

MSEC2023-104792

FORECASTING THE RANGE OF POSSIBLE HUMAN HAND MOVEMENT IN CONSUMER ELECTRONICS DISASSEMBLY USING MACHINE LEARNING

Hao-yu Liao

Graduate Research Assistant
Environmental Engineering Sciences
University of Florida, Gainesville, FL, 32611
haoyuliao@ufl.edu

Boyi Hu

Assistant Professor
Industrial and Systems Engineering
University of Florida, Gainesville, FL, 32611
boyihu@ise.ufl.edu

Yuhao Chen

Graduate Research Assistant
Industrial and Systems Engineering
University of Florida, Gainesville, FL, 32611
yuhaochen@ufl.edu

Xiao Liang

Assistant Professor
Civil, Structural, and Environmental Engineering
University at Buffalo, Buffalo, NY, 14260
liangx@buffalo.edu

Sara Behdad*

Associate Professor
Environmental Engineering Sciences
University of Florida, Gainesville, FL, 32611
sarabehdad@ufl.edu

ABSTRACT

Robotic technology can benefit disassembly operations by reducing human operators' workload and assisting them with handling hazardous materials. Safety consideration and predicting human movement is a priority in human-robot close collaboration. The point-by-point forecasting of human hand motion which forecasts one point at each time does not provide enough information on human movement due to errors between the actual movement and predicted value. This study provides a range of possible hand movements to enhance safety. It applies three machine learning techniques including Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Bayesian Neural Network (BNN) combined with Bagging and Monte Carlo Dropout (MCD), namely LSTM-Bagging, GRU-Bagging, and BNN-MCD to predict the possible movement range. The study uses an Inertial Measurement Units (IMU) dataset collected from the disassembly of desktop computers to show the application of the proposed method. The findings reveal that BNN-MCD outperforms other models in forecasting the range of possible hand movement.

Keywords: Human Motion Prediction, Gated Recurrent Unit, Bayesian Neural Network, Long Short-

Term Memory, Human-Robot Collaboration, Disassembly, Remanufacturing

1. INTRODUCTION

Human-robot collaboration in disassembly operations is receiving attention in recent years. Several topics such as disassembly sequence planning, object detection, human activity recognition, and human motion prediction, are important when it comes to the disassembly operation in human-robot collaboration.

The above-mentioned topics aim to facilitate human-robot collaboration from different aspects. The disassembly sequence planning determines the most suitable sequence for dismantling a product and sometimes specifies the task allocation between the human and the robot. Previous studies considered factors such as cost and safety when allocating disassembly tasks between humans and robots in human-robot collaboration [1]–[3]. The idea is to use the capabilities of robots for handling hazardous tasks and enhancing operator safety. Object detection allows the robot to identify the objects for grasping, picking, and holding actions [4]. Human activity recognition allows the robot to operate autonomously while increasing work productivity [5][6].

Besides disassembly sequence planning, object detection, and human activity recognition, human motion prediction plays an important role in enhancing the operator's safety [7][8]. One of the main challenges in human motion prediction is the complexity of considering uncertainties in human motion [9].

Previous studies have used a wide range of methods in addressing human motion prediction in different applications. To name a few, Wang et al. [10] used the Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) to predict human motion for objects such as cup, stone, sponge, spoon, and knife with different actions. Li et al. [11] applied Directed Acyclic Graph Neural Network (DA-GNN) to predict human motion in the CMU MOCAP and H3.6M dataset for actions such as walking and eating. Martinez et al. [12] used Gated Recurrent Unit (GRU) for human motion prediction using the H3.6M dataset, and Pavlo et al. [13] combined the Quarter Net framework with GRU to predict human motion using the H3.6M dataset. Zheng et al. [14] applied LSTM to forecast human arm motion on the generated data from a Franka Emika Panda Cobot. Wang and Shen [8] used the neural networks combined with Kalman filtering to predict human hand motion for picking actions. Wang et al. [15] applied LSTM for hand motion on the surface grinding plane. Zhang et al. [16] built a Recurrent Neural Network (RNN) model to predict motion trajectory prediction in the assembly process. Gril et al. [17] adopt the linear tensor regression model to predict the human motion in the assembly and disassembly operation of six pins, springs, and ball bearings repetitively. Liao et al. [18] combined Convolutional Long Short-term Memory (ConvLSTM) and You Only Look Once (YOLO) to predict human hand motion in the disassembly process of desktops.

Previous studies also have investigated the uncertainty of tasks and human motion in human-robot collaboration. To name a few, Burks et al. [19] proposed an assisted robotic planning and sensing framework and applied the online partially observable Markov decision process for semantic sensing and planning under uncertain environments. Sajedi et al. [20] applied the Bayesian neural networks to quantify the uncertainty for semantic segmentation of hands in human-robot collaboration. Furnari et al. [21] discussed the loss function incorporating uncertainty for the egocentric action anticipation and recognition methods. Farha and Gall [22] developed a framework for modeling the uncertainty of future activities and predicted the probability distribution of activities. Casalino et al. [23] developed a fuzzy approach for scheduling assembly tasks considering uncertain durations of tasks in a human-robot collaboration setting.

Although previous studies have extensively addressed human motion prediction, the literature on predicting the movement interval is still limited. This study aims to investigate the performance of three machine learning models - Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Bayesian Neural Network (BNN) - in combination with Bagging and Monte Carlo Dropout (MCD) techniques for estimating the potential range of human motion. Specifically, we examine the performance of three model variants: LSTM-Bagging, GRU-Bagging, and BNN-MCD. We also explore the unique application of electronic waste (e-waste) disassembly. Table 1 provides a comparison of this study with prior work.

E-waste is becoming a serious environmental and economic problem. In 2019, 53.6 million tons of e-waste were generated around the globe with a growth rate of 21% [24]. Product recovery solutions such as eco-design policies and facilitating disassembly operations are important for e-waste recovery [25].

TABLE 1: Comparison of literature and this study.

Reference	Methodology	Type of forecast	Experimental Process	Collision	HRC	E-waste
[8]	Neural Network with Kalman filtering	Point	Pick up tasks			
[10]	LSTM with CNN	Point	Objects manipulation	√	√	
[11]	Directed Acyclic Graph Neural Network	Point	CMU MOCAP and H3.6M			
[12]	GRU	Point	Human 3.6M			
[13]	QuaterNet with GRU	Point	Human 3.6M			
[14]	LSTM	Point	Arm motion	√	√	
[15]	LSTM	Point	Surface grinding		√	
[16]	RNN	Point	Assembly	√	√	
[17]	Linear tensor regression	Interval	Assembly & Disassembly	√	√	
[18]	ConvLSTM with YOLO	Point	Disassembly	√	√	√
This study	LSTM-Bagging GRU- Bagging BNN- MCD	Interval	Disassembly	√	√	√

E-waste disassembly is particularly unique since it involves the separation and recovery of a complex mix of materials ranging from metals to hazardous materials. Also, e-waste disassembly often requires considering small and complex parts that are difficult to dismantle. Further, the high variability in consumer electronics design makes disassembly challenging for the remanufacturing workforce. Also, chemical exposure and physical and ergonomic hazards increase the risk of disassembly operations for human workers. Thus, the disassembly of e-waste requires further investigation.

The focus of the study will be on disassembling desktop computers. This paper is organized as follows. Section 2 provides an overview of LSTM, GRU, and BNN models. Section 3 describes the dataset and data collection experiment. Section 4 provides the prediction results. Finally, Section 5 concludes the paper.

2. METHODOLOGY

This section describes the three machine learning models combined with Bagging and MCD.

2.1 Long Short-Term Memory (LSTM) with Bagging

LSTM network is proposed to alleviate the drawback of RNN on the vanishing gradient problem [26]. The LSTM consists of three gates namely forget gate, input, and output gates. The detail of LSTM equations can be found in [26].

We used PyTorch to construct the LSTM network [27]. The number of layers is 3 with 64 hidden sizes. The dropout probability is 20 in the last layer. The learning rate is set to 1e-3 with a weight decay of 1e-6, and the number of epochs is 100. We used the Adam optimizer and the squared l2 norm as the loss function.

Furthermore, we applied Bagging to LSTM. Bagging, also known as bootstrap aggregation [28], is an ensemble learning method for reducing the variance [29] and avoids the local optimal solution by repeating the training process [30]. Bagging has already shown his promise in previous literature [31]–[33]. In this paper, we trained LSTM 30 times. The ensemble prediction is the mean of 30 predicted values from LSTM.

2.2 Gated Recurrent Unit with Bagging

The GRU is a type of RNN introduced in 2014 by Cho et al. [34]. The GRU addresses the vanishing gradient issue in the standard RNN [35] which occurs when the gradient is too small to change the weight. The GRU formulation can be found in [34].

The parameters for the GRU model are set the same as the LSTM model. GRU has a simple structure for training and can address the issues of memory use, gradient disappearance, and gradient explosion [36]. As proven by previous studies [37], [38], Bagging can improve the performance of LSTM and GRU. Therefore, we utilized Bagging in both LSTM and GRU.

2.3 Bayesian Neural Network with Monte Carlo Dropout

The BNN estimators are formulated by the maximum a posteriori (MAP) estimator to decide the parameters. The weights of BNN are decided by the probability of the Gaussian process compared to the traditional neural networks with fixed weights [39][40]. According to [41], the parameters can be estimated from:

$$P(W|D) = \frac{P(D|W)P(W)}{P(D)} \quad (1)$$

W and D are the weight parameters and the observed data respectively. $P(W|D)$ is the posterior probability, $P(D|W)$ is the likelihood observation, and $P(W)$ is the prior probability for the weights. In this paper, we built the BNN model from the package by Lee et al. [42] and included two hidden layers.

The activation function of the first and second layers are ReLU and linear. Each layer has a dropout with a probability of 20%. The loss function by the default package is the combination of squared l2 norm and Kullback–Leibler divergence. The remaining parameters such as the optimizer, learning rate, and other hyperparameters are the same as the LSTM setting.

In addition, we applied MCD to reproduce the results from the BNN model. The MCD is proposed by Gal and Ghahramani [43]. It allows the activation of the dropout in the testing phase [44]. The MCD changes the model architecture each time when providing the prediction. This paper runs MCD on BNN 30 times. After conducting the 30 predictions, the ensemble prediction outcome can be computed. Monte Carlo Dropout combined with Bayesian inference has received attention in different fields due to its simplicity, scalability, and computational efficiency [45].

2.4 Possible hand movement area

After estimating the possible hand movement areas, the information can be provided to the robot control algorithm to avoid collisions. Figure 1 shows the concept of possible hand movement area from GRU with Bagging.

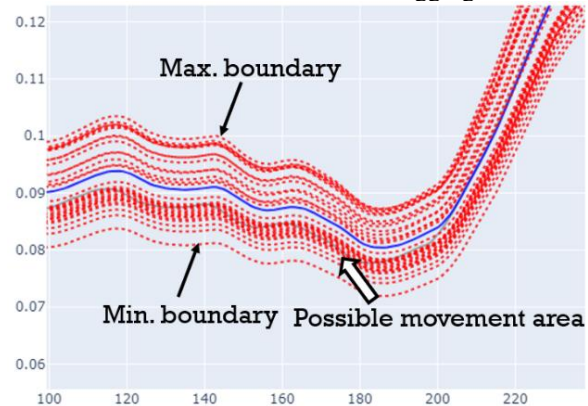


FIGURE 1: Possible hand movement area by applying Bagging or MCD.

The upper and lower bounds can be calculated from multiple runs of Bagging and MCD. Each model will run 30 times by applying Bagging or MCD. The number of predicted values is 30. Among these 30 predictions, the maximum and minimum values are considered as the boundary.

3. THE DISASSEMBLY DATASET

This section describes the data collection procedure and the disassembly experiment for a desktop computer.

3.1 Dataset of Dell OptiPlex 7050 Micro desktop for disassembling

The required dataset has been collected by using Inertial Measurement Units (IMU) sensors. Six sensors were deployed on a participant as shown in Figure 2. The product under disassembly is a Dell OptiPlex 7050 Micro desktop computer. Six components have been dismantled from the desktop in the following order: 1) screw, 2) cover, 3) hard disk drive, 4) fan, 5) heat sink, and 6) RAM (Figure 3).



FIGURE 2: Six IMU sensors on the participant.

1: Screw & 2: Cover

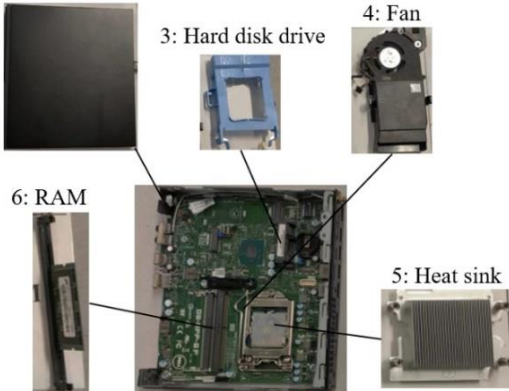


FIGURE 3: Each component of Dell OptiPlex 7050 Micro desktop.

The participant has completed the consent information and the experiment was authorized by the University of Florida Institutional Review Board (IRB 202200211). The frequency of IMU sensors is 60 HZ meaning the output of the sensor is 60 samples per second. The total samples are 6,686 with a total disassembly time of around 111 seconds. The duration between the samples is 16.67 milliseconds. The number of collected samples for 1) screw, 2) cover, 3) hard disk drive, 4) fan, 5) heat sink, and 6) RAM are 607 (10.1s), 335 (5.6 s), 464 (7.7 s), 734 (12.2 s), 3773 (62.9 s), and 775 (12.9 s), respectively. The proportion of training, validation, and testing is 70%, 20%, and 10% for each component. For example, the number of training, validation, and testing samples for the screw is 425 (70%), 121 (20%), and 61 (10%). The duration between the current time t and the predicted time for $t+1$, $t+2$, and $t+3$ is 16.67 ms, 33.34 ms, and 50.01 ms, respectively.

3.2 Time length for input and output

The hand's X, Y, and Z positions are collected by the sensors for the entire disassembling operation. The movement of collected samples is shown in Figure 4. The unit for hand positions is mm.

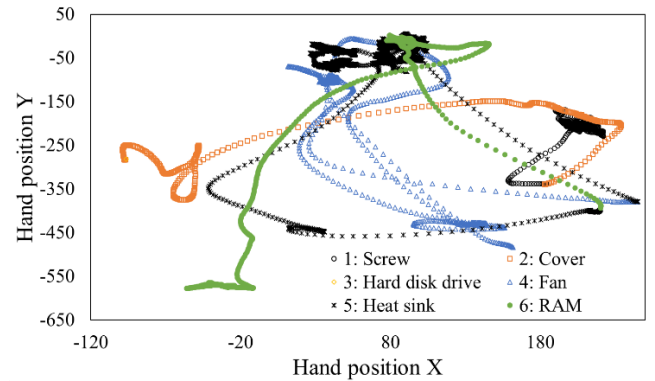


FIGURE 4: The visual representation of 6,686 sensor data collected for each component of the desktop.

According to Figure 4, the hand movement for each component is different. The length of the input window is decided based on the Pearson correlation coefficient (PCC) from the next time $t+1$ to the previous time e.g. $t-1, \dots, t-n$. One advantage of the Pearson correlation coefficient is that it can quantify the degree to which variables are linearly related and provide a measure of the proportion of variance shared between them.[46]. The input time length is selected when the PCC between the next time and the previous time is above 0.99. The higher length of input will increase the complexity and computation time. Therefore, we only selected lag features with 0.99 PCC.

The concept of the input window for input and output is described in Figure 5. For predicting the value of the hand's X position at time $t+1$ as output, the size of the input window is 7 time points, from $t-6$ to t , as input. Similarly, for predicting the Y position at time $t+1$, we used a longer

window of the previous 10 time points, from $t-9$ to t , as input. And for predicting the Z position at time $t+1$, we used a window of the previous 9 time points, from $t-8$ to t , as input. The PPC of each input time and output time is at least 0.99 above. When forecasting $t+2$, the input window will be shifted with 1 time lag without changing its size. For example, when forecasting $t+2$ in hand position X, the input will be from $t-5$ to $t+1$.

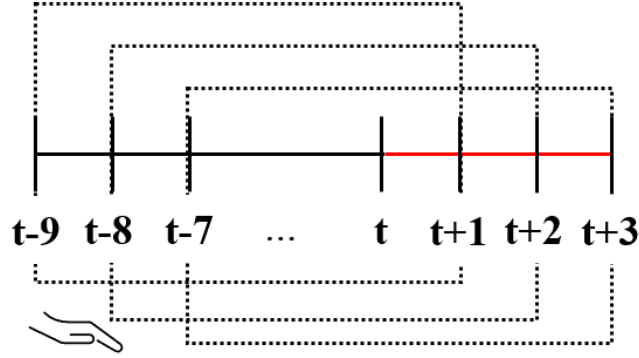


FIGURE 5: The input and output window for predicting the hand's Y position.

Figure 5 shows that the LSTM model uses a window of the previous 9 time points, from $t-9$ to t , as input for predicting the position of the hand's Y coordinate at time $t+1$. The predicted value for Y at time $t+1$, along with the input window from $t-8$ to t , are then used as inputs for predicting the hand's position at time $t+2$. This process is repeated for predicting the hand's position at time $t+3$. In Figure 6, the maximum hand movement between time t and $t+3$ is 53.26 mm (5.32 cm), meaning the hand moves 5.32 cm in 50 milliseconds. The hand movement is rapid as the hand can move 5.32 cm in 50 milliseconds and should be carefully predicted to avoid any collision.

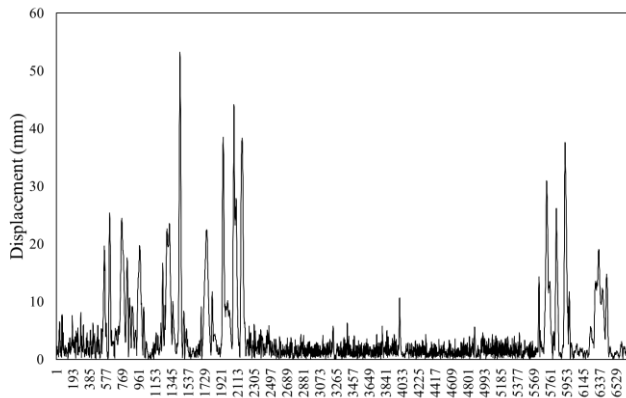


FIGURE 6: The displacement (x, y) between t and $t+3$.

4. THE RESULTS OF HUMAN HAND MOTION PREDICTION

This section discusses the results and compares the findings of the three models.

4.1 Human hand motion prediction

The mean absolute error (MAE), mean square error (MSE), and root mean square error (RMSE), are used to evaluate the performance of applying the three models for predicting $t+1$ to $t+3$. After training and validation, the testing results are listed in Tables 2-4. According to Table 2, the BNN-MCD model outperforms other models in predictions of the hand's X positions. The range of MAE, MSE, and RMSE is from 9.7 to 13.2, 166.3 to 293.6, and 12.9 to 17.1 for BNN-MCD.

According to Table 3, LSTM-Bagging outperforms other models in predicting the Y position. The MAE of the LSTM-Bagging model increases from 15.9 to 18.5 for $t+1$ to $t+3$, and the MSE increases from 371.5 to 653.4. The LSTM-Bagging has better results in terms of RMSE as well.

TABLE 2: The ensemble prediction results of each model for hand position X.

Model	Time	MAE	MSE	RMSE
LSTM-Bagging	$t+1$	12.5	310.8	17.6
GRU-Bagging	$t+1$	11.1	274.3	16.6
BNN-MCD	$t+1$	9.7	166.3	12.9
LSTM-Bagging	$t+2$	13.7	385.8	19.6
GRU-Bagging	$t+2$	13.9	377.7	19.4
BNN-MCD	$t+2$	10.2	221.8	14.9
LSTM-Bagging	$t+3$	15.9	475.8	21.8
GRU-Bagging	$t+3$	17.2	487.2	22.1
BNN-MCD	$t+3$	13.2	293.6	17.1

TABLE 3: The ensemble prediction results of each model for hand position Y.

Model	Time	MAE	MSE	RMSE
LSTM-Bagging	$t+1$	15.9	371.5	19.3
GRU-Bagging	$t+1$	16.7	445.6	21.1
BNN-MCD	$t+1$	27.7	1266.7	35.6
LSTM-Bagging	$t+2$	18.6	557.1	23.6
GRU-Bagging	$t+2$	19.1	614.9	24.8
BNN-MCD	$t+2$	40.4	2697.3	51.9
LSTM-Bagging	$t+3$	18.5	653.4	25.6
GRU-Bagging	$t+3$	19.7	721.9	26.9
BNN-MCD	$t+3$	47.4	3606.0	60.1

According to the results in Table 4, the LSTM-Bagging outperforms the other models in forecasting the hand position for the Z coordinate. Specifically, the range of MAE, MSE, and RMSE values for the LSTM-Bagging model are between 9.9 and 13.3, 222.9 and 376.1, and 14.9 and 19.4, respectively.

TABLE 4: The ensemble prediction results of each model for hand position Z.

Model	Time	MAE	MSE	RMSE
LSTM-Bagging	t+1	9.9	222.9	14.9
GRU-Bagging	t+1	14.2	343.8	18.5
BNN-MCD	t+1	29.1	1402.1	37.4
LSTM-Bagging	t+2	10.4	254.3	15.9
GRU-Bagging	t+2	16.3	457.6	21.4
BNN-MCD	t+2	51.5	3154.6	56.2
LSTM-Bagging	t+3	13.3	376.1	19.4
GRU-Bagging	t+3	18.6	576.6	24.0
BNN-MCD	t+3	61.8	4368.3	66.1

Figures 7-9 show the prediction results for the X position by BNN-MCD, the Y position by GRU-Bagging, and the Z position by LSTM-Bagging. The x-axis shows the time in 16.67 milliseconds between each interval and the order of the six disassembly tasks described in Section 3.1. Figure 10 shows the prediction performance for hand position X by BNN-MCD and LSTM-Bagging for hand positions Y and Z over time.

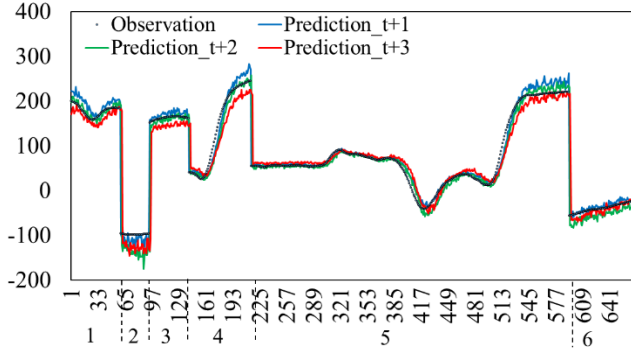


FIGURE 7: The testing results of BNN-MCD for the hand's X position.

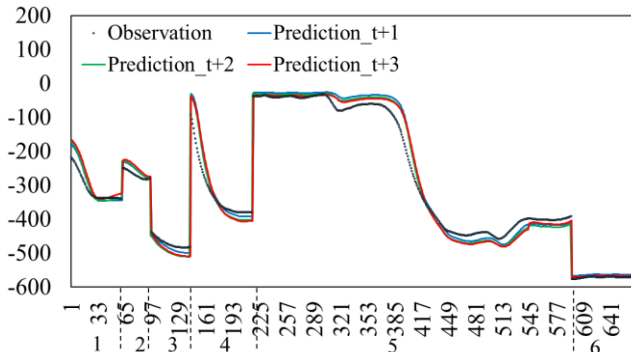


FIGURE 8: The testing results of GRU-Bagging for the hand's Y position.

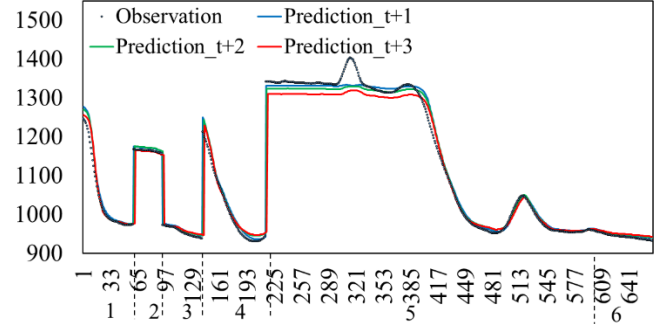


FIGURE 9: The testing results of LSTM-Bagging for the hand's Z position.

Although the prediction trend is similar to observations, there are still errors between the predicted values and observed values.

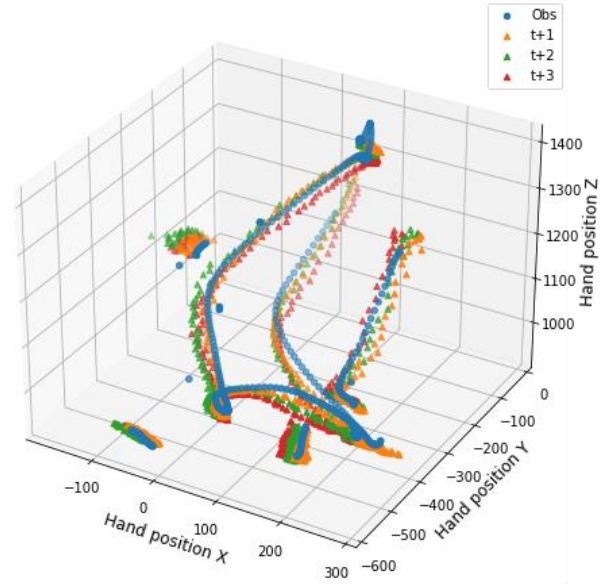


FIGURE 10: The prediction performance for hand position X by BNN-MCD, and LSTM-Bagging for hand positions Y and Z over time.

4.2 Prediction results of the potential range of motion for human hand

The upper and lower bounds of motion movement can be determined from the 30 prediction results by each model. Figures 11, 12, and 13 show the results for GRU-Bagging, LSTM-Bagging, and BNN-MCD, respectively. In Figures 11 and 12, the boundaries miss covering the hand position for some disassembly tasks, e.g. heat sink due to the errors between the predicted values and the observed values as the models overestimate or underestimate. However, the boundaries defined by BNN-MCD provide the reasonable movement area of hand position as shown in Figure 13. The MCD is a Gaussian process that randomly drops out neurons, while the BNN is a probabilistic model that treats

weights as random variables. The combination of these two models, both of which have uncertainty features, appears to be better suited for defining the possible movement area than fixed models like the GRU-Bagging and LSTM-Bagging.

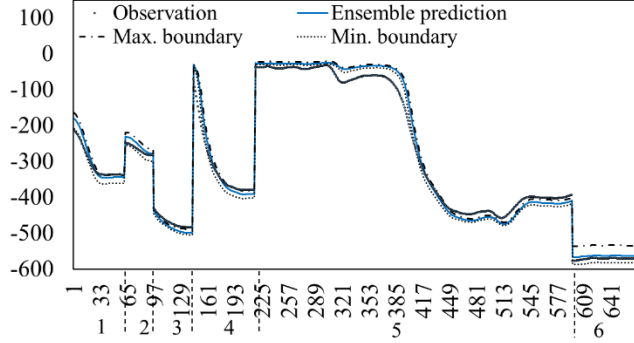


FIGURE 11: The testing results of GRU-Bagging on the possible movement area of the hand's Y position in time $t+1$.

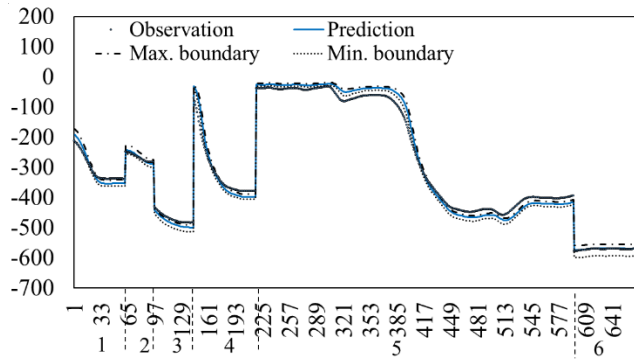


FIGURE 12: The testing results of LSTM-Bagging on the possible movement area of the hand's Y position in time $t+1$.

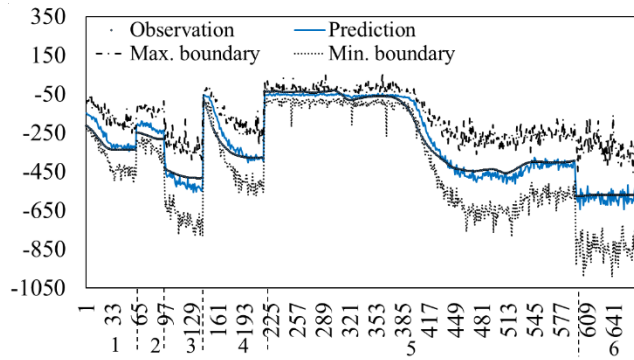


FIGURE 13: The testing results of BNN-MCD on the possible movement area of the hand's Y position in time $t+1$.

Table 5 presents partial testing samples of the BNN-MCD model, while Table 6 shows the number of observations that fall outside the boundary. Although some observations in Table 5 still do not fall within the boundary,

the BNN-MCD model performs better than LSTM-Bagging and GRU-Bagging, as it has fewer out-of-bounds observation points, as shown in Table 6. Specifically, for $t+1$, BNN-MCD has an error rate of only around 5% (31/669), while LSTM-Bagging and GRU-Bagging have error rates of 76% (506/669) and 66% (438/669), respectively. These results demonstrate that the BNN-MCD model is more effective in forecasting the possible areas of hand movements.

TABLE 5: The partial testing results of BNN-MCD for the first two prediction results on each task.

Task	Obs.	Pred.	Errors	Max. boundary	Min. boundary
1	-213	-147	66	-97	-211
1	-215	-150	65	-85	-237
2	-249	-212	37	-154	-281
2	-249	-201	48	-115	-295
3	-431	-432	1	-280	-596
3	-434	-409	25	-197	-633
4	-71	-51	20	-31	-79
4	-82	-56	26	-27	-87
5	-38	-51	13	23	-78
5	-38	-54	16	-27	-105
6	-578	-563	14	-264	-933
6	-578	-586	9	-315	-790

TABLE 6: The number of observation testing samples out of the upper and lower bounds for hand position Y (Total test samples: 669).

Time	LSTM-Bagging	GRU-Bagging	BNN-MCD
$t+1$	506	438	31
$t+2$	357	509	61
$t+3$	405	410	67

Figure 14 displays the PDF of the Gaussian distribution generated by BNN-MCD, which was run 30 times at each time point. The PDFs are normally distributed with mean and standard deviation calculated from 30 samples at each time. The width of the distributions is narrower in the range of approximately 200 to 400, indicating a smaller range of possible movement and lower uncertainty. In contrast, other ranges show a wider width of distributions, implying higher uncertainty and a larger range of possible movement.

It should be noted that since the PDF is drawn from simulation samples generated by BNN-MCD rather than real data, the actual observations may still fall outside the boundary, as shown in Table 6. This issue requires further discussion on how to improve forecasting accuracy in future research.

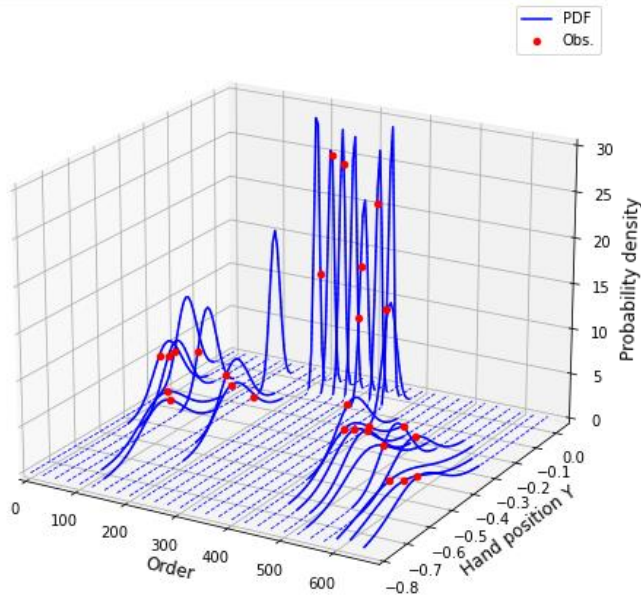


FIGURE 14: The Gaussian distribution of testing results for the hand position Y in time $t+1$, as plotted by every 25th order, using the BNN-MCD model.

5. CONCLUSION

This paper investigated the capability of three machine learning techniques including LSTM-Bagging, GRU-Bagging, and BNN-MCD for predicting the range of hand motion in the disassembly of consumer electronics. A case study of disassembling a desktop computer was used to show the application and IMU sensors were utilized to collect the required movement data. The Bagging and MCD procedures were performed 30 times, and the resulting ensemble prediction was calculated. The findings indicate that for forecasting hand position X, BNN-MCD outperforms LSTM-Bagging and GRU-Bagging. On the other hand, for hand positions Y and Z, LSTM-Bagging demonstrates better performance compared to the other models. The possible movement range is defined to enhance the safety of the human operator. In terms of defining the upper and lower bounds, BNN-MCD outperforms LSTM-Bagging and GRU-Bagging. The BNN model, which is a probabilistic model, is combined with MCD, a Gaussian process, to adjust the model's architecture to account for uncertainty.

The study can be extended in several ways. The current study analyzed each hand position separately to provide a detailed comparison of each model's performance. The results showed that BNN-MCD outperformed other models for position X, while LSTM-Bagging demonstrated superior performance for position Y. While comparing the three models separately provided different perspectives, it is computationally expensive. To decrease computation, it may be useful to consider all three positions together in future research, for instance, by inputting positions X, Y, and Z into each model and outputting the forecasting results for

all three positions. Also, currently, each disassembly operation is conducted once by a human operator, future work is needed to collect more samples across participants and across more complex disassembly tasks.

Moreover, the focus of data collection in this study was on the upper extremity and hand motion in disassembling tasks, however, the study can be extended to consider the whole-body motion. Besides IMU sensors, other sensors like RGB video images can be combined with IMU sensors to define different possible movement areas. Further, machine learning models can be combined with computer vision techniques to equip robots with more accurate scene monitoring techniques.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation–USA under grants #2026276 and #2026533. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] H. Liao, Y. Chen, B. Hu, and S. Behdad, "Optimization-Based Disassembly Sequence Planning Under Uncertainty for Human-Robot Collaboration." Jun. 27, 2022. doi: 10.1115/MSEC2022-85383.
- [2] S. Parsa and M. Saadat, "Human-robot collaboration disassembly planning for end-of-life product disassembly process," *Robot Comput Integr Manuf*, vol. 71, p. 102170, 2021.
- [3] K. Li, Q. Liu, W. Xu, J. Liu, Z. Zhou, and H. Feng, "Sequence planning considering human fatigue for human-robot collaboration in disassembly," *Procedia CIRP*, vol. 83, pp. 95–104, 2019.
- [4] Y. Y. Liao and K. Ryu, "Status Recognition Using Pre-Trained YOLOv5 for Sustainable Human-Robot Collaboration (HRC) System in Mold Assembly," *Sustainability*, vol. 13, no. 21, p. 12044, 2021.
- [5] T. Uzunovic, E. Golubovic, Z. Tucakovic, Y. Acikmese, and A. Sabanovic, "Task-based control and human activity recognition for human-robot collaboration," in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, 2018, pp. 5110–5115.
- [6] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, and J. Chen, "Deep learning-based human motion prediction considering context awareness for human-robot collaboration in manufacturing," *Procedia CIRP*, vol. 83, pp. 272–278, 2019.
- [7] Y. Kim, E. S. Choi, J. Seo, W. Choi, J. Lee, and K. Lee, "A novel approach to predicting human ingress motion using an artificial neural network," *J Biomech*, vol. 84, pp. 27–35, 2019.

- [8] J. Wang and L. Shen, "Semi-Adaptable Human Hand Motion Prediction Based on Neural Networks and Kalman Filter," in *Journal of Physics: Conference Series*, 2021, vol. 2029, no. 1, p. 12091.
- [9] Z. Ye, H. Wu, and J. Jia, "Human motion modeling with deep learning: A survey," *AI Open*, 2021.
- [10] Y. Wang, X. Ye, Y. Yang, and W. Zhang, "Collision-free trajectory planning in human-robot interaction through hand movement prediction from vision," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 305–310.
- [11] Q. Li, G. Chalvatzaki, J. Peters, and Y. Wang, "Directed acyclic graph neural network for human motion prediction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3197–3204.
- [12] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2891–2900.
- [13] D. Pavllo, C. Feichtenhofer, M. Auli, and D. Grangier, "Modeling Human Motion with Quaternion-Based Neural Networks," *Int J Comput Vis*, vol. 128, no. 4, pp. 855–872, 2020, doi: 10.1007/s11263-019-01245-6.
- [14] P. Zheng, P.-B. Wieber, J. Baber, and O. Aycard, "Human Arm Motion Prediction for Collision Avoidance in a Shared Workspace," *Sensors*, vol. 22, no. 18, p. 6951, 2022.
- [15] Y. Wang *et al.*, "Variable admittance control based on trajectory prediction of human hand motion for physical human-robot interaction," *Applied Sciences*, vol. 11, no. 12, p. 5651, 2021.
- [16] J. Zhang, H. Liu, Q. Chang, L. Wang, and R. X. Gao, "Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly," *CIRP annals*, vol. 69, no. 1, pp. 9–12, 2020.
- [17] L. Gril, P. Wedenig, C. Torkar, and U. Kleb, "A Tensor Based Regression Approach for Human Motion Prediction," *arXiv preprint arXiv:2202.03179*, 2022.
- [18] H. Liao, M. Zheng, B. Hu, and S. Behdad, "Human Hand Motion Prediction in Disassembly Operations," in *Proceedings of the ASME 2022 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2022.
- [19] L. Burks, H. M. Ray, J. McGinley, S. Vunnam, and N. Ahmed, "Semantic Sensing and Planning for Human-Robot Collaboration in Uncertain Environments," *arXiv preprint arXiv:2110.10324*, 2021.
- [20] S. Sajedi, W. Liu, K. Eltoumy, S. Behdad, M. Zheng, and X. Liang, "Uncertainty-assisted image-processing for human-robot close collaboration," *IEEE Robot Autom Lett*, vol. 7, no. 2, pp. 4236–4243, 2022.
- [21] A. Furnari, S. Battiato, and G. Maria Farinella, "Leveraging uncertainty to rethink loss functions and evaluation measures for egocentric action anticipation," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, p. 0.
- [22] Y. Abu Farha and J. Gall, "Uncertainty-aware anticipation of activities," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, p. 0.
- [23] A. Casalino, E. Mazzocca, M. G. Di Giorgio, A. M. Zanchettin, and P. Rocco, "Task scheduling for human-robot collaboration with uncertain duration of tasks: a fuzzy approach," in *2019 7th International Conference on Control, Mechatronics and Automation (ICCM)*, 2019, pp. 90–97.
- [24] V. Forti, C. P. Baldé, R. Kuehr, and G. Bel, "The Global E-waste Monitor 2020," *United Nations University (UNU), International Telecommunication Union (ITU) & International Solid Waste Association (ISWA), Bonn/Geneva/Rotterdam*, vol. 120, 2020.
- [25] R. Zuidwijk and H. Krikke, "Strategic response to EEE returns:: Product eco-design or new recovery processes?," *Eur J Oper Res*, vol. 191, no. 3, pp. 1206–1222, 2008.
- [26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [27] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Adv Neural Inf Process Syst*, vol. 32, 2019.
- [28] L. Breiman, "Bagging predictors," *Mach Learn*, vol. 24, no. 2, pp. 123–140, 1996.
- [29] A. Husejinovic, "Credit card fraud detection using naive Bayesian and c4. 5 decision tree classifiers," *Husejinovic, A.(2020). Credit card fraud detection using naive Bayesian and C*, vol. 4, pp. 1–5, 2020.
- [30] H. Y. Hsiao and K. N. Chiang, "AI-assisted reliability life prediction model for wafer-level packaging using the random forest method," *Journal of Mechanics*, vol. 37, pp. 28–36, 2021.
- [31] P. Huang, Y. Li, X. Lv, W. Chen, and S. Liu, "Recognition of common non-normal walking actions based on Relief-F feature selection and relief-bagging-SVM," *Sensors*, vol. 20, no. 5, p. 1447, 2020.
- [32] P. Chen *et al.*, "Multi-view real-time human motion recognition based on ensemble learning," *IEEE Sens J*, vol. 21, no. 18, pp. 20335–20347, 2021.
- [33] P. Chen *et al.*, "Through-wall human motion recognition based on transfer learning and ensemble

- learning,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.
- [34] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” *arXiv preprint arXiv:1409.1259*, 2014.
- [35] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM and other neural network architectures,” *Neural networks*, vol. 18, no. 5–6, pp. 602–610, 2005.
- [36] W. Cheng, J. Li, H.-C. Xiao, and L. Ji, “Combination predicting model of traffic congestion index in weekdays based on LightGBM-GRU,” *Sci Rep*, vol. 12, no. 1, p. 2912, 2022, doi: 10.1038/s41598-022-06975-1.
- [37] A. Ghalamzan-E, “Learning needle insertion from sample task executions,” *arXiv preprint arXiv:2103.07938*, 2021.
- [38] G. Petneházi, “Recurrent neural networks for time series forecasting,” *arXiv preprint arXiv:1901.00069*, 2019.
- [39] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, “Hands-on Bayesian neural networks—A tutorial for deep learning users,” *IEEE Comput Intell Mag*, vol. 17, no. 2, pp. 29–48, 2022.
- [40] X. Lin, H.-L. Zhen, Z. Li, Q. Zhang, and S. Kwong, *A Batched Scalable Multi-Objective Bayesian Optimization Algorithm*. 2018.
- [41] S. Maiti and R. K. Tiwari, “Neural network modeling and an uncertainty analysis in Bayesian framework: a case study from the KTB borehole site,” *J Geophys Res Solid Earth*, vol. 115, no. B10, 2010.
- [42] S. Lee, H. Kim, and J. Lee, “Graddiv: Adversarial robustness of randomized neural networks via gradient diversity regularization,” *IEEE Trans Pattern Anal Mach Intell*, 2022.
- [43] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” in *international conference on machine learning*, 2016, pp. 1050–1059.
- [44] K. Wang, H. Du, R. Jia, and H. Jia, “Performance Comparison of Bayesian Deep Learning Model and Traditional Bayesian Neural Network in Short-Term PV Interval Prediction,” *Sustainability*, vol. 14, no. 19, 2022, doi: 10.3390/su141912683.
- [45] Y. Wen, M. F. Rahman, H. Xu, and T.-L. B. Tseng, “Recent advances and trends of predictive maintenance from data-driven machine prognostics perspective,” *Measurement*, vol. 187, p. 110276, 2022.
- [46] S. Kulshrestha and S. Patel, “An efficient host overload detection algorithm for cloud data center based on exponential weighted moving average,” *International Journal of Communication Systems*, vol. 34, no. 4, p. e4708, 2021.