Vasilescu, M.M.; Borz, S.A. Time expenditure in computer aided time studies implemented for highly mechanized forest equipment. *Ann. For. Res.* **2016**, *59*, 129–143.

26. Borz, S.A. Development of a modality-invariant multi-layer perceptron to predict operational events in motor-manual willow felling operations. *Forests* **2021**, *12*, 406. [CrossRef]

27. Borz, S.A.; Talagai, N.; Cheța, M.; Chiriloiu, D.; Gavilanes Montoya, A.V.; Castillo Vizuete, D.D.; Marcu, M.V. Physical strain, exposure to noise and postural assessment in motor-manual felling of willow short rotation coppice: Results of a preliminary study. *Croat. J. For. Eng.* **2019**, *40*, 377–388. [CrossRef]

28. Borz, S.A.; Cheta, M.; Birda, M.; Proto, A.R. Classifying operational events in cable yarding by a machine learning application to GNSS-collected data: A case study on gravity-assisted downhill yarding. *Bull. Transilv. Univ. Bras.* **2022**, *15*, 13–32. [CrossRef]

29. Şofletea, N.; Curtu, L. *Dendrologie*, 2nd ed.; Pentru Viaţă Publishing House: Braşov, Romania, 2008; ISBN 978-973-85874-4-1.

30. Abrudan, I.V. *Împăduriri*; Transilvania University Press: Braşov, Romania, 2006.

31. Florescu, I.; Nicolescu, N.V. *Silviculture*; Lux Libris Publishing House: Brasov, Romania, 1996.

32. Castro Pérez, S.N.; Borz, S.A. Improving the event-based classification accuracy in pit-drilling operations: An application by neural networks and median filtering of the acceleration input signal data. *Sensors* **2021**, *21*, 6288. [CrossRef]

33. Ministry of Waters, Forests and Environment Protection; National Forest Administration. *Unified Work Time and Production Rates for Forestry, Ministry of Waters*; Forests and Environment Protection: Bucharest, Romania, 1997.

34. Spinelli, R.; Visser, R.J.M. Analyzing and estimating delays in wood chipping operations. *Biomass Bioenergy* **2009**, *33*, 429–433. [CrossRef]

35. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the 30th International Conference on Machine Learning (ICML 2013), Atlanta, GA, USA, 16–21 June 2013.

36. Nair, V.; Hinton, G.E. Rectified linear units improve restricted Boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning (ICML 2010), Haifa, Israel, 21–24 June 2010.

37. Kingma, D.P.; Ba, J.L. 2015: ADAM: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.

38. Fawcett, T. An introduction to ROC analysis. *Pattern Recogn. Lett.* **2006**, *27*, 861–874. [CrossRef]

39. Kamilaris, A.; Prenafeta-Boldu, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]

40. Cross Validation in Machine Learning. Available online: https://www.geeksforgeeks.org/cross-validation-machine-learning/ (accessed on 23 July 2024).

41. 5 Reasons Why You Should Use Cross-Validation in Your Data Science Projects. Available online: https://towardsdatascience.com/5-reasons-why-you-should-use-cross-validation-in-your-data-science-project-8163311a1e79 (accessed on 23 July 2024).

42. Bulling, A.; Blanke, U.; Schiele, B. A tutorial of human activity recognition using body-worn inertial sensors. *ACM Comput. Surv.* **2014**, *46*, 1–33. [CrossRef]

43. Borz, S.A.; Forkuo, G.O.; Oprea-Sorescu, O.; Proto, A.R. Development of a robust machine learning model to monitor the operational performance of fixed-post multi-blade vertical sawing machines. *Forests* **2022**, *13*, 1115. [CrossRef]

44. Marogel-Popa, T. Performance of Planting and Cultivation Operations in Poplar Forests Located in the Danube Meadow. Ph.D. Thesis, Transilvania University of Brasov, 2020. Available online: https://www.unitbv.ro/documente/cercetare/doctorat-postdoctorat/sustinere-teza/2020/marogel-tiberiu/REZUMAT_TEZA_DOCTORAT_MAROGEL-POPA_TIBERIU.pdf (accessed on 6 August 2024).

45. Neural Network. Available online: https://orangedatamining.com/widget-catalog/model/neuralnetwork/ (accessed on 6 August 2024).

46. Normalization vs Standardization. Available online: https://www.geeksforgeeks.org/normalization-vs-standardization/ (accessed on 6 August 2024).

47. Talagai, N.; Marcu, M.V.; Zimbalatti, G.; Proto, A.R.; Borz, S.A. Productivity in partly mechanized planting operations of willow short rotation coppice. *Biomass Bioenergy* **2020**, *138*, 105609. [CrossRef]

48. Manzone, M.; Balsari, P. Planters performance during a very short rotation coppice planting. *Biomass Bioenergy* **2014**, *67*, 188–192. [CrossRef]

49. Manzone, M.; Bergante, S.; Facciotto, G.; Balsari, P. A prototype for horizontal long cuttings planting in short rotation coppice. *Biomass Bioenergy* **2017**, *107*, 214–218. [CrossRef]

50. Bush, C.; Volk, T.A.; Eisenbies, M.H. Planting rates and delays during the establishment of willow biomass crops. *Biomass Bioenergy* **2015**, *83*, 290–296. [CrossRef]

51. Borz, S.A.; Nita, M.D.; Talagai, N.; Scriba, C.; Grigolato, S.; Proto, A.R. Performance of small-scale technology in planting and cutback operations of short-rotation willow crops. *Trans. ASABE* **2019**, *62*, 167–176. [CrossRef]