Action-Conditioned Generation of Bimanual Object Manipulation Sequences

Haziq Razali, Yiannis Demiris*

Personal Robotics Lab, Dept. of Electrical and Electronic Engineering,
Imperial College London
{h.bin-razali20,y.demiris}@imperial.ac.uk

Abstract

The generation of bimanual object manipulation sequences given a semantic action label has broad applications in collaborative robots or augmented reality. This relatively new problem differs from existing works that generate whole-body motions without any object interaction as it now requires the model to additionally learn the spatio-temporal relationship that exists between the human joints and object motion given said label. To tackle this task, we leverage the varying degree each muscle or joint is involved during object manipulation. For instance, the wrists act as the prime movers for the objects while the finger joints are angled to provide a firm grip. The remaining body joints are the least involved in that they are positioned as naturally and comfortably as possible. We thus design an architecture that comprises 3 main components: (i) a graph recurrent network that generates the wrist and object motion, (ii) an attention-based recurrent network that estimates the required finger joint angles given the graph configuration, and (iii) a recurrent network that reconstructs the body pose given the locations of the wrist. We evaluate our approach on the KIT Motion Capture and KIT RGBD Bimanual Manipulation datasets and show improvements over a simplified approach that treats the entire body as a single entity, and existing whole-body-only methods.

Introduction

Modelling human and object motion given a semantic action label has broad applications in human-robot interaction (HRI) (Chao et al. 2015) or virtual and augmented reality (AR/VR) (Chacón-Quesada and Demiris 2022). In the context of HRI, being able to forecast the hand and object trajectories would allow the robot to respond in a timely manner while avoiding collisions. For AR/VR, predictive computation facilitates systems to plan ahead on rendering with increased buffer time. Existing work however, have focused on modelling only the human motion for whole-body actions such as run, jump, walk, etc (Guo et al. 2020; Petrovich, Black, and Varol 2021). The same set of methods cannot be deployed in a setting that involves human-object interaction as there exists some spatio-temporal correlation between the human and object motion e.g., how a person orients the bottle relative to a cup during a pour action. Our goal in this work is thus to address the relatively new task of action-conditioned generation of bimanual object manipulation, where we take a semantic action label such as ”Pour” and generate a realistic sequence of both the human and object motion, from the moment the person approaches the objects and performs the pouring action, till the moment the objects are placed back on the table after completion. A first solution that can address said task would be to incorporate a densely connected graph where the nodes represent the human and object pose, and the edges their relation. However, one intuitive observation in the context of object manipulation is the varying degree each joint is involved during the action. We propose a network partitioned into three modules, each dedicated to the respective body parts or objects: (1) a graph recurrent network that models the wrist and object motion, (2) an attention-based recurrent network for the finger joints, and (3) another recurrent network for the body joints (Fig. 1). We show that this provides better performance versus representing the entire body pose as a single node.

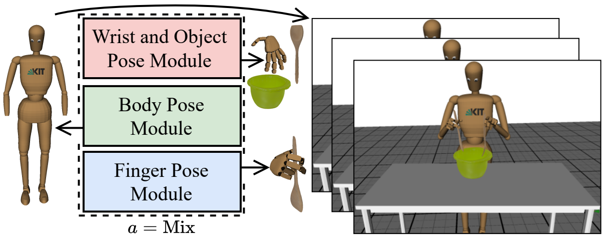

Figure 1: Given the action label a, and initial human and object pose, our network generates the entire manipulation sequence using 3 separate modules.

Next, existing works use a recurrent network built for action recognition to quantify the performance of their generative models. However, the absence of object relation in the recognition network makes it infeasible for use in our case. To address this, we propose a bidirectional graph recurrent network for bimanual action segmentation.

In summary, our contributions are as follows: (1) We introduce a novel neural network for the action-conditioned

[*] Yiannis Demiris is supported by a Royal Academy of Engineering Chair in Emerging Technologies.
Code at www.imperial.ac.uk/personal-robotics/software

generation of bimanual object manipulation sequences. To the best of our knowledge, ours is the first that tackles the problem of action-conditioned pose generation in the context of object manipulation. (2) We propose a bidirectional graph recurrent network to evaluate the performance of generative models tasked to generate bimanual actions.

Related Works

Human Pose Forecasting. Recent works on 3D human pose forecasting differ mainly in their architecture, opting for either a deterministic model (Martinez, Black, and Romero 2017; Guo and Choi 2019; Corona et al. 2020) or injecting stochasticity (Liu et al. 2021; Yuan and Kitani 2020; Kundu, Gor, and Babu 2019) via Variational Autoencoders (VAE) (Kingma and Welling 2013) or Generative Adversarial Networks (Goodfellow et al. 2014) in order to predict multiple plausible futures. These works receive as input a sequence of the past pose to output either the joint positions (Martinez, Black, and Romero 2017; Li et al. 2018) or joint rotations (Fragkiadaki et al. 2015; Pavllo, Grangier, and Auli 2018) that are then converted to positions via forward kinematics. More recent works go beyond body joints to instead, output the full SMPL body model (Loper et al. 2015; Taheri et al. 2022). Early deterministic models tend to use recurrent networks such as Gated Recurrent Units (GRU) (Chung et al. 2014) or fully convolutional layers. Several other works incorporate additional context such as eye gaze (Razali and Demiris 2021) or the object coordinates (Razali and Demiris 2022; Taheri et al. 2022). Lastly, the context-aware model (Corona et al. 2020) forecasts both the human pose and object motion and is related to our work, although there exist several notable differences. First, their model relies on the past 1 second to predict the next 2 seconds and does not take in an input action label, meaning the predicted sequence relies purely on the past motion without any controllability. Second, their model predicts the full pose at every timestep. By contrast, our method receives only the input action label and initial positions to first generate the wrist and object motion, before reconstructing the full body pose including the finger joints, from start to finish, and is thus more akin to synthesis.

Human Pose Synthesis. In contrast to human pose forecasting, methods developed for human pose synthesis typically do not receive a sequence of the past pose as input. Rather, the input may either be a zero-vector or the default standing pose. These models are then trained to generate the complete motion conditioned either on an audio signal (Li et al. 2021), a semantic action label (Guo et al. 2020; Petrovich, Black, and Varol 2021), or a sentence (Ahuja and Morency 2019). Autoregressive type methods (Guo et al. 2020) hold an advantage in that they can be easily repurposed for forecasting unlike the purely generative ones that accept only the input action label without the pose (Petrovich, Black, and Varol 2021; Ahuja and Morency 2019). Most similar to our work are Action2Motion (Guo et al. 2020) and Actor (Petrovich, Black, and Varol 2021). Action2Motion takes an action label to generate the human pose in an autoregressive manner using a VAE-GRU whereas Actor employs a VAE-Transformer (Vaswani et al. 2017) to generate the full sequence in one shot. A similarity shared by the abovementioned works is that they were built for whole-body motions such as run, walk, jump, etc without any object interaction which is our contribution in this paper.

Method

Given the one-hot action label a, initial object poses $X^0 = [x_1^0, x_2^0, ...x_N^0] \in \mathbb{R}^{N \times M \times 3}$ and their labels $L = [l_1, l_2, ...l_N]$ at time $t = 0$ for N objects represented by M points of the bounding box or motion capture markers, and the human pose $P^0 \in \mathbb{R}^{K \times 3}$ with K joints, our goal is to generate the complete human and object motion from the moment the person begins reaching for the object to perform the action, till the moment the objects are placed back on the table after completing said action. In short, we want to learn the expression $p(P^{1:T}, X^{1:T}|a, P^0, X^0, L)$.

However, not every joint of the human body is directly involved during bimanual object manipulation. The forearms, or more specifically the wrists act as the primary movers in reaching or moving the object throughout the action. Their movements may mirror each other e.g., when rolling dough with both hands, or uniquely, wherein one hand provides stability while the other makes precise movements e.g., when stirring the contents of a cup. The finger joints are then angled to ensure that the object is firmly held and oriented as required for the task such as the cutting of fruits. The remaining body joints are lastly positioned as naturally and comfortably as possible to perform the action. There is thus a higher degree of correlation between the objects and forearms. In light of this, we can partition the complete human pose P into the left and right wrists x_l, x_r, fingers $F = [f_l, f_r]$, and remaining body joints b. The wrists share the variable x and are subsumed into X as they will be treated as objects from here onwards. We can then further factorize our initial objective into three components:

$$\log p(X^{1:T}|a, X^0, L) \tag{1}$$
$$+ \log p(F^{1:T}|X^{1:T}, L) + \log p(b^{1:T}|x_l^{1:T}, x_r^{1:T})$$

The result is more reflective of the degree of interaction that occurs during bimanual manipulation in that it first generates the wrist and object pose sequence given the action label $p(X^{1:T}|a, X^0, L)$ before computing the required finger joint angles $p(F^{1:T}|X^{1:T}, L)$ at every timestep. The remaining body joints are then reconstructed independently of the object pose and finger joint angles $p(b^{1:T}|x_l^{1:T}, x_r^{1:T})$. Note that we assume the objects to already be within grasping distance. Figure 2 illustrates the framework of our system. In the following, we describe our method for all three modules.

Wrist and Object Pose Module

We formulate the task of wrist and object pose generation as a sequential modeling problem over a densely connected graph, where the wrists and objects are each represented by a vertex, and their locations recursively predicted over time. Specifically, we first define a graph $G(V, E)$ at time $t = 0$ that connects each vertex to all its neighbours, where each

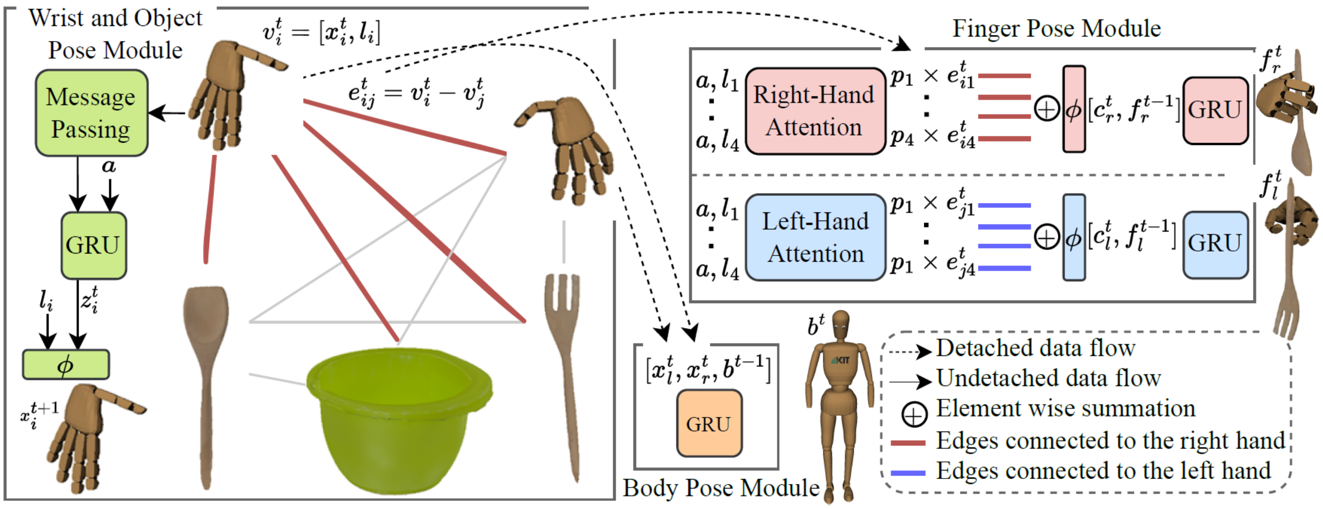

Figure 2: Overview of our method. The wrist and object motion are generated by the graph network. The outputs are then sent to the finger and body pose modules to generate the respective body parts. We detach the data flowing to the finger and body pose modules to simplify the training. The architecture can thus be viewed as 3 modular components that can be trained separately.

vertex $v_i^t = [x_i^t, l_i]$ concatenates the coordinates and label, and the edge $e_{ij}^t = v_i^t - v_j^t$ the difference between neighbours i and j. We then compute the representation for each vertex by running the graph through the edge-convolution (Wang et al. 2019) variant of the message passing scheme:

$$
\begin{aligned}
g_i^t &= \max_{j \in \mathcal{N}(i)} \phi_1([v_i^t, e_{ij}^t]) && (2)\\
&= \max_{j \in \mathcal{N}(i)} \phi_1([\phi_2([x_i^t, l_i]), \phi_2([x_j^t, l_j]) - \phi_2([x_i^t, l_i])])
\end{aligned}
$$

where ϕ denote linear layers and $[.,.]$ a concatenation. This operation encodes the relation between vertex i and all its neighbours through the edge while maintaining information about what and where the node is through the vertex information. We then concatenate the one-hot action label a to the vertex feature g_i^t and provide them as input to a GRU and subsequently a linear layer to produce the parameters of a Gaussian distribution:

$$h_i^t = \text{GRU}(h_i^{t-1}, [g_i^t, a]) \quad (3)$$

$$\mu_i^t, \sigma_i^t = \phi_3(h_i^t) \quad (4)$$

Intuitively, the GRU is tasked to generate the latent motion distribution for the object given its feature-wise proximity to all its neighbours and the action label. Lastly, we sample from the Gaussian distribution and concatenate the output to the object label, and send them to another linear layer to decode either the coordinates of the object or wrist.

$$z_i^t \sim N(\mu_i^t, \sigma_i^t) \quad (5)$$

$$x_i^{t+1} = x_i^t + \phi_4([z_i^t, l_i]) \quad (6)$$

Note that the object label l_i not only helps the vertex establish their relation to each other in equation 2 but is also useful in determining the coordinates of the bounding box corners or motion capture markers in equation 6 as it determines their size or positions relative to each other. The entire network is then run recursively until the variance of the wrist and object coordinates dip below a threshold, indicating that the action has been completed and the person has reverted to the default standing pose.

Finger Pose Module

As mentioned, the finger joint angles are a function of both the objects of interest, proximity between the wrists to said objects of interest, and action. Although information pertaining to proximity are encoded in the edge features, they need to be aggregated appropriately. To this end, we first attend to the most likely object given the action label:

$$P = \sigma(\frac{\phi_5(L)\phi_6(a)^T}{\sqrt{d}}) \quad (7)$$

where d represents the dimension of the embedded variables and P the vector of object probabilities. We then use the probability scores to scale the edge features returned by the graph network and concatenate the result to the finger pose from the previous timestep for prediction.

$$h_i^t = \text{GRU}(h_i^{t-1}, [\sum p_i \times e_{ij}^t, f^t]) \quad (8)$$

$$f^{t+1} = \phi_7(h_i^t) \quad (9)$$

where a GRU is used to enforce temporal smoothness. Since the handedness of an individual affects the object selected by the left and right hand, we incorporate a separate set of weights for the operations above for each hand to induce a bias towards the object frequently selected by the respective hand as shown in Fig. 2. Our finger pose module works as a deterministic model as there is very little to no in-hand manipulation of the objects. The finger module can be easily augmented with a VAE if there is a need for stochasticity.

Body Pose Module

Existing work has shown that the head has a significant lead before motor actions only if the objects are not situated in front of the person within the field of view, nor is it within grasping distance (Land 2006). Because we assume the converse, we find it sufficient to simply concatenate the locations of the left and right wrists, and the body at the previous timestep as input to predict it at the next timestep.

$$h_i^t = \mathrm{GRU}(h_i^{t-1}, [x_l^t, x_r^t, b^{t-1}]) \tag{10}$$

$$b^t = \phi_8(h_i^t) \tag{11}$$

Likewise, we find the GRU crucial in maintaining temporal smoothness.

Loss Function

Altogether, our architecture is trained end-to-end to minimize the following loss at every timestep:

$$\underbrace{\lambda_1 ||\hat{X}^t - X^t||_2^2 + \lambda_2 \mathrm{KL}(q(Z^t|X^{\leq t}, Z^{<t})||p(Z^t|X^{<t}, Z^{<t})}_{\text{Wrist and Object Pose Loss}}$$

$$+ \underbrace{\lambda_3 ||\hat{F}^t - F^t||_2^2 + \lambda_4 P \log(\hat{P})}_{\text{Finger Pose Loss}} + \underbrace{\lambda_5 ||\hat{b}^t - b^t||_2^2}_{\text{Body Pose Loss}} \tag{12}$$

where KL denotes the Kullback-Liebler divergence and the lambdas the tuning parameters. Note that our KL term does not assume the posterior to be a unit Gaussian and thus requires a posterior graph network that is removed at test time (Guo et al. 2020). We include the cross entropy for the most likely object-action pair to act as an auxiliary loss for the attention network in equation 7. Lastly, we simplify the training and selection of lambdas by detaching the gradients flowing from the finger and body pose decoders into the object decoder and setting all lambdas to 1. The architecture in Figure 2 can thus be viewed as 3 modular components that can be trained separately.

Experiments

Datasets

The **KIT Motion Capture Dataset** (Krebs et al. 2021) contains motion capture data of a right-handed person performing bimanual actions such as Cut, Pour, Stir, etc. We only select tasks with at least two interacting objects. The result is 995 sequences across 9 actions, each on average 8 seconds long that is temporally annotated with a total of 15 intra- or fine-action labels such as Approach, Hold, Cut, etc. The person stands directly in front of a table with the objects within both field-of-view and grasping distance. The person then picks up the objects to perform an action before placing them back to where they were originally picked from. We preprocess the dataset by centering the coordinates with respect to the table center, augment it by adding Gaussian noise in the horizontal XY plane to all entities except the table, and sample each sequence at 10 Hz. The added noise is constant throughout time and also such that the objects do not go beyond the table. We use the motion capture markers on each object to compute its oriented 3D bounding box. We split the dataset to obtain a train:test ratio of 70:30. The **KIT RGBD Dataset** (Dreher, Wächter, and Asfour 2019) differs in that it contains RGBD recordings instead of motion capture. We use the provided ground truths that are estimated using OpenPose (Cao et al. 2017) and YOLOv3 (Redmon and Farhadi 2018). We clip the video such that the person is already standing in front of the table immediately before performing the action and preprocess in the same way as above. The result is 480 sequences across 8 actions, each on average 10 seconds long that is temporally annotated with a total of 14 fine actions. We represent each object by its centroid as we find the training to be highly unstable if using the estimated 3D bounding box corners.

Evaluation Metrics

We follow performance measures first proposed in (Guo et al. 2020): accuracy, FID, diversity, and multimodality. FID and accuracy measure the model's ability to generate sequences that are semantically correct, while diversity and multimodality the overall and within-class variance respectively. However, because the dataset provides fine action labels, we compute said metrics per-frame instead of per-sequence and additionally compute the segmental F1 score (Lea et al. 2017) at an intersection over union of 0.5 (F1@0.5) and edit distance (Lea, Vidal, and Hager 2016). The segmental score penalizes over-segmentation errors while the edit distance predictions that are out-of-order.

We then noted that there exists no model that performs bimanual action segmentation given 3D coordinate data. The recognition model used in (Guo et al. 2020; Petrovich, Black, and Varol 2021) do not jointly model the human and object motion whereas the bimanual segmentation models proposed by (Morais et al. 2021) and (Dreher, Wächter, and Asfour 2019) operate on images and is built for online segmentation respectively. (Dreher, Wächter, and Asfour 2019) additionally requires symbolic relations which may not reflect the generative model's ability to replicate real-world data. An advantage presented to us is that our model effectively enables the classification at every timestep to use information from both the past and future. We thus train a modification of our graph network that uses only the wrist and object pose for segmentation and convert it to a bidirectional variant to utilize said information from both past and future. We do not use the body and finger pose since the motion of the wrist and object intuitively provides sufficient discriminatory information. Lastly, because our segmentation model does not utilize the body and fingers, we compute their MSE for the sake of evaluation. In summation, we report the per-frame accuracy, FID, diversity, multimodality, segmental F1 score, and edit distance for the wrist and object pose, and the MSE scores for the body and finger pose. We generate sets of sequences 20 times with different random seeds and report the mean and confidence interval at 95%.

Setup

We compare our approach to existing whole-body-only methods: Action2Motion (A2M) (Guo et al. 2020) and Actor (Petrovich, Black, and Varol 2021), and two variants of our model: one that treats the wrists and fingers as a single node in the wrist module without a separate finger pose module (WF), and another that treats the entire human pose as a single node in the graph without the finger and body pose modules (CA), with the latter being closely related to

		Kit Motion Capture Dataset						KIT RGBD Dataset					
a	l	Acc ↑	F1@0.5 ↑	Edit ↑	FID ↓	Div. →	Multi. →	Acc ↑	F1@0.5 ↑	Edit ↑	FID ↓	Div. →	Multi. →
✗	✗	$62.6^{\pm0.70}$	$49.5^{\pm1.12}$	$76.4^{\pm0.66}$	$36.9^{\pm0.87}$	$19.9^{\pm.12}$	$12.2^{\pm.16}$	$52.6^{\pm0.96}$	$43.4^{\pm0.81}$	$62.0^{\pm0.88}$	$49.9^{\pm1.19}$	$16.0^{\pm.13}$	$12.2^{\pm.13}$
✗	✓	$63.6^{\pm0.88}$	$53.7^{\pm1.18}$	$80.3^{\pm0.64}$	$30.0^{\pm1.04}$	$18.0^{\pm.10}$	$10.3^{\pm.23}$	$59.7^{\pm0.92}$	$48.7^{\pm0.82}$	$64.9^{\pm0.49}$	$40.0^{\pm0.75}$	$15.8^{\pm.09}$	$11.9^{\pm.08}$
✓	✗	$65.9^{\pm0.51}$	$55.4^{\pm0.81}$	$81.3^{\pm0.67}$	$27.3^{\pm0.69}$	$18.1^{\pm.17}$	$10.1^{\pm.09}$	$58.0^{\pm0.83}$	$47.6^{\pm0.92}$	$63.9^{\pm0.86}$	$42.5^{\pm0.60}$	$16.1^{\pm.10}$	$11.8^{\pm.09}$
✓	✓	$\mathbf{71.2}^{\pm0.38}$	$\mathbf{67.6}^{\pm0.86}$	$\mathbf{88.0}^{\pm0.60}$	$\mathbf{28.8}^{\pm0.87}$	$18.1^{\pm.13}$	$10.1^{\pm.10}$	$\mathbf{65.0}^{\pm1.19}$	$\mathbf{52.2}^{\pm0.90}$	$\mathbf{68.1}^{\pm1.01}$	$\mathbf{31.7}^{\pm0.73}$	$16.5^{\pm.08}$	$12.3^{\pm.13}$
Real		$90.0^{\pm0.21}$	$95.1^{\pm0.56}$	$97.2^{\pm0.44}$	$0.9^{\pm0.43}$	$19.1^{\pm.12}$	$10.2^{\pm.12}$	$76.5^{\pm0.47}$	$76.7^{\pm0.90}$	$78.8^{\pm0.68}$	$1.3^{\pm0.68}$	$17.7^{\pm.09}$	$12.4^{\pm.07}$
WF		$69.5^{\pm0.45}$	$64.1^{\pm0.87}$	$86.3^{\pm0.62}$	$29.2^{\pm0.74}$	$18.1^{\pm.13}$	$10.1^{\pm.09}$	$64.1^{\pm0.99}$	$51.2^{\pm0.88}$	$67.5^{\pm0.91}$	$32.1^{\pm0.66}$	$16.4^{\pm.09}$	$12.2^{\pm.10}$
CA		$66.9^{\pm0.62}$	$62.7^{\pm1.21}$	$85.9^{\pm0.66}$	$29.7^{\pm0.50}$	$18.1^{\pm.14}$	$10.1^{\pm.10}$	$62.7^{\pm0.25}$	$50.8^{\pm0.74}$	$66.5^{\pm0.44}$	$33.1^{\pm0.37}$	$18.6^{\pm.11}$	$13.3^{\pm.09}$
Actor		$59.0^{\pm0.88}$	$57.5^{\pm1.60}$	$80.0^{\pm0.73}$	$32.8^{\pm1.18}$	$18.2^{\pm.18}$	$9.7^{\pm.14}$	$57.5^{\pm1.09}$	$46.3^{\pm1.08}$	$63.5^{\pm0.87}$	$41.2^{\pm0.80}$	$16.4^{\pm.11}$	$12.0^{\pm.08}$
A2M		$57.2^{\pm1.17}$	$50.2^{\pm1.77}$	$80.6^{\pm1.65}$	$40.2^{\pm3.45}$	$18.1^{\pm.12}$	$11.0^{\pm.13}$	$51.9^{\pm3.10}$	$39.9^{\pm1.72}$	$58.8^{\pm1.91}$	$48.4^{\pm5.19}$	$16.4^{\pm.33}$	$12.9^{\pm.30}$

Table 1: We study the effect the action a and object labels l have on the wrist and object pose module and compare to previous work. The top rows ablate our method with the action and object labels. In the middle row, the results on real data, and in the bottom rows, all previous work. Our method with both a and l outperforms all previous work. → means motions are better when the metric is closer to real.

	Body Pose MSE ↓	
Body Pose Decoder	Kit Motion Capture	KIT RGBD
MLP (Ours)	$0.171^{\pm0.488}$	$0.044^{\pm0.009}$
RNN (Ours)	$\mathbf{0.057}^{\pm0.107}$	$\mathbf{0.038}^{\pm0.011}$
CA	$0.091^{\pm0.207}$	$0.051^{\pm0.031}$
Actor	$0.379^{\pm0.440}$	$0.192^{\pm0.045}$
A2M	$0.316^{\pm0.487}$	$0.187^{\pm0.010}$

Table 2: We study using an MLP vs RNN for reconstructing the body pose and compare to the state-of-the-art. Our method utilizing an RNN performs the best.

	Finger Pose MSE ↓	
Finger Pose Edge Agg.	Kit Motion Capture	KIT RGBD
Average (Ours)	$0.056^{\pm0.119}$	$\mathbf{0.032}^{\pm0.080}$
Attention (Ours)	$\mathbf{0.044}^{\pm0.081}$	$0.033^{\pm0.082}$
WF	$0.061^{\pm0.162}$	$0.033^{\pm0.091}$
CA	$0.097^{\pm0.078}$	$0.035^{\pm0.096}$
Actor	$0.404^{\pm1.344}$	$0.134^{\pm0.165}$
A2M	$0.466^{\pm1.762}$	$0.050^{\pm0.093}$

Table 3: We study various edge aggregating mechanisms in the finger pose module and compare to the state-of-the-art. Our method utilizing attention performs the best.

the Context-Aware model (Corona et al. 2020). We adapt both Action2Motion and Actor to accept the object label in addition to the action label. We train all models including ours for 1000 epochs using the ADAM optimizer (Kingma and Ba 2014) with an initial learning rate of 1e-3 and batch size of 32. The experiments were implemented using PyTorch 1.12.0 installed on an Ubuntu 20.04 machine with an NVIDIA RTX-2080. Finally, we have the number of parameters of all models match at approximately 2.5 million.

Quantitative Results

Effect of object and action labels. We ablate several components of our architecture while comparing to previous work. The first question we ask is whether the information provided by object and action labels to the graph network is beneficial. Note that the accuracy, segmental F1 score, edit distance, and FID are more important than diversity and multimodality as they reflect the quality of the sequences. In Table 1, we first see that our variant with both object and action labels has the best performance on both datasets by a considerable margin. The labels allow the model to generate sequences that are more representative of real data as evidenced by the better per-frame accuracy and FID. Likewise, the segmental F1 score and edit distance also suggest that the sequences are more recognizable temporal-wise. Next, we observe our method leading both the CA and WF variants. Our method learns better as it employs a single module that is tasked to learn only the spatio-temporal relationship between the wrists and objects. Lastly, the numbers slightly degrade on the more challenging KIT RGBD dataset and is principally due to the ground truths being obtained via detectors as opposed to the more accurate motion capture cameras.

Choice of body pose decoder. We next ask if our choice for the decoder matters when reconstructing the body by comparing a GRU against a simple MLP in Table 2. Note that because the graph recurrent network does not require the output of the body pose module in order for it to recursively predict the future, the body pose module thus effectively has access to the wrist information from $t = 0$ to $t = T$. The MSE suggests that the model using temporal information provides better reconstructions. We can attribute this mainly to the fact that there exist various ways for a person to naturally position his body given the same wrist locations. Using a GRU with the past pose as information would thus ensure that the reconstructions are temporally smooth. Likewise, our method with the RNN body decoder outperforms recent work.

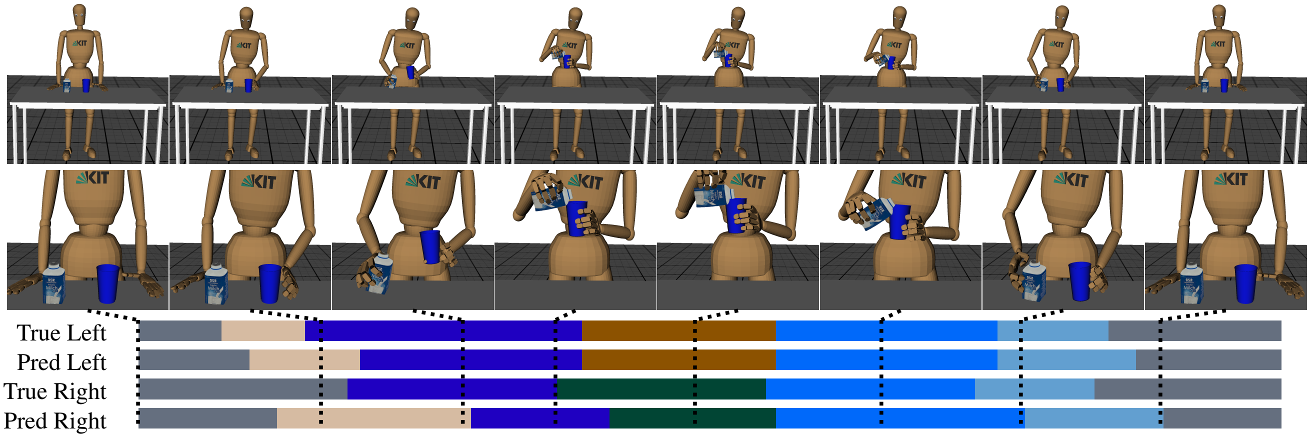

Figure 3: Output for the KIT Motion Capture dataset. Our model can generate a sequence that closely resembles the "Pour" motion with fingers that are appropriately angled. The colored bars show the true and predicted segmentation for the left and right hands with the lines referencing the corresponding time. ■ Idle ■ Approach ■ Move ■ Hold ■ Pour ■ Place ■ Retreat.

Choice of edge aggregating mechanism. We examine the importance of the aggregating mechanism when computing the finger joint angles given the edges. Table 3 shows that the variant with attention is able to achieve a lower MSE than the one that does a simple average on the KIT Motion Capture dataset. Naturally, the estimation of the finger pose is made easier if the model can first de-emphasize the objects each hand will not interact with through the attention mechanism. On the KIT RGBD dataset, the two aggregation methods are on par likely due to the noisy estimates returned by the detectors. Nevertheless, the lower numbers compared to previous work still indicate that our finger pose module learns better since it is not required to simultaneously predict the positions of the other less relevant joints and objects. The tables highlight the efficacy of our novel approach in modelling bimanual object manipulation. By grouping the joints and objects based on their degree of interaction with each other, and dedicating a single module to each group, the learning is made much easier. Whole-body-only methods are insufficient for bimanual object manipulation except when there is very low variation in the action. The context version may be improved by fine-tuning the lambdas for each body part and for each dataset but with increased training time. Our method, in contrast, is modular and free from tuning.

Qualitative Results

We present some visual results for the KIT Motion Capture dataset in Fig. 3 where the sequence in the top row illustrates the pour action, the next row a closer view of the fingers and objects, and the horizontal bar the predicted left and right-hand fine actions at every timestep with the lines referencing the respective timesteps. Our segmentation model is able to correctly predict the fine actions with the errors coming mainly from the exact moment the action transits from one label to the next. The fingers also appear to be appropriately angled. The figures show several limitations of our model. First, because there is no mechanism nor constraint that forces the object to remain attached to the hand after it is grasped, we can see in several frames the object being at an unnatural distance from the hand. Second, our model does not check for collision nor was it trained with the interpenetration loss as the data has not been annotated for contact. However, our modular design makes it easy to replace our finger pose module with GrabNet (Taheri et al. 2020). Fig. 4 illustrates our output on the KIT RGBD dataset where the 3D points are reprojected back onto the image sequence. A close inspection of the sequence will show the hand and object motion remaining stationary throughout the action, which is the consequence of minimizing the MSE given noisy or very fine data. This consequently also results in the segmentation model's inability to associate the motion to the very transient fine actions. Note that the predictions being off from the objects in image space does not equate to poor performance as it is ultimately a generative problem given only the initial positions. The reprojections are for the sake of visualization. All-in-all, the output is still very usable for HRI since it enables the robot to roughly predict the locations of the body pose, hand, object positions, and action over time given the first frame and uttered action. It can then refine its predictions as more data is presented over time.

Robustness Against Distractors

Our autoregressive model can also be used for forecasting with zero modifications i.e., it can accept a sequence of the past pose and action label to output the future sequence. In the context of forecasting for applications such as HRI, the issue of distractors becomes much more relevant. Distractors are defined as irrelevant objects within the scene that the person does not interact with both directly and indirectly given the action label. For instance, a rolling pin given the "Cut" action. A model without any mechanism to handle distractors will see its performance degrade in the real world. We assess our graph's robustness by training and testing it with increasing distractor counts. For each sequence, we randomly sample a set of objects from a different sequence and set their velocities to zero. To ease the assessment, we

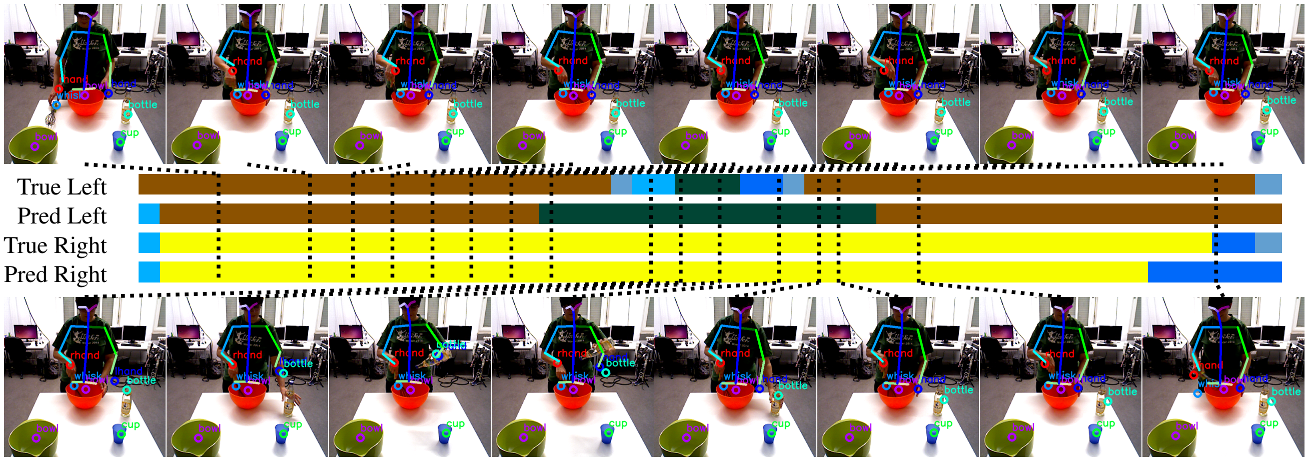

Figure 4: Our model is able to generate the semantic positioning of the various objects for the ”Cook” action on the more challenging KIT RGBD dataset. The colored bars show the true and predicted segmentation for the left and right hands with the lines referencing the corresponding time. ■ Approach ■ Lift ■ Hold ■ Stir ■ Pour ■ Place ■ Retreat.

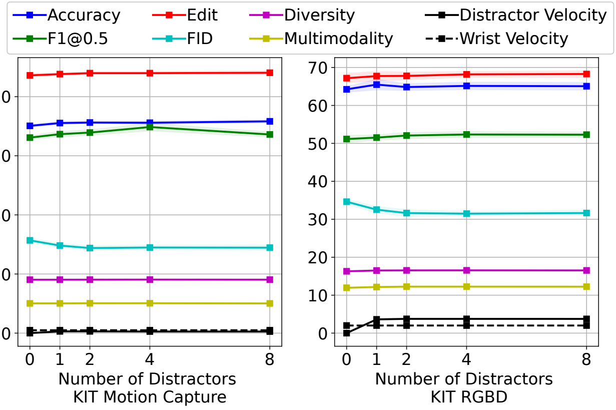

Figure 5: Performance with increasing distractors. The distractor velocity has been scaled by 100 to aid analysis.

set the distractor count to 4 during training and [0,1,2,4,8] during testing. We then have the generative model receive the first frame as input to forecast the complete sequence. We do not sample relevant objects that are already present in the scene e.g., an additional knife for a ”Cut” action as it is simply not possible to forecast which knife the person reaches for without more information in the input action text such as ”Cut 2 cucumbers using the left-most knife”. Some sequences, however, do contain relevant distractors but are always placed beyond grasping distance.

Fig. 5 shows our model's ability to ignore distractors. The recognition and segmentation metrics remain near constant. We also indicated the average wrist and distractor velocities to ensure that the result is not due to the segmentation model's ability to disregard distractors. Our model successfully maintains near constant distractor velocity with increasing distractor counts. A valid critique of our method is that the same performance can be achieved by using lookup tables to suppress distractors given the one-hot action label. However, we argue that learning is more beneficial especially for future work when handling open-vocabulary sentences that may not explicitly reference the objects in use.

Conclusion

We tackle the relatively new task of action-conditioned generation of bimanual object manipulation sequences. We proposed a novel neural network that splits the body joints into 3 separate parts according to their degree of interaction with the object, which shows improvements over prior work. We based our method on an autoregressive approach which can be used as a generative or forecasting model for AR or HRI respectively. Our method's modularity makes selection of lambdas easy, and allows swapping various components; for example, our finger pose module could be easily replaced with GrabNet if the focus was only on the generative setting and the object mesh is readily available. Limitations include the absence of any constraints to have the object remain attached to the grasping hand during manipulation. Actions with semantic variation such as pouring the contents from a bowl to a cup or vice-versa will also require two different one-hot encodings for improved performance. Lastly, a unique problem for this task is that evaluation becomes more challenging due to the need for fine action segmentation as opposed to the intuitively easier task of recognition given an entire whole-body sequence as done in (Guo et al. 2020; Petrovich, Black, and Varol 2021). It is easier to obtain near-perfect performance with recognition than it is with fine action segmentation. This then leads to an issue where the segmentation model must be first improved in order for the reported metrics to better reflect the performance of the generative or forecasting model. The analysis will not be as clear-cut if the discriminative power of the segmentation model is inadequate as it can result in the numbers being close given either a poor or an excellent generative model.

References

Ahuja, C.; and Morency, L.-P. 2019. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, 719–728. IEEE.

Cao, Z.; Simon, T.; Wei, S.-E.; and Sheikh, Y. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7291–7299.

Chacón-Quesada, R.; and Demiris, Y. 2022. Proactive Robot Assistance: Affordance-Aware Augmented Reality User Interfaces. *IEEE Robotics Automation Magazine*, 29(1): 22–34.

Chao, Y.-W.; Wang, Z.; He, Y.; Wang, J.; and Deng, J. 2015. HICO: A Benchmark for Recognizing Human-Object Interactions in Images. In *Proceedings of the IEEE International Conference on Computer Vision*.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Corona, E.; Pumarola, A.; Alenya, G.; and Moreno-Noguer, F. 2020. Context-aware Human Motion Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6992–7001.

Dreher, C. R.; Wächter, M.; and Asfour, T. 2019. Learning object-action relations from bimanual human demonstration using graph networks. *IEEE Robotics and Automation Letters*, 5(1): 187–194.

Fragkiadaki, K.; Levine, S.; Felsen, P.; and Malik, J. 2015. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, 4346–4354.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Guo, C.; Zuo, X.; Wang, S.; Zou, S.; Sun, Q.; Deng, A.; Gong, M.; and Cheng, L. 2020. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2021–2029.

Guo, X.; and Choi, J. 2019. Human motion prediction via learning local structure representations and temporal dependencies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 2580–2587.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Krebs, F.; Meixner, A.; Patzer, I.; and Asfour, T. 2021. The KIT Bimanual Manipulation Dataset. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, 499–506. IEEE.

Kundu, J. N.; Gor, M.; and Babu, R. V. 2019. Bihmp-gan: Bidirectional 3d human motion prediction gan. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 8553–8560.

Land, M. F. 2006. Eye movements and the control of actions in everyday life. *Progress in retinal and eye research*, 25(3): 296–324.

Lea, C.; Flynn, M. D.; Vidal, R.; Reiter, A.; and Hager, G. D. 2017. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 156–165.

Lea, C.; Vidal, R.; and Hager, G. D. 2016. Learning convolutional action primitives for fine-grained action recognition. In *2016 IEEE international conference on robotics and automation (ICRA)*, 1642–1649. IEEE.

Li, C.; Zhang, Z.; Sun Lee, W.; and Hee Lee, G. 2018. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5226–5234.

Li, R.; Yang, S.; Ross, D. A.; and Kanazawa, A. 2021. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13401–13412.

Liu, Z.; Lyu, K.; Wu, S.; Chen, H.; Hao, Y.; and Ji, S. 2021. Aggregated multi-gans for controlled 3d human motion prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2225–2232.

Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; and Black, M. J. 2015. SMPL: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6): 1–16.

Martinez, J.; Black, M. J.; and Romero, J. 2017. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2891–2900.

Morais, R.; Le, V.; Venkatesh, S.; and Tran, T. 2021. Learning asynchronous and sparse human-object interaction in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16041–16050.

Pavllo, D.; Grangier, D.; and Auli, M. 2018. Quaternet: A quaternion-based recurrent model for human motion. *arXiv preprint arXiv:1805.06485*.

Petrovich, M.; Black, M. J.; and Varol, G. 2021. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10985–10995.

Razali, H.; and Demiris, Y. 2021. Using Eye-Gaze to Forecast Human Pose in Everyday Pick and Place Actions. *International Conference on Robotics and Automation*.

Razali, H.; and Demiris, Y. 2022. Using a Single Input to Forecast Human Action Keystates in Everyday Pick and Place Actions. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3488–3492. IEEE.

Redmon, J.; and Farhadi, A. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Taheri, O.; Choutas, V.; Black, M. J.; and Tzionas, D. 2022. GOAL: Generating 4D Whole-Body Motion for Hand-Object Grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13263–13273.

Taheri, O.; Ghorbani, N.; Black, M. J.; and Tzionas, D. 2020. GRAB: A dataset of whole-body human grasping of objects. In *European conference on computer vision*, 581–600. Springer.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics (TOG)*.

Yuan, Y.; and Kitani, K. 2020. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, 346–364. Springer.