Visual Haptic Reasoning: Estimating Contact Forces by Observing Deformable Object Interactions

Yufei Wang, David Held, and Zackory Erickson

***Abstract*—Robotic manipulation of highly deformable cloth presents a promising opportunity to assist people with several daily tasks, such as washing dishes; folding laundry; or dressing, bathing, and hygiene assistance for individuals with severe motor impairments. In this work, we introduce a formulation that enables a collaborative robot to perform visual haptic reasoning with cloth—the act of inferring the location and magnitude of applied forces during physical interaction. We present two distinct model representations, trained in physics simulation, that enable haptic reasoning using only visual and robot kinematic observations. We conducted quantitative evaluations of these models in simulation for robot-assisted dressing, bathing, and dish washing tasks, and demonstrate that the trained models can generalize across different tasks with varying interactions, human body sizes, and object shapes. We also present results with a real-world mobile manipulator, which used our simulation-trained models to estimate applied contact forces while performing physically assistive tasks with cloth. Videos can be found at our project webpage.[1]**

***Index Terms*—Physically Assistive Devices; Deep Learning for Visual Perception; Perception for Grasping and Manipulation**

I. INTRODUCTION

ROBOTIC manipulation of highly deformable cloth presents a promising opportunity to assist people with many tasks, such as assisting an older adult with muscle atrophy or a physical disability to get dressed [1], bathing and hygiene assistance with a washcloth or towel [2], cleaning dishes with a dish towel, folding laundry [3], [4], or bed making [5], [6]. In each of these scenarios, it can be helpful for a robot to infer how cloth interacts with and applies forces to objects it makes contact with. For example, the applied force between a gown and human body can inform a robot if the gown is getting caught during assistive dressing, and the force between a washcloth and the cleaning surface can tell if the dirt on the surface is successfully removed during assistive bathing or dish cleaning. Besides inferring such task execution status, knowing the applied force is also vital to prevent harm and discomfort during physical interactions between robots and humans in these tasks.

In this paper, we introduce methods that enable collaborative robots to perform *haptic reasoning* with cloth by using only point cloud observations and robot kinematics. As shown in Fig. 1, *haptic reasoning* consists of inferring the distribution of applied forces as cloth physically interacts with other objects. Prior work has introduced methods for estimating contact forces purely from visual feedback as rigid objects undergo contact with robot end-effectors or human hands at a few discrete points [7]–[9]. In contrast, cloth lacks a succinct state representation, has inherently high-dimensional non-linear dynamics, and has contacts with objects over large surface areas, which collectively presents unique challenges for estimating the location and magnitude of applied forces as cloth interacts with other objects in the environment.

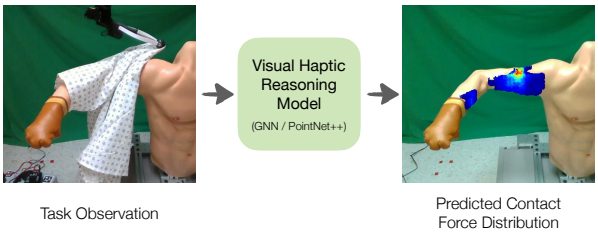

Fig. 1. An illustration of the predicted contact force distributions by the proposed visual haptic reasoning model as a deformable gown physically interacts with a manikin in the robot-assisted dressing task. Note that the model is trained entirely in physics simulation.

To overcome these challenges, we take a data-driven approach with physics simulation to model the physical interactions between cloth and other objects. We present two model formulations for doing haptic reasoning during cloth manipulation. The first introduces a graph neural network (GNN) [10] architecture that encodes the local interactions between the cloth and object in contact, whereas the second approach uses a PointNet++ [11] architecture. Both models form a mapping from a point cloud observation of the task to a 3D representation of the applied forces on an object. We conduct quantitative evaluations in physics simulation and contrast the predicted force distributions of both model formulations during robot-assisted dressing, bathing, and dish washing tasks. We perform ablation studies on these models and evaluate generalization performance as cloth interacts with a wide distribution of human body sizes and object shapes. Finally, through cloth manipulation studies in the real world, we demonstrate that haptic reasoning models trained in physics simulation can be transferred to a real-world mobile manipulator to infer applied contact forces.

In summary, we make the following contributions:

- We introduce model formulations for visual haptic reasoning, which enable a robot with visual sensing to infer the force distributions that cloth applies onto other objects during manipulation.
- We performed analysis of these haptic reasoning models in physics simulation across a number of tasks including robot-assisted dressing, bathing, and dish washing.
- We evaluate haptic reasoning in the real world and

Manuscript received: February, 25, 2022; Revised June, 1, 2022; Accepted July, 28, 2022.

This paper was recommended for publication by Editor Angelika Peer upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the National Science Foundation under Grant No. IIS-2046491, and LG Electronics. Corresponding author: Yufei Wang.

The authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA. Emails: yufeiw2@andrew.cmu.edu, dheld@andrew.cmu.edu, zackory@cmu.edu

Digital Object Identifier (DOI): see top of this page.

[1]https://sites.google.com/view/visualhapticreasoning/home

demonstrate these models with a mobile manipulator performing physically assistive tasks.

II. RELATED WORK

A. Deformable cloth manipulation for robotic assistance

Several assistive robotic tasks involve manipulating deformable objects like cloth around the human body. Examples include dressing assistance with hospital gowns, jackets, scarfs [1], [2], [12]–[14]; bathing assistance with a towel [2], [15], picking and placing garments on hangers [16], [17], laundry folding [3], [4], [18]–[25], and bedding assistance [5], [6]. Prior work [26] has showed how a robot could predict if an end effector trajectory would succeed in dressing a hospital gown sleeve onto a person by leveraging force measurements at the end effector. In contrast to these prior works, we introduce a methodology for an assistive robot to infer the location and magnitude of the forces that cloth applies onto the human body using point cloud observations.

B. Estimating Contact from Vision

Several previous works [7]–[9], [27]–[32] have explored estimating contact points and forces between objects in contact purely from vision using RGB, RGB-D or thermal images. Most of them focus on interactions between human hands and non-deformable objects [7], [9], [30], or between a rigid robot end effector and a non-deformable object [8], [27], [33], or between a rigid robot tool and a deformable organ [31], [32], where there are only a few points of contact. In contrast to these prior works, our work focuses specifically on estimating contact forces when manipulating deformable cloth around other rigid objects, which results in hundreds of points of contact and applied forces across a human body or object surface. Zhu *et al*. [28] are able to predict contact forces on every human mesh vertex when the human sits on a chair from RGB-D images, by using Finite Element Method (FEM) and modeling the human as a soft body. This approach is intractable for a collaborative robot that must make predictions in real time when physically interacting with people. We instead use neural network models trained entirely in physics simulation to predict the force that cloth applies onto other objects, which is much faster as it only needs a single forward pass of the network.

C. Estimating Contact Force during Interaction Involving Deformable Objects

One prior work [29] most relevant to ours estimates the contact forces between a hospital gown and human limbs during assistive dressing. The trained models rely on manual discretization of the human limb into a fixed set of contact locations, which limits generalization. In a following work [1] the estimated forces are used in model predictive control for assistive dressing. Instead, by using point cloud observations, we need no such discretization for the object and our trained model generalizes across different assistive tasks, human shapes and object sizes. Clever *et al*. [34] used physics simulation to compute the pressure of a human lying on a deformable bed, and trained a neural network model for estimating human pose in bed. We also use physics simulation to get high-resolution force distributions, but we focus on estimating the forces applied by deformable cloth to other objects in tasks such as robot-assisted dressing and bathing, instead of the forces applied from a lying rigid human to a deformable bed.

III. PROBLEM FORMULATION

Given a robot manipulation scenario where deformable cloth interacts with another fixed object (e.g., helping a person dress a garment), we aim to learn a model that predicts the applied normal forces between the cloth and the object based on a point cloud observation of the scene. Formally, given a depth image $I \in \mathbf{R}^{H \times W}$ (where H and W are the height and width of the image) of the scene, we transform the depth image to a point cloud $\boldsymbol{P} \in \mathbf{R}^{H \times W \times 3}$ using the camera's intrinsic matrix. We then segment the point cloud to obtain a set of points associated with the cloth $\boldsymbol{P}_C \subseteq \boldsymbol{P}$ and a set of object points $\boldsymbol{P}_O \subseteq \boldsymbol{P}$. In simulation, the classification and segmentation of points is provided directly by the simulator. In the real world, we use color thresholding on the RGB image of the scene to perform the segmentation. We aim to learn a haptic reasoning model that can predict the magnitude of the contact normal force $f_i \in \mathbb{R}$ that cloth applies to each point $\boldsymbol{p_i} \in \boldsymbol{P}_O$ on the object.

We make the following assumptions for this problem. First, we assume the object remains static during the interaction. This assumption helps address the visual occlusion of the object caused by cloth during the interaction, as with this assumption we can obtain the object point cloud $\boldsymbol{P}_O$ before the cloth occludes the object. Future work can investigate methods such as capacitive sensing [1], [13] to track the pose of an object or human limb under visual occlusion. In addition, we assume the critical contact areas between cloth and an object are observable through a partial point cloud. Future work could incorporate 3D model inference [35], [36] and force/torque sensing to predict force distributions over the full 3D model of an object.

IV. CLOTH MODELING AND CONTACT FORCE COMPUTATION VIA PHYSICS-BASED SIMULATION

We use physics-based simulation to compute the contact force between cloth and other objects, which are used as the ground-truth labels for training our proposed visual haptic reasoning models. Specifically, we choose NVIDIA FleX as our simulator, which uses position-based dynamics [37] for cloth simulation. We describe briefly here how position-based dynamics models cloth and how contact forces are computed, and refer the readers to [37]–[39] for full details. We also note that the proposed framework is orthogonal to the simulation techniques used to compute the contact forces.

In position-based dynamics, objects are represented using particles and constraints between them. A cloth can be created from a triangular mesh. A particle is created for each vertex in the mesh, and a stretching constraint is created for each edge [37], [38], which models a spring that tries to maintain its rest length. A bending constraint and a shearing constraint are created for pairs of adjacent triangles. Three stiffness parameters $k_{stretch}$, $k_{bending}$ and k_{share} are used to model how strong these constraints are. At each simulation step, after the positions of the particles are updated due to external forces such as gravity, the particle positions are projected to obey the constraints using a Guassian-Sidel algorithm.

Each particle in the cloth can have contacts with other objects, e.g., represented using another triangular mesh. The contact forces are solved using the classic Lagrangian dynamics, with a contact constraint in the form: $c(\boldsymbol{p}) = \boldsymbol{n}^T(\boldsymbol{p} - \boldsymbol{q}) - d \geq 0$,

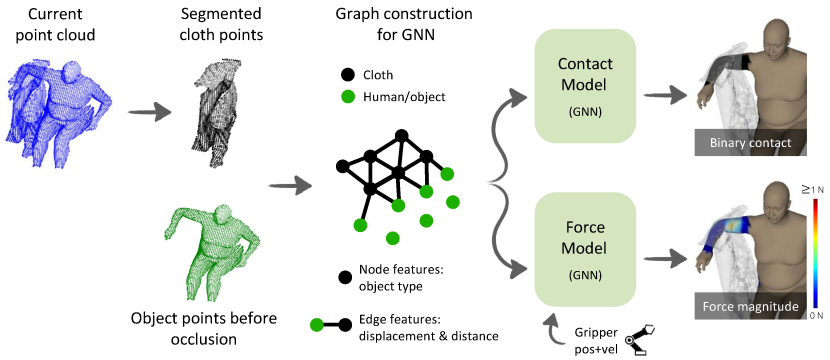

Fig. 2. An overview of the proposed visual haptic reasoning model, which estimates the location and magnitude of applied normal forces when deformable cloth (hospital gown in this example) physically interacts with other objects (human body in this example). The contact prediction model predicts if a point on the human body is in contact with the gown, denoted as the black regions. The force prediction model predicts the magnitude of applied force. Both models take the segmented point cloud as the input, and the force model also takes as input the gripper kinematics. In this example, the haptic reasoning models are instantiated as GNNs, and the middle part shows the graph construction process for GNNs.

where $\boldsymbol{p}$ is the position of the particle, $\boldsymbol{n}$ is the normal for the contact plane, and $\boldsymbol{q}$ is the point that the particle should not penetrate into, e.g., the vertex of the object mesh. d is a threshold parameter that the particle should maintain from the object point. After the dynamics is jointly solved with the constraint, the contact normal force is then $\boldsymbol{f} = \lambda \boldsymbol{n}$, and the Lagrangian multiplier λ represents the contact normal force magnitude.

We now describe how to get the magnitude of the contact normal force f_i for each point on the object $\boldsymbol{p_i} \in \boldsymbol{P_O}$, which are used as training labels for the proposed visual haptic reasoning models. In simulation, contact points between cloth and an object may not directly align with object points $\boldsymbol{P}_O$ from the observed point cloud. Given a contact between the cloth and an object that occurs at position $\boldsymbol{x}_j^C \in \mathbb{R}^3$ $(j = 1,...,N)$ with normal force $\boldsymbol{f}_j^C \in \mathbb{R}^3$, we distribute the applied force to all nearby observed object points that are within a distance ε of the contact position. This distribution of the contact force is inversely proportionate to the distance between $\boldsymbol{x}_j^C$ and $\boldsymbol{p}_i$. Formally, let $\boldsymbol{x}_i \in \mathbb{R}^3$ denotes the position of point $\boldsymbol{p}_i \in \boldsymbol{P}_O$, the contact normal force magnitude $F_j^C = ||\boldsymbol{f}_j^C||_2$ is distributed to a point $\boldsymbol{p}_i$ as follows:

$$F_{j\to i} = F_j \cdot \frac{w_i}{\sum_{k=1}^{H\times W} w_k}, \qquad w_i = \begin{cases} \frac{1}{||\boldsymbol{x}_j^C - \boldsymbol{x}_i||^{\xi}} & \text{if} ||\boldsymbol{x}_j^C - \boldsymbol{x}_i||_2 < \varepsilon \\ 0 & \textit{Otherwise} \end{cases}, \tag{1}$$

where $\xi > 0$ controls the smoothness of the distribution, which we set to be 0.5. The contact normal force magnitude f_i on point $\boldsymbol{p}_i$ is then the sum of all distributed forces: $f_i = \sum_{j=1}^{N} F_{j\to i}$. Our goal is to learn a model $h(\boldsymbol{P})$ that takes as input the point cloud $\boldsymbol{P}$, and predicts the force magnitude f_i for every point $\boldsymbol{p}_i \in \boldsymbol{P}_O$ on the object.

V. Visual Haptic Reasoning Model

A. Method Overview

An overview of the visual haptic reasoning framework is shown in Fig. 2. As the bottom branch shows, we train a force prediction model h_{force}, which takes the point cloud $\boldsymbol{P}$ (consisting of the object points $\boldsymbol{P}_O$ and cloth points $\boldsymbol{P}_C$), and the robot gripper kinematics as input, and predicts the magnitude of the contact force f_i at each point $\boldsymbol{p_i} \in \boldsymbol{P_O}$. Since the force prediction model is performing regression, it is common to predict small non-zero forces at many points on an object that are not in contact with cloth. To address this issue, we further introduce a contact prediction model $h_{contact}$, shown in the top branch of Fig. 2, to predict if a point $\boldsymbol{p} \in \boldsymbol{P}_O$ is having contact with cloth during manipulation, which removes the need for choosing a force threshold for determining contact at inference time. For training the contact model, a point $\boldsymbol{p}$ has a label of 1 if f_i is greater than 0, and a label of 0 otherwise.

The force prediction model h_{force} and the contact prediction model $h_{contact}$ can be instantiated using different neural network architectures (Fig. 2 shows a GNN instantiation). Several learning methods have been designed specifically for modeling point cloud data [11], [40]. In this paper we explore two distinct architectures for the haptic reasoning models, PointNet++ [11] and graph neural networks (GNN) [22], [41]. In the following subsections, we describe in detail how we instantiate the force and contact prediction models as PointNet++ or GNN.

B. Point Cloud Based Input Representation

We first describe how we construct the input to the PointNet++ and GNN haptic reasoning models based on the point cloud $\boldsymbol{P}$, which consists of the segmented cloth points $\boldsymbol{P}_C$, and object points $\boldsymbol{P}_O$ obtained before visual occlusion as described in Section V-B. We first filter the point cloud with a voxel grid filter by overlaying a 3D voxel grid over the observed point cloud and take the centroid of the points inside each voxel. This voxelization step is done both in simulation and in the real world, which makes the model agnostic to the density of the observed point cloud, and more robust to sim2real transfer. Then, given the voxelized point cloud $\boldsymbol{P}$ (we overload the notation $\boldsymbol{P}$ to refer to the voxelized point cloud in the rest of the paper), we remove object points from $\boldsymbol{P}_O$ that are sufficiently far from the cloth. We define this new point cloud $\tilde{\boldsymbol{P}}$ as:

$$\begin{aligned} \tilde{\boldsymbol{P}} &= \tilde{\boldsymbol{P}}_O \cup \{\boldsymbol{p}_{gripper}\} \cup \boldsymbol{P}_C, \\ \tilde{\boldsymbol{P}}_O &= \{\boldsymbol{p}_i | (\exists \boldsymbol{p}_i \in \boldsymbol{P}_O)(\exists \boldsymbol{p}_j \in \boldsymbol{P}_C)[||\boldsymbol{x}_i - \boldsymbol{x}_j||_2 < \tau]\}, \end{aligned} \tag{2}$$

where $\boldsymbol{x}_i$ represents the position of point $\boldsymbol{p}_i$, and $\tilde{\boldsymbol{P}}_O$ denotes object points that are within a distance τ of some cloth point. $\boldsymbol{p}_{gripper}$ is a single point at the position of the robot gripper.

$\tilde{\boldsymbol{P}}$ can be directly used as input to a PointNet++ model. To form a GNN model, we build a graph $G = \langle V, E \rangle$ from the point cloud, where V are the nodes and E are the edges of the graph. Fig. 2 visualizes the graph construction process.

The nodes V of the graph simply consist of all the points in $\tilde{\boldsymbol{P}}$. For edges, we connect an edge e_{jk} between a node j and node k when the following criteria are satisfied: 1) the distance between nodes is below a threshold α, i.e. $||\boldsymbol{x}_j - \boldsymbol{x}_k||_2 < \alpha$, where $\boldsymbol{x}_j$ denotes the position of node j, and 2) at least one node is a point on the cloth. Intuitively, message passing along the edges of the GNN can simulate the propagation of force from the gripper to the cloth, within the cloth itself, and from the cloth to the object.

C. Force Prediction Model

We describe two different instantiations for the force prediction model h_{force}, which is capable of predicting the forces that cloth applies onto objects in contact. The first instantiation uses a PointNet++, which can be directly applied to the cropped point cloud $\tilde{\boldsymbol{P}}$. The input includes the position $\boldsymbol{x}$ of each point $\boldsymbol{p} \in \tilde{\boldsymbol{P}}$ and a feature vector associated with each point $\boldsymbol{p}$. The feature vector consists of robot gripper velocity $\boldsymbol{v}$ and a 3-dimensional one-hot encoding vector $[\mathbf{1}_{object}(\boldsymbol{p}), \mathbf{1}_{cloth}(\boldsymbol{p}), \mathbf{1}_{gripper}(\boldsymbol{p})]$ indicating the type of the point (where $\mathbf{1}_y(\boldsymbol{p})$ is 1 if $\boldsymbol{p}$ belongs to y and 0 otherwise). We normalize the positions of the point to be zero-mean so that the model is invariant to translations of the point cloud.

To model force predictions with a GNN, the input to the force model is the graph built from the point cloud, as described in Section V-B. We use the GNS architecture [22], [41] for the GNN model. The features for each node in the graph include the one-hot encoding vector $[\mathbf{1}_{object}(\boldsymbol{p}), \mathbf{1}_{cloth}(\boldsymbol{p}), \mathbf{1}_{gripper}(\boldsymbol{p})]$ and the gripper velocity $\boldsymbol{v}$. The edge feature of an edge connecting node i and node j includes the distance vector $\boldsymbol{x}_i - \boldsymbol{x}_j$ and its L2 norm $||\boldsymbol{x}_i - \boldsymbol{x}_j||_2$. By using only relative distance in the edge features, the GNN model is also invariant to translations of the point cloud in Cartesian space. We do not include positions in the node features since the edge features are sufficient to capture the relative relationship between nodes.

As described in Section V-A, the output of the force prediction model h_{force} is the estimated magnitude of the contact normal force $\hat{f}_i$ for every point $\boldsymbol{p_i} \in \tilde{\boldsymbol{P}}_{\boldsymbol{O}}$. We train the force model using a Mean Squared Error loss between the predicted contact force $\hat{f}_i$ and the ground-truth contact force f_i, which can be obtained in simulation and computed using Eq (1). We mask the loss to be computed only on points that are having contact during the interaction, i.e., for points whose contact force f_i is greater than 0.

D. Contact Prediction Model

When training both PointNet++ and GNN models for contact prediction, we remove the gripper point $\boldsymbol{p}_{gripper}$ from the point cloud and reduce the point/node features to include only a 1-dim one-hot encoding $\mathbf{I}_{object}(p)$. A PointNet++ for contact prediction still includes normalized point positions as part of its input, and a GNN for contact prediction uses the same edge features as presented in Section V-C.

The output of the contact model $h_{contact}$ is the probability m_i of each point $\boldsymbol{p}_i \in \tilde{\boldsymbol{P}}_{\boldsymbol{O}}$ being in contact with the cloth. We train it with a Binary Classification Loss, where the ground-truth contact information can be obtained in simulation. For each point $\boldsymbol{p}_i \in \tilde{\boldsymbol{P}}_O$, the ground-truth label is 1 if it has non-zero contact force f_i (computed using Eq (1)), otherwise the label is 0. At test time, we combine the predictions of the contact and the force prediction model to compute the final contact normal force magnitude of a point $\boldsymbol{p}_i$:

$$f_i^{pred} = \begin{cases} \hat{f}_i & \text{if } m_i > \beta \\ 0 & \text{Otherwise} \end{cases}, \tag{3}$$

where β is a decision threshold.

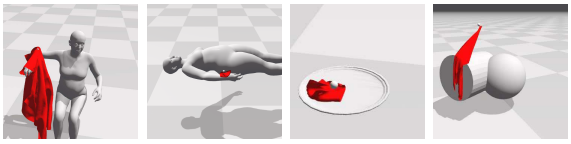

Fig. 3. Visuals of the 4 simulation tasks. From left to right: assistive dressing, assistive bathing, dish washing, and primitive shapes.

VI. Tasks and Dataset Collection

As described in Sec. IV, we use the NVIDIA FleX wrapped in SoftGym [42] as our physics simulation. As shown in Fig. 3, we build three representative robotic assistive and manipulation tasks that involve physical interactions between cloth and other objects: 1) Assistive dressing, where a robot must dress a hospital gown onto a person's right arm; 2) Assistive bathing, where the robot manipulates a washcloth to clean a person's arm; 3) Dish washing, where the robot uses a washcloth to clean the surface of a dish. To further evaluate visual haptic reasoning between cloth interacting with arbitrary rigid objects, we build a task where the robot pulls a rectangular towel over random combinations of objects with primitive shapes, including cylinders, cubes and spheres (referred to as the primitive shape task). In all tasks, we use a point grasp (visualized as a small white sphere), and the grasping is simulated by creating rigid anchors between the point grasp and the closest particles on the cloth.

We add the following variations for these tasks. For assistive dressing and bathing, we use the SMPL-X [43] model to generate 100 human body meshes varying in body shape and size. For dish washing, we randomly select 3 plates from the ShapeNet dataset [44]. For primitive shape, we vary the number of primitive shapes between 1 to 3, use a random selection of chosen shapes (cylinder, sphere, or cube), and vary the size and pose of each shape. For all tasks, we generate the gripper movement trajectories by linearly interpolating between some way points, where the way points are chosen differently for each task. For the dressing and bathing task, the waypoints consist of the fingertip, wrist, elbow, and shoulder of the human arm, with random deviation uniformly sampled from $[-5, 5]$cm added to these waypoints. For dish washing, we randomly sample way points on the surface of the plate. For primitive shape, we randomly sample waypoints on the surface contour of the combination of the primitive objects. For all tasks, we vary the gripper movement velocity across different trajectories. Depending on the gripper velocity and the waypoint locations, each simulated cloth manipulation trajectory can have between 200 to 800 time steps, where at each time step we store the point cloud, gripper velocity, and ground-truth contact normal forces between the cloth and the object as a data point for training. For assistive dressing, assistive bathing, and dish washing, we collect 600, 60, and 30 trajectories for training, validation and test, while due to the large task variation of primitive shape task, we collect 1800, 200, and 90 trajectories for training, validation, and test. The exact number of data

	Assistive Dressing	Assistive Bating	Dish Washing	Primitive Shapes
Training size	81379	187568	258624	624724
Validation size	7948	17978	24003	69413
Test size	3977	8989	11853	33889
average # cloth points	2361	79	76	353
average # object points	267	114	130	370

TABLE I
STATISTICS OF THE COLLECTED DATASET.

points for the collected datasets is summarized in Table I. The average number of object points and cloth points that will be inputted to our model, i.e., size of $\tilde{\boldsymbol{P}}_O$ and $\boldsymbol{P}_C$, are also reported in Table I.

VII. EXPERIMENTAL RESULTS

A. Evaluation Metrics and Baselines

For evaluation metrics, we use Mean Absolute Error (MAE) in Newtons for force prediction in relation to the ground-truth contact forces, and F1 score for contact prediction. For all learning methods, we search over the contact prediction decision threshold (β in Eq. (3)) on the validation dataset, and use the threshold that produces the highest F1 score.

We compare the proposed GNN and PointNet++ learning methods with the following baselines: *MLP*, where we trained two separate Multilayer Perceptrons that take as input a vectorized point cloud, one for contact prediction and one for force prediction. As the number of points in the point cloud can vary across tasks and trajectories, we take the maximum number of points as the fixed input dimension and pad the observation with 0 when it has fewer dimensions. *Constant Force Prediction*, a force prediction baseline which predicts a constant force for every point in contact. The constant force is chosen to be the median force on the training dataset which gives the lowest MAE on the training dataset. *Neighborhood Contact Prediction*, a contact prediction baseline that predicts a point on the object to be in contact if its distance to the closest point in the cloth is below a threshold. We perform a grid search over the threshold on the training dataset and evaluate with the one that results in the best F1 score.

For GNN, PointNet++, and MLP, we train two variants: the first is the task-specific model that is trained with data only from a single task, and the second is the task-agnostic model that is trained with data from all three assistive dressing, assistive bathing, and dish washing tasks. We hold out the primitive shape task and use it to evaluate the generalization performance of our task-agnostic models. In all evaluation tables, we use the suffix '-S' to denote the task-specific models, the suffix '-A' to denote the task-agnostic models, and we use bold text to denote the best result, and underlined text to denote the second best result.

B. Implementation Details

For PointNet++, we use the standard segmentation type of architecture in the original paper [11]. For GNN, we use the standard GNS [41] architecture. More details about the network architectures can be found on our project website. For the MLP baseline, we use a MLP of $[1024, 512, 256, 512, 1024]$ neurons and ReLU activation. We train the PointNet++, GNN and MLP models using the Adam [45] optimizer, with a learning rate of 0.0001 and batch size of 8. We train all models until they converge on the training dataset, and pick the model that has the lowest loss on the validation dataset for evaluating on the test dataset. For NVIDIA FleX simulator, the particle radius in the simulator is 0.625cm. Due to different sizes of cloth used in different tasks, we tune the stiffness of the stretch, bending, and shear constraints to ensure stable simulation behaviours of cloth for different tasks. The stiffness of these constraints are set to be $[1.7, 1.7, 1.7]$ for assistive dressing, and $[1.0, 0.9, 0.8]$ for the other three tasks. We set the threshold for contact (d in Sec. IV) to be 5mm. For hyper-parameters described in Sec.V, the value of ε used in Eq.(1) is set to be 3.12cm, the voxel size we used to voxelize the point cloud is 1.56cm, the value of τ in Eq.(2) is 6.25cm, and the value of α for determining the edge connection for GNN is 3.75cm.

Method \ Task	Assistive Dressing	Assistive Bating	Dish Washing	Primitive Shapes
GNN-S	0.200	**0.028**	<u>0.064</u>	**0.931**
GNN-A	0.192	0.035	**0.063**	1.460
PointNet++-S	**0.188**	<u>0.029</u>	0.066	<u>1.073</u>
PointNet++-A	<u>0.189</u>	0.034	0.065	1.398
MLP-S	0.640	0.057	0.134	1.503
MLP-A	0.696	0.057	0.122	1.502
Constant Force	0.555	0.059	0.118	1.367

TABLE II
FORCE PREDICTION MAE (IN NEWTONS) ON TEST DATASET.

C. Simulation Results

1) Force Prediction Result: The force prediction MAE of all methods over all tasks in simulation are shown in Table II. As shown in the first 3 columns, for assistive dressing, assistive bathing, and dish washing, for either task-agnositic or task-specific models, both GNN and PointNet++ models achieved significantly lower error than the constant force prediction baseline and the MLP baseline. Both GNN and PointNet++ models achieved similar performance overall, with GNN performing slightly better on bathing and dish washing, while PointNet++ achieved lower error for the assistive dressing task. Interestingly, for both GNN and PointNet++, the task-agnostic models achieved similar prediction errors compared with the task-specific models, indicating we can train a single model on multiple tasks instead of one for each task. When compared to task-specific models trained on primitive shapes, we observe from the 4^{th} column of Table II that task-agnostic force prediction models do not generalize as well to the primitive shape task, due to the large distribution shift. In addition, a task-specific GNN performs better than task-specific PointNet++ for inferring applied forces during the primitive shapes task, indicating that GNN architectures may be better suited for tasks with very large variations. We also report the percentage of error, i.e., the ratio between the force prediction MAE and the mean value of the ground-truth force of the dataset, for the best methods on each dataset. For assistive dressing, the PointNet++-S has an error percentage of 33.1%, for assistive bathing, the GNN-S has an error percentage of 34.0%, for dish washing, the GNN-A model has an error percentage of 43.2%, and for primitive shapes, the GNN-S model has an error percentage of 68% due to the large variation for this task. We later show in Fig. 6 that the force predictions are good enough for potential downstream tasks. On a NVIDIA 3090 GPU, the inference time for the GNN force model is $\sim$10ms, and the inference time for the PointNet++ force model is $\sim$20ms.

2) Contact Prediction Result: Table III shows the contact prediction F1 score of different methods on all tasks in

Algorithm \ Task	Assistive Dressing	Assistive Bating	Dish Washing	Primitive Shapes
GNN-S	0.888	0.910	0.955	0.930
GNN-A	0.890	0.912	0.954	0.872
PointNet++-S	**0.933**	**0.953**	**0.956**	**0.949**
PointNet++-A	0.930	0.946	0.948	0.866
MLP-S	0.790	0.748	0.902	0.781
MLP-A	0.788	0.728	0.888	—
Neighborhood	0.740	0.844	0.873	0.843

TABLE III
CONTACT PREDICTION F1 ON TEST DATASET.

Method \ Task	Assistive Dressing	Assistive Bating	Dish Washing	Primitive Shapes
GNN-S	0.200	**0.028**	0.064	**0.931**
GNN-S-particle	**0.160**	**0.026**	0.059	**0.622**
PointNet++-S	**0.188**	0.029	0.066	1.073
PointNet++-S-particle	0.193	0.029	0.067	0.734
GNN-A	0.192	0.035	**0.063**	1.460
GNN-A-particle	0.178	0.030	**0.055**	1.444
PointNet++-A	0.189	0.034	0.065	1.398
PointNet++-A-particle	0.194	0.034	0.034	1.431

TABLE IV
FORCE PREDICTION MAE: POINT CLOUD VS. CLOTH PARTICLES.

simulation. For all three assistive tasks shown in the first 3 columns, both GNN and PointNet++ achieved substantially higher F1 scores than the Neighborhood Contact Prediction and the MLP baselines. For contact prediction, PointNet++ is consistently better than GNN, for both task-agnostic and task-specific models. We also observe similar F1 scores for both task-agnostic and task-specific contact prediction models, indicating that we are able to train a single model on multiple tasks. In contrast to force prediction, the task-agnostic GNN and PointNet++ models generalized well to predicting cloth-object contact for the primitive shape task. The task-agnostic MLP model is unable to generalize due to its limiting way of handling point cloud data. The inference time for the GNN contact model is $\sim$10ms, and the inference time for the PointNet++ contact model is $\sim$20ms.

3) Visualizations of Contact and Force Predictions: Fig. 4 compares the predicted contact and force distributions from the task-agnostic PointNet++ model with the ground-truth for assistive dressing, assistive bathing, and dish washing tasks in simulation. As shown, the predicted contact areas and force magnitudes are qualitatively similar to ground-truth. In assistive dressing, the model accurately predicts large forces around the finger, elbow, and shoulder when the gown gets caught at those regions, and in bathing and dish washing, the model accurately predicts a spherical shape force distribution that decays from the center of the contact. More visuals on varying trajectories, human body shapes, and plate sizes can be found on the project webpage.

4) Comparison to Using Ground-truth Cloth Particles: We investigate how haptic reasoning models perform when we replace the cloth point cloud with ground-truth cloth particles in FleX simulation. Intuitively, since the cloth particles perfectly represent the underlying cloth mesh and its deformation, we would expect using particles to result in lower force prediction error and higher contact prediction accuracy. The object is still represented using point cloud.

Table IV shows the force prediction result with particles, and Table V shows the contact prediction result. Interestingly, we find that for both force and contact prediction, using particles with GNN models resulted in higher performance, yet using ground-truth particles did not improve performance

Method \ Task2	Assistive Dressing	Assistive Bating	Dish Washing	Primitive Shapes
GNN-S	0.888	0.910	0.955	0.930
GNN-S w/ particles	0.944	0.961	**0.970**	**0.973**
PointNet++-S	0.933	0.953	0.956	0.949
PointNet++-S w/ particles	0.932	0.950	0.954	0.961
GNN-A	0.890	0.912	0.954	0.872
GNN-A w/ particles	**0.945**	**0.962**	**0.970**	0.898
PointNet++-A	0.930	0.946	0.948	0.866
PointNet++-A w/ particles	0.932	0.950	0.954	0.893

TABLE V
CONTACT PREDICTION F1: POINT CLOUD VS. CLOTH PARTICLES.

with PointNet++ models. We attribute this to the message passing scheme in GNN, which may leverage the particle representations better compared with the abstraction layers in PointNet++. We also note that for both contact and force prediction, among all compared methods and for all tasks, the best-performing method is always a GNN with access to ground-truth cloth particles. This finding suggests an interesting direction for future work in tracking the underlying mesh particles of deformable cloth in the real world.

5) Sensitivity to Noise in Point Cloud: Since point clouds obtained in the real world are usually noisy, we test the sensitivity of our method to noise in the point cloud in simulation. We manually inject different levels of noise to the point cloud positions, which are modeled as Gaussian distributions with different standard deviations (Std) [46]. We test the GNN-A model on the assistive dressing task. The force model is robust up 3mm of noise, where the MAE slightly increases from 0.189 to 0.253, while the contact model is robust up to 10mm of noise, with the contact F1 score drops slightly from 0.889 to 0.808. The project website details more experiments on the models' sensitivity to noise.

D. Real World Results

1) Experimental Setup: We evaluated our model trained purely in simulation on the same three tasks in the real world with a Stretch RE-1 mobile manipulator. For assistive bathing, we use a medical manikin lying on a hospital bed for the human body. For assistive dressing, we use a silicone torso and arm model that is posed similar to human models in simulation. For dish washing, we pick a common dinner plate. Fig. 5 visualizes these three tasks. We use the same type of control trajectories as we used in simulation. An Intel D435i camera is used to capture a depth image of the scene, and we use color thresholding on the RGB image to segment the cloth and the object. Due to the challenge of sensorizing these environments with high-resolution force sensing arrays, we provide qualitative results of the contact and force predicted by our trained models in these three tasks. We conduct an additional experiment to illustrate the accuracy of the trained models. As shown in Fig. 6, we command the Stretch RE-1 robot to clean the blue powder on a black plate, and we use the trained model to predict which regions are cleaned after the robot action. A point on the plate is assumed to be cleaned if the predicted force magnitude is above a chosen threshold. Such predictions can be potentially used for downstream control tasks to clean specific area of the plate.

2) Qualitative Results: As shown in Fig. 5, the various haptic reasoning models trained entirely in simulation produce qualitatively reasonable contact and force distribution predictions for all three tasks. As the gripper and washcloth move over the edge of the plate during dish washing, our trained

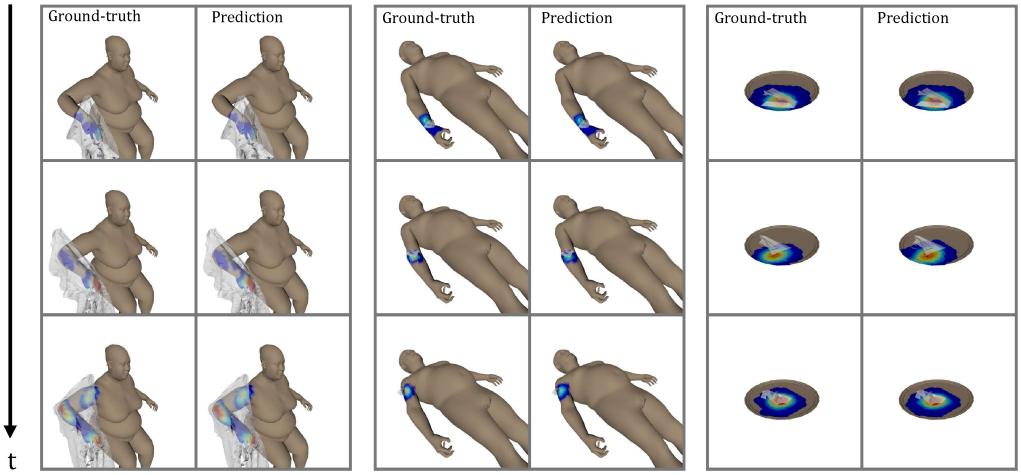

Fig. 4. Visuals of the force and contact predictions of the task-agnostic PointNet++ model for the assistive dressing, assistive bathing, and dish washing tasks in simulation, on the test dataset. For each task, the left column is the ground-truth and the right column is the prediction. The color map is scaled differently for each task due to different magnitudes of forces, but the same color map is used to visualize the ground-truth and predicted forces within each task. Better viewed zoomed-in.

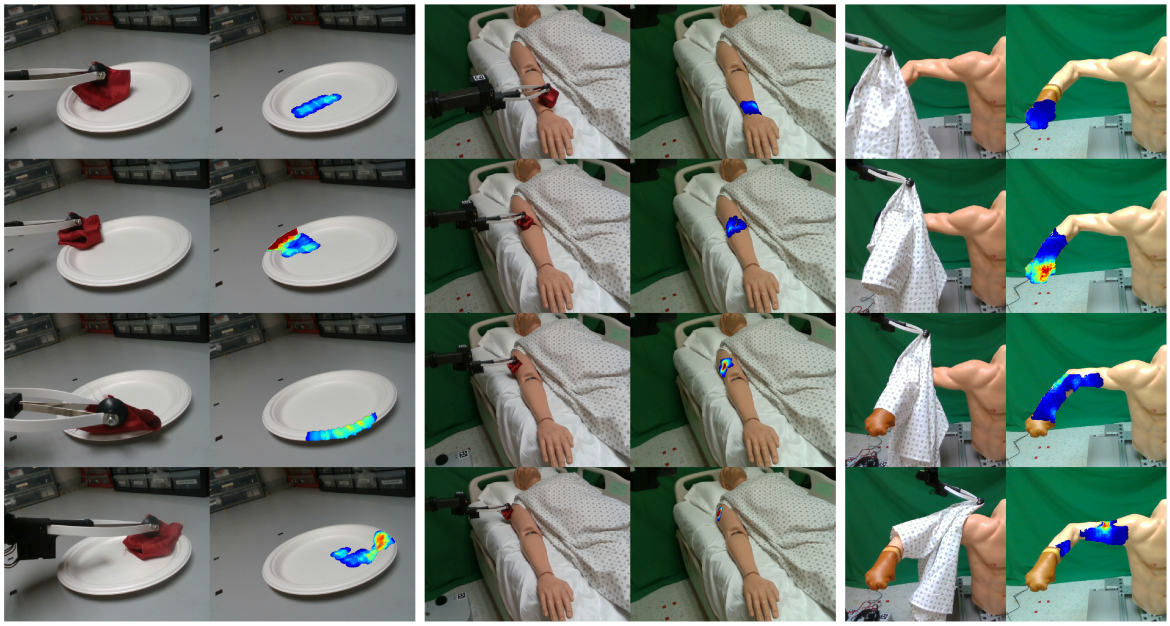

Fig. 5. Qualitative visualizations of the force and contact predictions of the trained visual haptic reasoning models for real-world tasks. Left: GNN-A for dish washing; Middle: PointNet++-A for bathing a manikin; Right: PointNet++-S for dressing a gown on a silicone arm. All models are trained entirely in simulation. Better viewed zoomed-in. The colors are scaled differently for each task to better show different magnitudes of forces in different tasks.

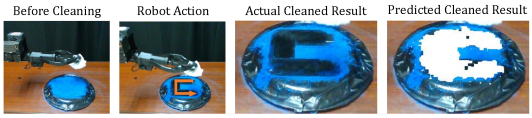

Fig. 6. Comparison of the predicted cleaned area (the white regions in the 4th plot) of the GNN-S model and the actual cleaned area (the 3rd plot) after the robot wipes the blue powder on the black plate with a wash cloth.

haptic reasoning models (GNN-A) predict a visually accurately force and contact distribution, with greater forces applied on the edges of the plate. For dressing assistance, we observe the contact model (PointNet++-S) predicts sizable regions of contact between the garment and body that align with visible observations. We observe greater forces on the hand as the hospital gown sleeve is initially pulled onto the hand and briefly snags, and we observe larger force predictions near the shoulder as the robot begins to stretch and pull the garment over the shoulder. More trajectories and some failure cases are in the supplement video and on the project webpage. As Fig. 6 shows, the trained model gives qualitatively correct predictions on which regions of the plate were cleaned by the wash cloth, demonstrating the potential of using visual haptic reasoning models for downstream control tasks. All these results indicate that haptic reasoning can be reasonably transferred from simulation to the real world, and presents a promising direction for continued research in robotic caregiving and robotic cloth manipulation.

VIII. CONCLUSION

We introduced a formulation for robots to compute haptic reasoning during cloth manipulation. Haptic reasoning enables a robot the infer the distribution of applied forces as cloth interacts with other objects or people in the environment. We presented two distinct model representations, including GNN and PointNet++ implementations, that enable haptic reasoning using only point cloud and robot kinematic observations. We conducted quantitative analyses of model performance during robot-assisted dressing, bathing, and dish washing tasks in simulation and demonstrated generalization performance across

human body sizes and object shapes. We also transferred simulation-trained models to a real environment and evaluated visual haptic reasoning with a mobile manipulator performing physically assistive tasks with cloth.

ACKNOWLEDGMENT

The authors would like to thank Fukang Liu for helping build the silicone torso and arm model, Eliot Xing, Kavya Puthuveetil, Akhil Padmanabha, Xingyu Lin, and Daniel Seita for helping provide feedback on the paper.

REFERENCES

[1] Z. Erickson, H. M. Clever, G. Turk, C. K. Liu, and C. C. Kemp, "Deep haptic model predictive control for robot-assisted dressing," in *ICRA 2018*. IEEE, 2018, pp. 4437–4444.

[2] Z. Erickson, V. Gangaram, A. Kapusta, C. K. Liu, and C. C. Kemp, "Assistive gym: A physics simulation framework for assistive robotics," in *ICRA 2020*. IEEE, 2020, pp. 10 169–10 176.

[3] T. Weng, S. M. Bajracharya, Y. Wang, K. Agrawal, and D. Held, "Fabricflownet: Bimanual cloth manipulation with a flow-based policy," in *CoRL*. PMLR, 2022, pp. 192–202.

[4] R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, and K. Goldberg, "Visuospatial foresight for physical sequential fabric manipulation," *Autonomous Robots*, pp. 1–25, 2021.

[5] D. Seita, N. Jamali, M. Laskey, A. K. Tanwani, R. Berenstein, P. Baskaran, S. Iba, J. Canny, and K. Goldberg, "Deep transfer learning of pick points on fabric for robot bed-making," *arXiv:1809.09810*, 2018.

[6] K. Puthuveetil, C. C. Kemp, and Z. Erickson, "Bodies uncovered: Learning to manipulate real blankets around people via physics simulations," *IEEE RA-L*, vol. 7, no. 2, pp. 1984–1991, 2022.

[7] T.-H. Pham, A. Kheddar, A. Qammaz, and A. A. Argyros, "Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces," in *CVPR*, 2015, pp. 2810–2819.

[8] W. Hwang and S.-C. Lim, "Inferring interaction force from visual information without using physical force sensors," *Sensors*, vol. 17, no. 11, p. 2455, 2017.

[9] K. Ehsani, S. Tulsiani, S. Gupta, A. Farhadi, and A. Gupta, "Use the force, luke! learning to predict physical forces by simulating effects," in *CVPR*, 2020, pp. 224–233.

[10] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, *et al.*, "Relational inductive biases, deep learning, and graph networks," *arXiv:1806.01261*, 2018.

[11] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[12] F. Zhang, A. Cully, and Y. Demiris, "Probabilistic real-time user posture tracking for personalized robot-assisted dressing," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 873–888, 2019.

[13] Z. Erickson, H. M. Clever, V. Gangaram, E. Xing, G. Turk, C. K. Liu, and C. C. Kemp, "Characterizing multidimensional capacitive servoing for physical human-robot interaction," *arXiv:2105.11582*, 2021.

[14] Y. Gao, H. J. Chang, and Y. Demiris, "Iterative path optimisation for personalised dressing assistance using vision and force information," in *2016 IEEE/RSJ IROS*. IEEE, 2016, pp. 4398–4403.

[15] C.-H. King, T. L. Chen, A. Jain, and C. C. Kemp, "Towards an assistive robot that autonomously performs bed baths for patient hygiene," in *2010 IEEE/RSJ IROS*. IEEE, 2010, pp. 319–324.

[16] R. Antonova, P. Shi, H. Yin, Z. Weng, and D. K. Jensfelt, "Dynamic environments with deformable objects," in *Thirty-fifth NeurIPS Datasets and Benchmarks Track (Round 2)*, 2021.

[17] F. Zhang and Y. Demiris, "Learning grasping points for garment manipulation in robot-assisted dressing," in *ICRA 2020*. IEEE, 2020.

[18] M. Cusumano-Towner, A. Singh, S. Miller, J. F. O'Brien, and P. Abbeel, "Bringing clothing into desired configurations with limited perception," in *ICRA 2011*. IEEE, 2011, pp. 3893–3900.

[19] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding," in *ICRA 2010*. IEEE, 2010.

[20] Z. Xu, C. Chi, B. Burchfiel, E. Cousineau, S. Feng, and S. Song, "Dextairity: Deformable manipulation can be a breeze," *arXiv preprint arXiv:2203.01197*, 2022.

[21] H. Ha and S. Song, "Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding," in *Conference on Robot Learning*. PMLR, 2022, pp. 24–33.

[22] X. Lin, Y. Wang, Z. Huang, and D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *CoRL*. PMLR, 2022, pp. 256–266.

[23] D. Triantafyllou and N. A. Aspragathos, "A vision system for the unfolding of highly non-rigid objects on a table by one manipulator," in *International Conference on Intelligent Robotics and Applications*. Springer, 2011, pp. 509–519.

[24] F. Osawa, H. Seki, and Y. Kamiya, "Unfolding of massive laundry and classification types by dual manipulator," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 2007.

[25] D. Seita, A. Ganapathi, R. Hoque, M. Hwang, E. Cen, A. K. Tanwani, A. Balakrishna, B. Thananjeyan, J. Ichnowski, N. Jamali, *et al.*, "Deep imitation learning of sequential fabric smoothing from an algorithmic supervisor," in *2020 IEEE/RSJ IROS*. IEEE, 2020, pp. 9651–9658.

[26] W. Yu, A. Kapusta, J. Tan, C. C. Kemp, G. Turk, and C. K. Liu, "Haptic simulation for robot-assisted dressing," in *ICRA 2017*, 2017.

[27] E. Magrini, F. Flacco, and A. De Luca, "Estimation of contact forces using a virtual force sensor," in *2014 IEEE/RSJ IROS*. IEEE, 2014.

[28] Y. Zhu, C. Jiang, Y. Zhao, D. Terzopoulos, and S.-C. Zhu, "Inferring forces and learning human utilities from videos," in *CVPR*, 2016, pp. 3823–3833.

[29] Z. Erickson, A. Clegg, W. Yu, G. Turk, C. K. Liu, and C. C. Kemp, "What does the person feel? learning to infer applied forces during robot-assisted dressing," in *ICRA 2017*. IEEE, 2017, pp. 6058–6065.

[30] S. Brahmbhatt, C. Ham, C. C. Kemp, and J. Hays, "Contactdb: Analyzing and predicting grasp contact via thermal imaging," in *CVPR*, 2019, pp. 8709–8719.

[31] N. Haouchine, W. Kuang, S. Cotin, and M. Yip, "Vision-based force feedback estimation for robot-assisted surgery using instrument-constrained biomechanical three-dimensional maps," *IEEE RA-L*, vol. 3, no. 3, pp. 2160–2165, 2018.

[32] A. Mendizabal, R. Sznitman, and S. Cotin, "Force classification during robotic interventions through simulation-trained neural networks," *International journal of computer assisted radiology and surgery*, vol. 14, no. 9, pp. 1601–1610, 2019.

[33] J. Liang and O. Kroemer, "Contact localization for robot arms in motion without torque sensing," in *ICRA 2021*. IEEE, 2021, pp. 6322–6328.

[34] H. M. Clever, Z. Erickson, A. Kapusta, G. Turk, K. Liu, and C. C. Kemp, "Bodies at rest: 3d human pose and shape estimation from a pressure image using synthetic data," in *CVPR*, 2020.

[35] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3d reconstruction in function space," in *CVPR*, 2019, pp. 4460–4470.

[36] C. Chi and S. Song, "Garmentnets: Category-level pose estimation for garments via canonical space shape completion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3324–3333.

[37] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff, "Position based dynamics," *Journal of Visual Communication and Image Representation*, vol. 18, no. 2, pp. 109–118, 2007.

[38] M. Macklin, M. Müller, N. Chentanez, and T.-Y. Kim, "Unified particle physics for real-time applications," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 1–12, 2014.

[39] M. Macklin, K. Erleben, M. Müller, N. Chentanez, S. Jeschke, and V. Makoviychuk, "Non-smooth newton methods for deformable multi-body dynamics," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 5, pp. 1–20, 2019.

[40] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *CVPR*, 2017, pp. 652–660.

[41] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia, "Learning to simulate complex physics with graph networks," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8459–8468.

[42] X. Lin, Y. Wang, J. Olkin, and D. Held, "Softgym: Benchmarking deep reinforcement learning for deformable object manipulation," in *Proceedings of (CoRL) CoRL*, 2020.

[43] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *CVPR*, 2019.

[44] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv:1512.03012*, 2015.

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[46] M. S. Ahn, H. Chae, D. Noh, H. Nam, and D. Hong, "Analysis and noise modeling of the intel realsense d435 for mobile robots," in *2019 16th International Conference on Ubiquitous Robots (UR)*. IEEE, 2019, pp. 707–711.