Asymmetric Identification Model for Human-Robot Contacts via Supervised Learning

Qasem Abu Al-Haija * and Ja'afer Al-Saraireh

Department of Computer Science/Cybersecurity, Princess Sumaya University for Technology,
Amman 11941, Jordan; j.saraireh@psut.edu.jo
* Correspondence: q.abualhaija@psut.edu.jo

Abstract: Human-robot interaction (HRI) occupies an essential role in the flourishing market for intelligent robots for a wide range of asymmetric personal and entertainment applications, ranging from assisting older people and the severely disabled to the entertainment robots at amusement parks. Improving the way humans and machines interact can help democratize robotics. With machine and deep learning techniques, robots will more easily adapt to new tasks, conditions, and environments. In this paper, we develop, implement, and evaluate the performance of the machine-learning-based HRI model in a collaborative environment. Specifically, we examine five supervised machine learning models viz. the ensemble of bagging trees (EBT) model, the k-nearest neighbor (kNN) model, the logistic regression kernel (LRK), the fine decision trees (FDT), and the subspace discriminator (SDC). The proposed models have been evaluated on an ample and modern contact detection dataset (CDD 2021). CDD 2021 is gathered from a real-world robot arm, Franka Emika Panda, when it was executing repetitive asymmetric movements. Typical performance assessment factors are applied to assess the model effectiveness in terms of detection accuracy, sensitivity, specificity, speed, and error ratios. Our experiential evaluation shows that the ensemble technique provides higher performance with a lower error ratio compared with other developed supervised models. Therefore, this paper proposes an ensemble-based bagged trees (EBT) detection model for classifying physical human–robot contact into three asymmetric types of contacts, including noncontact, incidental, and intentional. Our experimental results exhibit outstanding contact detection performance metrics scoring 97.1%, 96.9%, and 97.1% for detection accuracy, precision, and sensitivity, respectively. Besides, a low prediction overhead has been observed for the contact detection model, requiring a 102 μS to provide the correct detection state. Hence, the developed scheme can be efficiently adopted through the application requiring physical human–robot contact to give fast accurate detection to the contacts between the human arm and the robot arm.

Keywords: human–robot interaction (HRI); human–machine communication (HMC); robotics; machine learning; asymmetric classification

1. Introduction

Robots are considered asymmetric coworkers in industrial settings, assisting humans in complex or physically demanding tasks. As multi-purpose service assistants in homes, they will be an essential part of our daily lives [1]. Robots are beginning to migrate away from production lines, where they are separated from humans, and into environments where human-robot interaction (HRI) is unavoidable [2]. With this transition, human-robot collaboration has progressed to the point where robots can now work alongside and around humans on difficult tasks.

Human-robot interaction (HRI) has recently received more attention, intending to transform the manufacturing industry from inflexible traditional production techniques to a flexible and smart manufacturing paradigm [3,4]. The current industry needs a novel generation of robots to assist workers by leveraging responsibilities in standings of

flexibility and cognitive skill needs [3]. HRI applications include everything from robots that aid the elderly and disabled to entertainment robots at amusement parks. HRI is employed in a variety of areas, including space technology, military, industry, medical treatment, assisting the elderly, crippled, and entertainment [5,6].

Operationally, HRI has two modes of operation relationships: parallel relationships and hierarchical relationships. In a parallel relationship, a human and a robot are two independent and asymmetric entities that make their own decisions, which is also called peer-peer interaction. While in hierarchical relationships, either the human or the robot transfers part of the responsibility of decision making (such as detection or identification) to others by leveraging intelligent learning techniques. Typically, HRI works as a team to complete a physical task at both the cognitive and physical levels. Based on a cognitive model for HRI, the robot gathers inputs from users and the environment and elaborates and translates them into information that the robot can use [7]. However, the drawbacks of human-robot collaboration include synchronization, excessive interaction force, and insufficient motion compliance [5].

Robots can interact with humans by several means. Figure 1 illustrates the human-robot interaction. There are different types of interactions, such as physical, social emotions, or cultural context interactions. There are different techniques through which a human can interact with the robots, such as visual, tactile screen sensing, tactile skin sensing, voice, and audiovisual [8].
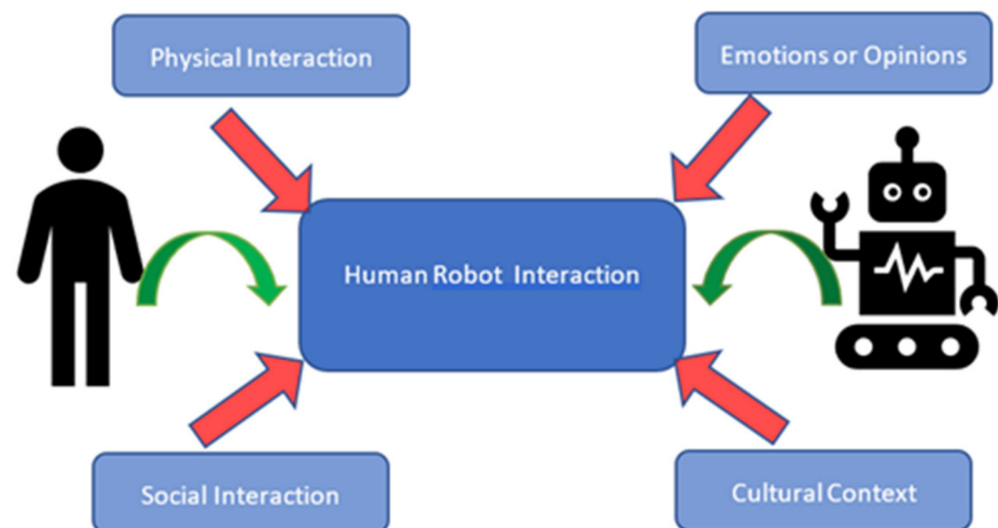

Figure 1. Framework Human-Robot Interaction.

In collaborative environments, the safe HRI necessitates robots to avoid potentially damaging collisions, while continuing to work on their primary mission wherever possible. Additionally, to avoid potentially dangerous collisions, robots must be able to accurately detect their surroundings and assimilate critical information in real-time [2]. In circumstances when people and robots must physically interact and/or share their workspace, these robots are expected to actively assist workers in performing complex tasks, with the highest priority placed on human safety [6]. Human-robot contact detection allows the robot to keep a safe distance from its human counterpart or the environment, which is an important requirement for ensuring safety in shared workspaces [6].

Designing a multi-objective, cascaded safety system that primarily prevents collisions while also limiting the force impact of a collision is unavoidable, assuring human safety when collaborating with a collaborative robot in a physical HRI. Human intention detection is one of the most important preconditions for addressing this concern [6]. However, when HRI circumstances become more regular, relying solely on collision avoidance will no longer suffice—contact is unavoidable and, in certain cases, desirable. When humans interact with the robot, for example, when touch is required, a simple knock can encourage

a worker to step away, allowing more movement spaces [2]. Safety is always a critical issue for HRI tasks [9]. As a result, the first issue that must be addressed is man-machine security [10].

Because collision avoidance cannot always be guaranteed in unexpected dynamic contexts, collision detection is the most basic underlying element for safe control of robot activity. Collision detection must be extremely accurate in order for the robot to react quickly. Due to their low bandwidth, exteroceptive sensors, such as cameras, are limited in their application. Furthermore, establishing detection using merely basic proprioceptive sensors has a lot of attraction in terms of on-board availability (no workspace constraints) and low costs.

The main strategies for safety HRI are collision avoidance, collision detection, and collision reaction. Collision avoidance is the interaction between humans and robots without contact. The robot modifies its predetermined trajectory to avoid collision with humans. Human wearable sensors, cameras, and sensors are commonly used for collision avoidance. Unfortunately, due to the limitations of sensors and robot motion capabilities, collision avoidance may fail, for example, if the human moves quicker than the robot can detect or counteract. When an unwanted physical collision is detected, the robot switches as quickly as possible from the control law associated with normal task execution to a reaction control law. A number of different reaction strategies have been considered and implemented [11].

Therefore, in this paper, we propose a new intelligent and autonomous human-robot contact detection model (HRCDM) aiming to improve the interaction between humans and robotics and thus help robots to actively assist workers in performing complex tasks with the highest priority placed on human safety. The proposed model is intended to provide a decision-making mechanism for the robot to identify the type of contact with humans in a collaborative environment. In addition, our proposed method considers the collaborative robots in which the torque signal is available, i.e., the robot requires motor torque. Therefore, it is assumed that the model is to be deployed for robots working in a collaborative environment with humans (and with each other as well). Thus, this method cannot be applied to any robot that contains only the position sensor. Specifically, our main contribution in this work can be summarized as follows:

- We developed a lightweight and high-performance human-robot contact detection model (HRCDM) using different supervised machine-learning methods.
- We discriminate and evaluate the performance of five supervised ML schemes EBT (ensemble bagging trees), FDT (fine decision trees), kNN (k-nearest neighbors), LRK (logistic regression kernel), and SDC (subspace discriminate) for HRCDS using new and inclusive contact detection dataset (CDD 2021).
- We provide extensive experimental results using several quality and computation indicators. In addition, we compare our best outcomes with existing approaches and show that our EBT-based HRCDS is more accurate than existing models by 3–15%, with faster inferencing.

2. Related Works

The human-robot interaction zone has recently placed a strong prominence on cyber-physical space in order to address the concerns accompanied by intelligent robots for a wide range of personal and entertainment applications. As such, collision or contact detection in the human–robotics collaborative environment is a major concern for several HRI domains that require the robot to infarct and move with the surroundings. A large number of approaches have been introduced and developed in the literature to address this issue. Collision avoidance methods based on sensors and cameras are presented in Section 2.1. While collision detection methods based on the dynamic model of the robot (model-based methods) are discussed in Section 2.2. These methods should be discussed and compared. The third part discusses collision detection methods based on data (data-based methods),

which use fuzzy logic, neural networks, support vector machines, and deep learning, and is presented in Section 2.3.

2.1. Collision Avoidance

The active collision avoidance approach proposed in [10] provides a workplace monitoring system to detect any human entering the robot's workspace. This proposed method is based on the calculated human skeleton in real-time to minimize measurement error and to estimate the positions of the skeleton points. A human's behavior is also estimated by an expert system. Robots use various methods to avoid human interaction, such as stopping and bypassing. In this approach, the AI method is used to generate a new path for the robot when it needs to execute in real-time while avoiding the human. The proposed method employs active collision avoidance to ensure that the robot does not come into contact with the human. The proposed work was to detect a human, analyze the human's motion, and then protect the human based on the motion [10].

To avoid collisions, [12] safety zones were created around a robot. For dynamic avoidance, the use of a 3D camera to inform the robot's movement in a dynamic environment was proposed. For safe HRI, a mixed perception approach was presented. A safety monitoring system was created to improve safety by combining the recognition of human actions with visual perception and the interpretation of physical HRI. Contact and vision data were gathered. When entering shared workspaces, the recognition system classified human actions by using the skeleton representation of the latter. When physical contact between a human and a robot occurs, the contact detection system is used to distinguish between intentional and unintentional interaction [6] by introducing on-board perception with proximity sensors [13]. This method relies on external sensing to keep a safe distance from humans [13]. The proximity data are used to avoid, rather than to make, a transition contact. Another noticeable work has been developed in [14], where a minimal robotic twin-to-twin transfusion syndrome (TTTS) approach was proposed. From a single monocular fetoscope camera image, CNN was used to predict the relative orientation of the placental surface. This work is based on the position of the object of interest as a cognitive ability model, and it employs vision for sensing. In this proposed study, the accuracy of adjusting the object of interest was 60%.

Unfortunately, due to the limitations of sensors and robot motion capabilities, collision avoidance may fail, for example, if the human moves quicker than the robot can detect or counteract. The steps of avoiding collision are to detect the human, analyze the motion of the human and then safeguard the human from a collision.

2.2. Collision Detection Methods Based on the Dynamic Model of the Robot

Moreover, the authors in [15] used a Gaussian mixture model with force sensing to encode catheter motions at both the proximal and distal sites based on cannulation data obtained from a single phantom by an expert operator. Non-rigid registration was used to map catheter tip trajectories into other anatomically similar trajectories, yielding a warping function. The robot detects and adjusts its end-effector using its distal sites. The proposed method produced more smooth catheter paths than the manual method, while the human positions the robot's end effector in a specific pose and holds the robot in place. In addition, the proposed approach in [16] is based on interactive reinforcement learning that learns a complete collaborative assembly process. This approach uses robot decision-making as a cognitive ability model and vision as a sensing technique. The action of the robot is based on observing the scene and understanding the assembly process. The proposed method was able to handle deviation during action execution.

A neural integrator was used as time-varying persistent activity of neural populations to model the gradual accumulation of sensory and other evidence [17]. In this approach, decision-making and vision sensing were used to assist a human in assembling a pipe system. While the robot looks at the human to understand the current stage of the process, it passes the pipe required by the human. Conversely, a nonparametric motion flow model

for HRI was proposed by [18]. The proposed method made use of a vision sensing and robot decision-making model as a cognitive ability model. By considering the spatial and temporal properties of a trajectory, the mean and variance functions of a Gaussian process were used to measure the motion flow similarity. The main human role in this approach is to grab the object held by the robot, but at a certain point in the interaction, the human decides to change the way he grabs it. A change in the user's motions was detected by the robot, and action was determined.

Furthermore, collaborators in [19] proposed a model learning and planning approach to trust-aware decision making in HRI. A computational model that incorporates trust into robot decision-making is used in this approach. A partially observable Markov decision process with human trust as a latent variable was learned from the data by the system. If the human does not trust the robot, the human must either put or prevent the robot from acting. The proposed model's results show that human trust corresponds to the robot's manipulation capabilities. In another HRI scenario, a neuro-inspired model based on dynamic neural fields was used to select an action [20]. Its foundation was task recognition and vision sensing. In this approach, a system of colored pipes was built in collaboration with the robot. The current step of the assembly sequence is understood by the robot, and it performs an action and advises the user on the next steps it should take. In addition, HRI using variable admittance control and the prediction was proposed by [21]. It used force sensing, reinforcement learning, and long short-term memory networks to obtain the optimal damping value of the admittance controller and to predict human intention. In this approach, the optimal damping value of the admittance controller was obtained by using reinforcement learning, while the human intention was predicted by using the long short-term memory networks. In this technique, the object held by the robot was dragged by the human. The intention of the human was detected by the robot and tracked the trajectory forced by the dragging of the object.

Additionally, the researchers used a model-based reinforcement learning model, a multi-layer perceptron, and force sensing [22]. A human must lift an object, while a robot assists in the lifting of an object. Based on bilateral control for HRI, Reference [23] presented imitation learning. This method made use of control variables, force sensing, and long-short rem memory. A model for predicting a human's ergonomics within an HRI was developed by [24]. The human must provide support for an object while rotating it. The robot looks at the human hand pose to provide the required rotation. Similarly, the authors in [25] presented a reinforcement learning-based framework for HRI. It was learned in an unsupervised and proactive manner, and it balances the benefits of timely actions with the risk of taking inappropriate actions by minimizing the total time required to complete the task. In this technique, machine learning methods such as reinforcement learning, long short-term memory, and variational autoencoders were used. The robot recognized the human action and then assisted the human in the assembly. This method avoids the time-consuming annotation of motion data and allows for online learning. Besides, the work proposed in [26] describes an artificial cognitive architecture for HRI that uses social cues to decipher goals. This work implemented a low-level action encoding with a high-level probabilistic goal inference and several levels of clustering on different feature spaces, paired with a Bayesian network that infers the underlying intention. In general, collision detection methods that are based on the dynamic model should obtain joint acceleration information, which is, however, complicated to calculate or measure.

2.3. Collision Detection Methods Based on Data

The approaches proposed by [27] based on collision detection and reaction do not include prior-to-contact sensing, and none of these approaches employ proactive perception of the environment to anticipate contact. A neural network was used to solve the synchronization movement problem in HRI. The radial basis function neural network (RBFNN) model was used in this approach to detect the collaborator's motion intention. The adaptive impedance control method was used to obtain training samples and to control

the robot during the data acquisition process [5]. In addition, ML methods are considered as a critical factor in HRI and interception. SVM, the Markov model, neural networks, and the Gaussian mixture model (GMM) have been used for HRI with accuracy ranging from 70% to 90% [28–30]. Deep learning algorithms are used to improve HRI recognition accuracy by using video stream data RGB-D images, 3D skeleton tracking, and joint extortion for the classification of arbitrary action [31–33].

Collision detection and identification for robot systems were solved by simple threshold logic and a hypothesis-tested method. Speech recognition and snoring identification and fault identification of the mechatronic system was used [34]. A deep learning neural network was used and tested using the system fault status [35]. In [36], different classifiers were used and tested using the recorded signal samples.

Time series and fuzzy logic modeling were used by [37] for collision detection between humans and robots. Collisions were detected statistically by observing large deviations between the actual and expected torque. The time series method is more efficient since it does not require an external sensor. The results show that both proposed methods can detect collisions fast. However, unexpected collisions and expected contacts of the robot with the human are not discussed in this research.

Time series and fuzzy logic modeling to detect collisions between a human and a robot were proposed by [37]. Collisions were detected statistically by looking for large differences in torque between the actual and expected values. Because it does not require an external sensor, the time series method is more efficient. The results show that both methods proposed can detect collisions quickly. However, unexpected collisions and expected human-robot contact were not addressed in this study. For detecting human–manipulator collisions, a neural network was proposed by [38]. The neural network is designed based on the manipulator's joint dynamics. The designed system's results achieve high effectiveness in detecting collisions between humans and robots. In [39], a practical aspect of collision detection was presented using simple neural architecture. In this study, a virtual force sensor that processes information about motor current with the help of an artificial neural network with four different architectures was used. The MC-LSTM architecture produced a mean absolute prediction error of about 22 Nm. The external tests conducted show that the presented architecture can be used as a collision detector. With the optimal threshold, the MC-LSTM collision detection f1 score was 0.85. In this study, isolation and identification were not considered; only the detection part was considered.

In [40], a deep learning method was used for online robot action planning and execution to parse visual observations of HRI and predict the human operator's future motion. The proposed method is based on a recurrent neural network (RNN) that can learn the time-dependent mechanisms that underpin human motions. In the same context, Dynamic neural fields were used to apply brain-like computations to endow a robot with these cognitive functions. In addition, HRI stiffness estimation and intention detection were proposed [41]. The endpoint stiffness of the human arm can be calculated based on the muscle activation levels of the upper arm and the human arm configurations. This method uses a neural learning algorithm to configure wrist recognition. The primary human role is to lift an object with the robot and move it to a new location. The robot's role is to detect human interaction and follow its movement to keep the object balanced. The results show that the robot was successful in completing its task.

In another context, the HRI framework proposed in [42] is based on estimating human arm impedance parameters with a neural network to improve tracking performance under variable admittance control for HRI. This method made use of force sensing. The human perceives different objects differently and holds one end of the saw, while the robot holds the other end of the saw and tracks its movement using recognition parameters. When compared to other approaches, the results of this research work show smoother performance.

Unlike aforementioned models, in this work, we propose a machine-learning-based self-governing human-robot contact detection system to support the decision making of a robotic system in classifying the type of contacts with the human in a collaborative HMI en-

vironment by characterizing the performance of five different ML-based models using eight different standard performance evaluation metrics. The selected ML model to develop the human-robot contact detection system is the optimal model that provides the best quality indication factors, minimum error rates, and best possible computational complexity.

3. HRC Detection Model

In this research, we have developed and evaluated our proposed machine learning-based human-robot contact detection using the MATLAB 2021b system along with Simulink toolboxes over a 64-bit Windows 11 Professional operating system, with the latest service packs. For improved simulation and efficient experimental assessment, we used a high-performance computing platform that employs an Intel 11th Gen Intel (R) Core (TM) i7-11800 H @ 2.30 GHz Processor, with 16 GB memory (DDR-4 RAM units), 1000-GB SD storage, and 4 GB graphical processing unit (GPU). The complete workflow diagram for the development and assessment of our proposed HRCDS is illustrated in Figure 2 below. Specifically, once the required data are accumulated, then, the HRCDM model is composed of three core components before the deployment and performance of the collision detection; these include the data formulation component, learning models' component, and the performance evaluation component.
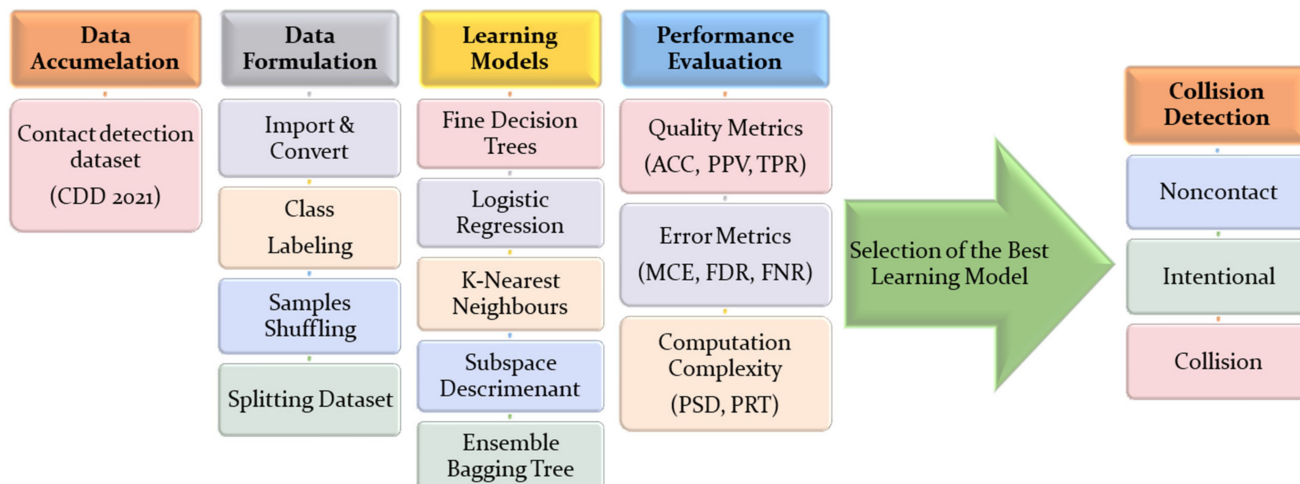

Figure 2. Workflow diagram to implement and validate the proposed HRCDS Model.

3.1. The CDD 2021 Dataset

Contact detection dataset (or CDD 2021 Dataset) is an up-to-date dataset that is assembled from a robot arm, Franka Emika Panda while performing frequent interactions and movements within a collaborative environment [43]. Each sample is composed of using 785 numerical features that are calculated from the 140-millisecond time-lapse sensor data, motor torque, external torque, joint position, and velocity, with a 200 Hz sampling rate (28 data points). The CDD 2021 dataset accumulates a total of 2205 samples distributed, as exhibited in Figure 3.

3.2. The Data Formulation Component

A data formulation component is a group of consecutive operations accountable for reformulating the accumulated data samples symmetrically and preparing a proper format that can be presented for the learning processes. The stages of this model component are described below and are summarized in Figure 4.
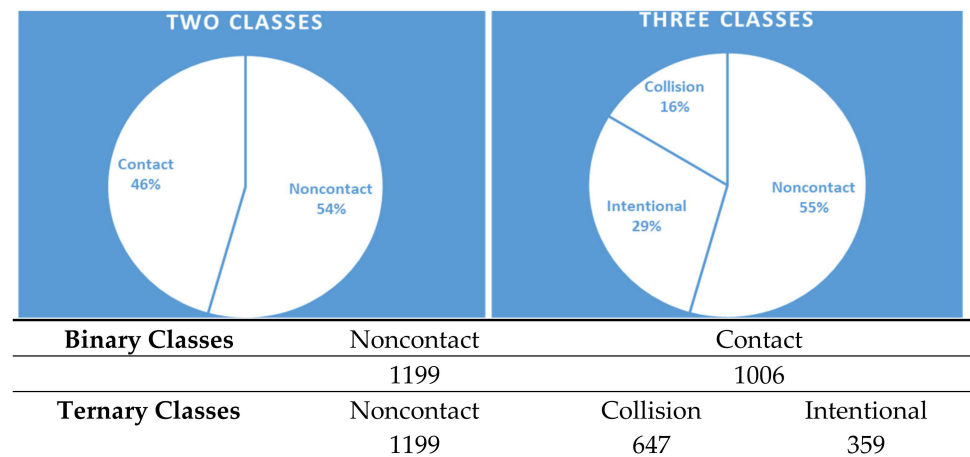
Binary Classes	Noncontact		Contact	
	1199		1006	
Ternary Classes	Noncontact	Collision		Intentional
	1199	647		359

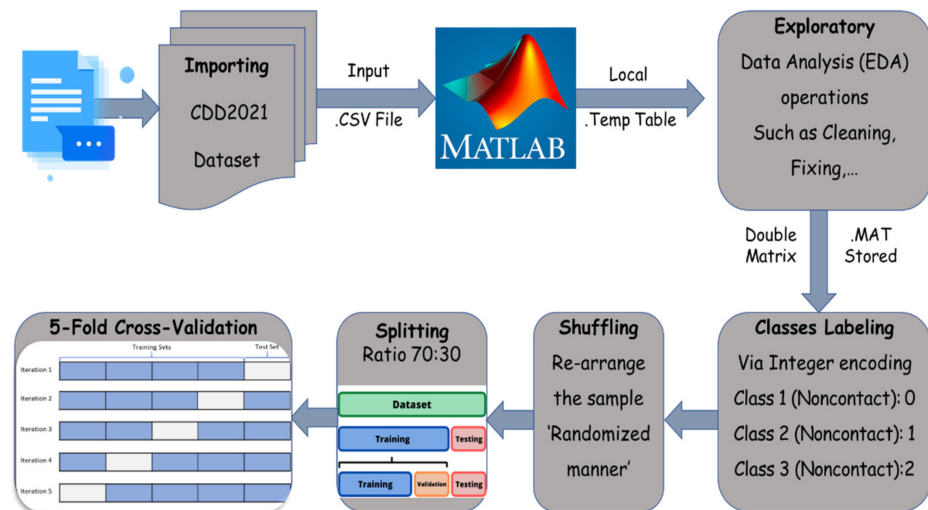**Figure 3.** CDD 2021 dataset: sample distribution.

Figure 4. Stages of the data formulation process.

- The first stage after accumulation of the intended data (i.e., CCD2021) is to read and host the CCD2021 records (which is available in .CSV format) using the MATLAB import tool. Once imported, MATLAB will host the data records in a temporary table in order to enable the exploratory data analysis (EDA) before the data are stored in the workplace to check the symmetry of samples features. At this stage, we have performed several EDA processes over the imported data including: (1) combining the datasets from different files in one unified dataset, (2) removing any unwanted or corrupted values, (3) fixing all missing values, (4) replacing nonimportable values or empty cells with zeros.

- The second stage of data formulation is the class labeling, where we have three classes at the output layer. We have three labels encoded using integer encoding as follows: "Noncontact: 0", "Intentional: 1", and "Collision: 2".

- The third stage of data formulation is samples randomization (also called shuffling). This process takes place by rearranging the positions for the sample in the dataset by using a random redistribution process. This process is essential to ensure that data are statistically unpredicted and asymmetric and to avoid any biasing by the classifier toward specific data samples.

- The fourth stage of data formulation is the splitting of the dataset into two subsets: training dataset that is composed of 70% of the samples in the original dataset and testing (validation) dataset that is composed of 30% of the samples in the original dataset. To ensure higher efficiency in the learning process (training + testing), we

used asymmetric k-fold cross-validation to distribute the dataset into training and testing datasets.

During the training process, we used 70% of the CDD 2021 dataset comprising approximately 1600 samples that are distributed in a balanced manner between the collision and non-collision classes. The rest of the samples, which is 30% of the CDD 2021 dataset, comprising approximately 410 samples, are used to test the model performance. The used dataset is quite enough to train the proposed model to provide binary/ternary classifications. To justify arguments for this experiment, we trained and tested the proposed model using several machine learning techniques (to be discussed in the upcoming sections), and most of these models have successfully achieved a high-performance detection capability in terms of several standard key performance indicators (KPIs), such as detection accuracy, detection sensitivity, and detection speed.

3.3. The Learning Models Component

In this work, we have developed our HRCDM model using five supervised algorithms in order to characterize their performance and then to pick up the optimal model to deploy the HRCDM model. The developed ML models include: the ensemble of bagging trees (EBT) model [44], the k-nearest neighbor (kNN) model [45], the logistic regression kernel (LRK) [46], the fine decision trees (FDT) [47], and the subspace discriminator (SDC) [48]. The summary of ML technique specifications and configurations used to implement the HRC detection model is illustrated in Table 1.

Table 1. Summary of ML techniques used to implement the HRC detection model.

Model Type	Description
EBT	Learning Family: Bagged Trees, Ensemble technique: Bag, Learner method: Decision tree, Number of learners: 30, Maximum number of DT splits: 2204, Learning rate = 0.1, Optimizer: Bayesian optimizer, number of epochs: 30 iterations.
kNN	Learning Family: Weighted KNN, Number of neighbors (k): 10, Learner method: nearest neighbor, Distance metric: Euclidean Distance, weight metric: Squared inverse, data standardization: true
LRK	Learning Family: Logistic Regression Kernel, Learner method: Logistic Regression, Number of expansion dimensions: Auto, Regularization strength (Lambda): Auto, Kernel scale: Auto, Multiclass method: One-vs-One, Iteration limit: 1000
FDT	Learning Family: Fine Tree, Learner method: Decision tree, Maximum number of splits: 100, Split criterion: Gini's diversity index, Surrogate decision splits Off, all features used in the model.
SDC	Learning Family: Subspace Discriminant, Ensemble technique: Subspace, Learner method: Discriminant, Number of learners: 30, Subspace dimension: 393, all features used in the model.

3.4. The Model Evaluation Component

The model evaluation in the machine-learning-based model is a vital stage of every model design [49]. This significantly requires tracing of the performance trajectories to provide insight into the solution approach and problem formulation, to compare the developed ML-based models to pick up the optimal model that maximizes the model performance and minimizes the model alarms and error thresholds, and finally to benchmark the best-obtained results with respect to those results reported in the literature within the same area of research. In this research, we have evaluated our models in terms of three quality indicating factors (ACC, PPV, and TPR), three alarms/error rates (MCE, FDR, and FNR), and two computation complexity factors (PSD and PRT). Besides, a confusion matrix analysis is provided to explore the statistical numbers of the samples classified correctly (TP, TN) and the samples classified incorrectly (FP, FN). Furthermore, the area under the ROC

(receiver operating characteristic) curve (in short, known as AUC) [50] is also investigated for every individual class (noncontact, intentional, and collision). Figure 5 demonstrates the summary of standard assessment factors involved in this research.

Quality Indication Factors	$ACC = \dfrac{TP+TN}{TP + TN + FP + FN} \times 100\%$	P represents the positive class
		N represents the negative class
	$PRC = \dfrac{TP}{TP + FP} \times 100\%$	TP represents the number of true positives
		TN represents the number of true negatives
	$RCL = \dfrac{TP}{TP + FN} \times 100\%$	FP represents the number of false positives
		FN represents the number of false negatives
Error Alarming Rates	$CER = 100\% - ACC\%$	ACC represents the classification accuracy
		PRC represents the classification precision
	$FDR = 100\% - PRC\%$	RCL represents the classification recall
		CER represents the classification error
	$FNR = 100\% - RCL\%$	FDR represents the false discovery rate
		FNR represents the false negative rate
Computation Complexity	$PSD = observations / second$	PSD represents the classification speed
	$PTM = (1 / PSD)$ in μsecond	PTM represents the classification time

Figure 5. Standard performance measurement factors.

3.5. Experimental Environment

To efficiently develop and evaluate the aforementioned models, we used a high-performance computing platform along with up-to-date software/programming utilities. For instance, in terms of software, the ML models stated above have been implemented and evaluated on a Windows 11 environment (64-bit operating system, x64-based processor) using the MATLAB 2021b computing system and machine learning toolbox (Simulink tool) along with the help of a parallel computation package to enable the parallel execution of model training and testing, which maximize the utilization of the available processing and computing devices. In terms of hardware, the developed (implemented) models have been executed on a commodity computer with high processing and computing capabilities. The computer is composed of central processing (11th Gen Intel (R) Core (TM) i7-11800 H @ 2.30 GHz), a graphical processing unit (NVIDIA GeForce RTX 3050 Ti Laptop GPU), and random-access memory (16.0 GB of RAM). In addition, the data (up-to-date data, 2021) used to train and test the stated ML models were obtained from Mendeley's data repository powered by Elsevier. Once the data are obtained, they undergo a series of data preprocessing operations such as data cleaning, transformation, encoding, standardization, shuffling, and splitting (into training and testing datasets). Once the data are preprocessed, the ML models are developed using MATLAB coding in addition to the machine learning toolboxes provided in the packages. Every model is customized and trained/validated using five-fold cross-validation using several parameters/hyperparameters configurations in order to pick up the optimal results of the learning process performance. Every model is evaluated using several standard performance measurement factors (stated above in Figure 5). Once all results from all models are generated, compared, and analyzed, then the best performant model is selected to be deployed with the robot to execute in real-time environments.

4. Results and Analysis

In this work, we developed a computational intelligence model for human-robot interactions (HRI) that can help the robot identify the asymmetric type of contact during its movements and interactions with humans. To do so, we have employed the five asymmetric aforementioned machine learning techniques at the implementation stage of the proposed HRCDS. The following subsections provide extensive results and analyses regarding the solution approach and model implementation.

4.1. Performance Evaluation

In this subsection, we present and analyze our experimental results for the developed and evaluated models. To begin, Table 2 along with Figure 6 compares the performance of HRCDS implementation utilizing various machine learning methods (i.e., HRCDS_EBT, HRCDS_kNN, HRCDS_LRK, HRCDS_FDT, and HRCDS_SDC) in terms of eight different performance indicators (i.e., ACC, PPV, TPR, MCE, FDR, FNR, PSD, and PRT). In terms of quality rates (accuracy (ACC), specificity (PPV), and sensitivity (TPR)), the ensemble learning-based model (HRCDS_EBT) exhibits the highest performance indicators with improvement factors of 0.5–20%, 1.5–23.4%, and 0.1–24.4% for ACC, PPV, and TPR, respectively, over the other implemented ML-based models. Consequently, HRCDS_EBT has recorded the lowest estimates in terms of error rates with 2.9%, 4.3%, and 4.9% for MCE, FDR, and FNR, respectively. Conversely, the lowest quality rates and thus highest error rates belong to the subspace discriminant learning-based model (HRCDS_SDC). However, it recorded relatively high-speed inferencing with a prediction time (PRT) of 135 μS. In terms of computational complexity, both HRCDS_EBT and HRCDS_FDT show the highest prediction speed with 9800 O/S and 15,000 O/S, respectively. On the contrary, the slowest predictive model is the HRCDS_LRK requiring a 909 μS for single sample inferencing. Although HRCDM_kNN is three times slower than HRCDS_EBT, it recorded a competent performance indicator, which can be a second high-performance alternative model for the HRCDM_EBT model.

Table 2. Summary of performance factors for HRC detection model using different ML Techniques.

	ACC	PPV	TPR	MCE	FDR	FNR	PSD	PRT
EBT	97.1%	96.9%	97.1%	2.9%	3.1%	2.9%	9800 O/S	102 μS
kNN	96.6%	94.2%	95.0%	3.4%	5.8%	5.0%	3300 O/S	303 μS
LRK	94.2%	92.0%	89.4%	5.8%	8.0%	10.6%	1100 O/S	909 μS
FDT	92.2%	71.7%	61.3%	7.8%	28.3%	38.7%	15,000 O/S	67 μS
SDC	77.1%	72.3%	70.7%	22.9%	27.7%	29.3%	7400 O/S	135 μS

Figure 6. Comparing the performance factors for several HRCDS variations.

4.2. Optimal Model Selection

Based on the above discussion, summary, and comparisons of performance factors for HRCDS variations using different asymmetric ML techniques provided in Table 2 and Figure 6, it can be inferred that overall, the best performance factors' results belong to the HRCDS-based ensemble bagged trees, and thus, HRCDS-EBT is efficiently selected for the robot in making intelligent decisions involving the asymmetric contacts with humans due to movements and interactions. Therefore, the following results and discussion will focus on the HRCDS-EBT. For instance, Figure 7 illustrates the trajectory of the misclassification rate (minimum classification error) for 30 iterations of the learning process of the HRCDM-EBT model. By observing the figure, the best point of the learning process is satisfied after the 18th iteration where the model reached its minimum classification error (2.9%) and was saturated to the end of the learning process. At this point, the best hyperparameters of the ensemble bagging trees were recorded (for example, the best learning rate, number of learners, etc.).
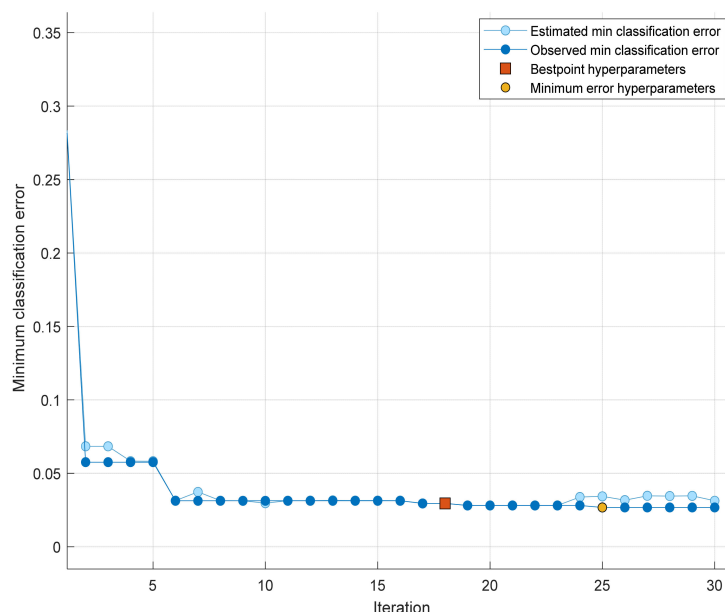

Figure 7. The performance trajectory for the minimum classification error vs. iteration number for the ensemble bagged trees (EBT).

4.3. Experimental Evaluation of the Optimal Model

Besides, Figure 8 exhibits the confusion matrix analysis for the three-classes classifier employing ensemble bagged trees (EBT) for the HRCDS. The diagonal of the matrix represents the number of correctly classified samples for each target class; that is, the number of correctly classified samples for the "collision" class is 319 samples out of 359 samples for this target class (i.e., 88.9% of classification accuracy for this class), the number of correctly classified samples for the "intentional" class is 623 samples out of 647 samples for this target class (96.3% of classification accuracy for this class), and the number of correctly classified samples for the "Noncontact" class is 1199 samples out of 1199 samples for this target class (100% classification accuracy for this class). Conversely, the non-diagonal cells of this matrix represent the number of incorrectly classified samples for each target class; that is, the number of incorrectly classified samples for the "collision" class is 40 samples out of 359 samples for this target class (i.e., 11.1% of minimum classification error for this class), the number of incorrectly classified samples for the "intentional" class is 24 samples out of 647 samples for this target class (3.7% of minimum classification error for this class), and the number of incorrectly classified samples for the "Noncontact" class is 0 samples out of 1199 samples for this target class (0.0% minimum classification error for this class).

However, the overall classification accuracy and minimum classification error will consider all numbers in the matrix as follows:

$$Overall\ accuracy\ for\ HRCDS-EBT\ =\ \frac{319+623+1199}{359+647+1199}\times 100\%\ =97.1\%$$

$$Overall\ minimum\ classification\ error\ =\ \frac{40+24+0}{359+647+1199}\times 100\%\ =2.90\%$$

Figure 8. The confusion matrix analysis for the three-classes classifier employing ensemble bagged trees (EBT).

Moreover, Figure 9 illustrates the sensitivity levels (TPR/recall) and specificity levels (PPV/precision) as well as the insensitivity levels (FNR) and imprecision levels (FDR) for the individual classes of the three-classes classifier of HRCDS-EBT. The "Noncontact" class, which represents 54% of the total number of samples (1199 samples out of 2205 samples), exhibits the best performance rates, scoring a 100% for sensitivity and specificity and 0.00% for insensitivity and imprecision. Similarly, the "Intentional" class, which represents 29% of the total number of samples (647 samples out of 2205 samples), exhibits excellent performance rates scoring 96.30% and 94.00% for sensitivity and specificity as well as 3.70% and 6.0% for insensitivity and imprecision, respectively. Conversely, the "Collision" class, which represents 16% of the total number of samples (359 samples out of 2205 samples), exhibits the lowest performance rates scoring 88.90% and 93.00% for sensitivity and specificity as well 11.10% and 7.0% for insensitivity and imprecision, respectively. However, the overall sensitivity and specificity for HRCDS-EBT are 97.1% and 96.9% with 2.9% and 3.1% recorded for the overall insensitivity and imprecision levels, respectively. The reason for such high-performance indicators is due to the impact of high-performance results obtained for the "Noncontact" class, which occupies the majority of the samples and thus highly contributed to the overall sensitivity and specificity levels. Another major reason is the use of ensemble learning, which is considered one of the best solutions for imbalanced data classification problems [51], enhancing the performance indicators toward the classes. The reasons for such high-performance indicators are due to the use of ensemble learning, which can enhance the performance indicators toward the classes that have been the majority of the samples, such as "Noncontact", which contributed to the overall sensitivity and specificity levels.
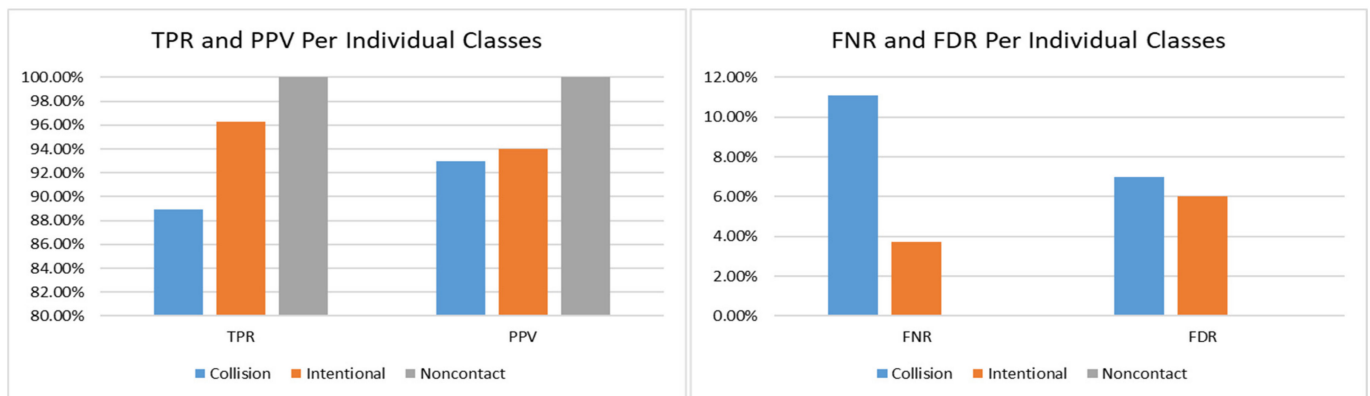
Figure 9. The sensitivity (recall) matrix analysis for the three-classes classifier employing ensemble bagged trees (EBT).

Additionally, Figure 10 illustrates the area under the ROC (receiver operating characteristic) curve (in short known as AUC) results forever individual class of the proposed three-class classifier. The AUC curve is a representational plot to present the analytical capability of the classifier model of distinguishing between the positive and negative classes. The AUC curve is created by plotting TPR against FPR to show the trade-off between sensitivity and specificity at different thresholds. For every classifier, the more its AUC curve becomes closer to the top-left corner, the better performance it has. Therefore, the best performant curve will pass through the point (0.0, 1.0) of ROC space providing a 100% ability in the discrimination of positive and negative classes (i.e., AUC = 1.0, best discrimination), whereas the closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test is. Therefore, the least accurate curve will pass through the point (0.5, 0.5) of the ROC space providing a 50% ability in the discrimination of positive and negative classes (i.e., AUC = 0.5, no discrimination). According to the obtained ROC curves for our classes, it can be seen that all classes illustrated an excellent discrimination capability, recording AUC values of 1.00, 0.98, and 0.93 for "Noncontact", "Intentional", and "Collision" classes, respectively. Overall, the average AUC of the proposed model exceeds 0.97, which reflects the high capability of our model in distinguishing between the positive and negative classes.
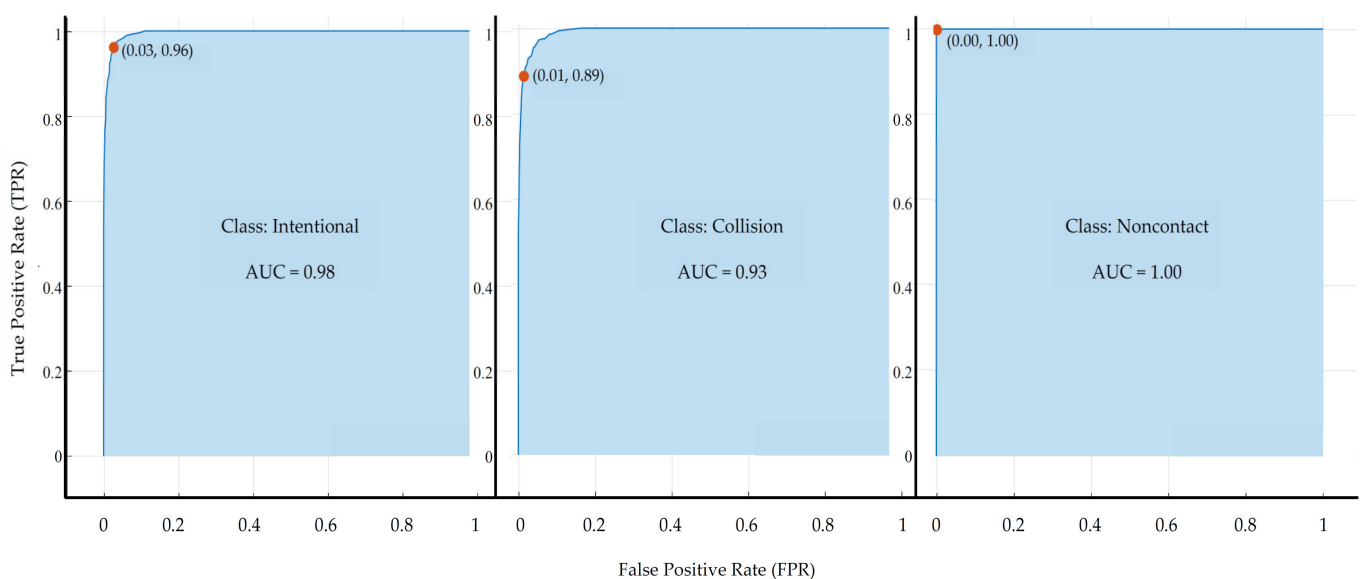

Figure 10. The ROC curves for each class of the three-classes classifier employing ensemble bagged trees (EBT).

4.4. Comparison with the State-of-the-Art Models

To the best of our knowledge, this is the first work that evaluates and characterizes the performance of machine learning approaches using the CDD 2021 dataset (contact detection) that is gathered from a robot arm of Franka Emika Panda (a well-known robot). However, we can still compare our empirical results with some of the results reported in some other related works in the same area of study. For instance, the intelligent HRI model proposed in [52] can provide collision detection between humans and robots and worker's clothing detection with a sensitivity level of 90% and an accuracy of 94%. Our model outperforms this model with a 7.1% sensitivity level and a 3.1% accuracy level. In addition, the authors of [53] developed an intelligent HRI model to classify human actions through the assembly process using convolutional neural network (CNN), and they achieved 82% accuracy, while our ensemble-based model indicates a higher accuracy of 97.1%. Furthermore, the authors of [54,55] characterized the performance of two deep learning-based collision detection systems, viz. collision net CNN [54] and force myography (FMHG) [55], which reported accuracy rates of 88% and 90%, respectively, introducing a detection overhead of 200 ms, while our ensemble model achieved 97.1% with 102μs detection overhead. In addition, the contributors in [38] developed a neural network-based safe human–robot interaction. In this work, the authors have characterized the performance of four different neural network models, including one multilayer feedforward NN (MLFFNN-1) architecture, MLFFNN architecture with two-hidden layers (MLFFNN-2), cascaded forward NN (CFNN), and recurrent NN (RNN). The best performance results belonged to the CFNN, scoring 92% and 84% for accuracy and sensitivity. However, our model performs significantly better in terms of both reported evaluation metrics. Finally, the researchers in [39] developed an intelligent robot collision detection system by employing the combination of convolution neural network and the long short-term memory (CNN-LSTM). Their empirical evaluation results reported, for their best model, an accuracy of 93.0%, the same for specificity, and 91.0% for sensitivity. All the reported results are lower than that reported by our model performance. Several other reported results are compared and exhibit the high performance of our proposed model, such as the support vector machine-based model reported in [56], which has an accuracy of 59% and sensitivity proportion of 72%. Lastly, Table 3 provides a summary of the effectiveness comparison of our system with other systems that exist in the literature in the same area of study.

Table 3. Comparing our best-obtained results with the existing model for HRI collision detection.

Ref.	Year	Model	Accuracy	Sensitivity	Specificity	Overhead
[38]	2021	CFNN model	92.0%	84.0%	-	NA
[39]	2021	(CNN-LSTM)	93.0%	91.0%	93.0%	NA
[52]	2022	CNN	94.0%	90.0%	NA	NA
[53]	2019	CNN	82.0%	NA	NA	NA
[54]	2019	CollisionNet	88.0%	NA	NA	18.5 ms
[55]	2019	FMG method	90.0%	NA	NA	200 ms
Proposed	2022	HRCDS-EBT	97.1%	97.1%	97.1%	102 μs

5. Conclusions

An intelligent self-governing human-robot contact detection model to support the decision making of a robotic model in classifying the type of contacts with humans in an HMI environment divided into three types of contacts, noncontact, incidental, or intentional, has been developed, implemented, and evaluated in this paper. The proposed solution approach characterizes the results of five machine-learning methods, including k-nearest neighbors (kNN), logistic regression kernel (LRK), fine decision trees (FDT), subspace discriminant (SDC), and ensemble bagging trees (EBT). The developed models have been evaluated on a contemporary and inclusive contact detection dataset (CDD 2021) that was assembled from the movements of the "Franka Emika Panda" robot arm when it was executing a repetitive movement. Our empirical results have been recorded for the

model-based EBT, which demonstrated remarkable performance metrics recording 97.1% and 97.0% for detection accuracy and the F1-score, respectively, with a low inference delay of 102 µS to provide the correct detection state. Therefore, the proposed model can be professionally adopted into physical human-robot contact applications to provide fast and accurate HRI contact detection.

For future work, we will try to address the collision detection framework for robotics, employing position sensors only where the torque signal is available and working in a collaborative environment with humans. In addition, we will try to investigate the employment of long, short-term memory (LSTM), which is flexible and has more control ability on the classification output and thus can have improved decision-making results for robotics systems. Furthermore, the proposed model can be deployed in a real HRI environment to provide collision detection in a real-time environment. More data targeting several other aspects of motion/interaction of the root system can be collected and recorded from the actual movements of the robot while interacting with humans and other robots.

References

1. Haddadin, S.; de Luca, A.; Albu-Schäffer, A. Robot collisions: A survey on detection, isolation, and identification. *IEEE Trans. Robot.* **2017**, *33*, 1292–1312. [CrossRef]
2. Escobedo, C.; Strong, M.; West, M.; Aramburu, A.; Roncone, A. Contact Anticipation for Physical Human-robot Interaction with Robotic Manipulators Using Onboard Proximity Sensors. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 7255–7262. [CrossRef]
3. Robla-Gomez, S.; Becerra, V.M.; Llata, J.R.; Gonzalez-Sarabia, E.; Torre-Ferrero, C.; Perez-Oria, J. Working Together: A Review on Safe Human–robot Collaboration in Industrial Environments. *IEEE Access* **2017**, *5*, 26754–26773. [CrossRef]
4. Nikolakis, N.; Maratos, V.; Makris, S. A cyber physical system (CPS) approach for safe human–robot collaboration in a shared workplace. *Robot. Comput. Integr. Manuf.* **2019**, *56*, 233–243. [CrossRef]
5. Liu, Z.; Hao, J. Intention Recognition in Physical Human-robot Interaction Based on Radial Basis Function Neural Network. *J. Robot.* **2019**, *2019*, 4141269. [CrossRef]
6. Amin, F.M.; Rezayati, M.; van de Venn, H.W.; Karimpour, H. A Mixed-Perception Approach for Safe Human–Robot Collaboration in Industrial Automation. *Sensors* **2020**, *20*, 6347. [CrossRef]
7. Semeraro, F.; Griffiths, A.; Cangelosi, A. Human–robot Collaboration and Machine Learning: A Systematic Review of Recent Research. *arXiv* **2021**, arXiv:2110.07448.
8. Rafique, A.A.; Jalal, A.; Kim, K. Automated Sustainable Multi-Object Segmentation and Recognition via Modified Sampling Consensus and Kernel Sliding Perceptron. *Symmetry* **2020**, *12*, 1928. [CrossRef]
9. Zhang, Z.; Qian, K.; Schuller, B.W.; Wollherr, D. An Online Robot Collision Detection and Identification Scheme by Supervised Learning and Bayesian Decision Theory. *IEEE Trans. Autom. Sci. Eng.* **2021**, *18*, 1144–1156. [CrossRef]
10. Du, G.; Long, S.; Li, F.; Huang, X. Active collision avoidance for human–robot interaction with UKF, expert system, and artificial potential field method. *Front. Robot. AI* **2018**, *5*, 125. [CrossRef]
11. Nascimento, H.; Mujica, M.; Benoussaad, M. Collision Avoidance in Human–robot Interaction Using Kinect Vision System Combined with Robot's Model and Data. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January 2021; pp. 10293–10298. [CrossRef]

12. Svarny, P.; Tesar, M.; Behrens, J.K.; Hoffmann, M. Safe physical HRI: Toward a unified treatment of speed and separation monitoring together with power and force limiting. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 7580–7587. [CrossRef]

13. Ding, Y.; Thomas, U. Collision Avoidance with Proximity Servoing for Redundant Serial Robot Manipulators. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 10249–10255. [CrossRef]

14. Ahmad, M.A.; Ourak, M.; Gruijthuijsen, C.; Deprest, J.; Vercauteren, T.; Poorten, E.V. Deep learning-based monocular placental pose estimation: Toward collaborative robotics in fetoscopy. *Int. J. Comput. Assist. Radiol. Surg.* **2020**, *15*, 1561–1571. [CrossRef]

15. Chi, W.; Liu, J.; Rafii-Tari, H.; Riga, C.V.; Bicknell, C.D.; Yang, G.-Z. Learning-based endovascular navigation through the use of non-rigid registration for collaborative robotic catheterization. *Int. J. Comput. Assist. Radiol. Surg.* **2018**, *13*, 855–864. [CrossRef]

16. Akkaladevi, S.C.; Plasch, M.; Pichler, A.; Ikeda, M. Toward Reinforcement based Learning of an Assembly Process for Human Robot Collaboration. *Procedia Manuf.* **2019**, *38*, 1491–1498. [CrossRef]

17. Wojtak, W.; Ferreira, F.; Vicente, P.; Louro, L.; Bicho, E.; Erlhagen, W. A neural integrator model for planning and val-ue-based decision making of a robotics assistant. *Neural Comput. Appl.* **2021**, *33*, 3737–3756. [CrossRef]

18. Choi, S.; Lee, K.; Park, H.A.; Oh, S. A Nonparametric Motion Flow Model for Human-Robot Cooperation. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 7211–7218. [CrossRef]

19. Chen, M.; Nikolaidis, S.; Soh, H.; Hsu, D.; Srinivasa, S. Trust-Aware Decision Making for Human-robot Collaboration. *ACM Trans. Human Robot Interact* **2020**, *9*, 1–23. [CrossRef]

20. Cunha, A.; Ferreira, F.; Sousa, E.; Louro, L.; Vicente, P.; Monteiro, S.; Erlhagen, W.; Bicho, E. Toward collaborative robots as intelligent co-workers in human–robot joint tasks: What to do and who does it? In Proceedings of the 52nd International Symposium on Robotics (ISR), Online, 9–10 December 2020; Available online: https://ieeexplore.ieee.org/abstract/document/9307464 (accessed on 6 February 2022).

21. Lu, W.; Hu, Z.; Pan, J. Human–robot Collaboration using Variable Admittance Control and Human Intention Prediction. In Proceedings of the 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), Hong Kong, China, 20–21 August 2020; pp. 1116–1121. [CrossRef]

22. Roveda, L.; Maskani, J.; Franceschi, P.; Abdi, A.; Braghin, F.; Tosatti, L.M.; Pedrocchi, N. Model-Based Reinforcement Learning Variable Impedance Control for Human–robot Collaboration. *J. Intell. Robot. Syst.* **2020**, *100*, 417–433. [CrossRef]

23. Sasagawa, A.; Fujimoto, K.; Sakaino, S.; Tsuji, T. Imitation Learning Based on Bilateral Control for Human–Robot Cooperation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6169–6176. [CrossRef]

24. Van der Spaa, L.; Gienger, M.; Bates, T.; Kober, J. Predicting and Optimizing Ergonomics in Physical Human–robot Cooperation Tasks. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 1799–1805. [CrossRef]

25. Ghadirzadeh, A.; Chen, X.; Yin, W.; Yi, Z.; Bjorkman, M.; Kragic, D. Human-Centered Collaborative Robots with Deep Reinforcement Learning. *IEEE Robot. Autom. Lett.* **2020**, *6*, 566–571. [CrossRef]

26. Vinanzi, S.; Cangelosi, A.; Goerick, C. The Role of Social Cues for Goal Disambiguation in Human–robot Cooperation. In Proceedings of the 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Naples, Italy, 31 August–4 September 2020; pp. 971–977. [CrossRef]

27. Mariotti, E.; Magrini, E.; De Luca, A. Admittance Control for Human–robot Interaction Using an Industrial Robot Equipped with a F/T Sensor. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 6130–6136. [CrossRef]

28. Villani, V.; Pini, F.; Leali, F.; Secchi, C. Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics* **2018**, *55*, 248–266. [CrossRef]

29. Xu, Z.; Zhao, J.; Yu, Y.; Zeng, H. Improved 1D-CNNs for behavior recognition using wearable sensor network. *Comput. Commun.* **2020**, *151*, 165–171. [CrossRef]

30. Xia, C.; Sugiura, Y. Wearable Accelerometer Optimal Positions for Human Motion Recognition. In Proceedings of the 2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech), Kyoto, Japan, 10–12 March 2020; pp. 19–20. [CrossRef]

31. Zhao, Y.; Man, K.L.; Smith, J.; Siddique, K.; Guan, S.-U. Improved two-stream model for human action recognition. *EURASIP J. Image Video Process.* **2020**, *2020*, 1–9. [CrossRef]

32. Gu, Y.; Ye, X.; Sheng, W.; Ou, Y.; Li, Y. Multiple stream deep learning model for human action recognition. *Image Vis. Comput.* **2019**, *93*, 103818. [CrossRef]

33. Srihari, D.; Kishore, P.V.V.; Kumar, E.K.; Kumar, D.A.; Kumar, M.T.K.; Prasad, M.V.D.; Prasad, C.R. A four-stream ConvNet based on spatial and depth flow for human action classification using RGB-D data. *Multimed. Tools Appl.* **2020**, *79*, 11723–11746. [CrossRef]

34. Rahimi, A.; Kumar, K.D.; Alighanbari, H. Fault detection and isolation of control moment gyros for satellite attitude control subsystem. *Mech. Syst. Signal Process.* **2019**, *135*, 106419. [CrossRef]

35. Shao, S.-Y.; Sun, W.-J.; Yan, R.-Q.; Wang, P.; Gao, R.X. A Deep Learning Approach for Fault Diagnosis of Induction Motors in Manufacturing. *Chin. J. Mech. Eng.* **2017**, *30*, 1347–1356. [CrossRef]

36. Agriomallos, I.; Doltsinis, S.; Mitsioni, I.; Doulgeri, Z. Slippage Detection Generalizing to Grasping of Unknown Objects using Machine Learning with Novel Features. *IEEE Robot. Autom. Lett.* **2018**, *3*, 942–948. [CrossRef]

37. Dimeas, F.; Avendaño-Valencia, L.D.; Aspragathos, N. Human–robot collision detection and identification based on fuzzy and time series modelling. *Robotica* **2014**, *33*, 1886–1898. [CrossRef]

38. Sharkawy, A.-N.; Mostfa, A.A. Neural networks' design and training for safe human–robot cooperation. *J. King Saud Univ. Eng. Sci.* **2021**. [CrossRef]

39. Czubenko, M.; Kowalczuk, Z. A Simple Neural Network for Collision Detection of Collaborative Robots. *Sensors* **2021**, *21*, 4235. [CrossRef]

40. Zhang, J.; Liu, H.; Chang, Q.; Wang, L.; Gao, R.X. Recurrent neural network for motion trajectory prediction in human–robot collaborative assembly. *CIRP Ann.* **2020**, *69*, 9–12. [CrossRef]

41. Chen, X.; Jiang, Y.; Yang, C. Stiffness Estimation and Intention Detection for Human–robot Collaboration. In Proceedings of the 2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA), Kristiansand, Norway, 9–13 November 2020; pp. 1802–1807. [CrossRef]

42. Chen, X.; Wang, N.; Cheng, H.; Yang, C. Neural Learning Enhanced Variable Admittance Control for Human–Robot Collaboration. *IEEE Access* **2020**, *8*, 25727–25737. [CrossRef]

43. Rezayati, M.; van de Venn, H.W. *Physical Human-Robot Contact Detection, Mendeley Data Repository*; Version 2; Elsevier: Amsterdam, The Netherlands, 2021. [CrossRef]

44. Abu Al-Haija, Q.; Alsulami, A.A. High-Performance Classification Model to Identify Ransomware Payments for Het-ero-geneous Bitcoin Networks. *Electronics* **2021**, *10*, 2113. [CrossRef]

45. Zhang, M.-L.; Zhou, Z.-H. A k-nearest neighbor based algorithm for multi-label classification. In Proceedings of the 2005 IEEE International Conference on Granular Computing, Beijing, China, 25–27 July 2005; Volume 2, pp. 718–721. [CrossRef]

46. Karsmakers, P.; Pelckmans, K.; Suykens, J.A.K. Multi-class kernel logistic regression: A fixed-size implementation. In Proceedings of the 2007 International Joint Conference on Neural Networks, Orlando, FL, USA, 12–17 August 2007; pp. 1756–1761. [CrossRef]

47. Feng, J.; Yu, Y.; Zhou, Z.H. Multi-layered gradient boosting decision trees. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 1–11.

48. Yurochkin, M.; Bower, A.; Sun, Y. Training individually fair ML models with sensitive subspace robustness. *arXiv* **2019**, arXiv:1907.00020.

49. Abu Al-Haija, Q. Top-Down Machine Learning-Based Architecture for Cyberattacks Identification and Classification in IoT Communication Networks. *Front. Big Data* **2022**, *4*, 782902. [CrossRef]

50. Abu Al-Haija, Q.; Ishtaiwi, A. Multiclass Classification of Firewall Log Files Using Shallow Neural Network for Network Security Applications. In *Soft Computing for Security Applications. Advances in Intelligent Systems and Computing*; Ranganathan, G., Fernando, X., Shi, F., El Allioui, Y., Eds.; Springer: Singapore, 2022; pp. 27–41. [CrossRef]

51. Chujai, P.; Chomboon, K.; Teerarassamee, P.; Kerdprasop, N.; Kerdprasop, K. Ensemble learning for imbalanced data classification problem. In Proceedings of the 3rd International Conference on Industrial Application Engineering (Nakhon Ratchasima), Kitakyushu, Japan, 28–31 March 2015.

52. Rodrigues, I.R.; Barbosa, G.; Filho, A.O.; Cani, C.; Dantas, M.; Sadok, D.H.; Kelner, J.; Souza, R.S.; Marquezini, M.V.; Lins, S. Modeling and assessing an intelligent system for safety in human-robot collaboration using deep and machine learning techniques. *Multimed Tools Appl.* **2021**, *81*, 2213–2239. [CrossRef]

53. Wen, X.; Chen, H.; Hong, Q. Human Assembly Task Recognition in Human–Robot Collaboration based on 3D CNN. In Proceedings of the 2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control and Intelligent Systems (CYBER), Suzhou, China, 29 July–2 August 2019; pp. 1230–1234.

54. Heo, Y.J.; Kim, D.; Lee, W.; Kim, H.; Park, J.; Chung, W.K. Collision Detection for Industrial Collaborative Robots: A Deep Learning Approach. *IEEE Robot. Autom. Lett.* **2019**, *4*, 740–746. [CrossRef]

55. Anvaripour, M.; Saif, M. Collision Detection for Human–robot Interaction in an Industrial Setting Using Force Myography and a Deep Learning Approach. In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 2149–2154.

56. Jain, A.; Koppula, H.S.; Raghavan, B.; Soh, S.; Saxena, A. Car that Knows Before You Do: Anticipating Maneuvers via Learning Temporal Driving Models. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3182–3190. [CrossRef]