

Unobtrusive hand gesture recognition using ultra-wide band radar and deep learning

Djazila Souhila Korti¹, Zohra Slimane²

¹SSL Laboratory, Department of Telecommunications, Faculty of Technology, Belhadj Bouchaib University, Ain Temouchent, Algeria

²STIC Laboratory, Department of Telecommunications, Faculty of Technology, Tlemcen University, Tlemcen, Algeria

Article Info

Article history:

Received Mar 18, 2023

Revised Jul 9, 2023

Accepted Jul 17, 2023

Keywords:

Extra trees

Hand gesture recognition

Impulse-radio ultra-wide band

Lighweight architecture

Multi-input convolutional

neural network

Optuna

ABSTRACT

Hand function after stroke injuries is not regained rapidly and requires physical rehabilitation for at least 6 months. Due to the heavy burden on the healthcare system, assisted rehabilitation is prescribed for a limited time, whereas so-called home rehabilitation is offered. It is therefore essential to develop robust solutions that facilitate monitoring while preserving the privacy of patients in a home-based setting. To meet these expectations, an unobtrusive solution based on radar sensing and deep learning is proposed. The multi-input multi-output convolutional eXtra trees (MIMO-CxT) is a new deep hybrid model used for hand gesture recognition (HGR) with impulse-radio ultra-wide band (IR-UWB) radars. It consists of a lightweight architecture based on a multi-input convolutional neural network (CNN) used in a hybrid configuration with extremely randomized trees (ETs). The model takes data from multiple sensors as input and processes them separately. The outputs of the CNN branches are concatenated before the prediction is made by the ETs. Moreover, the model uses depthwise separable convolution layers, which reduce computational cost and learning time while maintaining high performance. The model is evaluated on a publicly available dataset of gestures collected by three IR-UWB radars and achieved an average accuracy of 98.86%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Djazila Souhila Korti

SSL Laboratory, Department of Telecommunications, Faculty of Technology, Belhadj Bouchaib University
Ain temouchent, Algeria

Email: souhila.korti@univ-temouchent.edu.dz

1. INTRODUCTION

Hand paralysis occurs in approximately two-thirds of strokes [1], limiting post-stroke survivors from performing 80% of activities of daily living (ADLs) [2]. Hand function after stroke injuries is not regained rapidly and requires continuous physical rehabilitation for at least 6 months [3]. Due to the high burden on the health care system, assisted rehabilitation is primarily prescribed for a limited time, while home-based rehabilitation is offered [2]. Achieving the previous level of hand function in a home-based setting can be accomplished with appropriate physical therapy and remote assessment techniques. The development of an automatic and accurate recording approach for hand gesture recognition (HGR) using sensing technologies and artificial systems appears to be a flexible solution.

Impulse-radio ultra-wide band (IR-UWB) radar has recently emerged as one of the most effective and promising non-contact sensors for HGR [4]–[6]. It possesses the convenience of remote operation in a non-intrusive manner, granting users a sense of freedom and unrestraint. It provides a cost-effective and durable solution, characterized by low power consumption and excellent performance in both brightly lit and dark environments. Furthermore, it effectively overcomes the issue of occlusion by exhibiting remarkable

penetration capabilities through obstacles and walls. Alongside it is numerous benefits, IR-UWB radar also offers valuable insights, including range and velocity information, enabling precise detection of hand motion with exceptional accuracy. However, its vulnerability to orientation variations is a concern. This vulnerability becomes apparent in situations where the perception of IR-UWB radar becomes uncertain due to unfavorable aspect angles between the motion path and the radar line of sight [7]. This uncertainty makes it impractical to rely solely on a single sensor. To differentiate gestures more effectively, the recognition system incorporates multiple sensors [8]. By combining data from multiple IR-UWB radar or other sensors, more concise information is added [7], [9]. In case of missing or unreliable data provided by one sensor, the fusion model can still make informed decisions based on the data provided by the other sensors. This integration enhances the robustness of the system, making it more resistant to sensor failures. It also reduces ambiguity and uncertainty while boosting confidence.

Standard deep models, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been successfully applied to HGR using single-sensor data [10]–[17]. Their sophisticated structure allows them to work directly on raw data as they involve automatic feature detection and extraction. However, these standard models are designed to process data with consistent patterns under a single input and neglect the fusion mechanism. Consequently, using a set of radars to detect hand gestures can be difficult when using standard deep models. This difficulty is mainly due to the varying characteristics of the captured hand gestures. Different mounting configurations, including height, angle, and distance between radar and target, can affect the radar's field of view and spatial coverage, thus influencing the radar signal and capturing different aspects of the same gesture. Standard deep models may struggle to capture the full range of variation and learn how to integrate information from a variety of sensor sources efficiently. Therefore, the architecture of standard deep models should be appropriately designed to accommodate multiple data sources. Data from each sensor has to be processed separately to capture and preserve the relevant features independently and merge the information properly. However, it is important to mention that the separate processing of multiple sensor data requires more complex and higher-capacity models. Therefore, it is crucial to strike a balance between model complexity and desired performance [18].

The present paper proposes a novel multi-input multi-output (MIMO) approach based on the integration of extremely randomized trees (ETs) with CNN for HGR using IR-UWB sensors. Our proposed model does not require heavy data preprocessing or manual feature engineering. It eliminates the dependency on expert experience and prior knowledge. By leveraging the advantages offered by both CNN and ETs, we have developed a complete and effective model capable of automatically extracting discriminative features and producing more accurate results. To the best of our knowledge, MIMO architectures have not yet been used for the classification of hand gestures using IR-UWB radar. Unlike previous works where multiple sensor data were processed together [19]–[22], our multi-input multi-output convolutional eXtra trees (MIMO-CxT) can take data from multiple sensors as input and process them independently. The output of each CNN branch is combined before the prediction is made by the ETs. This process improves the feature extraction operation. By focusing on the data from each sensor separately, the ability to detect deeper and more useful patterns is increased. Processing data in parallel can effectively help extract complementary information about the same target across multiple sensors, allowing for a more comprehensive representation to be learned and a more efficient classifier to be obtained. In addition to its encouraging performance, our model has fewer parameters, which alleviates the computational complexity and cost issues. This makes it highly suitable for the development of an embedded radar-based HGR system that can be used as a telemedicine tool designed for the remote rehabilitation of stroke patients.

The paper's content is arranged as follows: section 2 describes the proposed model architecture and provides details about the experimental setup. Section 3 discusses the results obtained in the experimentation and presents a comparative analysis of the proposed model's performance. Section 4 concludes the paper and presents an outlook on possible future works.

2. METHOD

2.1. MIMO-CxT for multi-sensor systems

Several studies have demonstrated the effectiveness of using multiple IR-UWB radars or combining them with other sensors to increase the detection and classification performance of HGR [19], [20], [23]. Often, these sensors operate independently of each other, which means that they may capture data of different values, scales, or even natures in the case of heterogeneous sensor systems. It is therefore reasonable to process them separately in order to extract discriminative features from each sensor's data and preserve their characteristics without altering them. The MIMO-CxT consists of three parallel CNN branches used in a hybrid configuration with ETs backend. Unlike the standard CNN model, which mixes and processes the entire data at once, our model is designed to perform the feature extraction operation on the data of a certain sensor in each branch

independently. The outputs of all parallel CNN branches are concatenated, put in vector form, and fed to the ETs, which act as the classifier of the architecture. The structure of the MIMO-CxT is depicted in Figure 1.

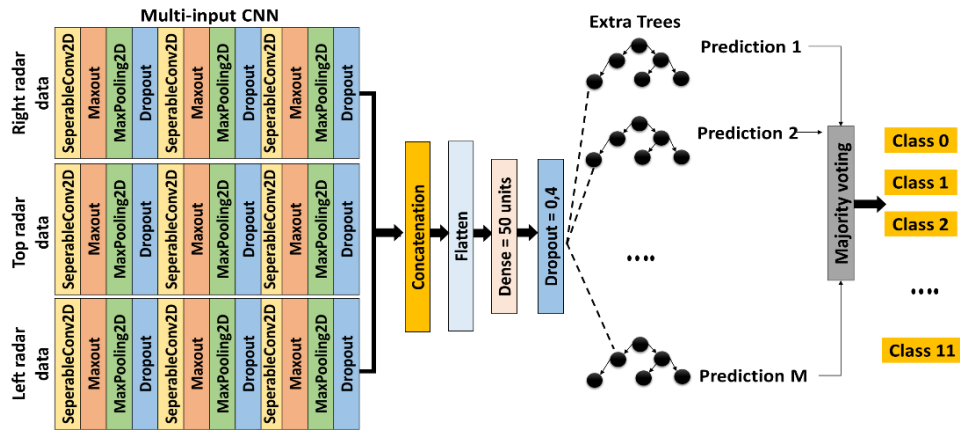


Figure 1. Architecture of the MIMO-CxT model

2.2. Multi-input CNN structure

A similar layer configuration is adopted between the three CNN branches, where each branch consists of an input layer of size $75 \times 75 \times 1$, corresponding to a single channel of the 75×75 input binary image. Table 1 provides a more detailed description of the structure of the three CNN branches. One of our objectives is to design an efficient network that reduces the amount of computation and the number of network parameters as much as possible without compromising classification performance. To achieve this, we propose the use of depthwise separable convolution. Unlike conventional convolution, which performs channel-wise and spatial-wise computations in one step, depthwise separable convolution factorizes the computation into two steps: depthwise convolution followed by pointwise convolution.

Table 1. Single CNN branch structure

Type	Filter shape	Padding	Strides	Parameters	Output shape
Input layer	-	-	-	0	(None, 75, 75, 1)
SeperableConv2D	$4 \times 4 \times 64$	Same	2×2	144	(None, 38, 38, 64)
Maxout	64	-	-	0	(None, 38, 38, 64)
MaxPooling2D	2×2	-	-	0	(None, 19, 19, 64)
Dropout	0.25	-	-	0	(None, 19, 19, 64)
SeperableConv2D	$4 \times 4 \times 64$	Same	2×2	5184	(None, 10, 10, 64)
Maxout	64	-	-	0	(None, 10, 10, 64)
MaxPooling2D	2×2	-	-	0	(None, 5, 5, 64)
Dropout	0.25	-	-	0	(None, 5, 5, 64)
SeperableConv2D	$4 \times 4 \times 64$	Same	2×2	5184	(None, 3, 3, 64)
Maxout	64	-	-	0	(None, 3, 3, 64)
MaxPooling2D	2×2	-	-	0	(None, 1, 1, 64)
Dropout	0.25	-	-	0	(None, 1, 1, 64)

The depthwise convolution takes an input feature map F with a size of $W_{in} \times H_{in} \times M$ (width, height, and number of filters) and generates an output feature map O with a size of $W_{out} \times H_{out} \times N$. It performs spatial convolution independently over every channel of the input, using a convolutional kernel \hat{K} with a size filter of $W_K \times H_K \times M \times N$. The depthwise convolution can be expressed as (1).

$$\hat{O}_{k,l,m} = \sum_{ij} \hat{K}_{i,j,m,n} \hat{O}_{k+i-1,l+j-1,m} \tag{1}$$

Next, the pointwise convolution is performed using a filter \tilde{K} with a size of 1×1 to combine the total generated output.

$$\hat{O}_{k,l,n} = \sum_{ij} \tilde{K}_{m,n} \hat{O}_{k-1,l-1,m} \tag{2}$$

The combination of depthwise and pointwise significantly reduces the number of network parameters, eliminating a large chunk of multiplication, resulting in a faster model for training and execution. Additionally, this combination ensures the classification accuracy of the model, making it less prone to overfitting. After each depthwise separable convolution layer, a Maxout layer is inserted. The Maxout layer is known for its effectiveness when trained with Dropout and for its robustness in improving network capacity. The Maxout activation function can be expressed as (3),

$$f(x) = \max_{j \in [1, k]} Z_{ij} \quad (3)$$

where $Z_{ij} = x^T W_{ij} + b_{ij}$, with x representing the input variables, W representing the weights, and b representing the biases. The model also incorporates a MaxPooling2D layer followed by a Dropout layer with a value of 25%. The outputs of the three CNN branches are concatenated, then flattened to form a vector of features, which is then passed through a dense layer with 50 units followed by a dropout of 40%.

2.3. Extremely randomized trees

Decision tree (DT) based ensemble methods aim to improve predictive performance by combining the outputs of multiple trees [24]. However, it is necessary to consider that the individual trees must be accurate and distinct from each other to obtain a more stable and robust classifier [24]. This can be achieved by using the randomization process, which helps reduce correlation and allows the trees to grow with greater diversity. ETs consist of a completely random and independent set of DTs, constructed using random subsets of features to minimize overfitting [24]. They make predictions about a target variable based on a sequence of rules defined in a forest-like structure. The implementation steps of ETs can be summarized as follows: Given a training dataset $X = \{x_1, x_2, \dots, x_i\}$, where a sample $x_1 = \{f_1, f_2, \dots, f_D\}$ is a D-dimensional vector with f_j as the feature and $j \in \{1, 2, \dots, D\}$. Three important parameters required for Extra-Trees are: the number of trees M , the number of randomly selected attributes K at each node, and the minimum sample size n_{min} required to split a node.

- a) A DT is constructed, consisting of a root node, split nodes, and leaf nodes. Starting with the root node, the DT grows in a top-down fashion, using entire training sample.
- b) At each internal node, the DT randomly selects K features $\{f_1, f_2, \dots, f_K\}$. For each feature k within the subset, its maximum and minimum values, fk_{max} and fk_{min} , are calculated. The optimal split value (cut point), which has the maximum variance reduction capabilities, is then selected from the range $[fk_{min}, fk_{max}]$. In detail, we use entropy as the score function, where the best split is determined by the feature with the least entropy and remains constant while the tree grows. Entropy is calculated using the (4),

$$Entropy = -\sum_t p_t \cdot \log_2(p_t) \quad (4)$$

where p_t represents the probability of class t .

- c) Iteratively, the subsets are split, and the trees are expanded until only pure nodes remain in terms of outputs or a minimum number of training samples needed for splitting (n_{min}) is reached. This concludes the partitioning process and creates a leaf node, which predicts the class label.
- d) Steps (a), (b), and (c) are repeated M times, generating an extreme random tree model composed of M independent DTs.
- e) Finally, by aggregating the predictions of the M trees, the final classification result is obtained through majority voting of each class at the leaf nodes.

Among various tree-based classification methods, there are several reasons why ETs are the most suitable choice for this paper. Firstly, the extreme randomization scheme of the ETs algorithm makes it significantly faster and more efficient in terms of training time. Secondly, using the full training set to train individual trees contributes to its strong generalization capability. The ensemble nature of ETs allows for the aggregation of diverse decision trees, each trained on a different subset of the data. This diversity helps improve the overall performance and robustness of the algorithm. Lastly, an advantage of using ETs is that their implementation does not heavily rely on hyperparameter tuning. ETs offer a more straightforward approach with less emphasis on hyperparameter selection.

2.4. Dataset

To validate the effectiveness of the MIMO-CxT presented in this study, we use a public dataset known as the UWB Gestures dataset, proposed by Ahmed *et al.* [11]. This dataset was acquired from 8 volunteers with an average age of 25.75 years using three XeThru X4 IR-UWB radars. As shown in Figure 2, the radars were positioned at three different locations: left, top, and right, and operated independently. Each volunteer was instructed to perform 12 predetermined gestures 100 times each. These gestures include left-right (LR) swipe,

right-left (RL) swipe, up-down (UD) swipe, down-up (DU) swipe, diagonal (diag)-LR-UD swipe, diag-LR-DU swipe, diag-RL-UD swipe, diag-RL-DU swipe, clockwise rotation, anti-clockwise rotation, inward push, and an empty gesture. In total, the dataset contains 288,000 RGB images that represent the range variation over time for each gesture.

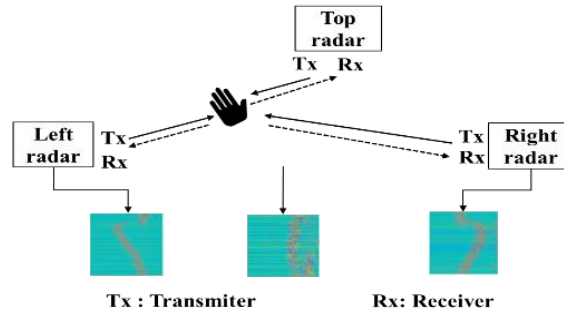


Figure 2. Diagram of radars location

2.5. Performance measures

In order to analyze the performance of our proposed model, various metrics have been used, including accuracy, precision, recall, and F1-score, defined by (5) to (8), respectively. These metrics are based on true positives T_p , true negatives T_N , false positives F_p , and false negatives F_N . Additionally, we have used graphical approach including precision-recall curve (PR) and receiver operating characteristic curve (ROC).

$$Accuracy = \frac{T_p + T_N}{T_p + F_p + T_N + F_N} \quad (5)$$

$$Precision = \frac{T_p}{T_p + F_p} \quad (6)$$

$$Recall = \frac{T_p}{T_p + F_N} \quad (7)$$

$$F1 - score = \frac{2T_p}{2T_p + F_p + F_N} \quad (8)$$

2.6. Implementation details

The model is implemented in Python using the Keras framework with TensorFlow backend. The hardware configuration is based on an Intel(R) Core (TM) i5 @ 2.40 GHz CPU, 16 GB of RAM, 1TB of hard disk, and Windows 10. All samples in the dataset were resized to 75×75 and converted to binary images as shown in Figure 3. The reasons for using binary images are faster inference, avoidance of unnecessary pre-processing, and lower storage requirements. Next, the converted samples were randomly split into 80% for training and 20% for testing. Additionally, the random seed parameter is used to ensure consistent test samples in each experiment. The Adam optimizer is used with a learning rate set to 0.001. The input labels are provided as integers; therefore, we use the sparse categorical cross-entropy loss function. This choice helps save memory and computation time compared to using vectors to encode the labels. To achieve optimal performance, the ETs hyperparameter settings are selected during the training process using the Optuna framework [25].

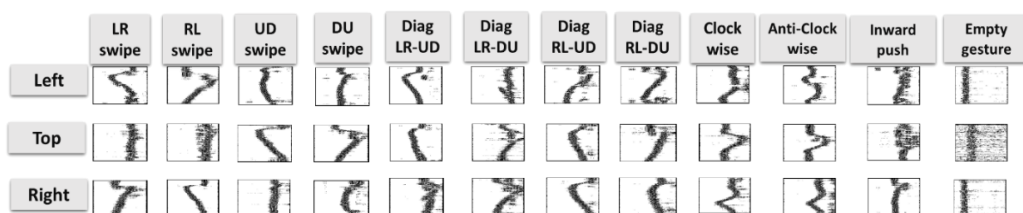


Figure 3. Example of binarized samples from the UWB Gestures dataset

3. RESULTS AND DISCUSSION

3.1. Results

3.1.1. Training process

The training process is conducted from scratch and can be divided into three main steps. Firstly, a CNN-Softmax model is used, where each CNN branch is fed with data from a specific sensor. Secondly, the CNN is trained with 15 epochs with different batch sizes. Finally, Softmax is replaced by the ETs classifier, which is fed with the extracted and merged features from the multiple CNN branches for classification. The results showed that the performance of the MIMO-CxT improves with smaller batch sizes. The best training accuracies were obtained with a batch size of 8 and 16. However, a batch size of 16 required relatively less training time compared to batch size of 8, and was therefore selected for the remaining experiments. Table 2 summarizes the results obtained for the different batch sizes.

Table 2. Train accuracy using different batch size

Metrics	Batch size				
	8	16	32	64	128
Train Accuracy (%)	99.67	99.55	99.30	99.21	98.98

Next, the MIMO-CxT is retrained using a batch size of 16 with the Optuna optimizer to fine-tune its hyperparameters. These hyperparameters include max depth, max features, min sample split, min samples leaf, n estimators, and max leaf nodes. Table 3 summarizes the optimal values for these hyperparameters. The best MIMO-CxT model achieved an accuracy of 99.7%, as shown in the optimization history plot in Figure 4.

Table 3. ETs hyperparameters

Hyperparameter	Value
max_depth	10
max_features	10
n_estimators	700
min_samples_leaf	5
min_samples_split	5

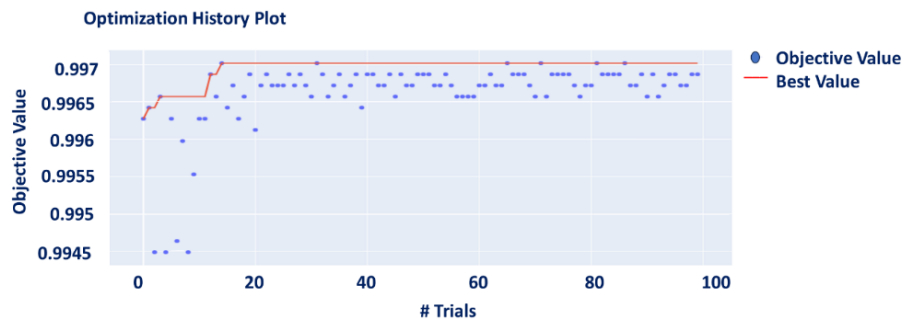


Figure 4. Optimization history plot

3.1.2. Evaluation process

For performance evaluation, the MIMO-CxT model configuration is used, along with the hyperparameters specified in Table 2. The results obtained on the test data are depicted in Figure 5, with the confusion matrix presented in Figure 5(a) and the classification report is presented in Figure 5(b). Although the obtained results are excellent, we also examine graphical approaches illustrated in Figure 6, including the PR curve presented in Figure 6(a) and the ROC presented in Figure 6(b).

3.1.3. Comparisons

a. Verification of data concatenation method

To demonstrate the feasibility and superiority of the proposed model, two different experiments were conducted. Firstly, the same model was fed with a single input from each sensor separately. Secondly, the single input model was fed with data from all three sensors simultaneously. Finally, the performance of these approaches was compared with the MIMO-CxT model. The results of these experiments are summarized in Table 4.

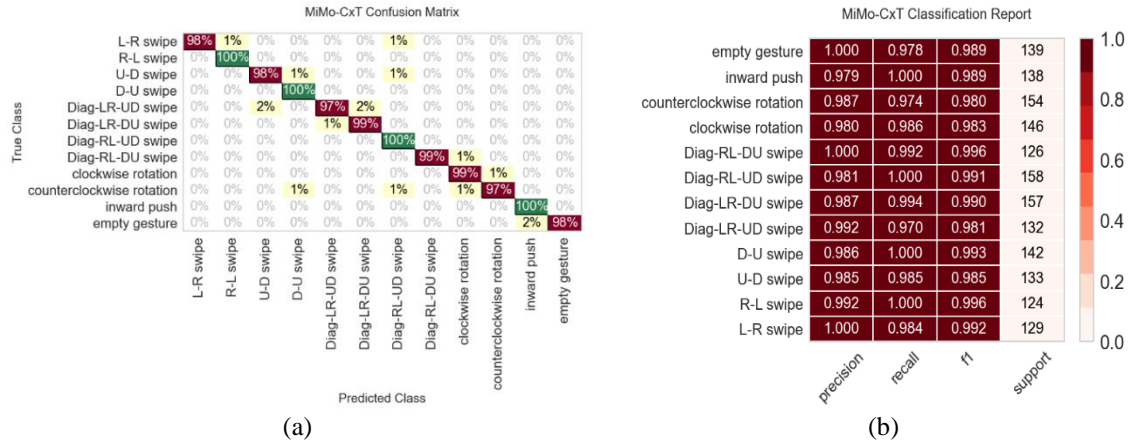


Figure 5. Classification performance of the MIMO-CxT on the test set (a) confusion matrix and (b) classification report

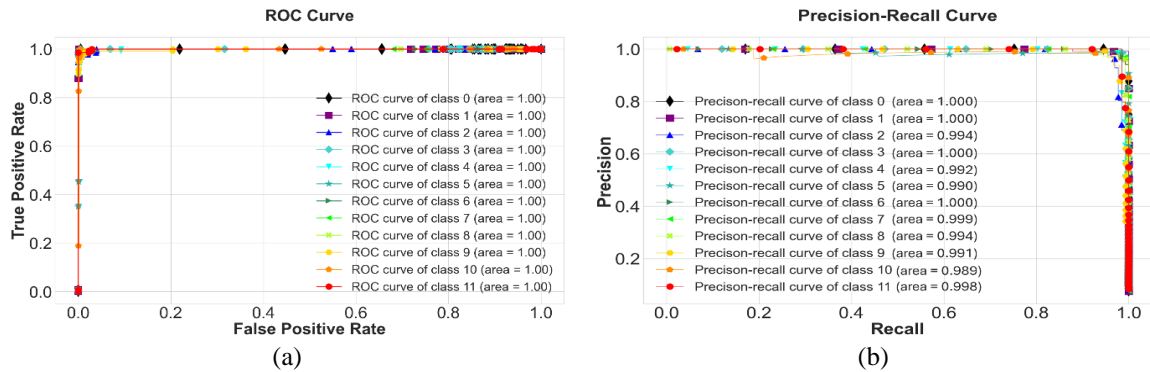


Figure 6. Graphical plots of MIMO-CxT on the test set: (a) ROC curve and (b) PR curve

Table 4. Comparative classification performance single/multiple inputs model

Metrics	Single-input multi-output CxT				MIMO-CxT
	Left	Top	Right	Left+Top+Right	
Train accuracy (%)	88.86	85.65	86.76	78.91	99.70
Test accuracy (%)	81.52	75.26	80.95	72.70	98.86
Precision (%)	82.14	76.21	80.43	73.62	98.90
Recall (%)	81.52	75.26	80.09	73.70	98.86
F1-score (%)	81.68	75.31	80.07	72.66	98.85

b. Verification of data preprocessing

In order to assess the effectiveness of utilizing binary images compared to RGB images, the MIMO-CxT model was trained and tested on both types of images. A comprehensive analysis was conducted, considering factors such as classification performance, model complexity, and computation times. The results of this evaluation are presented in Table 5.

Table 5. Comparative classification performance RGB/Binary images

Metrics	RGB images	Binary images
Train Accuracy (%)	98.73	99.70
Test Accuracy (%)	97.19	98.86
Precision (%)	97.23	98.90
Recall (%)	97.15	98.86
F1-score (%)	97.21	98.85
CNN parameters	32016	31536
Training time (s)	527.5141	450.3863
Prediction time (s)	2.7767	1.9582

c. Comparison to existing models

The third experiment aims to compare the performance of different classifiers against ETs when used in a hybrid configuration with a multi-input CNN. Additionally, as the first paper to propose a multi-input model for HGR with IR-UWB radar, we compared our model with single-input models used in the same context and multi-input models proposed in the literature. Table 6 summarizes the results obtained from different models trained on the public UWB Gestures dataset.

Table 6. Comparative classification performance of the MIMO-CxT over existing approaches

	Model	Train accuracy (%)	Test accuracy (%)
Single-input	Four-layer CNN [11]	90.47	78.90
	Six-layer CNN [10]	96.94	87.63
	Three-input CNN [26]	98.94	97.37
Multi-input	MIMO-CNNLSTM	96.11	97.79
	MIMO-CNNSoftmax	97.14	97.99
	Multi-stream CNN [27]	98.93	98.15
	MIMO-CNNSVM	98.94	98.18
Our proposed model	MIMO-CxT	99.70	98.86

3.2. Discussion

Employing multi-sensor systems for HGR offers numerous advantages such as increased accuracy, robustness, the ability to capture complex gestures, versatility, and adaptability to different needs. These benefits make multi-sensor systems preferable over single-sensor solutions. However, it is important to consider a solution capable of efficiently handling and integrating information from a variety of sensor sources. To address this, we present MIMO-CxT, an end-to-end hybrid deep learning approach for HGR based on multi-sensor systems.

The first experiment was conducted to analyze the performance of the model based on how the sensor data features are extracted: i) separately, ii) together, or iii) independently in parallel. Table 4 shows that the multiple-input model is able to recognize gestures much better than the single-input model in all tested scenarios. One of the main reasons for this improvement is the quantity of extracted features. It is evident that the multiple-input model achieves a high recognition rate due to its ability to extract and merge more diverse information concerning the same gesture across multiple sensors. Furthermore, we observed a significant difference in the model's performance when processing all data simultaneously and independently in parallel. We hypothesize that the reason for this poor performance is the presence of common features. Although the same gesture is represented differently by the three sensors, there is a strong similarity between the signals of different gestures. For example, the RL swipe of the left radar with the DU swipe of the top radar from Figure 3. This similarity in features between different gestures leads to confusion for the model and makes it error-prone. In addition, the single-input architecture is slower to converge and requires a significant amount of time for training. From the first experiment, we can reasonably conclude that processing data from multiple sensors in parallel and fusing their extracted features greatly improves the recognition rate while requiring less training time.

We conducted a second experiment in which we compared the performance of the model based on the image processing technique adopted in our work. Table 5 shows that using binary images results in lower inference and prediction times compared to red green blue (RGB) images. Using binary images helps filter out unnecessary information while retaining the main features of the gestures. Moreover, it reduces the number of trainable parameters while maintaining high model performance. Therefore, implementing our model does not require powerful computing hardware.

The same experiment was carried out on single/multiple input models already proposed in the literature, and the results are shown in Table 6. Comparing the performance of the single-input four-layer CNN [11] and six-layer CNN [10] models against MIMO-CxT, it is evident that the latter outperforms in terms of test accuracy and generalizability. Standard CNNs are designed for extracting features from a single input, which may result in insufficient capacity to capture the full range of data variations between different sensors. However, the implementation of MIMO-CxT with a multi-branch structure overcomes this limitation. The multi-branch structure acts as a regularizer, allowing the model to effectively capture the diverse variations present in the data. By utilizing different branches, the model can learn more robust and discriminative representations, ultimately leading to improved classification performance. The results were significantly improved with the use of multiple-input models. However, MIMO-CxT has several advantages compared to the models proposed in the literature. MIMO-CxT achieved an increase in accuracy of 1.49% and 0.71% when compared to the three-input CNN [26] and the multi-stream CNN [27], respectively. Both models [26], [27] have a deeper and more complex architecture than ours. They both use a large number of convolutional filters and several fully connected layers with a large number of units. This considerably increases the number of

trainable parameters, which in turn increases training time. We can also notice that combining CNN with LSTM slightly increases accuracy but still falls short of that achieved by MIMO-CxT. Taking into account the complex architecture with the sequential nature of the long short-term memory (LSTM), ETs are easy to implement and computationally more efficient. The training and inference times of ETs are faster, especially for large datasets. This computational efficiency is beneficial when working with real-time or near-real-time predictions. Although Softmax is a powerful classifier, the utilization of ETs enhanced accuracy by 0.86%. This improvement can be primarily attributed to the implementation of ensemble learning techniques by ETs. Employing individual trees enables the capture and learning of distinct information, while aggregating predictions from multiple trees leads to more reliable predictions. Using ETs as the final classifier yielded similar performance compared to support vector machine (SVM). However, ETs are inherently well-suited for multiclass classification tasks. Contrary to SVM, ETs can handle them directly without having to reduce them to several binary classification problems. Furthermore, ETs tend to be more scalable than SVM, especially when dealing with a large number of data points. The computational complexity of SVM grows rapidly as the number of data points increases.

In conclusion, ETs are an obvious choice for the proposed model to perform HGR. The results presented in Tables 4 to 6 indicate that the proposed MIMO-CxT model effectively recognizes multiple sensor-based gestures and outperforms existing approaches. MIMO-CxT, with its implementation of ETs, achieves good generalization performance while requiring fewer parameters. This implies that the model can efficiently classify and differentiate various hand gestures based on the acquired sensory signals from IR-UWB sensors.

4. CONCLUSION

Considering the high prevalence of upper limb paralysis after stroke, as well as the heavy burden on the healthcare system, there is a need for accurate and cost-effective telemedicine tools designed for the remote rehabilitation of stroke patients. To address this problem, we proposed a novel deep hybrid model, named MIMO-CxT, for HGR. The proposed architecture leverages the robustness of CNN to capture both shallow and deep features, as well as the simplicity and strong generalization capability of ETs for data classification. The advantages of the MIMO-CxT model include multi-source input, fast computational speed, low cost, and high generalization performance. The results were compared with those of conventional approaches, and it was found out that our model performs significantly better. Based on the performance of the MIMO-CxT, it can be considered as a promising solution to assist medical professionals as a home-based online monitoring tool for stroke patients. Future research in this field may explore other image processing and optimization techniques. Additionally, we plan to evaluate the proposed model on other recognition tasks, such as human activity.




REFERENCES

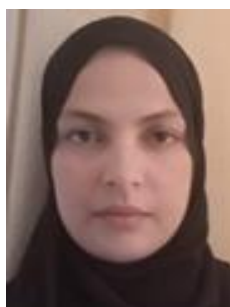
- [1] J. R. Villar, S. González, J. Sedano, C. Chira, and J. M. Trejo-Gabriel-Galan, "Improving human activity recognition and its application in early stroke diagnosis," *International Journal of Neural Systems*, vol. 25, no. 04, Jun. 2015, doi: 10.1142/S0129065714500361.
- [2] H. U. Nam *et al.*, "Effect of dominant hand paralysis on quality of life in patients with subacute stroke," *Annals of Rehabilitation Medicine*, vol. 38, no. 4, 2014, doi: 10.5535/arm.2014.38.4.450.
- [3] J. C. Martins, L. T. Aguiar, S. Nadeau, A. A. Scianni, L. F. Teixeira-Salmela, and C. D. C. de M. Faria, "Measurement properties of self-report physical activity assessment tools in stroke: a protocol for a systematic review," *BMJ Open*, vol. 7, no. 2, Feb. 2017, doi: 10.1136/bmjopen-2016-012655.
- [4] S. Ahmed, K. D. Kallu, S. Ahmed, and S. H. Cho, "Hand gestures recognition using radar sensors for human-computer-interaction: A review," *Remote Sensing*, vol. 13, no. 3, Feb. 2021, doi: 10.3390/rs13030527.
- [5] X. Li, Y. He, and X. Jing, "A survey of deep learning-based human activity recognition in radar," *Remote Sensing*, vol. 11, no. 9, May 2019, doi: 10.3390/rs11091068.
- [6] A. Sluÿters, S. Lambot, J. Vanderdonck, and R.-D. Vatavu, "RadarSense: Accurate recognition of mid-air hand gestures with radar sensing and few training examples," *ACM Transactions on Interactive Intelligent Systems*, Mar. 2023, doi: 10.1145/3589645.
- [7] H. Li, A. Mehul, J. Le Kerneç, S. Z. Gurbuz, and F. Fioranelli, "Sequential human gait classification with distributed radar sensor fusion," *IEEE Sensors Journal*, vol. 21, no. 6, pp. 7590–7603, Mar. 2021, doi: 10.1109/JSEN.2020.3046991.
- [8] S. Chioccarello, A. Sluÿters, A. Testolin, J. Vanderdonck, and S. Lambot, "FORTE: Few samples for recognizing hand gestures with a smartphone-attached radar," *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, no. EICS, pp. 1–25, Jun. 2023, doi: 10.1145/3593231.
- [9] H. Liu and Z. Liu, "A multimodal dynamic hand gesture recognition based on radar-vision fusion," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–15, 2023, doi: 10.1109/TIM.2023.3253906.
- [10] S. Ahmed, F. Khan, A. Ghaffar, F. Hussain, and S. Cho, "Finger-counting-based gesture recognition within cars using impulse radar with convolutional neural network," *Sensors*, vol. 19, no. 6, Mar. 2019, doi: 10.3390/s19061429.
- [11] S. Ahmed, D. Wang, J. Park, and S. H. Cho, "UWB-gestures, a public dataset of dynamic hand gestures acquired using impulse radar sensors," *Scientific Data*, vol. 8, no. 1, Apr. 2021, doi: 10.1038/s41597-021-00876-0.
- [12] Y. Li, X. Wang, B. Shi, and M. Zhu, "Hand gesture recognition using IR-UWB radar with ShuffleNet V2," in *Proceedings of the 5th International Conference on Control Engineering and Artificial Intelligence*, Jan. 2021, vol. 58, no. 22, pp. 126–131, doi: 10.1145/3448218.3448233.




- [13] S. Skaria, A. Al-Hourani, and R. J. Evans, "Deep-learning methods for hand-gesture recognition using ultra-wideband radar," *IEEE Access*, vol. 8, pp. 203580–203590, 2020, doi: 10.1109/ACCESS.2020.3037062.
- [14] F. M. Noori, M. Z. Uddin, and J. Torresen, "Ultra-wideband radar-based activity recognition using deep learning," *IEEE Access*, vol. 9, pp. 138132–138143, 2021, doi: 10.1109/ACCESS.2021.3117667.
- [15] J. Park, J. Jang, G. Lee, H. Koh, C. Kim, and T. W. Kim, "A time domain artificial intelligence radar system using 33-GHz direct sampling for hand gesture recognition," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 4, pp. 879–888, Apr. 2020, doi: 10.1109/JSSC.2020.2967547.
- [16] Z. Slimane, K. Lakhdari, and D. S. Korti, "Enhancing dynamic hand gesture recognition using feature concatenation via multi-input hybrid model," *International journal of electrical and computer engineering systems*, vol. 14, no. 5, pp. 535–546, Jun. 2023, doi: 10.32985/ijecs.14.5.5.
- [17] G. Park, V. K. Chandrasegar, and J. Koh, "Accuracy enhancement of hand gesture recognition using CNN," *IEEE Access*, vol. 11, pp. 26496–26501, 2023, doi: 10.1109/ACCESS.2023.3254537.
- [18] Y. Yang, J. Li, B. Li, and Y. Zhang, "MDHandNet: a lightweight deep neural network for hand gesture/sign language recognition based on micro-doppler images," *World Wide Web*, vol. 25, no. 5, pp. 1951–1969, Sep. 2022, doi: 10.1007/s11280-021-00985-1.
- [19] S. Skaria, D. Huang, A. Al-Hourani, R. J. Evans, and M. Lech, "Deep-learning for hand-gesture recognition with simultaneous thermal and radar sensors," in *2020 IEEE Sensors*, Oct. 2020, pp. 1–4, doi: 10.1109/SENSORS47125.2020.9278683.
- [20] S. Ahmed and S. H. Cho, "Hand gesture recognition using an IR-UWB radar with an inception module-based classifier," *Sensors*, vol. 20, no. 2, Jan. 2020, doi: 10.3390/s20020564.
- [21] L. Qiao, Z. Li, B. Xiao, Y. Shu, W. Li, and X. Gao, "Gesture-ProxyleSSNAS: A lightweight network for mid-air gesture recognition based on UWB radar," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pp. 1–10, 2023, doi: 10.1109/JSTARS.2023.3274830.
- [22] R. R. Sharma, K. A. Kumar, and S. H. Cho, "Novel time-distance parameters based hand gesture recognition system using multi-UWB radars," *IEEE Sensors Letters*, vol. 7, no. 5, pp. 1–4, May 2023, doi: 10.1109/LENS.2023.3268065.
- [23] F. Khan, S. K. Leem, and S. H. Cho, "In-sir continuous writing using UWB impulse radar sensors," *IEEE Access*, vol. 8, pp. 99302–99311, 2020, doi: 10.1109/ACCESS.2020.2994281.
- [24] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, Apr. 2006, doi: 10.1007/s10994-006-6226-1.
- [25] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul. 2019, pp. 2623–2631, doi: 10.1145/3292500.3330701.
- [26] Y. Sun, L. Zhu, G. Wang, and F. Zhao, "Multi-input convolutional neural network for flower grading," *Journal of Electrical and Computer Engineering*, vol. 2017, pp. 1–8, 2017, doi: 10.1155/2017/9240407.
- [27] J. A. Aghamaleki and V. Ashkani Chenarlogh, "Multi-stream CNN for facial expression recognition in limited training data," *Multimedia Tools and Applications*, vol. 78, no. 16, pp. 22861–22882, Aug. 2019, doi: 10.1007/s11042-019-7530-7.

BIOGRAPHIES OF AUTHORS



Djazila Souhila Korti    received her Master degree in telecommunication at the University of Abou Bekr Belkaid, Tlemcen (Algeria) in 2020. Currently she is a Ph.D. student at the University of Belhadj Bouchaib, Ain-Temouchent (Algeria) and a member of SSL laboratory. Her doctoral research center around ultra-wideband technology and its applications in in the field of human activity and gesture recognition. She can be contacted at email: souhila.korti@univ-temouchent.edu.dz and souhilakorti@gmail.com.



Zohra Slimane    received Magister Diploma (2008), Ph.D. (2012) and HDR (2017), in Telecommunication from Tlemcen University (Algeria). Since 2008, she has been researcher at STIC Laboratory and system engineer at Sonatrach Aval Research Group. In 2014, she joined Belhadj Bouchaib University Center (Algeria), where she is currently an Associate and Research Professor, responsible for LMD graduation and doctorate training in the telecommunications sector. Since 2019, she has been involved as project leader dedicated to indoor localization. Her research interests include location and imaging radars, UWB sensors, networking, mobile networks, ubiquitous internet, and next-generation networks. She can be contacted at email: zohra.slimane@univ-temouchent.edu.dz and zoh_slimani@yahoo.fr.