IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 August 2022, accepted 31 August 2022, date of publication 14 September 2022, date of current version 26 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3206436

RESEARCH ARTICLE

On EMG Based Dexterous Robotic Telemanipulation: Assessing Machine Learning Techniques, Feature Extraction Methods, and Shared Control Schemes

RICARDO V. GODOY[1], (Graduate Student Member, IEEE), ANANY DWIVEDI[2], BONNIE GUAN[1], (Graduate Student Member, IEEE), AMBER TURNER[3], DASHA SHIEFF[1], AND MINAS LIAROKAPIS[1], (Senior Member, IEEE)

[1]New Dexterity Research Group, Department of Mechanical and Mechatronics Engineering, The University of Auckland, Auckland 1010, New Zealand
[2]Chair of Autonomous Systems and Mechatronics, Friedrich-Alexander-Universität Erlangen–Nürnberg, 91054 Erlangen, Germany
[3]PricewaterhouseCoopers, Auckland 1010, New Zealand

Corresponding author: Ricardo V. Godoy (rdeg264@aucklanduni.ac.nz)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the University of Auckland Human Participants Ethics Committee (UAHPEC) under Reference No. 019043.

ABSTRACT Electromyography (EMG) signals are commonly used for the development of Muscle Machine Interfaces. EMG-based solutions provide intuitive and often hand-free control in a wide range of applications that range from the decoding of human intention in classification tasks to the continuous decoding of human motion employing regression models. In this work, we compare various machine learning and feature extraction methods for the creation of EMG based control frameworks for dexterous robotic telemanipulation. Various models are needed that can decode dexterous, in-hand manipulation motions and perform hand gesture classification in real-time. Three different machine learning methods and eight different time-domain features were evaluated and compared. The performance of the models was evaluated in terms of accuracy and time required to predict a data sample. The model that presented the best performance and prediction time trade-off was used for executing in real-time a telemanipulation task with the New Dexterity Autonomous Robotic Assistance (ARoA) platform (a humanoid robot). Various experiments have been conducted to experimentally validate the efficiency of the proposed methods. The robotic system is shown to successfully complete a series of tasks autonomously as well as to efficiently execute tasks in a shared control manner.

INDEX TERMS Muscle-machine interfaces, electromyography, shared control, intention decoding, telemanipulation, machine learning.

I. INTRODUCTION

Recent advances in the development of muscle-machine interfaces (MuMI) have made possible its use in several applications. One of the most used and successful approaches is to employ electromyography (EMG) signals to develop MuMIs. These electrical signals are generated during muscle contraction and carry vital information intrinsic to the movement of the muscles that are triggered to accomplish the motion performed by the subject. MuMIs based on EMG signals usually provide a hands-free, unobstructive solution, offering intuitive and natural operation of robotic devices in complex applications [1], such as the execution of teleoperation and telemanipulation tasks with robot arm hand systems [2], [3], [4], as shown in Fig. 1.

Machine learning (ML) techniques have been employed to decode EMG signals to perform both classification (e.g. decoding discrete human gestures) [5], [6], [7], [8] and

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Botter.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

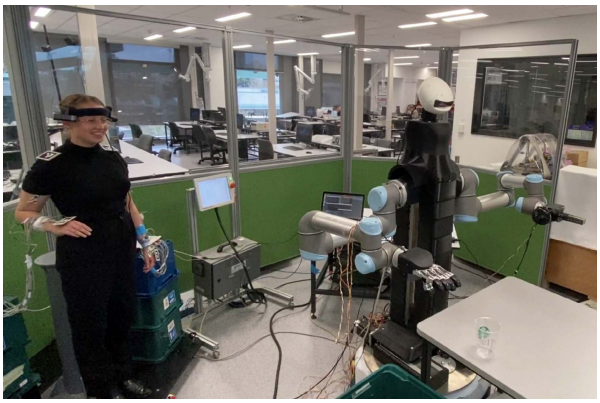

FIGURE 1. Example of a user teleoperating the new dexterity ARoA humanoid robot using an EMG based shared control framework for the execution of complex tasks.

regression (e.g. decoding continuous human motions) [9], [10], [11]. Classification results in a discrete output, for example, the discrete decision on the user's intention. In our previous work, we employed a Random Forest (RF) classifier in a framework that switches between autonomous and manual operation of robot systems to complete a task in a synergistic manner [2]. We also employed regression-based models to perform continuous decoding of the human motion using sEMG signals as input [12].

Classic machine learning techniques such as RF employ preprocessed signals and hand-crafted features as input to obtain the desired output. EMG features can be extracted from raw signals using Time Domain (TD) features, Frequency Domain (FD) features, or Time-Frequency Domain (TFD) features methods [13]. Past studies show that TD features result in more consistent performance over time than FD [14], [15]. In our previous study [16], we compared the performance of eight TD feature extraction techniques in discriminating between different grasping postures and gestures executed by the user of an EMG-based MuMI using five different ML techniques. It was found that the RF model presented the most consistent results. More robust models introduced in the deep learning (DL) [17], [18], [19] era can identify and incorporate patterns from processed data resulting in an increasingly complex system, often presenting a better performance in terms of correlation and accuracy compared to classic ML models [7], [20] at the cost of requiring greater computational power and potentially sacrificing real-time performance.

In this paper, we assess the ability of ML and DL models to solve regression and classification problems and enable the development of more functional MuMIs. To do this, we assess which hand-extracted features are most relevant for decoding dexterous, in-hand manipulation motions, and decoding human hand gestures. The performance of RF models, a Convolutional Neural Network (CNN) benchmark model, and a novel DL technique called Temporal Multi-Channel Transformers (TMC-T) are compared for eight different TD features. The TMC-T model is shown in Fig. 2. The models are evaluated in terms of accuracy and speed of calculation. In the second step, the model that presented the best performance for online applications (considering the trade-off between accuracy and processing time) is employed to develop a shared control framework for intuitive robotic telemanipulation.

II. EXPERIMENTS

In this section we present experiments conducted to evaluate the performance of the decoding models that were developed for the regression and the classification tasks. To do this, three different experiments were designed to evaluate the EMG-based decoding models in the regression and classification use cases: i) Decoding of Dexterous, In-Hand Manipulation Motions, ii) Hand Gesture Classification, and iii) Robotic Telemanipulation using the New Dexterity Autonomous Robotic Assistant (ARoA) humanoid platform [21].

A. DECODING OF DEXTEROUS, IN-HAND MANIPULATION MOTIONS

In this subsection, we present the guidelines employed for training the RF, CNNs, and TMC-T-based regression models to perform decoding of dexterous, in-hand manipulation motions using EMG signals as input.

1) DATASET

The regression models were trained on the dataset collected by the New Dexterity research group [12]. For this dataset, before the start of data collection, each participant was asked about any disabilities that may affect the quality of the data. Finally, myoelectric activations were acquired from 11 non-disabled subjects from 16 different muscles (8 on the hand and 8 on the forearm of the subject). For the hand, three electrodes were placed on the palm measuring the activity of the Lumbrical muscles, four electrodes were placed at the back of the palm measuring the activity of the Interossei and one electrode was placed on the base of the thumb to measure the myoelectric activations of the Opponens Pollicis muscle. For the forearm, three electrodes were placed on the Extensor Digitorum site, three were placed on the Flexor Digitorum site, one was placed on Abductor Pollicis Longus, and the final one was placed to measure the myoelectric activations of the Extensor Pollicis Brevis. The ground electrode was placed on the elbow, where the muscular activity becomes minimal. For data acquisition, two double-differential EMG electrodes were employed to measure the myoelectric activations. The EMG signals were acquired at a sampling rate of 1200 Hz by the bioamplifier, which bandpass filtered the data using a Butterworth filter (5 Hz - 500 Hz). The electric line noise was filtered out using a notch filter of 50 Hz.

For each subject, ten trials were recorded for each motion. For each trial, the subjects performed a 3-dimensional equilibrium point manipulation task using a Rubik's cube, a chips can from the Yale-CMU-Berkeley (YCB) grasping object set [22], and a custom-made off-centered mass cube.

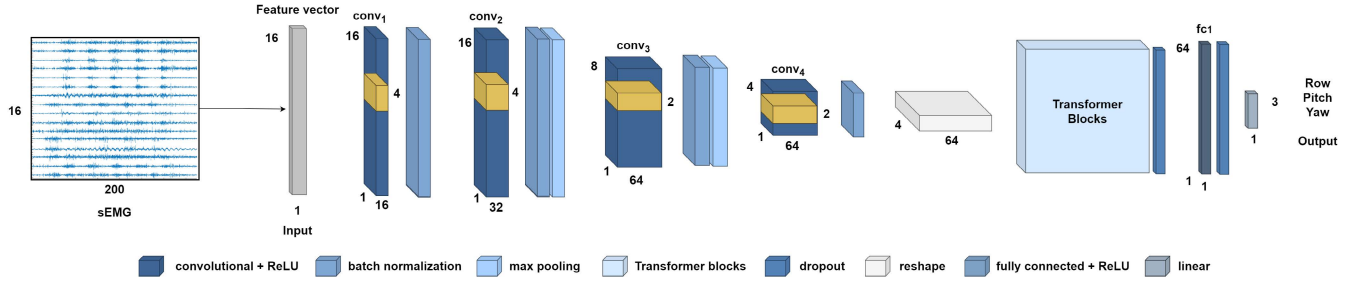

FIGURE 2. TMC-T model. The feature vector obtained using the feature extraction methods is provided to four convolutional layers, which extract features and learn embeddings from the input feature vector. Batch normalization layers follow the convolutional layers. Two max-pooling layers reduce the EMG channels dimension while keeping the most relevant information. After the convolutional blocks, the output is reshaped and provided to transformer blocks and fully-connected layers. The output from the TMC-T model is the Row, Pitch, and Yaw motions.

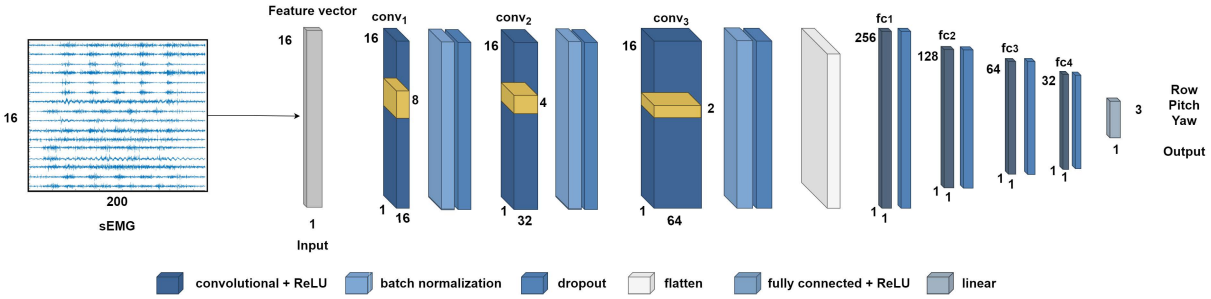

FIGURE 3. CNN model. *Conv* stands for convolutional and *fc* for fully-connected. First, the feature vector is obtained using the feature extraction methods discussed in Section III-A. Three convolutional layers extract features using convolutional kernels. The convolutional layers are followed by batch normalization and dropout layers. After the convolutional blocks, the output is flattened and provided to fully-connected layers. The output from the CNN model is the roll, pitch, and yaw motions.

Each trial started with 5 seconds of rest (where the object is held in a stationary pose) and was followed by five repetitions of each motion. The manipulation tasks performed during the experiments were:

- Pitch: a coordinated movement of the fingers that creates the pitch motion;
- Roll: a coordinated movement of the fingers that creates a roll motion;
- Yaw: a coordinated movement of the fingers that creates a yaw motion.

In the dataset, the myoelectric activations are the independent variables, and the motion of the object while being manipulated in the hands of the subjects is the dependent variable. The motion of the object refers to its rotation along the x-axis, y-axis, and z-axis, where the origin of the motions is set to the center of the object when it was in the stationary pose during the 5 seconds rest period.

2) TRAINING AND EVALUATION

Our models were trained on a Google Colab Pro virtual machine with GPU. The models were developed in Python using Tensorflow and Keras. The DL models employed the mean squared error (MSE) loss function and Adam as the optimizer [23]. Leave-one-out cross-validation was used to evaluate the RF, CNN, and TMC-T models.

The trained RF, CNN and TMC-T models' efficiency was assessed using the Pearson correlation coefficient and the percentage of the Normalized Mean Square Error (NMSE) for accuracy. The NMSE value of 0% implies a bad fit, whereas the NMSE value of 100% implies that the two motions are identical. The NMSE value is defined as follows:

$$NMSE(\%) = 100 * \left(1 - \frac{||x_r - x_p||^2}{||x_r - mean(x_r)||^2}\right) \quad (1)$$

where ||.|| indicates the 2-norm of a vector, x_r is the actual reference motion and x_p refers to the predicted motion.

B. HAND GESTURE CLASSIFICATION

In this subsection, we present the experiments that were conducted to evaluate the efficiency of the hand gesture classification models.

1) DATASET

For this set of experiments, we utilize the dataset collected by the New Dexterity research group in [16]. In this dataset, each subject was instructed to alternate between a rest state and a gesture state. In total, six gestures were recorded: i) a pinch grasp, ii) a tripod grasp, iii) a power grasp, iv) an open hand configuration with abducted fingers, v) co-contraction of all muscles, and vi) rest state. The myoelectric activations were acquired from 8 different muscle groups of the human arm and hand. The details regarding the electrode placement can be found in [16].

For data acquisition, an appropriate package in the robot operating system (ROS) [24] based framework was implemented. The beginning and termination of each gesture were prompted to the subject with a visual cue on a computer

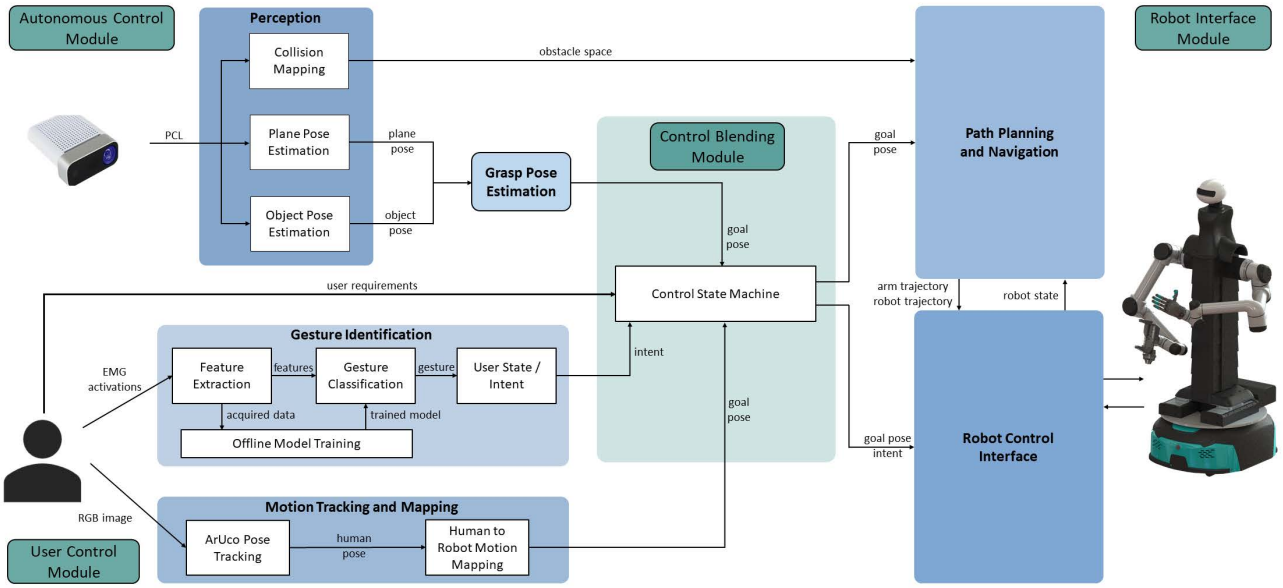

FIGURE 4. The robot system operates autonomously using the autonomous control module, whereas the manual operation of the robot is achieved through the user control module. The robot interface connects to the robot hardware. Shared control of the robot system is provided by the control blending module that merges the autonomous and user control modules.

screen in the form of a three-second timer. For the labeling of the data, a software trigger was sent to the data recording script to isolate the rest phase from the gesture phase, when the visual cues were presented to the subject. For each performed gesture, ten seconds of rest state were recorded followed by another ten seconds of gesture execution with five repetitions each.

2) TRAINING AND EVALUATION

The sparse categorical cross-entropy loss function was employed to develop the DL hand gesture decoding models. Once again, leave-one-out cross-validation was used for model evaluation. For models performance, the percentage of correct classifications (accuracy) was used as metric.

In order to choose the best model for robotic telemanipulation, we defined a metric σ to evaluate the accuracy and execution time trade-off. To do this, we selected two thresholds: the minimum desired accuracy (A) and the maximum accepted execution time (E). These two threshold values may vary depending on the application. For our robotic telemanipulation task, accuracy higher than 90% and an execution time (in *ms*) faster than the EMG data collection, so the user does not experience latency, are desired. Hence, for our application, $A = 0.9$ and $E = 0.0833$.

For a $\delta_A = \alpha - A$ and $\delta_E = E - \epsilon$ where α is the accuracy obtained and ϵ is the execution time of the model, the accuracy-execution time trade-off is defined as follows:

$$\sigma(\alpha, A, \epsilon, E) = \frac{\delta_A + |\delta_A|}{2\delta_A} \cdot \frac{\delta_E + |\delta_E|}{2\delta_E} \cdot \frac{\alpha}{\epsilon} \tag{2}$$

The first two terms of Equation 2 guarantee that the model meets the requirements. If the obtained accuracy is smaller than the accuracy threshold, the first term of the equation is zero, making the proposed evaluation metric $\sigma = 0$. If the model presents an execution time bigger than the desired, the second term, again making $\sigma = 0$. If both accuracy and execution time requirements are met, σ is the ratio between α and ϵ. Hence, the higher the accuracy and the lower the execution time obtained by the model, the higher the accuracy-execution time trade-off metric.

C. ROBOTIC TELEMANIPULATION USING THE ARoA PLATFORM

In this subsection, we present the experiments performed to evaluate a shared control framework for the real-time operation of the ARoA humanoid platform in the execution of telemanipulation tasks. To do this, the EMG-based shared control framework presented in Fig. 4 was developed so as to allow the user to take control of the robot platform and perform complex tasks whenever autonomous execution is not feasible. The proposed framework is divided into four main modules, namely, the **Autonomous Control Module**, the **User Control Module**, the **Robot Interface Module** and the **Control Blending Module**. Each module is represented in block layout, where each block conceptually corresponds to a node implemented within ROS.

1) AUTONOMOUS CONTROL MODULE

This module facilitates task execution without human intervention. To do this, it utilizes the *Perception* and *Grasp Pose Estimation* sub-modules. The *Perception* sub-module allows it to identify objects in 3D space and map the environment so as to build the obstacle space needed for an effective interaction with the robot surroundings. The *Grasp Pose Estimation* sub-module takes the poses of the objects and derives the goal pose of the robot end-effector for successfully grasping the object.

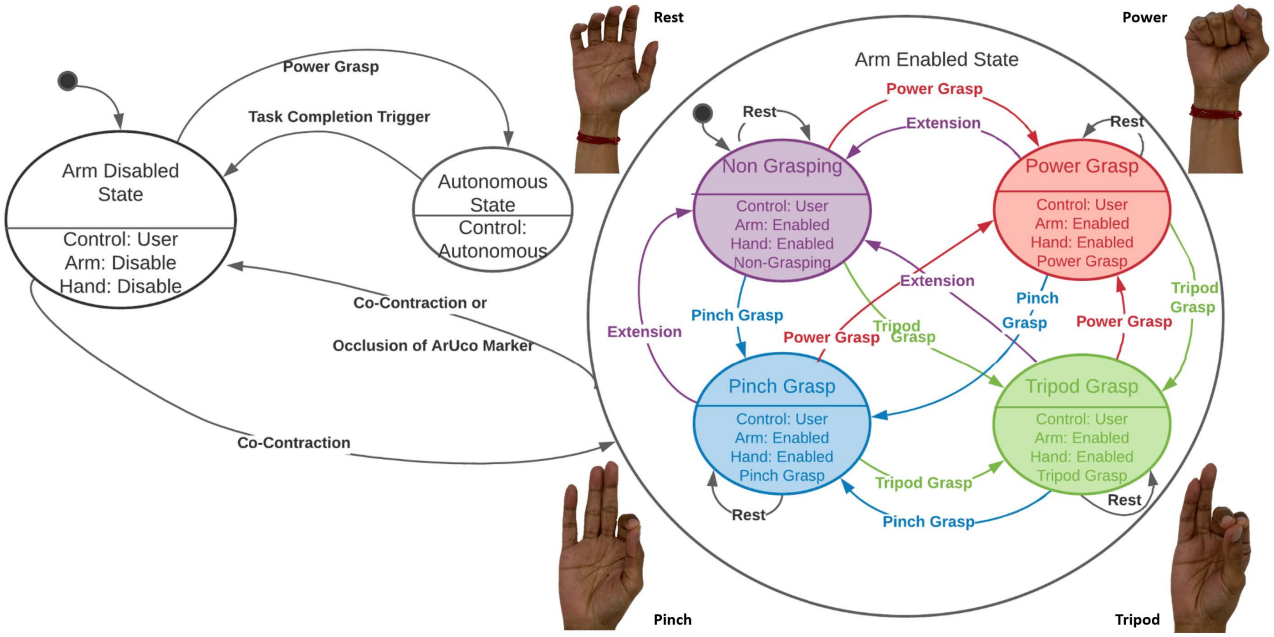

FIGURE 5. State machine scheme. Control actions based on different grasp types allow the robot arm and hand control to be switched between autonomous and user-operated, when needed.

2) USER CONTROL MODULE

This module enables the user to take manual control of the robot to execute tasks. It comprises of two sub-modules: *Gesture Identification* and *Motion Tracking and Mapping*. Using the *Gesture Identification* sub-module, the intention of the user can be interpreted. The *Motion Tracking and Mapping* sub-module is used to map the user's motion to the robot. This is done by tracking two fiducial (ArUco) markers attached on the clothes of the user (one at the back of the hand and the other one on the shoulder), using a head-mounted camera. Using the tracked poses of the ArUco markers, the relative position of the user's hand can be calculated with respect to their shoulder. This relative motion of the user's hand can be mapped on the robot. More information about the mapping can be found in [2].

3) CONTROL BLENDING MODULE

This module is responsible for switching between the autonomous task execution and the manual control of the robot by a user (teleoperation mode). It receives high level user requirements as inputs regarding the autonomous task execution (e.g., tidy the table or wipe the whiteboard). It also comprises of the Control State Machine block, which keeps track of the current state of the task execution and is updated by the user through execution of different gestures. When user control is active, both the hand and arm controls are enabled simultaneously. The robot hand used in this experiment is the New Dexterity NDX-A anthropomorphic robot hand [25] and can be controlled in the execution of pinch, power, and tripod gestures and grasps while decoding of the rest gesture maintains the previous state. To bring the hand in an open state, the extension gesture should be executed. When the arm control is enabled, the user can control the motion of the robot. To enable and disable the manual control of the robot arm, the co-contraction gesture is used. When manual control of the arm control is disabled, the control can be handed over to the robot using a power gesture that triggers autonomous operation. When manual control is disabled, the robot's current end-effector pose is maintained and used as the start reference frame. An overview of the control state machine is presented in Fig. 5.

4) ROBOT INTERFACE MODULE

This module enables communication with the robot hardware of the ARoA platform. The robot interface module comprises the *Path Planning and Navigation* sub-module, which is responsible for autonomous task execution and collision avoidance, and the *Robot Control Interface*, an interface to the robot hardware. The Control State Machine communicates with the *Path Planning and Navigation* for autonomous task executions, which passes goal trajectories to the *Robot Control Interface*. In the case of manual user control, the Control State Machine communicates directly with the *Robot Control Interface*.

III. METHODS

This section describes the TD features extracted from the raw EMG data, along with the different ML methods used to develop intention decoding models. The ML model that achieves the best predicting performance in terms of accuracy and processing speed, evaluated through the σ metric, is the RF classification. The RF model was then used to perform

TABLE 1. Summary of the features employed with corresponding formulae and descriptions.

Feature Extraction Methods	RMS	WL	ZC	IEMG
Formula	$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$	$WL = \sum_{i=1}^{N-1} \lvert x_{i+1} - x_i \rvert$	$x_k < 0 \quad \&\& \quad x_{k+1} > 0$ $\lvert\lvert\lvert$ $x_k > 0 \quad \&\& \quad x_{k+1} < 0$ $\&\&$ $\lvert x_k - x_{k+1} \rvert > V_t$	$IEMG = \sum_{i=1}^{N} \lvert x_i \rvert$
Description	Represents the average power of the signal.	Measure of complexity of the EMG signal.	Indicator of fatigue in muscles.	Detects the onset of muscle activity.
Feature Extraction Methods	**MAV**	**WAMP**	**VAR**	**LogD**
Formula	$MAV = \frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert$	$WAMP = \sum_{i=1}^{N-1}[f(\lvert x_n - x_{n+1} \rvert)]$ $\begin{cases} 1 & x \geq threshold \\ 0 & otherwise \end{cases}$	$VAR = \frac{1}{N-1}\sum_{i=1}^{N} x_i^2$	$LOG = e^{\frac{1}{N}\sum_{i=1}^{N} log(\lvert x_i \rvert)}$
Description	Measures the contraction level of muscles.	Measures the MUAP and the contraction level of muscles.	Measure of EMG signal power.	Provides an estimate of the muscle contraction force.

in a shared control manner robotic telemanipulation with the New Dexterity ARoA platform.

A. FEATURE EXTRACTION

To extract meaningful information from the raw EMG signals, eight TD features were extracted. To do this, the raw EMG signals were segmented using a sliding window of 166.67 ms with increments of 16.67 ms [26]. The performance of the decoding models depends on the window size and the stride. A very large window size is not ideal for real-time applications since the response rate of the system becomes slower, while a very small window leads to noisy predictions due to high biases and variance [13]. The eight different TD features were extracted from each EMG channel to develop machine learning-based models for decoding different hand gestures. The features examined are as follows: Root Mean Square Value (RMS), Waveform Length (WL), Zero Crossings (ZC), Integrated EMG (IEMG), Mean Absolute Value (MAV), Willison Amplitude (WAMP), Variance (VAR), and Log Detector (LogD) [12], [27], [28], [29], [30].

1) *Root Mean Square:* The RMS of the EMG signal is one of the most commonly used values in the TD. It represents the square root of the average power of the signal for the given time period.
2) *Waveform Length:* WL measures the complexity of the signal. It represents the waveform's amplitude, frequency, and duration in a single parameter.
3) *Zero Crossings:* ZC provides rough FD information and represents the number of times the signal crosses the zero value in a given time period. ZC can also be used to estimate the fatigue in the muscles.
4) *Integrated EMG:* IEMG is the summation of the absolute values of the EMG signal amplitude. It is generally used as onset index to detect muscle activity.
5) *Mean Absolute Value:* MAV can be calculated by taking the average of the absolute value of the EMG signal. It is similar to IEMG, which detects the onset of muscle activity. It also provides information regarding muscle contraction levels.
6) *Willison Amplitude:* WAMP represents the number of times that the difference between EMG signal amplitude among two consecutive values exceed a predefined threshold. WAMP is related to the firing of motor unit action potentials (MUAP) and the muscle contraction level.
7) *Variance:* VAR of the myoelectric activations measures the power of the signal. Variance is the mean value of the square of the deviation of that variable.

TABLE 2. Average motion decoding correlation (C) and accuracy (A) across all subjects for RF-based models developed for specific objects using 1, 25, 50, 75, and 100 trees for each TD feature investigated in this study. For 1 tree, random forests behave like a classic decision tree. The highest accuracy for each object and feature is highlighted in blue while the lowest in yellow.

Cube																
Trees	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
1	65.25	37.63	68.93	42.92	47.96	18.46	65.62	38.76	66.22	39.23	63.01	35.56	65.76	38.81	65.38	38.22
25	78.85	59.32	80.71	62.15	61.64	37.83	79.24	59.62	79.21	59.55	75.35	54.55	78.81	59.28	77.79	57.39
50	79.45	59.97	80.93	62.57	62.26	38.47	79.64	60.16	79.61	60.09	75.64	54.91	79.42	59.87	78.25	**57.94**
75	79.21	**60.42**	80.51	**63.05**	64.99	**41.88**	75.79	**63.01**	77.30	**60.38**	75.99	**56.11**	79.17	**60.24**	76.92	56.92
100	79.70	56.17	81.11	58.73	62.47	35.64	79.86	56.66	79.84	56.65	75.77	52.13	79.66	56.14	78.43	55.51
Cylinder																
Trees	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
1	55.88	24.19	59.29	28.54	40.99	12.49	56.32	25.12	56.61	25.39	55.96	24.39	55.44	24.17	55.70	24.83
25	70.38	49.30	72.89	52.70	56.42	31.68	70.49	49.41	70.61	49.62	68.71	46.77	70.61	49.58	69.23	47.85
50	70.92	49.98	73.36	53.32	57.10	32.33	71.10	50.15	71.23	50.28	69.15	47.40	71.04	50.14	69.80	48.45
75	68.69	47.03	72.26	51.74	54.61	29.51	70.33	47.95	69.24	47.88	67.59	45.53	68.70	47.05	68.49	46.63
100	71.34	**52.38**	73.42	**55.46**	57.45	**36.55**	71.43	**52.62**	71.39	**52.57**	69.27	**49.67**	71.13	**52.36**	70.09	**50.79**
OffCenterCube																
Trees	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
1	58.27	28.25	61.04	31.98	44.20	14.87	58.53	28.59	58.70	28.81	58.36	29.24	57.05	26.48	58.35	27.93
25	74.18	52.92	76.29	56.40	59.37	33.61	74.45	53.48	74.47	53.48	72.02	50.95	74.28	52.99	73.59	52.09
50	74.79	**53.65**	76.94	57.20	60.15	34.37	75.03	**54.19**	75.05	**54.19**	72.57	51.58	74.81	**53.66**	73.93	**52.54**
75	72.50	50.31	75.20	54.94	59.16	32.67	73.44	51.95	73.42	51.94	71.52	50.29	72.51	50.31	73.24	51.59
100	74.97	53.53	77.07	**57.34**	60.47	**37.30**	75.31	53.85	75.31	53.87	72.83	**52.60**	75.00	53.57	74.22	52.40

8) *Log Detector:* LogD also provides an estimate of the muscle contraction force.

A summary of all the different features that were calculated and examined in this study along with details regarding the calculation of each feature can be found in Table 1.

B. MACHINE LEARNING METHODS

To decode dexterous in-hand manipulation motions and hand gestures, we trained a series of regression and classification models using three different machine learning methods: RF, CNN, and TMC-T. For each learning method examined, eight different decoding models were developed using each one of the extracted features at a time. The ability of the models to discriminate between the different grasping postures and gestures was evaluated employing the leave-one-out cross-validation method, testing on each subject at the time and training for the rest of them. The DL hyperparameters were tuned by executing cross-validation with 10% of the available training data for optimization by empirical evaluation. The DL classification and regression models share the same hyperparameters. The difference between DL classification and regression models is in the loss function used (as explained in Sections II-A2 and II-B2) and in the last layer: the classification models employ a softmax layer with six neurons (to discriminate between six gestures). The regression models employ a dense layer with linear activation and three neurons (to decode the three motions).

1) RANDOM FOREST

RF is an ensemble regression method that is based on a combination of multiple decision trees. This is a classic ML technique in which the output is the most popular class among the decisions of the individual trees for the classification case or the average of the estimations of the individual trees in the regression case [12], [31], [32]. RF models offer good predictive performance and are extremely fast, at the cost of not being as robust as most DL techniques.

2) CNN

CNN is a well-established DL technique, representing the state-of-the-art in several tasks and application fields due to its ability to identify patterns and extract spatial characteristics of the data. In this work, we employed the CNN as a DL benchmark model to be compared with the RF and TMC-T models, since this technique is widely used for both classification and regression using EMG signals as input [8], [20], [33], [34]. The proposed CNN model is composed of three convolutional blocks. Each convolution block contains a convolutional layers, followed by a batch normalization [35] and a 0.1 dropout layer [36]. The filters sizes are shown in Fig. 3. A stride of *(1,1)* was used, and padding was applied so the output has the same size as the input. After the convolutional blocks, four dense layers employed with 256, 128, 64, and 32 neurons. A final linear with three neurons performs the motion decoding. The CNN model is shown in Fig. 3.

TABLE 3. Pearson correlation (C) and accuracy (A) for tested models employing 8 TD features as input for the cube.

Subject	Cube															
	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
TMC-T																
1	85.07	69.03	80.80	90.90	74.03	54.47	85.52	69.59	84.70	68.45	88.31	76.17	84.51	68.00	84.56	68.64
2	86.54	71.58	68.60	84.35	67.78	42.25	84.99	68.82	84.84	68.65	76.37	55.92	85.82	69.85	82.91	65.77
3	83.48	68.66	79.00	89.25	69.97	46.34	84.52	70.27	84.16	69.26	82.43	66.78	52.97	27.48	81.65	64.98
4	85.56	71.94	70.17	85.36	74.14	54.40	85.06	71.52	85.51	72.16	75.65	56.06	83.85	69.28	81.07	65.09
5	87.26	74.90	78.83	89.48	59.23	31.82	87.58	75.68	88.16	76.97	83.66	68.57	86.92	74.82	87.63	75.81
6	83.82	68.02	70.20	85.09	53.29	24.88	84.27	69.64	84.72	69.16	78.05	56.64	82.44	66.81	81.79	63.94
7	79.77	62.30	60.11	78.40	52.19	24.12	78.87	60.52	78.78	60.33	72.21	51.32	77.03	57.43	73.14	51.79
8	64.14	34.26	33.75	63.35	45.89	16.94	61.14	29.53	60.90	30.72	49.87	16.40	58.73	27.60	57.35	24.61
9	85.23	67.05	0.89	75.21	83.86	67.77	75.84	0.00	85.38	66.76	84.36	65.26	35.33	0.00	83.11	63.55
10	86.34	69.71	0.86	71.33	63.37	31.29	87.10	71.86	86.97	71.42	84.07	65.88	83.77	65.33	84.60	67.16
11	80.80	59.80	0.80	58.37	52.04	26.54	82.02	60.52	81.76	60.40	75.63	52.40	78.42	55.48	80.42	59.40
AVG	82.55	65.21	49.46	**79.19**	63.25	38.25	81.54	58.90	82.35	64.93	77.33	57.40	73.62	52.92	79.84	60.98
CNN																
1	83.30	65.90	89.98	77.64	71.71	50.24	83.40	66.42	83.76	66.63	86.93	72.89	82.00	63.78	82.59	64.99
2	83.80	66.02	85.96	70.49	67.08	41.15	84.11	66.96	83.31	65.53	86.10	72.09	82.18	63.48	81.01	62.16
3	82.62	66.68	88.21	77.30	67.03	43.11	81.54	63.72	83.06	66.69	82.71	67.33	82.43	66.78	79.46	60.71
4	84.73	70.70	85.52	71.94	72.03	52.69	85.02	71.44	84.65	70.77	76.09	58.20	83.48	69.27	81.15	65.84
5	86.72	74.35	88.86	78.03	56.23	30.08	87.05	74.76	86.70	74.20	83.14	68.35	85.86	72.72	86.41	74.15
6	83.28	67.64	84.87	69.89	51.83	27.80	84.32	68.87	84.33	68.89	77.16	57.48	72.42	47.71	81.27	63.67
7	78.79	60.68	78.00	59.45	52.16	25.17	78.25	60.12	78.82	60.91	70.77	49.61	66.33	39.74	74.10	54.05
8	66.87	39.05	62.95	31.79	46.10	17.81	67.26	39.94	66.43	38.64	58.31	29.88	71.37	46.89	60.94	30.07
9	88.32	75.29	90.17	79.22	84.23	68.91	88.47	75.73	88.03	74.98	84.02	67.78	58.11	31.71	85.88	71.23
10	85.33	69.53	86.90	72.79	65.34	38.90	85.94	70.51	85.29	69.64	72.89	49.30	74.80	52.99	84.05	67.09
11	80.20	57.28	82.20	60.83	51.57	23.94	81.28	58.51	80.16	57.47	74.91	50.60	77.40	53.47	80.29	57.45
AVG	82.18	64.83	83.96	**68.12**	62.30	38.16	82.42	65.18	82.23	64.94	77.55	58.50	76.03	55.32	79.74	61.04
RF																
1	84.69	67.45	90.59	78.56	70.62	50.03	84.67	67.91	84.49	67.57	87.30	72.37	84.64	67.41	84.17	67.36
2	84.04	62.31	84.66	67.17	64.64	37.85	84.16	62.34	84.02	62.10	78.12	58.45	84.11	62.39	82.72	61.64
3	82.36	66.49	86.40	73.25	71.22	49.22	82.80	67.24	82.60	66.93	82.52	66.94	82.19	66.31	80.59	64.15
4	81.65	67.01	80.16	62.91	76.03	59.03	81.13	66.31	81.05	66.29	74.48	54.85	81.68	67.02	79.43	63.91
5	83.12	67.61	87.69	75.59	58.63	33.40	85.00	70.61	85.04	70.80	80.93	63.59	83.18	67.73	85.34	71.23
6	80.85	64.77	81.09	65.45	51.04	26.25	80.62	64.51	80.41	64.13	74.32	54.95	80.80	64.66	76.59	57.70
7	76.64	56.99	75.53	55.94	53.42	26.04	76.39	56.50	76.77	56.92	72.00	50.20	76.71	57.00	73.99	52.98
8	52.50	22.58	56.36	24.51	46.29	16.81	52.38	22.29	52.20	22.07	42.08	11.38	52.55	22.42	52.44	21.23
9	86.70	71.86	86.55	72.04	81.87	66.25	85.69	70.13	85.64	70.10	83.98	66.20	86.76	72.00	82.92	64.48
10	81.70	59.94	83.32	62.24	64.50	35.40	82.50	61.17	82.58	61.39	80.95	59.75	81.68	59.91	82.46	61.89
11	79.69	52.64	77.88	50.59	46.63	22.91	80.71	52.75	80.93	52.74	75.33	45.32	79.28	51.78	80.12	50.79
AVG	79.45	59.97	80.93	**62.57**	62.26	38.47	79.64	60.16	79.61	60.09	75.64	54.91	79.42	59.87	78.25	57.94

3) TEMPORAL MULTI-CHANNEL TRANSFORMERS

The TMC-T [37] is a novel DL technique based on the Transformer architecture [38]. The TMC-T was adapted to use temporal signals with multiple channels as input, such as EMG. For this, convolution layers are used over the raw data to extract the embeddings supplied to the Transformer blocks. Along with convolution layers, max-pooling layers are employed to reduce the input data size. This step is of paramount importance since the attention mechanisms scale quadratically with the input length. The Transformer blocks are based only on attention mechanisms, which creates attention-based representation for each element in the input sequence.

Transformer-based models are designed to process sequential data without suffering from vanishing gradients like the recurrent neural networks and without presenting such complexity as the Long-short term memory (LSTM) networks or the impossibility of parallelization inherent to these recurrent techniques. Our TMC-T models benefit from the ability of the convolution layers to learn the spatial and temporal characteristics of the input data and the Transformer's ability to perform parallel computing and a faster training time. The TMC-T model is shown in Fig. 2.

IV. RESULTS

In this section, we present the results of the three sets of experiments conducted in this study. The three experiments are: i) Decoding of Dexterous, In-Hand Manipulation Motions, ii) Hand Gesture Classification, and iii) Robotic Telemanipulation with the New Dexterity ARoA Humanoid Platform.

TABLE 4. Pearson correlation (C) and accuracy (A) of tested models employing 8 TD extracted features as input for the cylinder.

Subject	Cylinder															
	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
TMC-T																
1	78.24	59.21	68.00	84.44	57.08	32.96	78.75	59.62	78.45	59.33	79.08	59.34	78.19	58.80	74.45	52.90
2	70.83	43.50	51.93	74.36	59.91	34.49	70.13	42.44	90.08	79.34	69.14	45.57	69.41	43.07	68.16	40.93
3	72.28	50.58	50.37	72.08	53.18	26.09	74.12	52.99	74.62	53.61	66.58	40.91	49.67	16.51	72.77	50.51
4	80.83	63.09	69.82	84.73	67.57	44.85	80.60	62.72	80.82	62.86	79.85	62.22	80.07	62.46	80.24	62.79
5	77.47	59.65	58.12	76.81	35.26	11.16	78.43	61.27	78.58	61.72	68.02	46.08	76.07	58.61	76.20	57.80
6	72.54	57.88	60.61	74.20	54.03	32.19	73.39	58.98	73.01	58.58	72.70	58.96	70.90	55.42	69.90	53.42
7	77.41	55.46	53.87	76.89	40.27	13.32	76.74	53.71	76.85	54.77	74.19	51.41	75.09	52.91	73.48	49.01
8	69.52	45.69	45.11	69.81	62.95	34.81	65.69	40.20	66.94	42.87	56.21	19.36	68.01	44.43	63.00	37.88
9	61.74	37.93	0.64	41.67	56.88	31.78	63.27	40.45	63.81	40.64	66.61	45.99	62.98	41.53	64.92	42.82
10	78.26	56.50	0.82	63.75	73.25	50.26	79.51	58.59	80.51	60.59	80.75	61.11	76.22	54.53	80.90	60.46
11	73.76	52.95	0.80	61.41	67.83	44.88	73.22	53.71	72.08	51.90	75.66	54.36	69.82	49.71	67.33	46.47
AVG	73.90	52.95	41.83	**70.92**	57.11	32.44	73.99	53.15	75.98	56.93	71.71	49.57	70.58	48.91	71.94	50.45
CNN																
1	79.59	61.97	85.22	70.46	58.96	35.35	78.60	60.11	78.40	59.83	80.13	62.38	77.82	59.13	74.51	53.55
2	70.97	47.15	73.35	51.53	60.03	35.38	71.21	47.47	70.33	45.60	69.71	47.04	71.19	47.18	69.28	45.21
3	73.53	52.69	72.37	50.35	49.34	22.85	71.40	48.35	73.07	51.61	82.71	67.33	67.37	44.72	69.36	46.51
4	79.43	62.43	83.62	69.22	63.48	38.95	78.84	61.61	78.97	61.81	79.55	62.78	79.14	62.23	77.81	59.74
5	76.28	58.86	76.73	59.44	28.52	7.83	77.35	60.92	77.81	61.84	83.14	68.35	73.39	51.91	46.53	49.74
6	71.34	56.32	73.41	59.21	51.72	29.95	72.41	58.27	72.69	58.36	71.57	56.92	66.18	49.02	66.77	47.04
7	76.06	55.61	77.01	56.97	36.55	14.67	76.31	56.57	75.94	55.63	73.88	53.03	76.96	56.49	73.27	51.99
8	66.28	41.44	75.20	54.03	62.47	32.89	68.26	43.99	67.85	43.69	44.75	17.80	64.82	35.87	66.12	40.61
9	64.13	44.29	66.18	47.19	56.72	33.66	65.02	46.23	65.27	45.86	64.42	45.45	83.88	63.01	65.55	45.33
10	81.51	63.98	84.44	69.62	71.05	49.19	83.13	66.67	83.02	67.02	81.46	62.80	84.20	67.20	82.57	65.85
11	70.57	51.38	77.66	59.51	67.43	44.21	70.83	51.59	70.19	50.85	74.06	54.55	69.82	49.71	68.53	48.79
AVG	73.61	54.19	76.84	**58.87**	55.12	31.36	73.94	54.71	73.96	54.74	73.22	54.40	74.07	53.31	69.12	50.40
RF																
1	74.64	54.92	80.86	63.35	57.85	35.39	74.41	53.93	74.45	54.01	74.38	54.33	74.34	54.66	71.91	50.01
2	69.25	45.87	74.09	51.75	61.02	35.08	68.68	44.77	69.02	45.33	70.18	45.76	69.41	46.17	67.46	43.30
3	67.45	44.00	71.17	49.34	54.34	27.38	67.13	43.30	66.91	42.88	65.20	41.54	68.13	44.76	61.99	37.66
4	78.67	62.01	82.12	67.28	64.62	40.89	78.17	61.26	78.39	61.55	80.05	63.88	78.30	61.51	77.02	59.51
5	72.94	52.53	73.10	53.29	38.06	11.74	73.80	53.87	73.65	53.59	68.52	47.14	73.57	53.39	72.44	52.24
6	69.31	53.08	69.50	55.28	49.03	26.70	71.27	56.07	71.36	56.15	66.18	51.59	69.54	53.34	71.98	55.95
7	50.36	74.34	49.05	73.15	15.18	38.52	51.33	74.37	51.58	74.51	52.96	74.13	49.95	74.17	51.11	73.43
8	58.78	33.22	68.05	43.58	67.63	38.74	59.75	33.72	60.29	34.13	53.70	18.48	59.22	33.79	58.11	31.51
9	62.80	39.62	63.70	41.56	55.01	27.74	63.03	39.46	63.32	39.78	62.71	41.79	62.84	39.67	63.35	39.92
10	78.38	59.34	80.19	61.53	72.28	49.79	78.65	60.02	79.00	60.46	78.37	58.33	78.37	59.31	79.61	60.99
11	73.52	54.88	71.02	50.53	69.68	46.97	72.86	53.87	72.67	53.60	67.20	45.58	73.59	54.97	70.51	50.72
AVG	68.74	52.16	71.17	**55.51**	54.97	34.45	69.01	52.24	69.15	52.36	67.22	49.32	68.84	52.34	67.77	50.48

A. DECODING OF DEXTEROUS, IN-HAND MANIPULATION MOTIONS

This subsection presents and discusses the results for decoding dexterous, in-hand manipulation motions. Results will be discussed focusing on accuracy (A), correlation (C), and time required to predict the output. To do this, two different DL methods and RF-based regression methods were compared.

1) DL MODELS

The two DL models compared in this experiment are CNNs and TMC-T. Details about these models are discussed in Section III-B2 and Section III-B3.

2) RF MODELS

Five models were developed to select the appropriate number of trees for the RF-based models, using a different number of trees for each subject, object, and one TD EMG feature per model as input. The trees evaluated in this study are: 1, 25, 50, 75, and 100. For 1 tree, Random Forests behave like a classic decision tree. The results obtained are presented in Table 2. Due to a consistent performance by RF-based models developed using 50 trees across all the conditions, they were selected to be compared with the DL models.

3) COMPARISON BETWEEN RF AND DL MODELS

Table 3, Table 4, and Table 5 present correlation and accuracy of the decoded motion with the actual motion for the two DL models and the RF-based models developed using 50 trees. In most cases, the DL models perform better in terms of correlation and accuracy than the RF model. The model that presents the best correlation and accuracy was the TMC-T.

TABLE 5. Pearson correlation (C) and accuracy (A) for tested models using 8 TD features as input for the off-center object.

Subject	Off-Center															
	RMS		WL		ZC		IEMG		MAV		WAMP		VAR		LogD	
	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A
TMC-T																
1	88.47	76.54	82.00	91.55	70.39	45.77	88.25	76.26	88.52	76.30	87.57	74.04	87.39	74.65	85.52	70.02
2	90.56	80.42	79.10	89.95	63.96	36.35	88.99	77.23	90.08	79.34	85.85	72.20	87.00	71.79	87.53	73.53
3	67.37	46.26	55.66	74.95	54.59	31.29	66.97	46.10	67.64	47.12	67.01	39.71	64.41	42.55	66.45	45.86
4	78.62	60.14	60.85	80.95	67.26	44.72	75.91	56.32	77.04	58.21	81.00	64.16	74.65	54.50	72.30	50.81
5	77.24	58.14	59.49	77.67	51.85	26.86	78.60	60.48	78.61	60.66	68.55	44.86	76.10	57.08	75.62	56.02
6	82.26	65.08	65.54	82.27	60.67	30.75	83.35	66.27	82.42	65.56	79.05	60.71	77.95	56.83	79.79	60.20
7	75.60	54.29	60.70	78.83	57.81	31.66	77.03	56.52	76.80	56.34	68.28	44.16	72.70	48.10	74.66	52.84
8	59.42	33.56	36.71	62.39	39.18	11.17	60.44	34.33	57.67	30.37	46.95	18.39	57.39	31.00	58.23	32.48
9	78.57	59.01	0.78	58.35	66.31	40.68	78.92	58.45	79.16	59.99	72.73	49.29	75.43	54.88	78.56	59.55
10	80.28	62.47	0.80	58.82	56.87	31.04	80.52	61.02	81.73	64.77	81.07	62.80	77.92	58.45	80.01	61.81
11	73.49	46.83	0.87	72.23	69.82	46.49	73.86	47.95	75.61	51.37	79.15	58.67	67.43	41.55	77.21	55.41
AVG	77.44	58.43	45.68	**75.27**	59.88	34.25	77.53	58.27	77.75	59.09	74.29	53.55	74.40	53.76	75.99	56.23
CNN																
1	86.84	72.54	90.86	80.13	67.97	42.98	87.06	73.36	87.46	74.07	86.10	72.09	85.92	71.50	83.86	67.69
2	88.09	76.61	89.03	78.32	63.21	37.46	88.97	77.84	88.67	77.62	82.74	67.45	85.90	68.67	83.62	67.86
3	71.54	50.87	77.02	58.71	67.03	43.11	70.31	50.13	70.31	50.32	68.74	46.47	66.10	44.15	68.09	47.46
4	76.08	58.42	82.22	66.23	65.72	44.49	74.67	56.36	74.08	55.73	79.55	62.78	73.53	54.30	71.36	52.42
5	78.02	59.85	78.85	61.30	53.35	28.58	74.67	56.36	79.88	62.81	68.51	46.27	77.31	58.78	77.23	58.84
6	79.59	59.32	79.78	59.85	51.14	22.34	79.80	62.88	79.23	58.41	71.57	56.92	66.33	39.74	78.13	56.78
7	75.99	54.71	75.56	55.47	57.90	31.85	78.62	57.60	75.30	54.16	65.49	41.97	72.62	49.49	71.95	49.39
8	59.48	33.09	64.79	38.69	37.28	12.58	75.91	55.64	59.85	33.72	84.02	67.78	64.49	38.93	60.33	33.77
9	78.04	57.58	80.43	61.46	63.93	36.73	59.00	32.56	78.02	58.26	72.89	49.30	62.82	40.86	78.43	58.12
10	82.02	64.05	84.91	69.73	56.29	30.77	83.76	67.58	83.62	66.45	80.32	61.45	77.37	56.68	81.76	63.70
11	67.58	37.77	86.61	70.97	68.18	44.88	61.39	23.34	59.49	22.12	77.08	53.91	67.43	41.55	62.12	29.75
AVG	76.66	56.80	80.91	**63.72**	59.27	34.16	75.83	55.79	75.99	55.79	76.09	56.95	72.71	51.33	74.26	53.25
RF																
1	85.60	69.67	86.14	70.69	67.26	42.70	85.84	70.15	85.82	70.10	81.11	64.11	85.37	69.36	84.20	66.94
2	78.46	59.93	85.14	70.78	60.45	35.11	79.56	61.28	79.79	61.66	81.62	65.29	77.81	59.02	77.97	58.79
3	64.76	44.57	73.04	52.98	54.12	31.82	65.46	45.99	65.79	46.25	64.89	42.85	64.86	44.67	65.69	46.60
4	73.05	53.84	83.12	67.94	69.54	48.90	70.65	50.68	70.57	50.64	79.72	64.12	73.23	54.01	71.36	51.86
5	75.65	55.02	73.74	53.41	53.10	28.41	76.13	56.28	76.53	56.83	69.53	47.41	76.00	55.53	71.57	50.47
6	79.51	58.47	80.00	60.66	63.17	29.83	80.05	58.50	80.28	58.86	77.74	57.09	79.55	58.43	80.15	57.69
7	70.76	46.85	73.57	52.69	58.89	32.75	72.68	49.42	72.16	48.49	68.50	46.60	71.01	47.15	71.26	46.91
8	57.86	31.36	55.45	28.47	43.65	18.25	56.14	28.88	55.94	28.57	45.41	19.76	58.02	31.50	58.27	31.88
9	75.35	51.44	73.03	50.27	64.93	35.22	76.28	53.31	75.88	52.81	70.06	45.54	75.29	51.27	75.27	52.48
10	80.41	60.68	81.15	62.76	58.18	32.10	81.20	62.58	81.29	62.60	80.41	60.11	80.46	60.74	78.34	58.61
11	81.28	58.39	81.95	58.53	68.31	43.00	81.39	59.04	81.56	59.31	79.23	54.54	81.36	58.59	79.18	55.76
AVG	74.79	53.65	76.94	**57.20**	60.15	34.37	75.03	54.19	75.05	54.19	72.57	51.58	74.81	53.66	73.93	52.54

Table 6 presents the time taken (in *ms*) by the RF models and DL models to predict one sample. The RF-based models have a very high response rate (∼800 Hz), allowing them to function in real-time even in the presence of unaccounted time delays (e.g., un-optimized code, processing delays). The TMC-T model, despite having a better accuracy and correlation than the RF model, takes approximately 13 times longer to predict a sample.

TABLE 6. Time (in *ms*) for predicting one sample achieved by the regression models.

Model	Prediction time (ms)
RF 50 Trees	0.00124
CNN	0.01204
TMC-T	0.01624

B. HAND GESTURE CLASSIFICATION

In this subsection, we present the gesture decoding performance of TMC-T, CNN, and RF-based classifiers. This set of experiments develops the gesture decoding models using each extracted EMG feature. The gesture classes were balanced for each gesture to avoid any biases due to an imbalanced dataset. Therefore, it was ensured that the training validation sets have the same number of data points for each

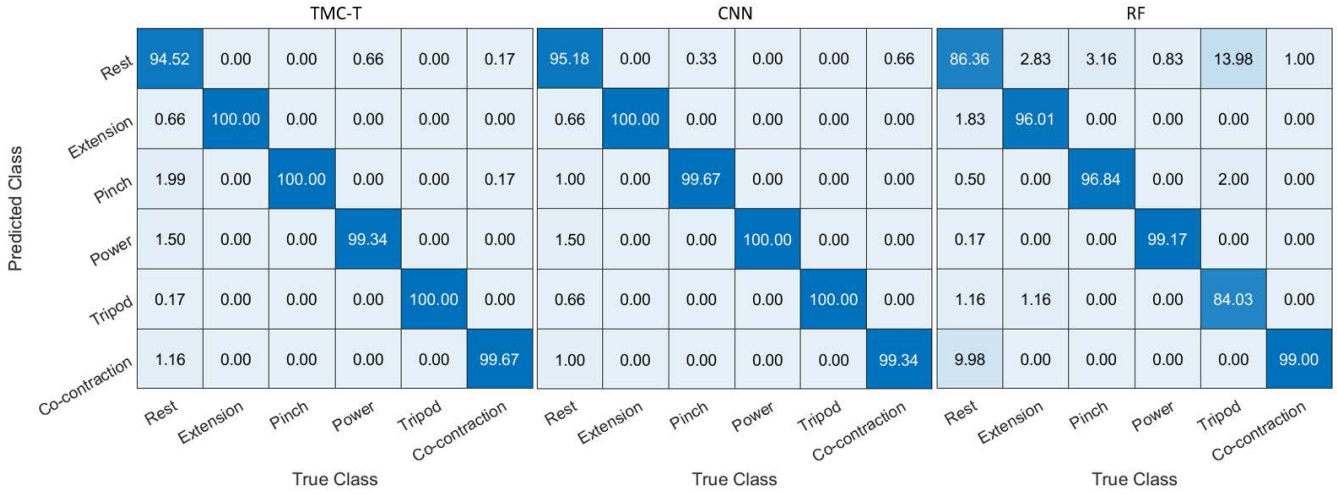

FIGURE 6. Confusion matrices for the three assessed learning models. The TMC-T and CNN were trained using WL feature as input, whereas RF used VAR feature as input.

grasp type. The presented results are the average values calculated using the 5-fold cross-validation. The accuracies of decoding the gestures are shown in Table 7. The best and the worst performing features are highlighted for each decoding model (which are trained for each subject using only one feature at a time). The performance of each feature was independently tested to isolate the ones with low performance to make informed decisions in the future regarding feature selection. It can be noticed that MAV and IEMG have similar performances for five of six tested subjects. This can be attributed to MAV being just a scaled version of IEMG. It can also be noticed that models developed using ZC as the input feature have the worst performance. The two features that presented the best average classification accuracy for the DL models are WL and WAMP. For the RF model, the best features were VAR and IEMG. In Fig. 6 we show the confusion matrices for each tested machine learning technique.

Table 8 presents the time taken (in *ms*) by the RF models and DL models to predict one sample. Table 8 also show the accuracy-execution time trade-off metric σ. This metric was measured for the best model of each ML technique. The RF-based models have a very high response rate (∼800 Hz). The TMC-T model took approximately 13 times longer to predict a sample. The RF model achieved the highest σ, followed by the CNN and TMC-T models. Even though the DL models showed better classification and regression accuracy, by evaluating the trade-off metric, RF-based models were selected for real-time robotic telemanipulation experiments because of its combination of accuracy and speed of code execution that are of paramount importance for real-time applications. However, in applications where real-time requirements are not strict but accuracy of predictions is a key performance evaluation metric, then, TMC-T models should be employed. It must also be noted that a shared control scheme has the human operator in the loop who can compensate for any decoding errors. This has been tested for the hand gesture classification case. Future work will focus on real-time EMG-based execution of dexterous manipulation tasks with the ARoA platform.

TABLE 7. Gesture classification accuracy for TMC-T, CNN, and RF using each extracted feature. The best and worst performing features are highlighted for each subject. Highest accuracy for each subject is bold and highlighted in blue while the lowest accuracy is highlighted in yellow.

Subjects	RMS	WL	ZC	MAV	IEMG	WAMP	VAR	LogD
TMC-T								
1	96.63	96.89	81.81	96.89	**97.03**	96.26	96.77	96.46
2	94.83	**95.17**	87.85	95.62	95.13	91.84	94.93	95.00
3	86.64	91.07	79.59	86.29	87.39	**91.98**	85.88	84.78
4	93.95	**96.96**	77.61	93.60	94.16	95.57	90.43	92.91
5	97.81	**98.38**	91.46	97.87	97.95	98.35	97.93	97.77
6	97.09	97.70	85.44	97.53	97.41	**97.76**	97.02	97.15
AVG	**94.49**	**96.03**	**83.96**	**94.63**	**94.85**	**95.29**	**93.83**	**94.01**
CNN								
1	95.84	96.49	82.03	96.40	96.59	96.10	**96.77**	96.70
2	94.32	94.84	87.50	94.65	**94.97**	92.26	94.22	93.77
3	85.65	89.97	78.37	85.93	84.10	**91.34**	80.99	83.93
4	93.82	**96.58**	76.70	92.89	92.19	94.95	92.41	91.60
5	97.75	**98.24**	91.37	98.19	97.87	98.20	97.81	97.83
6	96.89	97.66	85.39	97.28	96.95	**97.89**	96.34	97.35
AVG	**94.05**	**95.63**	**83.56**	**94.22**	**93.78**	**95.12**	**93.09**	**93.53**
RF								
1	94.76	95.12	84.95	95.05	95.21	**96.55**	94.81	95.62
2	90.15	85.92	84.16	90.35	90.54	83.79	**91.62**	90.28
3	83.83	**86.17**	73.90	84.06	84.34	85.09	83.79	82.97
4	90.86	**93.36**	72.82	90.44	90.24	92.52	90.33	89.32
5	93.82	94.48	91.23	93.62	93.69	**96.40**	94.71	93.62
6	93.12	87.99	86.12	91.09	**95.17**	90.63	94.68	87.86
AVG	**91.09**	**90.51**	**82.20**	**90.77**	**91.53**	**90.83**	**91.66**	**89.95**

C. TELEMANIPULATION WITH AN INTELLIGENT ROBOT SYSTEM

In this experiment, we validate the performance of the selected RF-based gesture classification models in a real-time shared control framework for the intuitive control of the New Dexterity ARoA humanoid platform in the execution

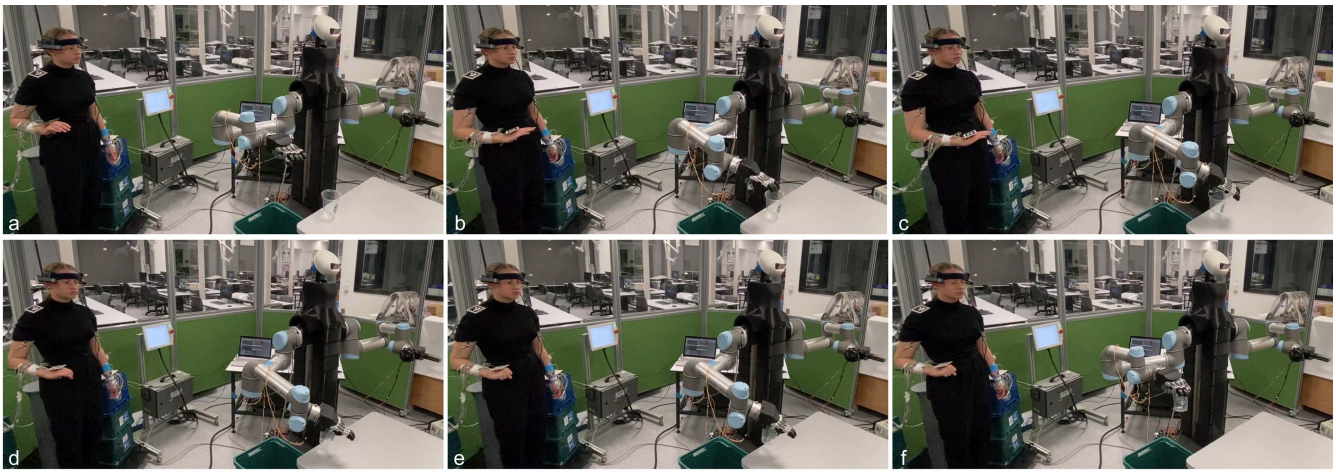

FIGURE 7. Task sequence from the shared task execution experiment. The experiments were conducted using the New Dexterity ARoA platform [21]. Subfigures a), b), and c) show the user controlling the motion of the robot platform to reach the object to be grasped. In subfigure d) we see the user executing "Pinch" gesture to grasp the object. After the successful grasp the user executes "Power" gesture (subfigure e) to hand-over the control back to the *Autonomous Control Module* to complete the task execution (subfigure f).

TABLE 8. Time (in *ms*) for predicting one sample achieved by the classification models. The accuracy-execution time trade-off is shown for the each model.

Model	Prediction time (ms)	Accuracy-execution time trade-off (σ)
RF 50 Trees	0.00126	72746
CNN	0.01362	7021
TMC-T	0.01765	5440

of complex telemanipulation tasks. The performance of the proposed framework was validated in five different scenarios.

1) HEAD MOTION COMPENSATION AND ARM MOTION TRACKING

In the first scenario, we show the effects of the user's head motion on the motion of the robot's end-effector. The motion of the head results in the shifting of the estimated marker poses in the image frame. This motion is compensated in practice by considering the relative pose between the shoulder and the wrist marker instead of taking an absolute pose of a marker. This experiment also demonstrates the robot arm teleoperation in one and two dimensions.

2) ENABLING AND DISABLING ARM MOTION TRACKING

In this scenario, we demonstrate the use of the muscle co-contraction action to enable and disable the teleoperation function of the robot arm.

3) GRASP BASED RELINQUISHING OF MANUAL CONTROL

For this scenario, the hand-over by the user from the manual teleoperation control of the robot system to an autonomous task execution is demonstrated. This is done by first disabling the manual teleoperation by executing the muscle co-contraction gesture, followed by a power grasp gesture to hand-over the control to the autonomous system. In this case, the robot finishes the task in a completely autonomous manner without requiring any user intervention.

4) AUTONOMOUS TASK EXECUTION

In order to test the autonomous task execution capabilities of the proposed framework and the employed platform, the robot was given the task of tidying up a table, where the objective was to grasp and move all the objects that were on the table into a bin.

5) SHARED CONTROL BASED TASK EXECUTION

Finally, the complete shared control framework was tested in the fifth scenario. In this case, the human and the robot controller complete the task in a synergistic manner. To demonstrate the capabilities of the proposed framework, a special table cleaning task was considered where the perception of the robot system fails due to a transparent or irregular object. In such a case, the robot system is unable to execute the task in an autonomous manner since it is not capable to identify where the object is on the table or it is unable to find an efficient grasping strategy to execute the task. Therefore, assistance from a human-in-the-loop is required to help the robot grasp the object and then pass the control back to the robot for autonomous task execution. Fig. 7 shows instances of the experiment to demonstrate the real-time teleoperation performance of the proposed shared control framework.

6) VIDEO OF EXPERIMENTAL VALIDATION

All the experiments were recorded, and the compiled video is available in HD quality at the following URL:

www.newdexterity.org/emgtelemanipulation

V. CONCLUSION

We compared various machine learning and feature extraction methods for the creation of EMG-based telemanipulation frameworks, and we showed that although TMC-T provided

the best decoding performance, the RF model combined very good decoding performance with excellent speed of execution that is needed for real-time applications.

More precisely, eight time-domain features extracted from EMG were tested as input for the first experiment. The TMC-T presented the best correlation and accuracy results between the tested models, however taking approximately 13 times more time to generate a sample output. The RF model presented consistently good results with a very high response rate (approximately 800 Hz). The TMC-T, CNN, and RF models performed human hand gesture decoding in the second set of experiments. The models were evaluated in terms of accuracy and execution time. We also defined an accuracy-execution time trade-off metric to further assess the models. Even though the TMC-T achieved the highest accuracy, based on the trade-off metric, the RF-based model was selected for real-time telemanipulation experiments executed with the New Dexterity ARoA humanoid platform. The discussed shared control framework allows the user to switch between manual and autonomous control of the robot system employing EMG-based gesture decoding and fiducial markers-based pose tracking. The efficiency of all different components was assessed, and the framework was experimentally validated in the execution of complex everyday life tasks.

Regarding future work and directions, the same experiments could be performed using alternative MuMIs, such as wearable forcemyography and lightmyography armbands. Moreover, the TMC-T model can be optimized in order to reduce the prediction time, making it a good candidate method for real-time applications.

REFERENCES

[1] P. Artemiadis, "EMG-based robot control interfaces: Past, present and future," *Adv. Robot. Autom.*, vol. 1, no. 2, pp. 1–3, 2012.

[2] D. Shieff, A. Turner, A. Dwivedi, G. Gorjup, and M. Liarokapis, "An electromyography based shared control framework for intuitive robotic telemanipulation," in *Proc. 20th Int. Conf. Adv. Robot. (ICAR)*, 2021, pp. 806–811.

[3] P. K. Artemiadis and K. J. Kyriakopoulos, "EMG-based control of a robot arm using low-dimensional embeddings," *IEEE Trans. Robot.*, vol. 26, no. 2, pp. 393–398, Apr. 2010.

[4] J. Vogel, C. Castellini, and P. van der Smagt, "EMG-based teleoperation and manipulation with the DLR LWR-III," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 672–678.

[5] A. Dwivedi, G. Gorjup, Y. Kwon, and M. Liarokapis, "Combining electromyography and fiducial marker based tracking for intuitive telemanipulation with a robot arm hand system," in *Proc. 28th IEEE Int. Conf. Robot Hum. Interact. Commun. (RO-MAN)*, Oct. 2019, pp. 1–6.

[6] E. Ceolini, C. Frenkel, S. B. Shrestha, G. Taverni, L. Khacef, M. Payvand, and E. Donati, "Hand-gesture recognition based on EMG and event-based camera sensor fusion: A benchmark in neuromorphic computing," *Frontiers Neurosci.*, vol. 14, p. 637, Aug. 2020.

[7] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori, "Comparison of six electromyography acquisition setups on hand movement classification tasks," *PLoS ONE*, vol. 12, Oct. 2017, Art. no. e0186132.

[8] U. Cote-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 760–771, Apr. 2019.

[9] A. Dwivedi, Y. Kwon, A. J. McDaid, and M. Liarokapis, "EMG based decoding of object motion in dexterous, in-hand manipulation tasks," in *Proc. 7th IEEE Int. Conf. Biomed. Robot. Biomechatronics (Biorob)*, Aug. 2018, pp. 1025–1031.

[10] C. Castellini, P. van der Smagt, G. Sandini, and G. Hirzinger, "Surface EMG for force control of mechanical hands," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2008, pp. 725–730.

[11] M. Sim ao, N. Mendes, O. Gibaru, and P. Neto, "A review on electromyography decoding and pattern recognition for human-machine interaction," *IEEE Access*, vol. 7, pp. 39564–39582, 2019.

[12] A. Dwivedi, Y. Kwon, A. J. McDaid, and M. Liarokapis, "A learning scheme for EMG based decoding of dexterous, in-hand manipulation motions," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 10, pp. 2205–2215, Oct. 2019.

[13] M. A. Oskoei and H. Hu, "Myoelectric control systems—A survey," *Biomed. Signal Process. Control*, vol. 2, no. 4, pp. 275–294, 2007.

[14] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Syst. Appl.*, vol. 40, no. 12, pp. 4832–4840, 2013.

[15] C. Kendell, E. D. Lemaire, Y. Losier, A. Wilson, A. Chan, and B. Hudgins, "A novel approach to surface electromyography: An exploratory study of electrode-pair selection based on signal characteristics," *J. Neuroeng. Rehabil.*, vol. 9, no. 1, pp. 1–8, 2012.

[16] A. Turner, D. Shieff, A. Dwivedi, and M. Liarokapis, "Comparing machine learning methods and feature extraction techniques for the EMG based decoding of human intention," in *Proc. 43rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Nov. 2021, pp. 4738–4743.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[18] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Proc. IEEE 25th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2015, pp. 1–6.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[20] S. Shen, K. Gu, X.-R. Chen, C.-X. Lv, and R.-C. Wang, "Gesture recognition through sEMG with wearable device based on deep learning," *Mobile Netw. Appl.*, vol. 25, pp. 2447–2458, Dec. 2020.

[21] G. Gorjup, C.-M. Chang, G. Gao, L. Gerez, A. Dwivedi, R. Yu, P. Jarvis, and M. Liarokapis, "The ARoA platform: An autonomous robotic assistant with a reconfigurable torso system and dexterous manipulation capabilities," in *Proc. EEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2021, pp. 4103–4110.

[22] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: Using the Yale-CMU-Berkeley object and model sett," *IEEE Robot. Autom. Mag.*, vol. 22, no. 3, pp. 36–52, Sep. 2015.

[23] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

[24] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, 2009, p. 5.

[25] G. Gao, A. Dwivedi, N. Elangovan, Y. Cao, L. Young, and M. Liarokapis, "The new dexterity adaptive, humanlike robot hand," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2019.

[26] A. Dwivedi, D. Shieff, A. Turner, G. Gorjup, Y. Kwon, and M. Liarokapis, "A shared control framework for robotic telemanipulation combining electromyography based motion estimation and compliance control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May/Jun. 2021, pp. 9467–9473.

[27] B. Zhang and S. Zhang, "The estimation of grasping force based on the feature extracted from EMG signals," in *Proc. IEEE Adv. Inf. Manag., Communicates, Electron. Autom. Control Conf. (IMCEC)*, Xi'an, China, Oct. 2016, pp. 1477–1480.

[28] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, "EMG feature extraction for tolerance of white Gaussian noise," in *Proc. Int. Workshop Symp. Sci. Technol.*, 2008, pp. 178–183.

[29] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *J. NeuroEng. Rehabil.*, vol. 7, no. 1, p. 21, Dec. 2010.

[30] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, "A novel feature extraction for robust EMG pattern recognition," 2009, *arXiv:0912.3973*.

[31] T. K. Ho, "Random decision forests," in *Proc. 3rd Int. Conf. Document Anal. Recognit.*, vol. 1, 1995, pp. 278–282.
[32] E. M. Kleinberg, "Stochastic discrimination," *Ann. Math. Artif. Intell.*, vol. 1, no. 1, pp. 207–239, Sep. 1990.
[33] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers Neurorobot.*, vol. 10, pp. 1–10, Sep. 2016.
[34] Y. Chen, C. Dai, and W. Chen, "Cross-comparison of EMG-to-force methods for multi-DoF finger force prediction using one-DoF training," *IEEE Access*, vol. 8, pp. 13958–13968, 2020.
[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[37] R. V. Godoy, G. J. G. Lahr, A. Dwivedi, T. J. S. Reis, P. H. Polegato, M. Becker, G. A. P. Caurin, and M. Liarokapis, "Electromyography-based, robust hand motion classification employing temporal multi-channel vision transformers," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10200–10207, Oct. 2022.
[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.

RICARDO V. GODOY (Graduate Student Member, IEEE) received the Bachelor of Engineering degree in mechatronics engineering, in 2019, and the M.Sc. degree in mechanical engineering from the University of São Paulo, São Carlos, Brazil, in 2021. He is currently pursuing the Ph.D. degree in mechatronics engineering with the New Dexterity Research Group, The University of Auckland, New Zealand. He works on the analysis and development of novel human–machine interfaces (HMI) for the control of robotic and bionic devices while focusing on the challenges and limitations in the use of HMI for robust grasping and decoding of dexterous, and in-hand manipulation tasks. He also works on the development of novel and robust deep learning-based solutions for bio-signals analyses and processing.

ANANY DWIVEDI received the B.Tech. degree in electronics and communication engineering from the LNM Institute of Information Technology, Jaipur, India, in 2015, and the M.S. degree in robotics engineering from the Worcester Polytechnic Institute (WPI), Worcester, MA, USA, in 2017. He is currently pursuing the Ph.D. degree with the New Dexterity Research Group, The University of Auckland, New Zealand. Here, his work focused on deciphering the myoelectric activity of the muscles of the human forearm and hand to decode dexterous manipulation motions in both real and virtual environments. He is also a Postdoctoral Researcher at the Chair of Autonomous Systems and Mechatronics, Friedrich-Alexander-Universität-Erlangen–Nürnberg, Germany, where his work focuses on exploring bi-directional human–machine interfaces to increase presence and embodiment of the user during interaction with various environments while considering human-centered design principles.

BONNIE GUAN (Graduate Student Member, IEEE) received the Bachelor of Engineering degree (Hons.) in mechatronics engineering from The University of Auckland, New Zealand, in 2022, where she is currently pursuing the Ph.D. degree with the New Dexterity Research Group. Her work focuses on exploring novel methods for developing human–machine interfaces to facilitate an intuitive and dexterous control of robotic and prosthetic devices.

AMBER TURNER received the Bachelor of Engineering degree (Hons.) in mechatronics engineering from The University of Auckland, New Zealand, in 2020. Her final year project focused on the design and development of novel muscle machine interfaces for the shared control of robotic and prosthetic devices. She is currently working at PricewaterhouseCoopers as a Technology Consulting Analyst, specifically with the Intelligent Automation Team.

DASHA SHIEFF is currently pursuing the master's degree in engineering science with The University of Auckland, New Zealand. She worked with the New Dexterity Research Group since her final undergraduate year developing and tuning machine learning frameworks. Her research interests include automation and machine learning.

MINAS LIAROKAPIS (Senior Member, IEEE) received the Diploma degree in computer engineering from the University of Patras, Patras, Greece, the M.Sc. degree in information technologies in medicine and biology from the National Kapodistrian University of Athens, Athens, Greece, and the Ph.D. degree in mechanical engineering from the National Technical University of Athens, Athens. He is currently a Senior Lecturer (Above the Bar) with the Department of Mechanical and Mechatronics Engineering, The University of Auckland, New Zealand, and the Director of the New Dexterity Research Group (www.newdexterity.org). Previously, he was a Postdoctoral Associate with the GRAB Laboratory, Yale University, USA. He is the Founder of OpenBionics initiative (www.openbionics.org) and a Co-Founder of OpenRobotHardware (www.openrobothardware.org) and HandCorpus (www.handcorpus.org). He is interested in providing robotics solutions to everyday life problems, modeling, designing, and controlling novel robotics and bionics hardware.