Unsupervised Human Activity Recognition Learning for Disassembly Tasks

Xinyao Zhang, Daiyao Yi, *Student Member, IEEE*, Sara Behdad and Shreya Saxena, *Member, IEEE*

***Abstract*— Large volumes of used electronics are often collected in remanufacturing plants, which requires disassembly before harvesting parts for reuse. Disassembly is mainly conducted manually with low productivity. Recently, human-robot collaboration is considered as a solution. For robots to assist effectively, they should observe work environments and recognize human actions accurately. Rich activity video recording and supervised learning can be used to extract insights; however, supervised learning does not allow robots to self-accomplish the learning process. This study proposes an unsupervised learning framework for achieving video-based human activity recognition. The framework consists of two main elements: a variational autoencoder-based architecture for unlabeled data representation learning, and a hidden Markov model for activity state division. The complete explicit activity classification is validated against ground truth labels; here, we use a case study of disassembling a hard disk drive. The framework shows an average recognition accuracy of 91.52%, higher than competing methods.**

***Index Terms*— Human activity recognition (HAR), unsupervised learning, disassembly tasks, variational autoencoder (VAE), hidden Markov model (HMM).**

I. Introduction

Robots are becoming an inevitable element of the intelligent manufacturing industry, where they team up with humans to implement various tasks. In order to achieve a safe and efficient collaborative environment, human activity recognition (HAR) has gradually gained a lot of attention. The recognition of human activity is intended to allow robots to proactively provide assistance. To accomplish this, robots need to understand human behaviors and states based on the observed information during operational processes.

Manuscript received June 27, 2022; revised October 17, 2022 and December 30, 2022; accepted March 24, 2023. This work was supported by the National Science Foundation–USA under grants 2219876 and 2026276. 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. Paper no. TII-22-2763. Xinyao Zhang and Daiyao Yi contributed equally to this work and are considered co-first authors. (Corresponding authors: Sara Behdad; Shreya Saxena.)

Xinyao Zhang and Sara Behdad are with the Environmental Engineering Sciences Department, University of Florida, Gainesville, FL 32611 USA (e-mail: xinyaozhang@ufl.edu; sarabehdad@ufl.edu).

Daiyao Yi and Shreya Saxena are with the Electrical and Computer Engineering Department, University of Florida, Gainesville, FL 32611 USA (e-mail: yidaiyao@ufl.edu; shreya.saxena@ufl.edu).

A. Research Motivation

To enhance environment observation, the use of sensors or vision systems has become popular for data acquisition while humans are engaged in a task. However, wearable sensors pose several concerns, such as the need for continuous intrusive monitoring [1]. Smartphone sensors have proved to serve as flexible candidates for data acquisition, but the recorded data typically contains noise and requires more discriminatory representation [2], [3]. Due to their non-intrusive and accessible nature, vision-based systems have been widely developed for HAR. A robot manipulator is typically equipped with a camera to collect human motion data and further learn tracking tasks [4]. Videos from manufacturing settings contain spatio-temporal information about human motion, which can help identify human actions [5]. Consequently, a video recording synchronized with human operation can capture rich information about the surrounding environment, including the behavior of humans interacting with products or tools.

Although video data contains a wealth of information, extracting human behaviors in an actionable way remains challenging. Allowing robots to understand human behaviors requires obtaining features from video data [6]. The solutions are mainly focused on the application of neural networks. For example, by adjusting the weights of the neural network, a growing self-organizing map was achieved to represent human activities. [7]. Similarly, convolutional neural networks (CNN) were applied to learn discriminative features in daily activities [8]. Accurate feature extraction from video data is a prerequisite for representing human activities. Since people behave very differently in work environments as compared to their ordinary daily activities, this poses difficulties for human activity learning in manufacturing scenarios. In the disassembly workspace, human workers' actions are related to the active workflow. Therefore, it is impractical to identify only independent action data without considering the correlation between actions. Although supervised-learning based recognition works well in some environments, it cannot properly deal with action uncertainties in settings such as remanufacturing plants where they receive products with different conditions, models, and quality. Due to the high degree of variability in used products and in activity performance by different subjects, disassembly operations can hardly be defined as fixed patterns.

Given the large volume of used consumer electronics ready for recycling and reuse, efficient disassembly has become a necessity. In order to achieve an efficient robotic-assisted

TABLE I: Comparison of Different HAR Approaches for Video Data

	Mechanism		Model		Method			Task	
Ref	Supervised learning	Unsupervised learning	With explicit feature extraction	Without explicit feature extraction	CNN	RNN	HMM	Assembly	Disassembly
[9], [10], [11]	×			×	×			×	
[12], [13]	×		×		×			×	
[14]	×			×			×	×	
[15]	×			×		×			×
Our framework		×	×		×	×	×		×

disassembly for end-of-use electronics, two aspects are investigated in this study. First, unsupervised learning of human tasks for robots to be familiar with the disassembly process. Second, addressing the impact of the unknown state of electronics on the disassembly process.

B. Main Contributions

To address the difficulty of recognizing human activity in the remanufacturing domain, this paper proposes a framework for unsupervised learning of action features directly from videos for action recognition. Notably, we introduce a novel model for unsupervised feature extraction from videos that combines the ability of CNNs to parse spatial image data and RNNs to handle temporal data, here termed a sequential variational autoencoder (Seq-VAE).

The research contributions are reflected in the following.

1) The study proposes a novel unsupervised learning framework for vision-based end-to-end human activity recognition. In contrast to supervised HAR approaches, the proposed framework significantly reduces the manual annotation effort as well as importantly shifts vision-based HAR from manual-intensive to model-automated. Moreover, identifying discrete disassembly activities is not sufficient, so we use video streaming to explore the sequential relationships between disassembly activities. Also, since the products to be disassembled have been in use for a long time, we also discuss the impact of the uncertainty of the state of the disassembled products on the disassembly process.

2) In the proposed deep framework, we design state-of-the-art algorithms integrating a VAE-based feature extraction model, a hidden Markov model (HMM) for continuous action division, and a nonlinear support vector machine (SVM) kernel for recognition validation. The complete workflow achieves a balance between requiring fewer resources and effective activity recognition learning.

3) To approximate uncertainties of end-of-use electronics received at remanufacturing sites, we collect data sets by designing multiple disassembly tasks. The experimental results demonstrate the performance of the entire framework by evaluating it on real datasets. Moreover, we individually evaluate the modules for visual feature representation and continuous activity segmentation.

The rest of the paper is structured as follows. Section II compares our work to related HAR-themed research, from top-level learning mechanisms to bottom-level implementation approaches. Section III describes the latent feature representation learning architecture, followed by human activity recognition learning. Section IV introduces the data set designed for the experiments and describes the experimental results. This section also discusses the stage results after each module and the comparison with other unsupervised HAR methods. Section V concludes the paper and extends to potential future work.

II. Related Work

Table I compares different HAR approaches that have been used for processing video data dedicated to assembly and disassembly tasks. The criteria chosen are based on the learning mechanisms at the top level, the model categories at the middle level, and the implementation methods at the bottom level.

As summarized in Table I, previous studies have used different state-of-the-art supervised methods for processing the video data, including CNN, recurrent neural network (RNN) and HMM. Several studies have applied CNN streams for extracting spatial features from human motions [9]–[13]. Besides CNN, aiming to capture action-based temporal information, previous researchers used RNN sequence model to predict incomplete movements [15], [16]. Furthermore, the modeling of hierarchical relationships between classification results was discussed in [14]. In terms of the learning mechanism, most of the earlier studies have selected supervised learning [17], [18]. Although supervised learning has been successful at handling tasks in which the labels can conveniently be designated by humans, such as static classification, the models' performance may degrade while encountering recognition of time series data. Moreover in robotic applications, it is difficult for robots to self-supervise complex tasks [19]. To address this challenge, learning video demonstrations in an unsupervised manner directly from observations can be a solution. Unsupervised

TABLE II: Comparison of Supervised and Unsupervised Learning

	Supervised learning	Unsupervised learning
Use-case	Static classification, More human interference	Dynamic recognition, Less human interference

TABLE III: Comparison of Multiple Architectures

	CNN	RNN	HMM
Use-case	Spatial-based activities	Temporal-based activities	Segmented sequential activities

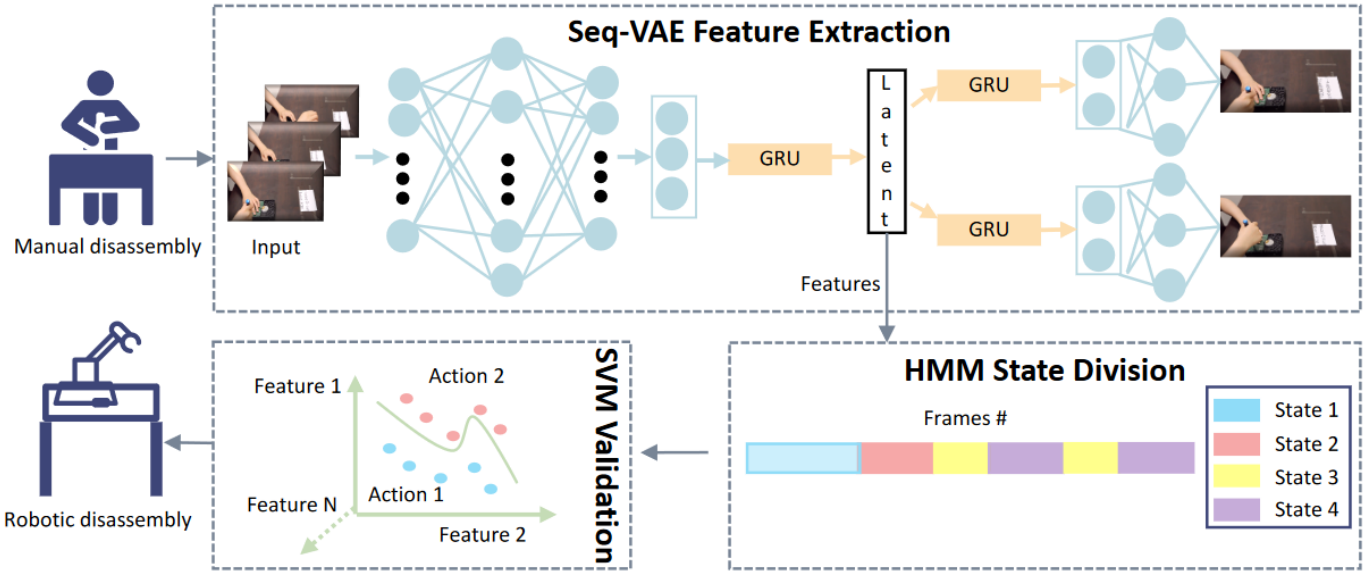

Fig. 1: Workflow of the proposed unsupervised HAR framework.

learning is more accessible and scalable for robot learning and dynamic recognition. Furthermore, unsupervised learning provides a label- free learning pathway compared to computationally expensive supervised learning models, as summarized in Table II.

HAR models can be separated into two types: with explicit feature extraction [20] and without explicit feature extraction [21]. A HAR model works on the principle of starting with feature extraction and ending with action recognition. The result of feature selection determines the performance of the final action recognition. Features obtained from video images encode spatio-temporal information about human activities or trajectories. Explicit feature extraction modules reveal hidden or less obvious features that are used to distinguish actions in the original videos.

As highlighted in Table I, the HAR state of the art includes different implementation models ranging from deep learning to Markov models. Unlike the conventional machine learning methods that extract shallow features, deep learning-based HAR models are robust to image variations. A popular architecture in deep learning, CNNs have been shown to have the capability to classify features related to subjects' actions [5], [22], [23]. In particular, human actions unfold sequentially, with people completing assignments in manufacturing scenarios based on a time-series workflow. At this point, the superiority of RNN, which is more suitable for analyzing time series data, is clearly demonstrated. RNN-based models can extract temporal features more efficiently than CNN [20]. As the manufacturing activities are connected, we need models such as RNNs that can process sequence dependencies over a long-range. In addition to deep learning-based feature learning methods, an HMM has been applied to perform action recognition. HMMs is a successful model for segmentation of video data encapsulating behavior [23]. The HMM model has been evaluated for its ability to divide continuous human behaviors [24]. Table III lists the comparison for different architectures.

Note that recognizing human activity when performing a disassembly task is a dynamic process, which cannot be estimated in advance as in supervised learning. As an alternative, unsupervised learning with explicit feature representation can learn features automatically and also allows the robot to easily observe human actions. We take advantage of different architectures - CNN for spatial feature learning, RNN for sequential feature learning, and HMM for active state segmentation - by integrating all of them into a single unsupervised learning framework.

III. Methodology

In this section, we first provide an overview of the proposed framework structure. Then, we describe the Seq-VAE model built to extract features from unlabeled video data, and the HMM model used to distinguish actions from the motifs.

A. Framework Structure

The detailed framework is illustrated in Fig. 1. The unsupervised HAR workflow consists of three main modules. First, after receiving unlabeled videos of consecutive actions taken by operators while completing a specific task, the Seq-VAE architecture extracts deeply embedded spatio-temporal features from the data and represents them in latent space. Second, considering the dependencies between consecutive actions, an HMM uses the low-dimensional latent representation features to automatically delineate the states associated with the active process. Third, based on the results of state segmentation, the SVM classifier is trained to validate feature-activity matching for clear human activity recognition. The choice of the classification algorithm should not largely affect the performance in recognizing the actions; we chose SVM due to its simplicity and accuracy in most of the classification tasks. The end-to-end human activity recognition learning framework outputs unlabeled human activity information to facilitate robot operations.

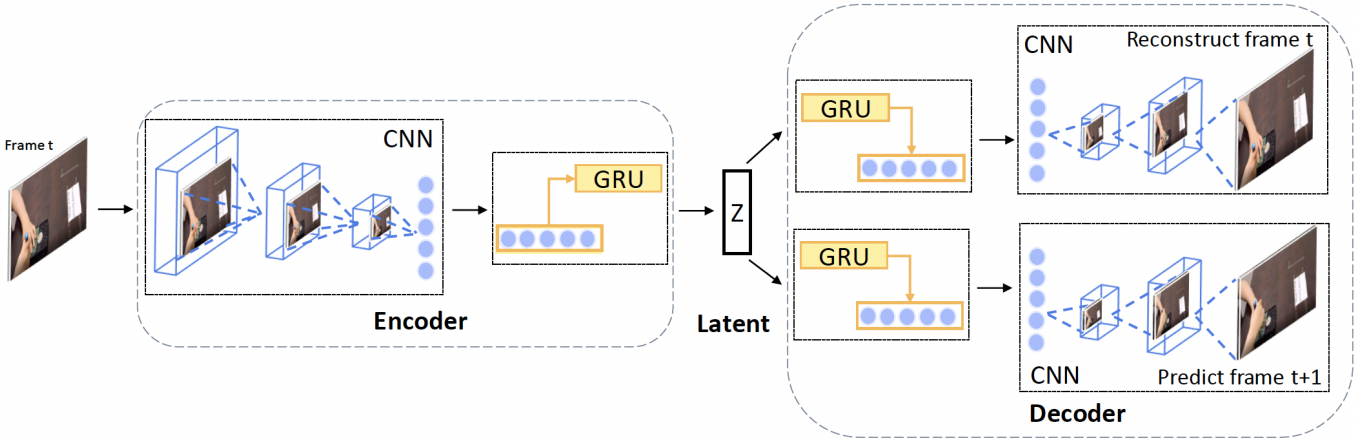

Fig. 2: Structure of Seq-VAE: the frames are fed into the CNN-GRU-based encoder model for feature extraction and reconstructed in the CNN-GRU-based decoder model to force the encoder to learn richer representations.

B. Sequential Variatonal Autoencoder (Seq-VAE)

Here we developed a model that can directly extract useful action motifs from sequential video data (Fig. 2). The work is inspired by [25], in which the authors aimed to reconstruct sequential pose estimation data at the current time point while also reconstructing n step ahead future data. In our case, instead of using poses as the input data, we directly use video frames as the input and predict the frames one step ahead. We provide the mathematical details of the model below.

First, we define the input frame sequence at time t as $X_t = \{x_{t-W}, x_{t-W+1}, ..., x_t\}$, and the corresponding one step ahead frame sequence as X_{t+1}. The corresponding reconstructed variables are denoted as $\hat{X}_t$ and $\hat{X_{t+1}}$, respectively. The latent space variable at time t is denoted as z_t.

Our model aims to learn a d dimensional latent z_t which captures both the spatial and temporal information about the input frames x_t and x_{t+1}. The mapping from images to latent is achieved by the following steps. First, the current input frames x_t go through a series of convolutional layers to capture essential features from the frames. Then, a forward GRU layer is implemented to capture the temporal information. The latent is regularized using a KL divergence as detailed in the following paragraph, and then is passed through two decoders in parallel. Both of these decoders contain a GRU layer followed by a series of CNN layers. This process results in the reconstruction of the current input frame and predictions of the future input frames. As a result, the latent z_t at time t captures both the past, current and future information about the action.

We denote the encoding process described above as f_e while the decoding process as $f_{d\_current}$ and f_{d_future}; therefore, the latent z_t is expressed as:

$$z_t = f_e(X_t) \tag{1}$$

The decoder output for current reconstruction:

$$\hat{X}_t = f_{d\_current}(z_t) \tag{2}$$

The decoder output for future prediction:

$$\hat{X}_{t+1} = f_{d\_future}(z_t) \tag{3}$$

In a VAE model, the latent variables $p(z|X)$ are regularized through the KL divergence. The goal is to minimize the distance between the unknown distribution $p(z|X)$ and a prior normal distribution $p(z)$.

To deal with the unknown distribution $p(z|X)$, the KL divergence can be calculated through the Evidence Lower Bound (ELBO) [26]:

$$\mathscr{L}_{ELBO} = \mathbb{E}_{q(z|X)}[\log(p(X|z)] - KL[q(z|X)||p(z)] \tag{4}$$

The first term in the loss above is defined as the loss over frames, $\mathscr{L}_{current\_frames}$. Since the latent z_t can predict the future input X_{t+1}, we add another loss term in the above equation 4 to calculate the loss over the anticipated frames $\mathscr{L}_{future\_frames}$. The loss function can be expressed as:

$$\begin{aligned}\mathscr{L}_{ELBO} &= \mathscr{L}_{current\_frames} + \mathscr{L}_{future\_frames} + \mathscr{L}_{KL} \\ &= \mathbb{E}_{q(z_t|X_t)}[\log(p(X_t|z_t)] + \mathbb{E}_{q(z_t|X_t)}[\log(p(X_{t+1}|z_t)] \\ &\quad - KL[q(z_t|x_t)||p(z_t)]\end{aligned} \tag{5}$$

We minimize this loss function to obtain model parameters and latent variables that we treat as features in downstream steps.

We use principal component analysis (PCA) and autoencoders (AEs) as comparisons to the Seq-VAE. In an AE model, there is no regularization on the latent variables; the loss is directly the reconstruction error of the frames, which is calculated as the following, where N is the total number of frames.

$$\mathscr{L}_{MSE} = \sum_{t}^{N}(x_t - \hat{x}_t)^2/N \tag{6}$$

C. Human Action Recognition

Here, we introduce a model to identify the motifs in the data through an HMM. Our HMM consists of a set of discrete hidden states $S_t \in \{1,2,..K\}$ and observation sequences $\{Z_1, Z_2, .., Z_M\}$ where $Z_m = \{z_{m_1}, z_{m_2}, ..z_{m_T}\}$, $z_{m_t} \in \mathbb{R}^d$. In our model, z are the latent variables as identified by the Seq-VAE, $m \in \{1,2,..,M\}$ is the trial number, m_t represents the t^{th} time

step in trial m, M is the total number of trials, K is the number of hidden states, and d is the observation dimension.

In the HMM, there are three sets of parameters. The state transition matrix $A \in \mathbb{R}^{N\times N}$ where each entries $A_{i,j}$ represents the probability to switch from state i to state j:

$$A_{i,j} = P(S_{t+1} = j | S_t = i) \tag{7}$$

The emission matrix $B \in \mathbb{R}^{N\times S}$ where $B_{i,j}$ represents the probability that the observation is i given that we are in state j:

$$B_{i,j} = P(z_t = i | S_t = j) \tag{8}$$

The probability distribution $P \in \mathbb{R}^T$ where $P_t(S_k)$ represents the probability of being in state S_k at time t. We use the Expectation-Maximization (EM) algorithm to fit the HMM parameters and obtain the hidden states from the given observations; mathematical details can be found in [27].

We adopted the open source software from Linderman et al. [28] for implementing the algorithm. The HMM fits with the observations as the latent variables from the Seq-VAE. After obtaining the HMM parameters using the training set, we feed the model with the new testing set and generate the hidden states.

The HMM typically results in sub-actions performed by the human, at a temporal scale that may be shorter than human-annotated actions. Thus, sub-actions obtained using HMM may be combined to produce a manually-defined dissembling action. We trained a classifier to validate whether the hidden state generated by the HMM is consistent with the ground-truth (human-annotated) actions. The process is described as follows. We first annotate the actions at each time step as $a_t \in \{1,2,...,J\}$, with J being the total number of actions. We then train an SVM model to output ground truth action labels a_t given the HMM states as inputs. Specifically, as inputs at time t, we augment the current HMM state S_t with the length of the current state, as well as the last two HMM states visited in the past and the length of these states. Given this 6-dimensional vector as input, we classify the current action label a_t, and note the accuracy using our framework as compared to other methods. Thus, we validate the HMM states using the ground truth labels.

IV. Use-Case and Experiments

This section introduces a case study of end-of-use hard disk drive (HDD) disassembly. We first demonstrate that applying the Seq-VAE for feature extraction separates the human actions through time. In addition, we compare our model with two baselines- AutoEncoder (AE) and PCA. We apply the latent to an HMM for motif identification and human action recognition. Finally, we validate our framework with an SVM for comparison to ground truth labels.

A. Task Design and Data Collection

The experiment is designed as follows. Only one user is in charge of the disassembly workstation. An end-of-use open-case HDD is used as the experiment's target, which is assumed to be of good internal condition. Moreover, we

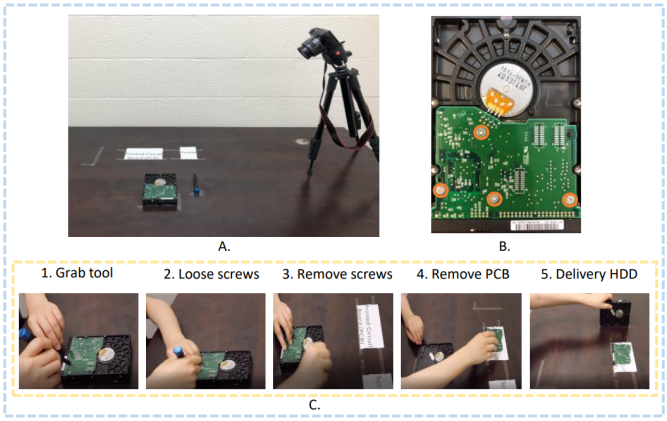

Fig. 3: Experimental design: A. Experimental workstation. B. Disassembly target. C. Disassembly action arrangement.

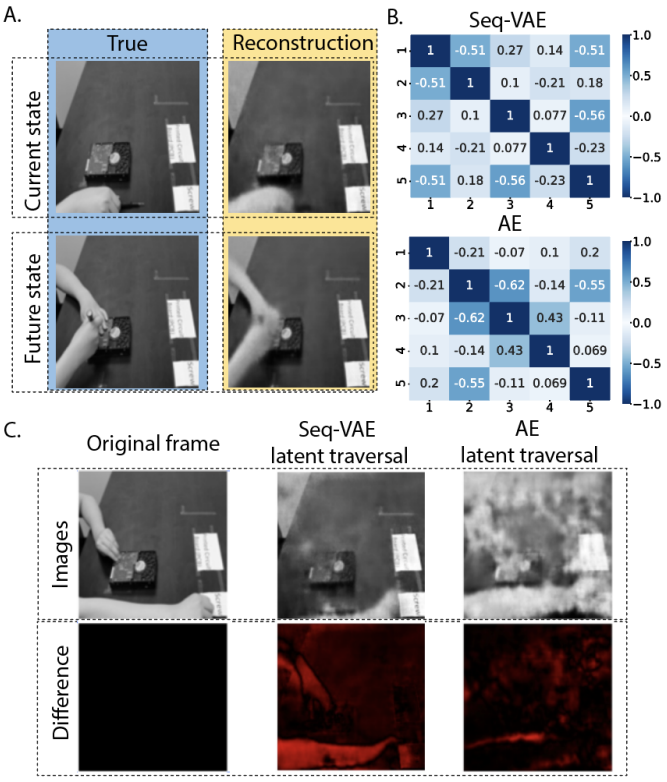

Fig. 4: A. The model performance on reconstructing the current state image and predicting the future state image. B. The correlation heatmap for latents generated by Seq-VAE and AE. C. The latent traversals for the Seq-VAE and AE.

represent uncertainties of an end-of-use device by designing three different cases, including an HDD with 2 screws, 3 screws, and 4 screws, illustrated in Fig. 3 A, B. The number of screws affects the overall disassembly planning as thousands of electronics are sent to remanufacturing sites every day. Small differences in the used electronics can also make a difference to the collaboration between human operators and robots, as effective collaboration requires small or no gaps among different disassembly steps. In order to eliminate bias in the movements, the user was tasked to perform disassembly by a

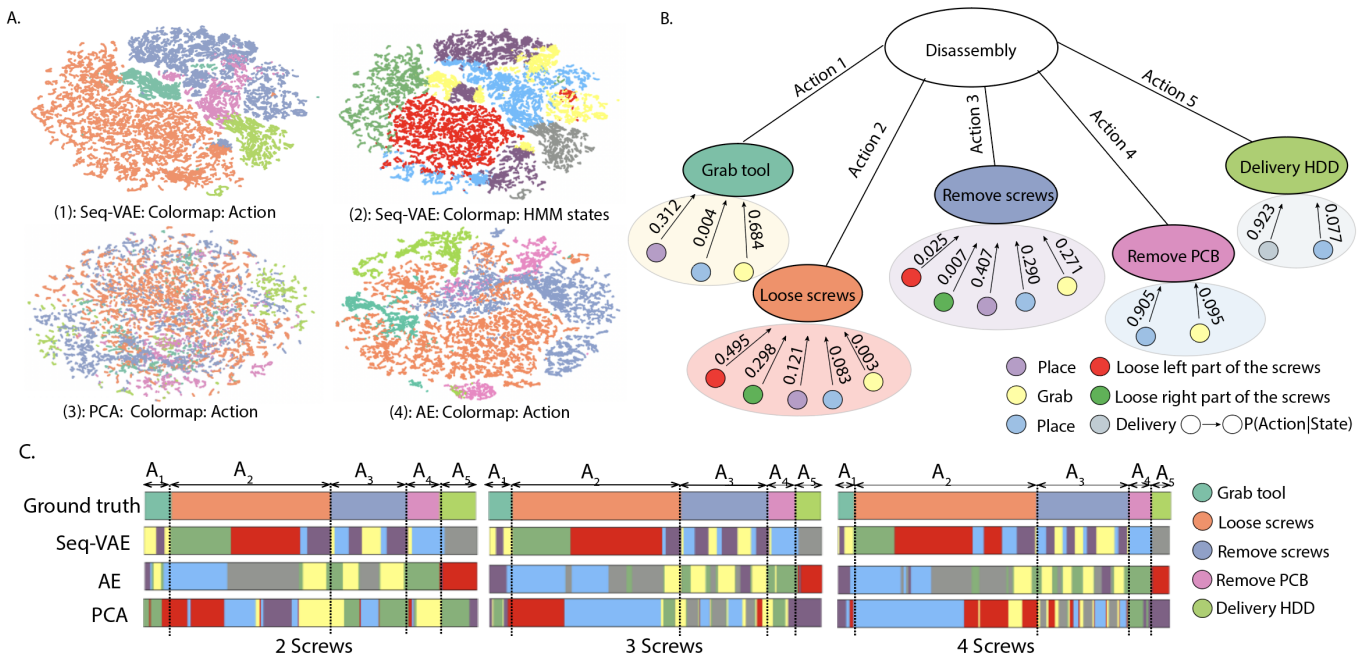

Fig. 5: t-SNE plots showing the relationship between the manually annotated actions and the recovered HMM states. A. t-SNE for the latent generated by different models with different colormaps. (1,3,4): Different colors represent different actions, labeled using manual annotation (ground truth). (2): Different colors represent different HMM states, where HMM was applied on the Seq-VAE latent variables. B. Action tree that reveals the relation between the manually annotated actions and the recovered HMM states. The color of the actions nodes correspond to the color in part A (1,3,4). The color of the HMM states correspond to the color in part A (2). The purple placement action represents the placement tool, while the blue action represents the placement component. C. The HMM results and the ground truth comparison for the three feature extraction models. Only in the Seq-VAE model can the state transitions successfully reflect the changes in activity.

specific sequence, including actions of grabbing a screwdriver, loosing screws, removing screws, removing printed-circuit-board (PCB), and finally delivering HDD. Each case scenario is repeated 15 times to collect sufficient data, as illustrated in Fig. 3 C.

For data pre-processing, the raw video is split into frames that were recorded by a digital camera at 29.97 frames per second (fps). These frames were converted from RGB with a resolution of 1920×1080 to gray scale images of 128×128 pixels. The inputs to the model are two series of pre-processed images: one defined as the current input and the other as a future input, one frame later than the current input. They both have a window size of W. Note that W is an important hyperparameter and needs to be carefully defined, as detailed in Sec. IV-B (2).

In order to match the divided action states to human interpretable activity categories, we used a labeling strategy with less human interference. Specifically, since the transitions of the hidden states of the HMM represent changes in motion, we referred to the starting and ending positions of the states in the HMM to reduce the labeling workload.

B. Model Setup

1) Model Training: The feature extraction model was developed using TensorFlow and Keras. We applied a symmetric image encoder and decoder, each with 14 convolution layers. We applied the Adam optimizer with the learning rate fixed to be 10^{-3}, batch size to be 64 and trained for 100 epochs on a single Nvidia 3080 GPU. The experiment was carried out while splitting the trials into training and testing: 80% for training and 20% for testing.

2) Hyperparameters Selection: Three coefficients need to be determined in our model as indicated above: $\{W, d, K\}$. In our case, W is chosen to be 20 which well captures the action dynamics through time as compared to smaller quantities; moreover, it is relatively computationally efficient as compared to the larger quantities. d is determined by visualizing the correlation matrix for each latent; one can get useful latents when the correlation between the individual latents is lower than 0.5. We showed that d equals to 5. K can be chosen by computing the log-likelihood as a function of number of states. Too few states do not allow for a multi-scale representation of behavior, and more states do not yield better models when comparing log-likelihood functions [23], [25]. Here, we show results for K=6; results are similar for K=5 and K=7. The combination of each lower-level action forms the desired action.

C. Behavior Reconstruction

We performed the experiment 5 times with different training and testing sets each time by randomly shuffling the trials. The Mean Square Error (MSE) for each individual pixel given by the Seq-VAE model is as follows. For the training set, the error is $7.90 \pm 0.23 \cdot 10^{-9}$ for the current state reconstruction while

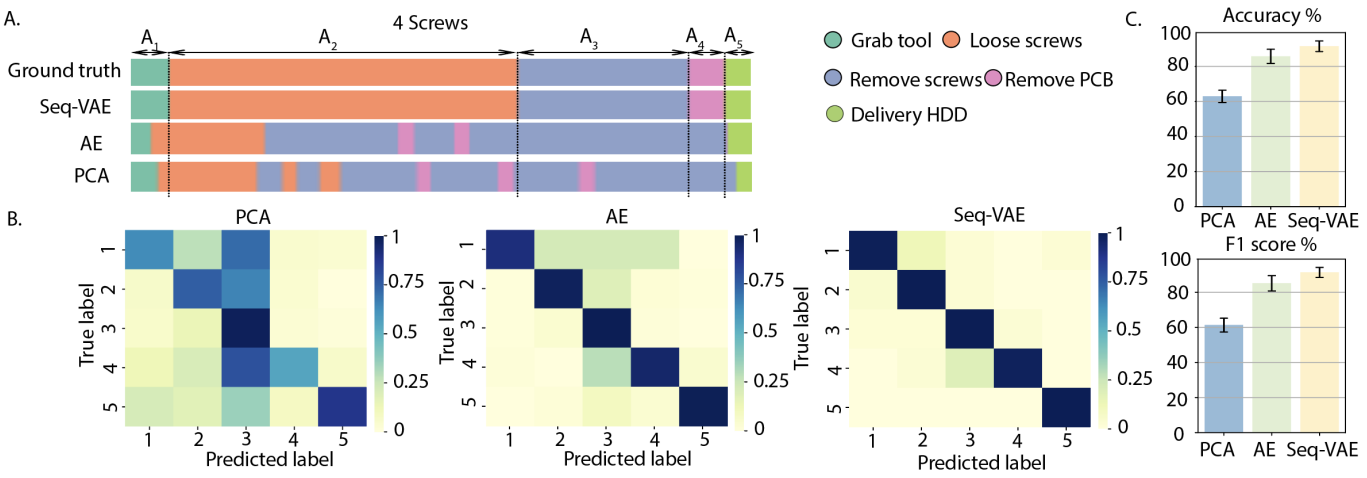

Fig. 6: SVM classification results on the latent generated by different models. A. Ethograms of an example trial given by SVM classification based on different latent variable models. B. Confusion matrix for the SVM classification accuracy and a comparison between different models. C. The SVM classification accuracy and F1 score comparison.

$7.86 \pm 0.13 \cdot 10^{-9}$ for the future state prediction. For the test set, the values are $11.26 \pm 0.69 \cdot 10^{-9}$ and $11.46 \pm 0.68 \cdot 10^{-9}$, respectively. The reconstruction images are shown in Fig. 4A. We compared the current reconstruction error with the error generated by using the autoencoder. The MSE error per pixel given by AE is $1.42 \pm 0.65 \cdot 10^{-10}$ on the test set. Our model produces comparable MSE results.

D. Latent Space Representation

We evaluated the independence of each latent obtained from Seq-VAE by showing the correlation matrix of each latent. From the Fig. 4B, the latent variables are largely uncorrelated with each other after training the Seq-VAE, similarly to the AE model, which points to a well-regularized solution in both cases. Next, we examined the interpretability of the latent variables. In our experimental setting, the movements of the human, here the arms, are the most important part of the video. We performed latent traversals on our latent variables to identify what each latent variable corresponds to in the video, and if any of the latents successfully captures the arms' movement. The term 'latent traversal' refers to the study of how the image varies with a change in latent [22]. Briefly, a base image is chosen by randomly picking one frame from the videos, and the goal is to visualize and quantify the effect of changing a specific latent at a time. Practically, we changed the value of one latent to achieve its maximum value across all frames, and this new set of latents form the input to the decoder. We obtained the corresponding output from the decoder as the 'latent traversal' image. Finally, we visualized the difference between the 'latent traversal' image and the base image, here denoted in red in the Fig. 4C. We showed the latent traversal for one of the latents in Seq-VAE and the closest corresponding latent in the AE model; the Seq-VAE latent successfully captures the movement of the arms while the AE latent encodes the image in a distributed way.

Finally, for visualization of the latent space, we performed t-SNE on the latent variables and plot these in a two-dimensional space (Fig. 5A). Here, each dot corresponds to one frame in the trial. The frames manually labeled as a specific ground truth action are clustered together in the Seq-VAE latent space, and show adequate separation across different actions. The same is not as clear in the AE or PCA latent space, where much more overlap between different actions.

E. Action Clustering

To automatically identify different discrete actions, we used a hidden Markov model on the Seq-VAE latent variables. The resulting discrete states are shown in Fig. 6A as an ethogram, along with a comparison to ground truth actions as well as discrete states identified by applying HMMs to AE and PCA latents. Here, one can easily observe that the combinations of different Seq-VAE states can be recognized as various actions. While the ground truth labels encompass long actions such as 'grab tool', these actions can be further broken down into 'pick up tool' and 'place tool'. We see in the ethogram that the Seq-VAE states consistently subdivide the ground truth actions into different sub-actions. In Figure 5B, we show a hierarchical tree showing the breakdown of ground truth actions into the different sub-actions. The hierarchical mapping maintains consistency between the different levels. The different colors in the bottom level represent different states, which are automatically segmented by the HMM. Due to complex motions contained in a single activity, each human-annotated activity (middle level of the tree) consists of multiple states found by the HMM. The conditional probabilities of states to activities are calculated based on the frequency of HMM states present in that activity. At the topmost level, sequential activities form a disassembly task. We see in Fig. 5A (1) and (2) that the Seq-VAE states also capture actions that are potentially missed by human annotation. For example, the human annotation labels the overall action of 'loose screws', but the Seq-VAE states capture the different sub-actions involved, such as 'loose left part of screws', and 'loose

right part of screws'.

In Fig. 6A, we show the ethograms for disassembling the HDD with 2, 3, and 4 screws. The pattern of the Seq-VAE states is maintained across the three conditions. In addition, by comparing the ethogram with the ground truth states, one can observe that they are well overlapped with each other.

Although the t-SNE analysis shows that different actions separate well in the Seq-VAE latent space and HMM on these latents captures the sub-actions well, we validate these Seq-VAE states by automatically identifying the labeled ground truth actions from the states. To do this, we train an SVM to directly classify the human actions. We show the results on held-out data in Fig. 6B,C. Here, we compared with classifying the previously extracted latents and states using PCA and AE. We see that using PCA systematically results in action 3 ('remove screws') being over-represented. Moreover, the model has a hard time recovering the final action, 'delivery HDD', using the PCA or the AE latents. Our unsupervised framework, Seq-VAE, recovers the true label with high accuracy across the five actions. With the SVM classification accuracy being $62.93 \pm 3.53\%$, $85.77 \pm 4.08\%$, and $91.52 \pm 3.04\%$, and the F1 score being $61.32 \pm 3.96\%$, $85.12 \pm 4.39\%$, and $91.33 \pm 2.87\%$ for feature extraction using PCA, AE, and Seq-VAE, respectively, our methods succeed in capturing the human actions.

F. Model Robustness to Unseen Scenarios

New experiments were conducted under the same experimental workstation setup shown in Fig. 3A. The first case (a) is: the operator grabs a screwdriver, loses 2 screws, but after putting down the screwdriver, the screwdriver starts to roll, the operator puts the screwdriver back in place, and then removes the screws at the end. The second case (b) is: the operator grabs a screwdriver, loses 4 screws, removes the screws and removes the PCB, but the PCB is fastened to the HDD and the operator needs to remove it with force and then deliver the HDD at the end. In both cases, activities that are not related to the previously defined disassembly actions include putting the rolling screwdriver back in place B_1 and removing the PCB with force B_2. Activities such as these are very likely to occur at the remanufacturing site and to affect the disassembly process.

To test the performance of our model for unseen actions, we directly input the data with two experiments into the model for testing, without retraining the model. Based on the HMM results in Fig. 7A, we distinguish unseen actions by looking for state changes that differ from the sequence of states in seen actions. Specifically, B_1 is distinguished because the red state occurs, and B_2 is distinguished because the combined yellow and blue states occur. In addition, we further verify how the SVM corresponds to the generated latent. In Fig. 7B, the region boundaries of the SVM are obtained from the training data and visualized using PCA applied to the augmented HMM states. The square points are the test data in the new scenarios. We can observe that the position corresponding to B_1 is far from the positions of the previous A_2 (orange) and the next A_3 (blue). Similarly, the position of B_2 is far from A_4 (pink) and

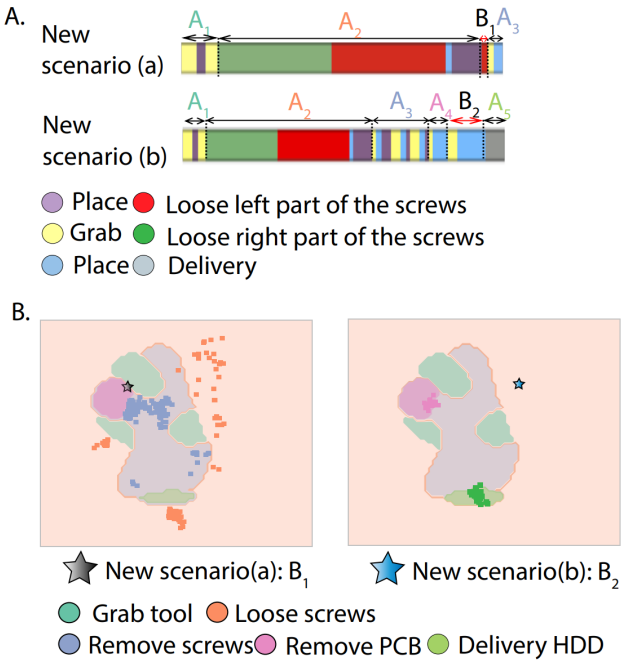

Fig. 7: HMM and SVM results on new scenarios B_1 and B_2 show that these unseen scenarios are distinguishable by the model. A. HMM results applied to the new scenarios show that unseen actions are distinguishable by considering state changes. B. SVM boundaries visualized using PCA on the augmented HMM states, with squares indicating the position of the states surrounding the scenarios B_1 and B_2.

A_5 (green). Thus, our framework has ability to successfully distinguish between unseen and predefined actions.

When it comes to the potential of applying our framework to human-robot interactions, we believe that accurately identifying different demolition activities allows the robot to provide assistance proactively. Specifically, given the safety issues that may arise if the robot moves while the human operator is working, it is desirable to have the robot assist after the human has completed an activity. In this way, the question of when the robot should provide assistance needs to be addressed in the context of human-robot collaboration. In our case study, the robot can help recycle the disassembled part immediately after the disassembly activity is completed, as indicated in Fig. 6A. The robot can help retrieve the disassembled parts whenever the related activities, such as A_3, A_4, A_5, are completed.

V. Conclusions and Future Work

This paper proposes an end-to-end unsupervised framework to efficiently perform human action recognition on a disassembly task by combining explicit feature extraction and multiple algorithms integration. The proposed framework consists of two main steps. First, a novel Seq-VAE architecture has been developed to process video data and extract spatio-temporal features of behaviors from video streams. Second, an HMM is used to identify the hidden discrete states, termed sub-actions. Finally, to understand the relationship to manually-labeled human motions, a nonlinear SVM kernel has been applied. The

proposed framework has been validated with experimental data extracted from disassembly tasks of a hard disk drive under unknown use-cases. The proposed scheme displays an advantage in identifying continuous complex activities compared to other widely applicable unsupervised learning methods.

In this paper, an end-of-use open-case HDD is used as the experiment's target, which is assumed to be of good internal condition. Since we focus on extracting features related to the operator's actions, we propose an unsupervised learning framework where temporally varying features are key towards distinguishing human activities. Large variations in the quality of the product may have an effect on the proposed unsupervised learning framework. However, we posit that this variation in quality will lead to variations in the duration of each activity that should be captured by our model. We will validate this in future work.

The work highlights the potential of unsupervised learning for the recognition of human disassembly tasks. These studies can be extended to consider advanced self-supervised learning methods such as reinforcement learning. Directly learning rich input representations can facilitate robots to adapt disassembly skills from unlabeled video data. In the future, further experimental studies can be conducted by enrolling robots in disassembly tasks to validate this framework for human-robot collaboration. Another future goal is to extend this framework to enable robots to learn disassembly tasks in a self-supervised manner.

REFERENCES

[1] E. Kim, "Interpretable and accurate convolutional neural networks for human activity recognition," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 7190–7198, 2020.

[2] Z. Chen, Q. Zhu, Y. C. Soh, and L. Zhang, "Robust human activity recognition using smartphone sensors via ct-pca and online svm," *IEEE transactions on industrial informatics*, vol. 13, no. 6, pp. 3070–3080, 2017.

[3] Z. Chen, C. Jiang, and L. Xie, "A novel ensemble elm for human activity recognition using smartphone sensors," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2691–2699, 2018.

[4] H. Su, W. Qi, Y. Hu, H. R. Karimi, G. Ferrigno, and E. De Momi, "An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1864–1872, 2020.

[5] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, "Transferable two-stream convolutional neural network for human action recognition," *Journal of Manufacturing Systems*, vol. 56, pp. 605–614, 2020.

[6] A. Nguyen, D. Kanoulas, L. Muratore, D. G. Caldwell, and N. G. Tsagarakis, "Translating videos to commands for robotic manipulation with deep recurrent neural networks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3782–3788, IEEE, 2018.

[7] R. Nawaratne, D. Alahakoon, D. De Silva, H. Kumara, and X. Yu, "Hierarchical two-stream growing self-organizing maps with transience for human activity recognition," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7756–7764, 2019.

[8] M. Zeng, T. Yu, X. Wang, L. T. Nguyen, O. J. Mengshoel, and I. Lane, "Semi-supervised convolutional neural networks for human activity recognition," in *2017 IEEE International Conference on Big Data (Big Data)*, pp. 522–529, IEEE, 2017.

[9] P. Wang, H. Liu, L. Wang, and R. X. Gao, "Deep learning-based human motion recognition for predictive context-aware human-robot collaboration," *CIRP annals*, vol. 67, no. 1, pp. 17–20, 2018.

[10] J. Zhang, P. Wang, and R. X. Gao, "Hybrid machine learning for human action recognition and prediction in assembly," *Robotics and Computer-Integrated Manufacturing*, vol. 72, p. 102184, 2021.

[11] H. Petruck and A. Mertens, "Using convolutional neural networks for assembly activity recognition in robot assisted manual production," in *International Conference on Human-Computer Interaction*, pp. 381–397, Springer, 2018.

[12] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, "Transferable two-stream convolutional neural network for human action recognition," *Journal of Manufacturing Systems*, vol. 56, pp. 605–614, 2020.

[13] W. Tao, M. C. Leu, and Z. Yin, "Multi-modal recognition of worker activity for human-centered intelligent manufacturing," *Engineering Applications of Artificial Intelligence*, vol. 95, p. 103868, 2020.

[14] A. Roitberg, N. Somani, A. Perzylo, M. Rickert, and A. Knoll, "Multimodal human activity recognition for industrial manufacturing processes in robotic workcells," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pp. 259–266, 2015.

[15] 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.

[16] Y. Zhang, C. Mitelut, G. Silasi, F. Bolanos, N. Swindale, T. Murphy, and S. Saxena, "Uncovering the effect of different brain regions on behavioral classification using recurrent neural networks," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, pp. 6602–6607, 2021.

[17] M. Al-Amin, R. Qin, M. Moniruzzaman, Z. Yin, W. Tao, and M. C. Leu, "An individualized system of skeletal data-based cnn classifiers for action recognition in manufacturing assembly," *Journal of Intelligent Manufacturing*, pp. 1–17, 2021.

[18] D. J. Rude, S. Adams, and P. A. Beling, "Task recognition from joint tracking data in an operational manufacturing cell," *Journal of Intelligent Manufacturing*, vol. 29, no. 6, pp. 1203–1217, 2018.

[19] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1134–1141, IEEE, 2018.

[20] X. Li, Y. Wang, B. Zhang, and J. Ma, "Psdrnn: An efficient and effective har scheme based on feature extraction and deep learning," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6703–6713, 2020.

[21] Q. Zhu, Z. Chen, and Y. C. Soh, "A novel semisupervised deep learning method for human activity recognition," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 3821–3830, 2018.

[22] M. R. Whiteway, D. Biderman, Y. Friedman, M. Dipoppa, E. K. Buchanan, A. Wu, J. Zhou, N. Bonacchi, N. J. Miska, J.-P. Noel, E. Rodriguez, M. Schartner, K. Socha, A. E. Urai, C. D. Salzman, T. I. B. Laboratory, J. P. Cunningham, and L. Paninski, "Partitioning variability in animal behavioral videos using semi-supervised variational autoencoders," *bioRxiv*, 2021.

[23] D. Yi, S. Musall, A. Churchland, N. Padilla-Coreano, and S. Saxena, "Disentangled multi-subject and social behavioral representations through a constrained subspace variational autoencoder (cs-vae)," *bioRxiv*, 2022.

[24] P. Asghari, E. Soleimani, and E. Nazerfard, "Online human activity recognition employing hierarchical hidden markov models," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 3, pp. 1141–1152, 2020.

[25] K. Luxem, P. Mocellin, F. Fuhrmann, J. Kürsch, S. Remy, and P. Bauer, "Identifying behavioral structure from deep variational embeddings of animal motion," *bioRxiv*, 2022.

[26] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013.

[27] L. Rabiner and B. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[28] S. Linderman, B. Antin, D. Zoltowski, and J. Glaser, "SSM: Bayesian Learning and Inference for State Space Models," 10 2020.

Xinyao Zhang received the B.E. degree in Construction Engineering from the Chang'an University, Xi'an, China, in 2019, the M.S. degree in Construction Engineering from the University of Michigan, Ann Arbor, USA, in 2020. She started working toward a Ph.D. degree in the Department of Environmental Engineering Sciences from the University of Florida, Gainesville, USA in 2021.

Her current research interests include using machine learning and robotics to perform analysis of disassembly operations and to understand how emerging technologies can improve the efficiency and sustainability of remanufacturing systems.

Daiyao Yi is a second-year PhD student in the Department of Electrical and Computer Engineering at the University of Florida. She received her M.S. degree in Biomedical Engineering from the University of Michigan, Ann Arbor, USA, in 2020 and her B.E. degree in Electrical and Electronic Engineering from the University of Nottingham Ningbo China, Ningbo, China, in 2019.

Her current research focuses on behavioral modeling and pose estimation from high-dimensional video data across subjects.

Sara Behdad is an Associate Professor at the Engineering School of Sustainable Infrastructure Environment at the University of Florida. She received her Ph.D. in Industrial and Enterprise Systems Engineering from the University of Illinois at Urbana-Champaign and her B.S. and M.S. in Industrial Engineering from Tehran Polytechnic.

Her research interests include product lifecycle engineering, design for sustainability, and remanufacturing.

Shreya Saxena is broadly interested in the neural control of complex, coordinated behavior. She is an Assistant Professor at the University of Florida's Department of Electrical and Computer Engineering. Shreya was a Swiss National Science Foundation Postdoctoral Fellow at Columbia University's Zuckerman Mind Brain Behavior Institute in the Center for Theoretical Neuroscience. She did her PhD in the Department of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology. Shreya received a B.S. in Mechanical Engineering from the Ecole Polytechnique Federale de Lausanne (EPFL), and an M.S. in Biomedical Engineering from Johns Hopkins University. She is honored to have been selected as a Rising Star in both Electrical Engineering (2019) and Biomedical Engineering (2018).