Learning Based Exteroception of Soft Underwater Manipulator With Soft Actuator Network

Kailuan Tang , *Graduate Student Member, IEEE*, Shaowu Tang , *Graduate Student Member, IEEE*,
Chenghua Lu , *Graduate Student Member, IEEE*, Shijian Wu , Sicong Liu , *Member, IEEE*,
Juan Yi , *Member, IEEE*, Jian S. Dai , *Fellow, IEEE*, and Zheng Wang , *Senior Member, IEEE*

Abstract—**Interactions with environmental objects can induce substantial alterations in both exteroceptive and proprioceptive signals. However, the deployment of exteroceptive sensors within underwater soft manipulators encounters numerous challenges and constraints, thereby imposing limitations on their perception capabilities. In this article, we present a novel learning-based exteroceptive approach that utilizes internal proprioceptive signals and harnesses the principles of soft actuator network (SAN). Deformation and vibration resulting from external collisions tend to propagate through the SANs in underwater soft manipulators and can be detected by proprioceptive sensors. We extract features from the sensor signals and develop a fully-connected neural network (FCNN)-based classifier to determine collision positions. We have constructed a training dataset and an independent validation dataset for the purpose of training and validating the classifier. The experimental results affirm that the proposed method can identify collision locations with an accuracy level of 97.11% using the independent validation dataset, which exhibits potential applications within the domain of underwater soft robotics perception and control.**

Index Terms—**Underwater soft manipulator, exteroception, signal processing, neural network, collision detection.**

I. INTRODUCTION

PERCEPTION plays a crucial role in the closed-loop control of soft robots and is also a vital feedback mechanism for detecting their interaction with the environment and estimating their internal states [1], [2], [3], [4]. By utilizing additional external sensors [5] and visual systems [2], [6], [7], previous studies have equipped robots with an extensive array of sensing methods to perceive and monitor objects and obstacles in their vicinity dynamically, enabling the identification of potential collision risks [6], touchless interactive teaching of soft manipulator [8], and sensing contact shape to guide robot grasping by tactus [9].

There are also many reported works on terrestrial soft robots with realization of both internal and external perceptions [8], [10], [11], [12], achieving excellent performance in various tasks. However, due to the limitations of the underwater environment, most sensors designed for use in the air cannot work properly underwater due to the electromagnetic shielding, corrosion, water flow and turbulence, light absorption and scattering, and difficulty in maintenance and deployment. Consequently, the use of some optical and electromagnetic sensors is restricted [4], [7], [13].

To overcome the limitations imposed by underwater environments on sensing technology and avoid complex sensor design, the exteroception methods, which utilizes the proprioceptive information within a manipulator to obtain external interaction events, are being extensively explored. Pioneering works such as tension sensors in cables in cable-driven system [14], [15], pressure sensors in hydraulic actuation system [16], [17], and optical sensors embedded in soft actuators in the manipulator [18] have been reported. These methods are also used to estimate the interaction force [1], [19] or position of the soft robotic system with various approaches. In our previous works, we utilized the inherently-presented pneumatic feedback as proprioception [20], [21], enabling exteroception capabilities in soft robots. Model-based methods are developed to estimate the interaction force and position of soft gripper [22] and soft manipulator [23]. And data-driven approach is also proposed to calculate position and orientation of soft manipulator joint [24].

In these previous works, the deformation caused by interaction or actuation in multi-actuator systems is transmitted through rigid components existing between the actuators, thus simultaneously affecting multiple proprioceptive pressure signals. As a result, it's possible to extract multi-modal perception information from the indirect sensing signals. Meanwhile, in the rigid robot field, there also have been reported works of using joint torque [5], [25], motor current [26], and hydraulic pressure [27] as proprioception signals to determine the robot interaction state [6], [28], [29]. Consequently, utilizing the effect on proprioception signals to sense the interaction state of UW soft manipulator warrants further investigation.

In this work, we summarize the SAN concept from our previous works, and propose a novel model-free exteroperception
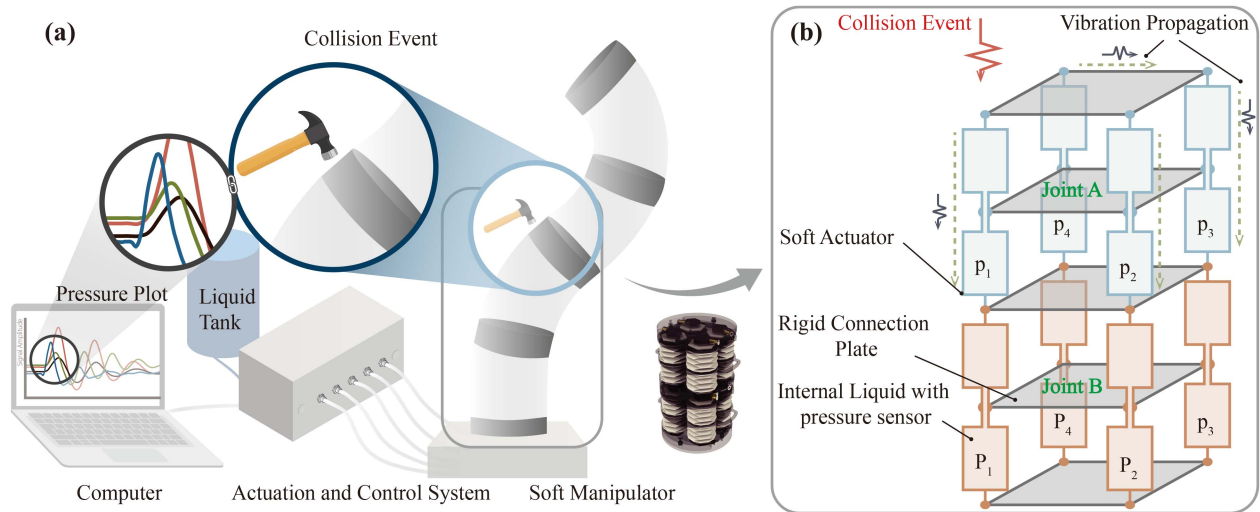
Fig. 1. Total concept of the collision detection and localization in a hydraulic manipulator. (a) External collision events on the manipulator will be reflected in the pressure data inside each soft actuator. (b) The concept of a soft actuator network (SAN), which is a physical structure network where external collisions and vibrations on the manipulator will be transmitted along the soft actuators.

method for identifying collision interaction events and their locations at specific points on the Planarly-Bundled and Overly-Constrained (PBOC) soft UW manipulator from preceding research [16], based on the characteristics of the SAN interacting with the environment. Therefore the sensory capabilities of hydraulic soft UW manipulator could be expanded with the proposed method.

The main contributions of this work are as follows:

1) A new model-free collision localization method for soft manipulators based only on the easily accessible internal pressure signals of the actuators is proposed.
2) According to the characteristics of manipulator vibration transmission along the SAN, the features of multi-channel pressure data generated by vibration and collision events are extracted, and an FCNN-based classifier is designed to identify the collision location.
3) A demonstration of the proposed method is presented by constructing an experimental device with the PBOC manipulator and hydraulic pressure sensors, and the experimental results show that the proposed method can effectively identify the collision position.

The rest of this paper is organized as follows. Section II introduces the concept of the SAN, the proposed collision detection and localization method and the collision waveform feature extraction method. Section III presents the experimental setup and the experimental results. Section IV concludes the paper.

II. COLLISION LOCALIZATION METHOD WITH SAN

A. Collision Interaction Analysis

In hydraulic-driven soft robotic manipulator systems [17][16], collision events generate abnormal pressure signals within the control system (see in Fig. 1), which possess potential utility. However, these types of signals (such as collisions or hydraulic hammer effects) are considered noise in traditional systems and filtered out to maintain control stability. This study aims to harness these signals by analysing their characteristics for precise perception of collision events and

their locations. This approach not only enhances the system's sensing capabilities but also opens up new possibilities for the application of hydraulic-driven soft manipulators in complex underwater environments.

1) Analysis of Soft Actuator Network: The soft manipulator consists of soft actuators and rigid connection parts, which can be regarded as a SAN, as shown in Fig. 1(b) because these soft actuators are connected to the rigid parts in a series or parallel configuration. Since they are assembled as a whole, a collision event at any point on the manipulator will generate vibrations that propagate through the SAN. When these vibrations reach the soft robotic actuators, the pressure signal oscillations of the liquid medium inside them occur simultaneously. The oscillation transfer function of soft actuators is difficult to establish due to the complex compliance, hysteresis and flexibility of soft actuators. This problem further makes it difficult to establish oscillation models of the SAN containing soft actuators and rigid components, because collision events generate vibrations that propagate through various paths within the SAN.

2) Proprioception Network in SAN: In a hydraulic soft robot manipulator, each soft actuator is connected to a pressure sensor to form a hydraulic control loop, thus these sensors configure a distributed proprioception network (PN) in the manipulator, corresponding to the SAN composed of soft actuators. As described in 1), there is a strong coupling between the actuators in the SAN, and pressure sensor signals also exhibit mutual interference. On the other hand, the information from sensors possesses a high degree of redundancy. The propagation of vibrations in SAN after a collision event incorporates high-dimensional external perception information related to the collision location into the multi-channel pressure signals obtained from PN. Since the theoretical models of collision localization of soft manipulators is difficult to establish and the perception signals are also complex, we propose a learn-based method to distinguish the collision occurrence location of soft robot arms without relying on theoretical models. Specifically, we propose a model-free method that extracts waveform features from the multi-channel pressure signals of PN and employs a neural network classifier to identify the collision occurrence location of the soft manipulator.
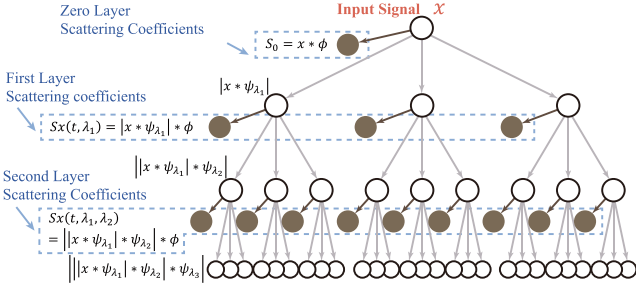
Fig. 2. Tree structure of the wavelet scattering transform, which will be used for feature extraction of the pressure data inside the manipulator.

B. Feature Extraction From Collision Signal

The propagation of the collision-induced vibrations in the PBOC manipulator potentially encompasses intricate pathways and variable transfer functions, consequently, these internal pressure signals from soft actuators at various positions exhibit distinctive waveforms. Many detailed features related to the position information of collision are contained in these pressure signals, including each single channel signal and the differences between them. However, the waveform features in the time domain are difficult to use to locate collision events. It is necessary to extract these features from the raw sensor signals into a suitable format for the classification.

Wavelet Scattering Network (WSN) is a novel computational method that utilizes wavelet transforms to effectively process complex signals and data without losing information [30][31]. The WSN allows for the extraction of features from real-valued time-series and image data with time invariants. They have emerged as a powerful tool in a variety of application areas, including image processing, audio/signal analysis, and machine learning [30][31].

The WSN is a multi-layer tree structure that decomposes the input signal $x(t)$ and the result of one layer is the input for the next layer [32][33]. We define N as the maximum level of layers, Q as the number of wavelets per octave, ϕ as the low-pass filter or scaling function and $*$ denotes the convolution operator. The WSN decomposes input signals by proceeding with the three operations after calculating the zero-order scattering coefficients: wavelet convolution, taking modulus and averaging with scaling function.

$\psi(t)$ is the mother wavelet as a band-pass filter with a center frequency normalized to 1 and bandwidth of Q^{-1}, whose Fourier Transform is $\hat{\psi}(\omega)$. $\psi_{\lambda_n}(t)$ is the n-th wavelet filter bank, which is dilated from the mother wavelet as

$$\psi_{\lambda_n}(t) = \lambda_n \psi(\lambda_n t),$$
$$\hat{\psi}_{\lambda_n}(\omega) = \hat{\psi}(\omega/\lambda_n), \tag{1}$$

where $\lambda_n = 2^{k/Q}, k \in \mathbb{Z}, \lambda_n \in \Lambda_n, 1 \leq n \leq N, n \in \mathbb{Z}, \Lambda_n$ is the grid of all wavelet center frequencies λ. As shown in Fig. 2, the input of WSN is $x(t)$, in the zero layer, a time-average operation is calculated by

$$S_0 x(t) = x * \phi(t), \tag{2}$$

which removes all high frequencies.

The first layer takes $x(t)$ as input and then performs wavelet convolutions on the input signal with each wavelet filter ψ_1 in

the first filter bank Λ_1, in this order, resulting in a modulus of $|x * \psi_{\lambda_1}|$.

The first-order scattering coefficients are computed by averaging the wavelet modulus coefficients with ϕ

$$S_1 x(t, \lambda_1) = |x * \psi_{\lambda_1}| * \phi(t). \tag{3}$$

The second layer takes $|x * \psi_{\lambda_1}|$ as input of the second layer and calculates the wavelet modulus coefficients $||x * \psi_{\lambda_1}| * \psi_{\lambda_2}|$, then the scattering coefficients is calculated as

$$S_2 x(t, \lambda_1, \lambda_2) = ||x * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi(t). \tag{4}$$

Generally, the higher-order wavelet scattering coefficients can be calculated by repeating these three operations in sequence. In this paper, the Gabor wavelet is used as the mother wavelet to proceed with WSN decomposition. The calculated scattering coefficients are utilized as the waveform features to construct the proprioception collision dataset for classifier training and the independent validation dataset for independent predictive tests.

C. Design of the Collision Position Classifier

After using the WSN method to transform the temporal waveforms of signals into feature matrices in the feature domain, in this section, we further employ this method to extract feature data from the temporal waveforms of pressure oscillation signals collected by the PN sensor nodes when a collision occurs in the SAN. Based on the core task of collision location, we designed a classifier for classification and recognition. Since the signal features extracted using the WSN method are related to the signal spectra, in order to preserve the integrity of the feature information without affecting the generalization ability of the neural network, we directly use the the raw feature data as input. Furthermore, we designed a fully connected neural network with a structurally simple architecture to perform classification and recognition on the feature data, thereby highlighting the effectiveness of the WSN method (see in Fig. 3). Due to the high dimensionality of the input layer, when receiving the original feature data, the hidden layer of the neural network is set to two layers to avoid the underfitting problem.

III. EXPERIMENTAL VALIDATION AND RESULTS

A. Experimental Setup and Data Collection

Our previous work proposed a PBOC manipulator with a hydraulic actuation and control system [16]. Here we retain the relative pressure sensors (24PCDFA6D, Honeywell) in the hydraulic control loop and remove other actuation components since the data is collected in the passive manipulation state. The overall schematic diagram of experiments is shown in Fig. 4(f) and the experimental setup is shown in Fig. 4(a). In this setup, soft actuators in the PBOC manipulator are filled with water. Each sensor is connected to a soft actuator via hydraulic pipeline, and all sensors are sampled by the data acquisition device (DAQ device, USB5630-D, ART Technology) at a frequency of 2 kHz. The collision points and their directions for the experiments and verification are shown on the left side of Fig. 5. The right side of Fig. 5 shows the number and name of each collision event, which will be used in the subsequent work on feature extraction and neural network training. A set of data examples are shown in Fig. 6 where we present pressure signals sampled from four actuators in the upper joint during a collision event at position
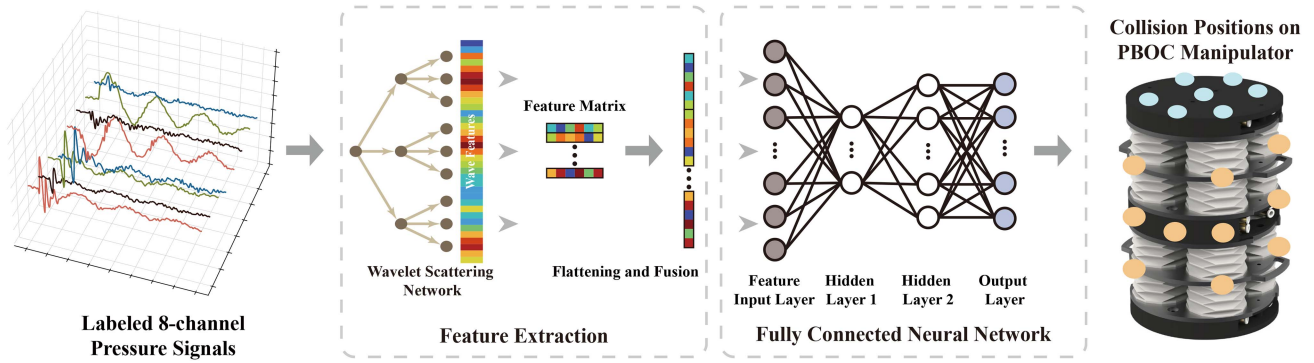
Fig. 3. Complete process of using internal pressure information to locate external collision events, including signal data acquisition, the feature extraction of 8-channel pressure information, the construction and training of a fully-connected neural network, and finally the location of the collision event.
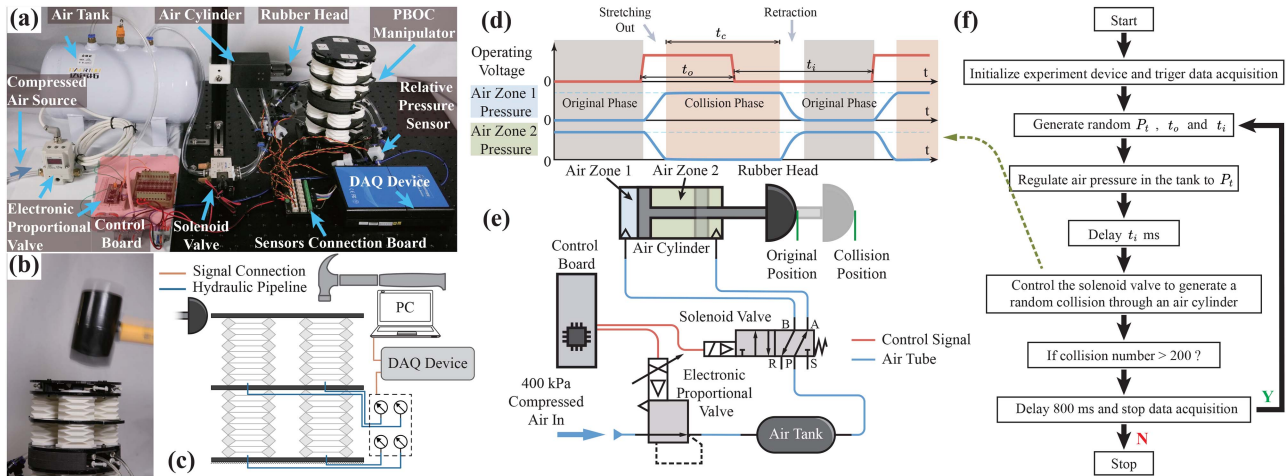

Fig. 4. Collision data collection experiment. (a) Automatic collision test device, the air-driven rubber head mechanically collides with the soft manipulator. (b) Manual collision experiment, which was carried out by three independent experimenters. (c) Schematic diagram of the experimental design for automatic and manual collision signal acquisition. Controller processes (d), air pipe connection (e) and the workflow (f) in automatic collision experiments.
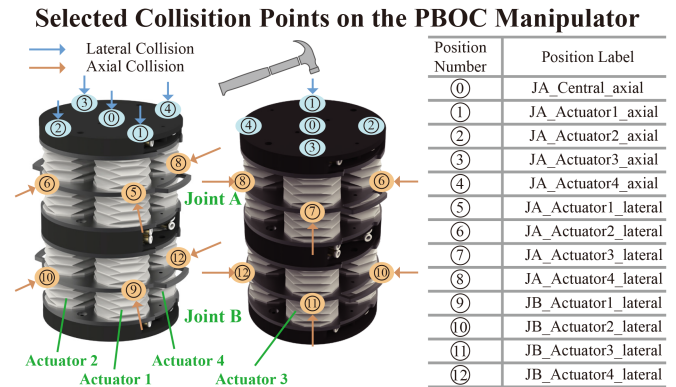
Fig. 5. Positions and directions of collisions on the arm, where are used to collect training data and detect collision events, including 5 axial points on the top and 8 lateral points. The table shows the position numbers and labels.

Fig. 6. Sampled signal waveforms from actuators in the top joint in the manipulator, the collision event occurs on position (a) (JA_Actuator1_axial).

(a). These signal waveforms and the differences between them show the potential to locate the collision time through analysis and training. The pressure signals are calculated from the voltage signals of the sensors with the following equation:

$$p = \frac{U_{actuator} - 2.5}{1.65} \times 206, \qquad (5)$$

where p is the hydraulic pressure value inside the actuator, $U_{actuator}$ is the voltage signal collected by the sensor, and the remaining constants are the characteristic parameters of these sensors.

Fig. 7. Analysis of pressure data for different collision events and locations. (a)-(d) and (f) shows the waveforms of internal pressure signals inside the manipulator actuators. These pressure signals have been preliminarily denoised. (e) shows the position and direction of collisions on the manipulator.

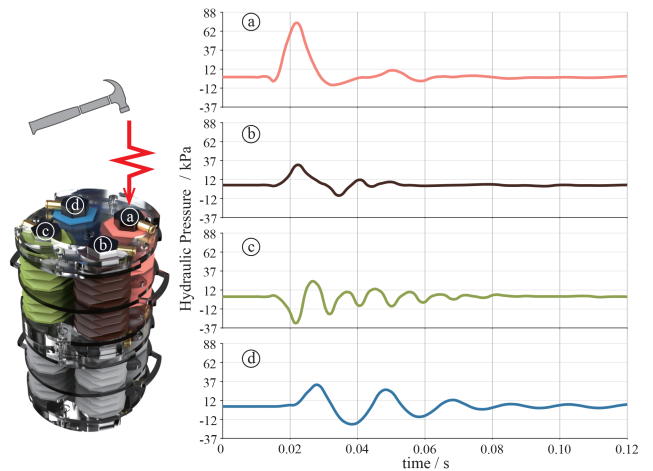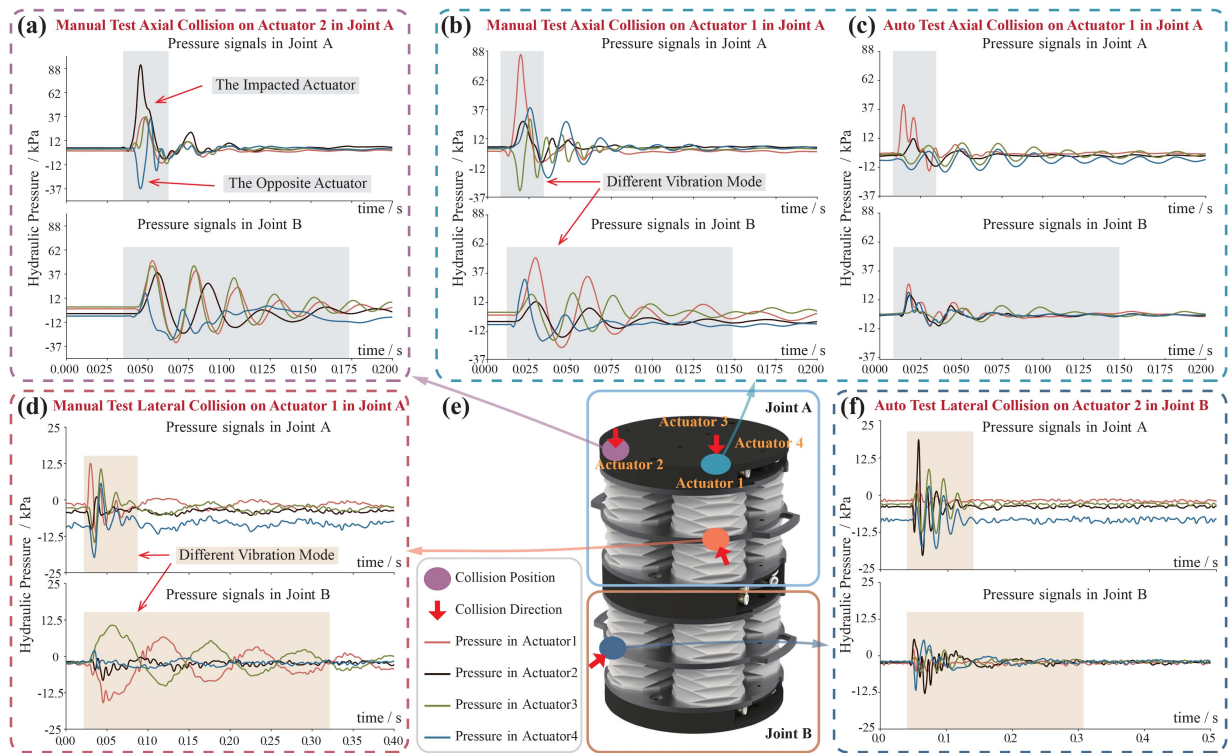To conduct the proprioception training dataset, we designed both automatic and manual experiments, we also selected several collision positions, these positions and their labels are shown in Fig. 5, which are also used as position classification labels.

The entire collision test process and data acquisition includes the following two parts: automatic test and manual test.

1) Automatic Test: The collision events are generated by an air cylinder in this test mode, with pipeline connection showing in Fig. 4(e). The proportional valve (IVT1050, SMC) regulates the air pressure in the tank, and the solenoid valve (4V310-10, Airtec) controls the airflow in the test device. The compressed air is connected to the air zone 2 in the cylinder when the solenoid valve is defective, which keeps the piston on the left side, and thus the rubber head keeps its original position. When the valve is activated, the compressed air is connected to air zone 1, and the piston moves to the right side, thus the rubber head produces a collision event against the PBOC manipulator, performing the role of a pneumatic hammer. The operating voltage of the solenoid valve is generated by the control board (STM32F103C8T6, STMicroelectronics).

The test procedure is specified in Fig. 4(f), where the duration of the random collision is defined as t_o, the time interval between two random operations is defined as t_i, and target air pressure p_t defined in Fig. 4(d) is generated. In the actual automatic test, the range of t_o is [35 ms, 125 ms], and the range of t_i is [500 ms, 2000 ms], and p_t ranges from 80 kPa to 220 kPa. Due to the inherent response time delay of solenoid valves, the rise and fall of air pressure lag behind the reversal time of the operating voltage, resulting in a collision time being greater than the opening time of the solenoid valves.

2) Manual Test: In the manual test, three independent participants took part in the hammer test of the soft manipulator.

The participants were asked to use a rubber hammer and a metal hammer to hit the soft manipulator 100 times at intervals of approximately 1.5 s, applying different degrees of force.

B. Analysis of Collision Signals in the PN

After the collection of raw data, in this section, we will present and analyze several raw signals of impact events with different locations or sources (automatic or manual hammering) in PN.

1) Analysis and Labeling of Collision Signals in the PN: The collision stimuli vibration propagated in the PBOC manipulator could have complex paths and transfer functions, resulting in different waveforms of the pressure signal in each soft actuator. The pressure signals are numbered according to the actuator position for easier observation of the differences between the signals. Significant differences between these four pressure signals within actuators can be observed, as shown in Fig. 6.

It is shown that the peak and valley voltage values are always present in the actuator that is directly struck and its opposite actuator. (Fig. 7(a) and (b)), and the signals in the two joints have notable vibration mode (Fig. 7(a), (d) and (f)). And the different collision tools could also achieve observable differences in pressure waveform (Fig. 7(b) and (c)).

2) Feature Extraction and Proprioception Collision Dataset Construction: The total 8-channel collision signals are sampled with position labels, and then the unique features of the pressure signals in the time domain were extracted by applying the WSN method. To compute the features of the 8-channel pressure signals, a dataset construction program was developed with MATLAB Wavelet Toolbox. The WSN in the program is constructed with two orders of filter banks, the first one has a quality factor $Q = 8$, and the second one has $Q = 1$. The signal

Training test with the proprioception collision dataset of 93.7% accuracy

True Label	JA_Central_axial	JA_Actuator1_axial	JA_Actuator2_axial	JA_Actuator3_axial	JA_Actuator4_axial	JA_Actuator1_lateral	JA_Actuator2_lateral	JA_Actuator3_lateral	JA_Actuator4_lateral	JB_Actuator1_lateral	JB_Actuator2_lateral	JB_Actuator3_lateral	JB_Actuator4_lateral	
JA_Central_axial	1200	0	62	0	0	0	0	0	1	0	0	0	0	95.01%
JA_Actuator1_axial	20	1307	41	0	5	0	0	1	2	0	0	0	1	94.92%
JA_Actuator2_axial	1	0	1294	0	0	0	0	0	2	0	0	0	0	99.77%
JA_Actuator3_axial	46	0	9	1400	1	0	1	0	0	0	0	0	0	96.09%
JA_Actuator4_axial	41	0	13	0	1298	0	0	0	0	0	18	0	0	94.74%
JA_Actuator1_lateral	0	0	0	0	0	1275	0	6	6	0	2	0	2	98.76%
JA_Actuator2_lateral	0	0	0	0	0	0	987	0	60	0	7	0	0	93.64%
JA_Actuator3_lateral	0	1	0	0	0	2	0	1251	24	0	0	1	2	97.66%
JA_Actuator4_lateral	0	0	2	0	0	0	5	0	1265	0	0	0	0	99.45%
JB_Actuator1_lateral	1	1	17	2	0	17	1	0	92	1044	2	11	0	87.88%
JB_Actuator2_lateral	0	0	7	0	0	0	68	0	95	0	1088	0	26	84.74%
JB_Actuator3_lateral	9	4	19	0	0	3	0	49	116	2	1	1048	4	83.51%
JB_Actuator4_lateral	0	0	13	0	3	0	11	0	125	0	48	0	1079	84.36%
Predicted Label														**(a)**

Prediction with the independent validation dataset of 97.11% accuracy

True Label	JA_Central_axial	JA_Actuator1_axial	JA_Actuator2_axial	JA_Actuator3_axial	JA_Actuator4_axial	JA_Actuator1_lateral	JA_Actuator2_lateral	JA_Actuator3_lateral	JA_Actuator4_lateral	JB_Actuator1_lateral	JB_Actuator2_lateral	JB_Actuator3_lateral	JB_Actuator4_lateral	
JA_Central_axial	239	0	1	0	1	0	0	0	0	0	0	0	0	99.17%
JA_Actuator1_axial	0	241	0	0	0	0	0	0	0	0	0	0	0	100.0%
JA_Actuator2_axial	0	0	240	0	0	0	0	0	0	0	0	0	0	100.0%
JA_Actuator3_axial	2	0	0	239	0	0	0	0	0	0	0	0	0	99.17%
JA_Actuator4_axial	0	0	0	0	243	0	1	0	0	0	0	0	0	99.59%
JA_Actuator1_lateral	0	0	0	0	0	233	0	5	0	0	0	0	0	97.90%
JA_Actuator2_lateral	0	0	0	0	0	0	240	0	0	0	3	0	0	98.77%
JA_Actuator3_lateral	0	0	0	0	0	10	0	233	0	0	0	0	0	95.88%
JA_Actuator4_lateral	0	0	1	0	0	0	27	0	220	0	1	0	0	88.35%
JB_Actuator1_lateral	0	0	0	0	0	0	0	0	0	241	0	0	0	100.0%
JB_Actuator2_lateral	0	0	1	0	0	0	0	0	0	0	208	0	31	86.67%
JB_Actuator3_lateral	0	0	0	0	0	0	0	0	0	0	6	237	0	97.53%
JB_Actuator4_lateral	0	0	0	0	0	0	0	0	0	0	1	0	244	99.59%
Predicted Label														**(b)**

Fig. 8. Proprioception accuracy results of the neural network after training. These two confusion matrix shows (a) the classification results of the test part in the proprioception collision dataset, and (b) shows the prediction results of the independent validation dataset, respectively.

length is much longer than the limitation of the signal sampling rate. The time interval between two collision events is less than 2 s, so we choose the default scattering transform invariance scale $t_{vari} = 2$ s. The generated features are stored in a large and high dimension matrix with the size of $R^{F \times T \times N}$, which are sorted by feature vector length F, number of channels N and number of time windows T. The matrix is flattened into a 2D matrix (as shown in Fig. 3) with the size of $R^{FN \times T}$ to fuse feature data of all channels as the training or predicting dataset with one row for a feature group. We used the calculated and fused features from both automatic test data and manual test data to construct the proprioception collision dataset, which was used to train the FCNN classifier.

It should be explicitly noted that in order to ensure that the neural network classifier is insensitive to the time-domain deformation, stretching, and time shift of the signal, and has a larger recognition tolerance, the signal is not split into signal segments that contains only a single time-domain collision signal for the WSN transformation. Because collisions of soft manipulators do not occur continuously and regularly, the splitting operation also breaks the randomness of the collision events in the experiments. The collision recognition accuracy is defined as

$$A_s = \frac{N_r}{N_T}, \quad (6)$$

where N_r is the number of correctly recognized collision time windows, and N_T is the total number of collision time windows in the dataset.

C. Signal Classifier Training and Self Test Results

The FCNN classifier that matched the proprioception collision dataset was developed. The number of nodes in the feature input layer matches the number of features in the WSN output, and the output of the FCNN classifier matches the position number. The network had 34 nodes in the first hidden layer and 100 nodes in the second hidden layer.

The proprioception collision dataset was divided into training (70%) and testing (30%) sets randomly, ensuring that the

model's performance can be assessed on unseen data during training. To keep the classification accuracy, no normalization was applied to the dataset. The network was trained on a single CPU (AMD Ryzen R7-5800X@4.5 GHz) with Adam optimizer for a maximum of 30 epochs and a minimum batch size of 50. The total training spent 298 s CPU time. The testing confusion matrix is in Fig. 8(a), and the total accuracy of the testing dataset of the network is $A_s = 93.7\%$.

D. Model Validation With Independent Manual Collision Data

After completing the training process, the neural network is rigorously evaluated using an independent validation dataset that was not used during the training phase. The independent validation dataset was generated and sampled by the three participants in manual testing in another independent experiment, and the same feature extraction process was adopted to check the prediction accuracy.

The neural network achieves remarkable performance in efficiently classifying and identifying patterns within independent data sets. The confusion matrix of the independent prediction is shown in Fig. 8. From the results, the accuracy of the classification exceeds $A_s = 97.11\%$, demonstrating the network's robustness and generalization capabilities.

E. Discussion

1) Prediction Accuracy of the FCNN Classifier: To verify the separability of discrete collision locations, we employed the t-SNE method for dimensionality reduction, thereby visualizing the proprioceptive collision dataset and an independent validation dataset, as shown in the analysis diagrams of Fig. 9(a) and (b). Classifying collision positions of this passive manipulator is a relatively complex task, due to the high dimensionality of the dataset and the overlapping of the feature distributions of different classes, which contains noise and outlier data. The independent validation dataset for classifiers with one hidden layer consistently shows poor prediction accuracy in our classifier training test, with results below 80%. However, adding an
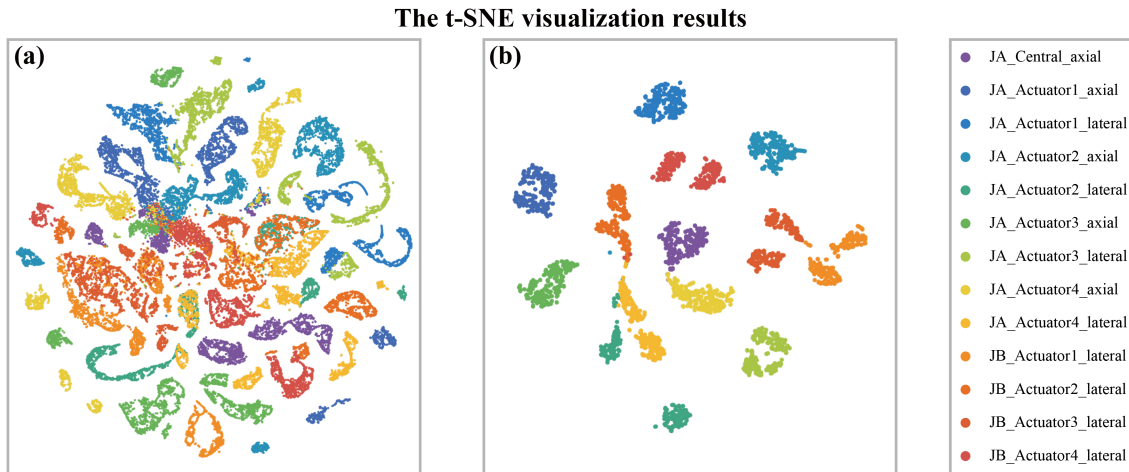
The t-SNE visualization results

Fig. 9. Data visualization via t-SNE method for (a) the proprioceptive collision dataset and (b) the independent validation dataset. The color of the dots represents the location of the collision. It provides a concise pattern to exhibit the separability of the two dataset.

additional hidden layer to the classifier significantly improves the classification accuracy of the independent validation dataset, with results above 95%.

To avoid the problem of overfitting and weak generalization, we incorporated additional noise during the data collection process. Specifically, we introduced randomness in the following aspects of collision event generation:

1). Automatic experiments: We utilized randomly generated collision forces and time intervals for PN data collection.

2). Manual experiments: The direction, force, and time interval of the impact were randomly determined by the experimental participants.

By incorporating this additional randomness, the range of feature distributions is expanded in the proprioception collision dataset, thereby improving the classifier's generalization ability. As a result, the trained network achieves higher classification accuracy on the independent validation task than the cross-validation accuracy during training.

Moreover, serval hyperparameter scanning experiments based on Bayesian optimization are carried out by using the Experiment Manager in MATLAB Deep Learning Toolbox. The results obtained include the number of neurons set for the classifier in Section III-C, the initial learning rate of 0.0384, and the $L2$ regularization coefficient of 7.767×10^{-6}. With this parameter combination, the FCNN classifier achieved a balanced recognition accuracy while avoiding overfitting or underfitting problems.

2) The Real-Time Capability of the Proposed Perception Method: The primary purpose of the experimental design is to validate the effectiveness of the proposed perception method, hence, a real-time collision position perception system was not constructed. However, it is essential to investigate the real-time perception capability of this novel sensing approach.

As depicted in Fig. 3, the trained FCNN classifier necessitates complete signal feature vectors as its input to predict the collision position. Therefore, the evaluation of the real-time capability of this method consists of three parts in total, including the time required for data acquisition, feature vector extraction, and the forward inference process of the FCNN classifier.

The collection time must be greater than or equal to $T_{inv} = 2000$ ms to ensure that it meets the requirements of the WSN and is capable of computing the feature vectors, thereby supporting

real-time inference. The average time consumption of extracting 2000 ms signal features by the WSN is 35.4 ms running on CPU, and the average forward inference time of the FCNN classifier is 0.03 ms running on CPU. Therefore, the total time consumption of the calculating period is 35.4 ms + 0.03 ms = 35.7 ms ≪ 2000 ms. Therefore, using a sliding time window of $T_{inv} = 2000$ ms in length, the perception algorithm can infer the collision position based on the pressure signal of the past 2000 ms in a period of no more than 100 ms to achieve real-time capability. This requires at least 2000 ms of signal collection for real-time inference.

IV. CONCLUSION AND FUTURE WORK

Based on the transmission characteristics of vibrations in SAN, this paper introduces a model-free collision interaction localization method using internal pressure signals of soft robotic arms. Through analysis of SAN and PN, the interconnection between soft actuators allows collision-induced vibrations to propagate via multiple paths, generating hydraulic pressure fluctuations detectable by PN sensors. WSN techniques extract waveform features from these pressure signals, creating a dataset for training a neural network classifier. This enables model-free collision localization perception. As a result, the test accuracy rate reaches 93.7% after training on the proprioception collision dataset. To further validate the accuracy and generalization capability of the classifier, an independent manual collision experiment is designed, and a similar method is employed to construct the independent validation dataset. The prediction accuracy rate of the classifier on the independent validation dataset reaches 97.11%, demonstrating its excellent classification capability. This indicates that utilizing WSN to directly analyze the PN multi-channel signal characteristics of soft robot arms to obtain high-dimensional external perception information is feasible.

Future work will focus on methods to further improve the collision perception capabilities of soft underwater manipulators, exploring collision perception when the manipulator is actively actuating in various states, and enhancing the real-time capabilities. Manipulator control and dynamic interactive response based on this proprioceptive methods will also be further investigated.

REFERENCES

[1] O. Faris et al., "Proprioception and exteroception of a soft robotic finger using neuromorphic vision-based sensing," *Soft Robot.*, vol. 10, no. 3, pp. 467–481, Oct. 2022.

[2] H. Wang, H. Xu, Y. Meng, X. Ge, A. Lin, and X. -Z. Gao, "Deep learning-based 3D pose reconstruction of an underwater soft robotic hand and its biomimetic evaluation," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 11070–11077, Oct. 2022.

[3] W. Dou et al., "Soft robotic manipulators: Designs, actuation, stiffness tuning, and sensing," *Adv. Mater. Technol.*, vol. 6, no. 9, Sep. 2021, Art. no. 2100018.

[4] J. McConnell et al., "Perception for underwater robots," *Curr. Robot. Rep.*, vol. 3, no. 4, pp. 177–186, Oct. 2022.

[5] A. Kouris, F. Dimeas, and N. Aspragathos, "A frequency domain approach for contact type distinction in human–robot collaboration," *IEEE Robot. Automat. Lett.*, vol. 3, no. 2, pp. 720–727, Apr. 2018.

[6] A. Mohammed et al., "Active collision avoidance for human-robot collaboration driven by vision sensors," *Int. J. Comput. Integr. Manuf.*, vol. 30, no. 9, pp. 970–980, Sep. 2017.

[7] D. Q. Huy et al., "Object perception in underwater environments: A survey on sensors and sensing methodologies," *Ocean Eng.*, vol. 267, Jan. 2023, Art. no. 113202.

[8] W. Liu et al., "Touchless interactive teaching of soft robots through flexible bimodal sensory interfaces," *Nat. Commun.*, vol. 13, no. 1, Aug. 2022, Art. no. 5030.

[9] C. Lu, K. Tang, M. Yang, T. Yue, H. Li, and N. F. Lepora, "DexiTac: Soft dexterous tactile gripping," *IEEE/ASME Trans. Mechatron.*, early access, Mar. 25, 2024, doi: 10.1109/TMECH.2024.3384432.

[10] H. Zhao et al., "Optoelectronically innervated soft prosthetic hand via stretchable optical waveguides," *Sci. Robot.*, vol. 1, no. 1, Dec. 2016, Art. no. eaai7529.

[11] R. L. Truby et al., "Soft somatosensitive actuators via embedded 3D printing," *Adv. Mater.*, vol. 30, no. 15, Apr. 2018, Art. no. 1706383.

[12] S. Shu et al., "Machine-learning assisted electronic skins capable of proprioception and exteroception in soft robotics," *Adv. Mater.*, vol. 35, no. 18, May 2023, Art. no. 2211385.

[13] Y. Cong et al., "Underwater robot sensing technology: A survey," *Fundam. Res.*, vol. 1, no. 3, pp. 337–345, May 2021.

[14] F. Xu, H. Wang, J. Wang, K. W. S. Au, and W. Chen, "Underwater dynamic visual servoing for a soft robot arm with online distortion correction," *IEEE/ASME Trans. Mechatron.*, vol. 24, no. 3, pp. 979–989, Jun. 2019.

[15] F. Xu, H. Wang, Z. Liu, and W. Chen, "Adaptive visual servoing for an underwater soft robot considering refraction effects," *IEEE Trans. Ind. Electron.*, vol. 67, no. 12, pp. 10575–10586, Dec. 2020.

[16] K. Tang et al., "A strong underwater soft manipulator with planarly-bundled actuators and accurate position control," *IEEE Robot. Automat. Lett.*, vol. 8, no. 11, pp. 7559–7566, Nov. 2023.

[17] Z. Shen et al., "An underwater robotic manipulator with soft bladders and compact depth-independent actuation," *Soft Robot.*, vol. 7, no. 5, pp. 535–549, 2020.

[18] Z. Shen et al., "Soft origami optical-sensing actuator for underwater manipulation," *Front. Robot. AI*, vol. 7, 2021, Art. no. 616128.

[19] F. Feng et al., "A learning-based tip contact force estimation method for tendon-driven continuum manipulator," *Sci. Rep.*, vol. 11, no. 1, Sep. 2021, Art. no. 17482.

[20] Z. Fang et al., "Multi-dimensional proprioception and stiffness tuning for soft robotic joints," in *Proc. Int. Conf. Robot. Automat.*, Philadelphia, PA, USA, 2022, pp. 10973–10979.

[21] Y. Su et al., "A high-payload proprioceptive hybrid robotic gripper with soft origamic actuators," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 3003–3010, Apr. 2020.

[22] L. Wang et al., "Mechanoreception for soft robots via intuitive body cues," *Soft Robot.*, vol. 7, no. 2, pp. 198–217, Apr. 2020.

[23] Y. Su, X. Chen, Z. Fang, D. Liu, J. Lam, and Z. Wang, "Spatial position–force perception for a soft parallel joint via pressure-deformation self-sensing," *IEEE/ASME Trans. Mechatron.*, vol. 29, no. 5, pp. 3590–3601, Oct. 2024.

[24] L. Wang et al., "Soft robot proprioception using unified soft body encoding and recurrent neural network," *Soft Robot.*, vol. 10, no. 4, pp. 825–837, Mar. 2023.

[25] A.-N. Sharkawy et al., "Neural network design for manipulator collision detection based only on the joint position sensors," *Robotica*, vol. 38, no. 10, pp. 1737–1755, Oct. 2020.

[26] T. Zhang et al., "Collision detection method for industrial robot based on envelope-like lines," *Ind. Robot: Int. J. Robot. Res. Appl.*, vol. 46, no. 4, pp. 510–517, Jun. 2019.

[27] J. Vorndamme, M. Schappler, A. Toedtheide, and S. Haddadin, "Soft robotics for the hydraulic atlas arms: Joint impedance control with collision detection and disturbance compensation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Daejeon, Korea (South), 2016, pp. 3360–3367.

[28] A.-N. Sharkawy et al., "Human-robot collision detection based on neural networks," *Int. J. Mech. Eng. Robot. Res.*, vol. 7, no. 2, pp. 150–157, 2016.

[29] N. Briquet-Kerestedjian, A. Wahrburg, M. Grossard, M. Makarov, and P. Rodriguez-Ayerbe, "Using neural networks for classifying human-robot contact situations," in *Proc. 18th Eur. Control Conf.*, Naples, Italy, 2019, pp. 3279–3285.

[30] R. N. Khushaba and A. J. Hill, "Radar-based materials classification using deep wavelet scattering transform: A comparison of centimeter vs. millimeter wave units," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2016–2022, Apr. 2022.

[31] J. Anden and S. Mallat, "Deep scattering spectrum," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4114–4128, Aug. 2014.

[32] S. Mallat, "Group invariant scattering," *Commun. Pure Appl. Math.*, vol. 65, no. 10, pp. 1331–1398, Oct. 2012.

[33] T. D. Pham, "Classification of motor-imagery tasks using a large EEG dataset by fusing classifiers learning on wavelet-scattering features," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 1097–1107, 2023.