Learning Garment Manipulation Policies towards Robot-Assisted Dressing

Fan Zhang*, Yiannis Demiris

Personal Robotics Laboratory
Department of Electrical and Electronic Engineering
Imperial College London, U.K.
*Corresponding author. E-mail: f.zhang16@imperial.ac.uk

Assistive robots have the potential to support people with disabilities in a variety of activities of daily living (ADL), including dressing. In the present work, we develop a dressing pipeline intended for bedridden people who have completely lost their upper limb movement functionality, and experimentally validate it on a medical training manikin. The pipeline is comprised as follows: the robot would grasp a hospital gown hung naturally on a rail, fully unfold the gown, navigate around a bed, and lift up the user's arms in sequence to finally dress the user. To automate this pipeline, we address two fundamental challenges: (i) learning manipulation policies to bring the garment from an uncertain state into a configuration that facilitates robust dressing; (ii) transferring the deformable object manipulation policies learned in simulation to real world to leverage cost-effective data generation, which is sensitive to fabric physical properties. We tackle the first challenge by proposing an active pre-grasp manipulation approach that learns to isolate the garment grasping area prior to grasping. The approach combines prehensile (i. e., grasping) and non-

prehensile actions, and thus bypasses grasping-only behavioral uncertainties (e. g., multiple-layer garment grasping). We bridge the sim-to-real gap of deformable object policy transfer by approximating the simulator to real-world garment physics. A contrastive neural network is introduced to compare pairs of real and simulated garment observations, measure their physical similarity and account for simulator parameters inaccuracies. The proposed method enables a bi-manual robot to put back-opening hospital gowns onto a medical manikin with reliability greater than 90%.

Summary This work uses sim-to-real transfer to learn garment physics and manipulation for robot-assisted dressing.

Introduction

Dressing assistance is a basic assistive activity in the daily life of elderly people, and people who suffer from impairments. Studies indicate that although over 80% of people in skilled nursing facilities require assistance with dressing (*1*), there are growing concerns regarding the increased costs of daily care and lack of nursing staff (*2*). A cross-sectional study (*3*) of $14,500$ Medicare beneficiaries shows that 8.2% of those beneficiaries have reported difficulties with dressing. (*3*) also reports that of all activities of daily living, dressing has shown the highest burden on caregiving staff and the lowest use of assistive technologies. This work focuses on robot-assisted dressing, which is a growing area of research that has the potential to alleviate this problem.

Background and challenges

Recent state-of-the-art research has made great progress towards addressing the challenges of using a robot to perform dressing (*4–19*). This work focuses on developing a pipeline towards dressing bedridden people who have completely lost the ability to move their limbs, and validating the approach on a medical training manikin.

Dressing a bedridden person is an essential clinical skill in the Certified Nursing Assistant (CNA) Practice Test, which is a compulsory exam that defines training standards for nurse assistants who work in nursing homes or hospitals (*20*). For reference, public tutorial videos of dressing bedridden people in CNA tests can be found here (*21, 22*). In this work, we endeavor to equip a robot with the equivalent ability to provide dressing assistance in putting on back-opening hospital gowns for bedridden people in a manner that emulates a CNA. We use the same settings as in the CNA test, including back-opening gowns normally used in hospitals and a professional training manikin lying on a hospital bed, to simulate an adult bedridden person. The patient care manikin (weighting 18kg), equipped with natural movement of the joints, arms and legs for realistic positioning, represents a life-sized adult (174cm). Movie 1 shows the complete dressing sequence: the robot would grasp a hospital gown hung naturally on a rail, fully unfold the gown, navigate around a bed, lift up the manikin's arms in sequence to finally dress the manikin.

Several state-of-the-art surveys have identified open challenges on perceiving and handling deformable objects (*23–25*), two of which are faced in this work as well to endow robots with the ability to autonomously execute such a dressing pipeline: 1) manipulation of the garment, which has complex dynamical and high-dimensional states, to bring it from an uncertain state into a configuration that facilitates robust dressing. In our case, the proposed pipeline involves learning six grasping/manipulation policies in sequence to fulfill the dressing task (Fig. 1). Simulation is leveraged to learn garment grasping/manipulation policies, and thus reduce the

constraints of costly real data collection or robot experiments (*26–28*). However, this introduces a second challenge: 2) deformable object policies learned in simulations suffer from a lack of transferability onto the real world, especially in the physical domain.

For the first challenge of manipulating a garment that has large dimensional configuration space, most studies on robot-assisted dressing setup the initial robot configuration before dressing by manually attaching the garments on the robot end-effector (*11–14*). Recent work on grasping has formulated this problem as computing suitable grasping points on garments, either through learning grasping points in a supervised end-to-end manner (*29–33*), or using reinforcement learning to self-explore grasping points (*34, 35*). In these approaches, only prehensile action (i. e., grasping) is adopted prior to any garment manipulations, which could lead to issues including multiple-layer garment grasping. Unlike the successful garment hanging or folding tasks in the above research, such issues could induce the occlusions of the sleeve opening, and consequently cause garment dressing failures. Multi-step manipulation that combines both prehensile and non-prehensile (e. g., pushing) is an area of research that has received less scientific attention (*23*). Typical combinations applied on rigid objects include push-and-grasp (*36*) and slide-to-wall-and-grasp actions (*37*). Deploying manipulation-and-grasp policies is especially challenging for deformable objects because garment manipulation induces further deformations during the manipulation procedure. Solutions to this problem include garment grasp-and-fling actions for unfolding (*38*) and edge trace-slide-and-grasp strategies (*39, 40*).

Our second challenge, that of deformable object policy transfer, is an area of research that has been less explored (*41*). One solution of bridging the simulation and reality gap is to synthesize realistic-looking images with generative models or domain randomization aiming to use robust representations in the visual domain (*42–44*). However, visual appeal does not equate to physical realism as a visually realistic garment could behave in a physically unrealistic manner due to occasional object details deletion and addition (*45*). Therefore, (*23*) has pointed

out another potential solution, which is to accurately modeling garment states in simulation by identifying the underlying physics correspondence between simulated and real observations. Understanding of the underlying physics is pertinent to deformable object manipulation as their deformations are sensitive to the physical properties of the object itself (*46–55*). Recent work focuses on inferring garment physical attributes by either comparing handcrafted garment features (*56–59*), or directly from visual observations of object dynamic behaviors using deep learning method (*60–63*). However, both methods remain challenging as the physical models of garments tend to have high numbers of unknown parameters, and bear intricate coupling of intrinsic and extrinsic forces. Even fewer studies have explored sim-to-real transfer deformable object policies (*41, 64*). Instead of physics understanding, these approaches focus on the dynamics randomization, which could be biased by the expertise of the practitioner.

Contributions

To overcome the first challenge, we propose an active pre-grasp manipulation learning framework, which aims to isolate the garment grasping area prior to actual grasping. Different to most state-of-the-art work on deformable object grasping that only adopts prehensile action, our pre-grasp manipulation policy continuously learns to select among motion primitive behaviors that combines both non-prehensile and prehensile actions, including move-left-grasp, move-right-grasp and direct-grasp, as shown in Fig. 2. The approach involves interactively training pixel-wise neural networks that map from multi-view visual observations of the garment to affordance-based pre-grasp manipulation actions through robot exploration and exploitation. Our proposed approach bypasses uncertainties of the grasping-only behavior (e. g., multiple-layer garment grasping, sleeve opening occlusion), and thus facilitates robust dressing pipeline.

In regards to the second challenge, we propose to tailor the simulator to the real-world

garment physics by selecting more realistic simulator parameter values. Our solution is to compare pairs of real and simulated garment observations, learn their physical similarity and account for simulator parameters inaccuracies. We propose to learn the garment physical similarity metric using contrastive loss (*65*), which could efficiently map pairs of physically similar observations to nearby points, whereas dissimilar pairs are pushed apart in the embedding space. This loss leverages the easily-acquired simulated garment physics information to learn the physical similarity, without observing a real-world manifestation of the phenomenon. Then, we refine physics simulations from a single real-world observation towards maximizing the physical similarity between the real-world behavior and its simulated counterpart. Our experimental results will show that this method outperforms several state-of-the-art methods in terms of creating a realistic physical simulation that facilitates a more efficient sim-to-real garment grasp/manipulation policy transfer.

With the above proposed methods, we finally introduce a dressing pipeline for bedridden people. Each grasping/manipulation policy in Fig. 2 is learned in a sim-to-real manner: in the physics domain, physical attributes of the real gown have been learned through our proposed contrastive learning approach, and applied to the synthetic gown in simulation; while in the image domain, classic domain randomization is employed to change colors and textures of the garment, manikin, bed and rail when synthesizing data. The pipeline and the corresponding garment manipulation in each stage could be summarized as follows:

1. Stage A: the robot would grasp a gown (Fig. 1A) that is naturally hung on a rail with the hanging point randomly around the collar, while the gown is segmented out using a Mask R-CNN network (*66*);

2. Stage B: the robot would fully unfold the gown (Fig. 1B), while the manipulation policy is learned using our proposed active pre-grasp manipulation approach;

3. Stage C: the robot would navigate around the bed to grasp the garment (Fig. 1C), lift up and dress the user's arms in sequence, while the garment grasping policy is learned using supervised pixel-wise Convolutional Neural Networks (CNNs);

4. Stage D: the robot would spread the gown to cover user's upper body (Fig. 1D), while the garment segmentation is learned in the manner as Stage A.

The main contributions of this paper can be summarized as follows:

1. A robot-assisted dressing pipeline intended for bedridden people, which is comprised of a series of garment grasping and manipulation, robot navigation and user upper-body dressing, and validated on a medical training manikin.

2. An active pre-grasp manipulation approach that learns to isolate the garment grasping area prior to grasping to facilitate a more robust dressing.

3. A contrastive learning method to create a realistic physical simulation that facilitates a more efficient sim-to-real garment grasp/manipulation policy transfer.

Results

This section is divided into three main experiments. The first experiment evaluates the overall dressing pipeline on a medical manikin, followed by the second and third experiments respectively performing statistical comparisons of the proposed garment contrastive physics learning and pre-grasp manipulation methods against several ablations and state-of-art baselines.

Experiments on dressing a medical training manikin

In this experiment, we evaluate the performance of the complete dressing pipeline on a medical manikin. Three back-opening hospital gowns are used in our experiments. For each gown, its

physical attributes are estimated using the proposed contrastive learning methods, and the pre-grasp manipulation policy (stage B) as well as other grasping policies (stage A, C and D) are trained in the sim-to-real manner.

First, we carry out ablation studies of each stage in the pipeline. To evaluate the performance of each stage independently, for each stage, we assume its previous stage has been completed. For example, in stage C of dressing, we manually put the garment flatly on the manikin to simulate that the stage B of garment unfolding has been achieved. 200 trials of each stage execution have been carried out independently. Table 1-second row presents the success rates of each stage.

Then, 200 trials of the complete pipeline are carried out. To finish the whole dressing pipeline, all intermediate stages have to be performed correctly to be considered as a success. For instance, if stage B fails, then stage C would not be executed. The experimental results show that 181 trials have been successful, yielding a overall success rate of 90.5%. Here, we also present the actual success rate of each stage in this overall experiment, which is measured as (successful trials / totally executed trials of each stage, specifically $198/200$ of stage A, $192/198$ of stage B, $183/192$ of stage C and $181/183$ of stage D), as shown in Table 1-third row.

The achieved overall success rate is lower than the performance of each stage in the pipeline because failures can occur in various stages and affect the complete process. These failures include but not limited to: 1) stage B: inaccurate estimation of manipulation actions in some challenging scenarios, for example, the manipulation area is partially or completely occluded by the other parts of the gown, and 2) stage C: the sleeve getting caught by the manikin's inflexible hand. Details of failure analysis are presented in the supplementary materials.

Patient safety is one of the priorities that have been taken into account during the design of our dressing pipeline. We have taken into account four aspects of user's safety: 1) The robot is operated in a moderate speed, taking into account a tradeoff between the safety consideration

and the user's comfort. Thus most of experimental time is spent by robot physical movements, but not by perception or reasoning. 2) We design the motion of the robot lifting the manikin's arm to be perpendicular to the real-time tracked arm posture, which is executed in a slow speed for 5 seconds to ensure that no large force is applied to the user's arm and shoulder joint. 3) A moderate maximum grip force of the gripper is set to ensure that the gripper is firm enough to hold and lift the user's arm, without applying large force on the user. 4) The robot dressing motion planning is realized using the hierarchical controller proposed in our previous work (*19*). This controller has described the dressing task as two hierarchical subtasks. The high-priority subtask adapts robot motions to minimize the force applied between the user and the robot. The low-priority subtask completes the real-time updated dressing trajectory without affecting user's comfort and safety.

Contrastive learning real garment physics evaluation

Fig. 3-top illustrates the framework of the proposed garment physics learning method in two phases. In the first phase, depth video clips of garment dropping, generated in the Blender simulation engine with various physical properties, are used as visual observations of garment dynamic behaviors. These videos are mapped onto the embedding space with contrastive loss to learn garment physical similarity. In the second phase, the parameter error is learned from the embedded representations in a supervised manner. The real-world garment observations are then mapped onto the same embedding space for physical properties estimation.

Ablation studies of garment physical similarity learning and properties estimation

We first investigate the performance of physical similarity learning in phase one. It has been observed in our preliminary experiments that some properties have larger impact on garment deformation in this particular physical phenomenon of garment dropping. Therefore, six phys-

ical properties are selected as the input of the simulation in our experiments, as presented in Table 2. The simulation properties dataset is described as:

$$\mathbf{\Theta}_s \equiv \left\{ \boldsymbol{\theta}_s^i = \left[P_1^i, \ldots, P_6^i \right] \right\}^{i=1,2,\cdots,N}$$

where $[P_1, \ldots, P_6]$ denotes six physical properties, $\boldsymbol{\theta}_s$ is one set of properties, and N denotes the number of sets.

The contrastive loss consists of two "opposing forces": positive and negative (*67*). As similar properties always lead to similar garment behaviors in the simulation, we define that video clips generated from similar physical properties are considered as positive pairs, while negative pairs have dissimilar physical parameters. In total, 100 classes of simulation properties are generated as isotropic Gaussian blobs with 300 sets of properties in each class. Properties in the same class are considered to be similar, whereas dissimilar properties are from different classes. We use these properties to generate $30,000$ video clips in Blender simulation engine, which are split with a 60%-20%-20% percentage for training, validation and testing respectively. All properties are generated between $[0, 1]$ and mapped to the actual value afterwards. The standard deviation of the clusters is set as 0.005 for each set of parameters to make sure parameters in the same class are closer to each other than to other classes. Fig. 4A-left visualizes the generated simulation properties using 2D t-SNE (*68*).

Fig. 4A-right shows the 2D t-SNE visualization of a subset of the learned embedding space using the training dataset. It can be observed that the learned embeddings with the same color (i. e., video clips sharing similar physical parameters from the same class) tends to be closer to each other. This demonstrates that the proposed method with contrastive loss is capable of efficiently mapping physically similar garment data near to each other, while dissimilar data being pushed further. We then evaluate the performance of parameter residual estimation in the second phase. Mean absolute percentage error (MAPE) is adopted as the metric due to the

different value scale of each physical property, which is described as:

$$M_{eg} = \text{dis}(\boldsymbol{\theta}_e, \boldsymbol{\theta}_g) = \frac{1}{N}\sum_{k=1}^{N}\left|\frac{P_g^k - P_e^k}{P_g^k}\right|,$$

where $\boldsymbol{\theta}_g$ is the ground truth simulation parameters, and $\boldsymbol{\theta}_e$ denotes the estimated parameters. Our experiments on the testing dataset show that the parameter residual network achieves 1.57% median value of MAPE on $6,000$ synthetic testing dataset, which shows a high accuracy of physics inference.

Comparisons with baselines

Next, we benchmark the proposed method against five baselines, including: 1) B1: parameter identification using a classic supervised neural network (*69*); 2) B2: parameter identification with contrastive loss (*70*); 3) B3: parameter residual estimation without contrastive loss (*71*); 4) B4: manually tuned parameters (*26*); and 5) B5: randomized parameter (*41, 64*). For comparison purposes, the embedding functions are all formatted as the same network structures in the above baselines.

We first use only synthetic data to evaluate the performance of physics learning in this experiment. Fig. 4C presents the results of estimated physical attributes. Three conclusions can be drawn from this figure: 1) The proposed method achieves the smallest MAPE (1.57%); 2) Baseline 3, which estimates physical parameter residual without contrastive loss, obtains 78.99% median value of MAPE on the synthetic testing dataset. Instead of training the network with contrastive loss in two phases, baseline 3 uses the classic Mean Squared Error loss and trains the network in one phase. This result can be explained by the effectiveness of contrastive loss, which maps physically similar examples to nearby points in the embedding space, and thus guarantees that the small parameter residual (residual network output) also corresponds to the embedded representations that are close to each other (residual network input). The embedding space in baseline 3 without contrastive loss may not capture such information; 3)

Baseline 2 obtains 13.88% median value of MAPE on the $6,000$ synthetic testing dataset. In baseline 2, instead of using a nearest-paired embedded sample to estimate parameter residual in phase two, it predicts the parameters directly from embedded representations. This is usually achieved by a supervised neural network or Bayesian optimization methods. This result reflects the nature of contrastive loss that it learns 'how similar' the input data are, instead of 'how different'. The contrastive loss function ignores the distances among the different negative classes and thus, does not assure the optimization among the different negative class embeddings (*72*). Therefore, using the combination of the test data and its nearest embedding could include more useful information for parameter residual estimation than using whole embedded representations. Similar approaches have also been adopted by state-of-the-art research (*73, 74*) to predict deformable objects states transitions by finding the close embeddings in the latent space that are learned in a contrastive manner, instead of the whole embedded data.

Real garment physics estimation analysis

A depth video clip of one real garment dropping has been collected, as shown in Fig. 4B. The garment physical parameters are respectively predicted using the proposed method and the five baselines described in last section. Fig. 4D visually shows a depth image of the real garment from one point of view, and the depth images of synthetic garments animated in the simulation engine using the estimated garment physics achieved by the proposed method and five baselines. It can be observed that our method simulates a more similar shape to the real garment than the five baselines. Fig. 4B shows the garment dropping snapshots from the real garment and the synthetic garment with our estimated physics, which also indicates identical deformation. As the ground truth parameters of the real garment are challenging to obtain, in the following experiments, we will quantify the performance of real garment physics learning using garment manipulation success rate on physical robots.

Garment pre-grasp manipulation evaluation

Fig. 3-bottom illustrates the framework of the proposed active pre-grasp manipulation learner. Multi-view RGB-D images of the hospital gown are fed into three fully convolutional networks to respectively infer pixel-wise affordance for three motion primitive behaviors. The robot action space is defined as a set of end-effector-driven motion primitives. Each pixel represents a different location on which to execute the primitive, and each camera view orientation defines the primitive orientation. The action with highest affordance is selected for robot execution with ϵ-greedy exploration strategy. Our system bootstraps its learning of the affordance function from demonstration data to guide and accelerate the agent towards good behaviors. The full learning process is: 1) the policy is pre-trained using simulation demonstration data and fine-tuned with trial and error in simulation; 2) based on the training in simulation, the policy is transferred to real world by training with real demonstration data and finally fine-tuned with trial and error with the real robot and garment.

Effect of active learning

We compare our method with three ablations: 1) A1: training using only real demonstration data; 2) A2: training using only real active trial; and 3) A3: training using only real demonstration data and active trial. For our method, $3,000$ sets of demonstration states ($2,000$ data in simulation, $1,000$ in real world, $3,000 * 6$ RGB-D images in total) have been collected by five human participants. For the ϵ-greedy exploration strategy, ϵ is initialized at 0.5 in simulation and annealed over time, while initialized at 0.1 in real world to highlight the immediate performance of sim-to-real policy transfer.

Fig. 5-top presents the training process of learning a manipulation action (Fig. 1B-1) on one hospital gown. The performance is measured by the pre-grasp success (reward=1) rate over the last $j = 200$ attempts. From the lines of A1-A3 ablations and the proposed method in this figure,

we could draw four conclusions: 1) The proposed method is capable of learning effective pre-grasping policies – achieving a success rate more than 92% in real-world experiments; 2) For ablation 1, we train the network using only real demonstration data and test on 1500 trials on the real robot. It achieves a success rate that is approximately 22.5% lower than ablation 3, which demonstrates that the active exploration steps enable the algorithm to explore other pre-grasp solutions beyond what it has learned from demonstrations; 3) Through the comparisons between ablation 2 and ablation 3, we can see that the demonstration data not only helps the algorithm learn faster (higher performance in the early training stage), but also helps the algorithm learn better (higher performance after active trial); 4) The comparisons between ablation 3 and the proposed method show that learned policy from simulation has been transferred to the real world, especially when real-world data is insufficient.

Comparisons with a grasping-only policy

We also investigate whether the proposed pre-grasp manipulation leads to more robust dressing than the grasping-only policy. For the latter method, we train a supervised CNN network to estimate the grasping point directly from each image with same amount of demonstration data as in our method. This method has also been adopted in (*27, 29*). We test both methods on the physical robot performing the stage B of garment unfolding and stage C manikin dressing in sequence. For each method, 50 replications of trials have been carried out. The results in Fig. 5-bottom show our proposed pre-grasp manipulation (represented as a star) significantly outperforms the grasping-only policy (green dot) in terms of dressing success rate. This result is expected as the grasping-only policy may successfully grasp the garment, but could cause multi-layer cloth grasping that occludes sleeve openings and thus yield dressing failures.

Sim-to-real pre-grasp manipulation analysis

In this experiment, we investigate how the estimated real garment physics affect garment unfolding performance. We deploy the estimated physical properties from our method and five baselines to train the garment pre-grasp manipulation policies respectively following the same procedure. From the lines of B1-B5 baselines and the proposed method in Fig. 5-top, we can see a drop of success rate at the beginning stage of real data training for each method. This result is expected as garment deformation is a complicated mechanism that is unlikely to be fully represented by six physical parameters. The internal garment modeling in the simulation engine could affect the behavior of deformation. We also need to consider the sim-to-real gap in the visual domain. However, the pre-grasping performance using the estimated parameters achieved by our approach still surpasses that of the other five baselines, which shows more gained knowledge from simulation has been transferred to the real world.

Sim-to-real robotic-dressing analysis

Lastly, we deploy the above learned policies on the physical robot to perform dressing. Specifically, the robot performs the stage B (unfolding) and C (dressing) of the pipeline in sequence. Three back-opening hospital gowns with similar style but different physical attributes are used in our experiments. We respectively estimate their physical properties using our method and five baselines, and train the corresponding pre-grasping policies. The inferred embeddings of the three gowns are visualized in Fig. 4A-right. For each learned policy, 50 trials of garment unfolding and dressing have been carried out. Fig. 5-bottom shows our method outperforms all other five baselines. Note here that the open-loop robot dressing motion is designed as the robot pulling the gown along the arm from the real-time tracked hand to elbow and shoulder positions in sequence. Thus the dressing performance is highly dependent on whether the garment is properly unfolded in stage B and if the grasping point is accurately estimated in stage C.

Discussion

This work has addressed necessary subtasks for a complete dressing pipeline intended for bedridden people and validated on a medical manikin. Experiments have shown high success rates with various hospital gowns considering the complexity of the pipeline. Since no *a priori* knowledge has been assumed on the dynamic models of the robot for our proposed dressing, it is envisaged that the method could be readily transferred to different dual-arm mobile robotic platforms.

In terms of dressing robustness, the presented garment pre-grasp manipulation substantially advances the grasping-only policy. Our method is based on a pixel-wise version of deep networks that combines active learning with affordance-based manipulation. The pixel-wise parameterization of both state and action enables convolutional features to be shared across locations and orientations instead of explicitly memorized by the network. Thus the policy could be more easily trained and extended to new state and action pairs. Such model-free pixel-wise parameterization also allows the proposed method to be easily generalized to different deformable objects manipulation tasks by only redefining motion primitives without changing the learning framework.

Our method for measuring intrinsic and extrinsic physical correspondence between real and simulated garment effectively reduces the sim-to-real gap of transferring deformable object policies in the physics domain. We have used a dropping motion to represent garment dynamic behaviors. It is envisaged that other dynamic observations, for instance wind blowing the cloth, might lead to similar physics estimation using the same approach. Another strength of this approach lies in modeling physical similarity in an intuitive way that is realistic enough to achieve high precision, while keeping tractability in mind. Thus, the approach has the potential to be generalized to different simulation engines and physical properties, even other sim-to-real

tasks dealing with soft objects, such as garment dragging and folding, fluid pouring or granular materials gathering.

Limitations and future work

We also see limitations and opportunities for future work. In our pre-grasp manipulation approach, we have shown that having human demonstration data is more effective on pre-training manipulation policies than using only trial and error data from the robot since the demonstration data contains significantly more diverse and successful pre-grasping examples. An interesting question for future work is to investigate how to significantly decrease the amount of required demonstration data with reinforcement learning, or even completely remove demonstration so that the robot could learn the pre-grasp manipulation policy with self-supervised learning.

For garment physics learning, our method leverages only simulation data to learn garment physical similarity, and thus requires only one-time training before perceiving real garments with similar styles but different physical attributes. An interesting question for future work could be rapidly learning a generic model to capture the morphological properties between different types of garments. We train the pre-grasp manipulation policy for each gown in simulation with its estimated physics and efficiently transfer the policy to the real world. In our future work, we would investigate whether other approaches (e.g., transfer learning) could generalize the learned physics and manipulation policies to held-out gowns.

As the procedures of the dressing pipeline are designed empirically in this work, one possible extension would be incorporating real-time reasoning approaches (e. g., high-level temporal logic language), performed concurrently with manipulation. The proposed pipeline comprises specific scenarios, including a hospital gown with short sleeves hung on a rail with the hanging point randomly around the collar of the gown. One interesting research direction would be investigating on relaxing these prerequisites, for instance, picking up different types of clothes

from a pile of crumpled garments.

We have used a life-sized, professional training manikin weighting 18kg to simulate bedridden people with weak or contracted arms, as in the Certified Nursing Assistant (CNA) Practice Test. In our future work, we intend to extend our experiments to include real end-users with physical impairments. Note here that most robotic arms (including Baxter robot) suffer from limited payload to lift up the real user's arm to dress the upper body or hold the real end-user's whole upper body to tie the hospital gown on the back. Thus we would require robot platforms with higher payload for real human's experiments. Additional safety measures would be required for experiments on end-users. For instance, we might consider adding foam on the robot gripper's surface or using soft grippers to lift the user's arm to ensure comfort and safety. We could also implement fail-safe strategies to recover the robot from failures like sleeves caught on the user's arm. A more complex pipeline could be explored considering potentially bent elbows of real patients.

Materials and Methods

Dressing pipeline learning details

We use a dual-arm robot (Baxter) equipped with a mobile base (Clearpath Ridgeback), two grippers (Robotiq 2F-85) and two RGB-D cameras (Realsense LiDAR L515) to execute the dressing pipeline. The medical manikin meets the international technical standards of medical equipment (*75*).

Stage A of grasping the garment on the rail

A map of the room is pre-built using the LiDAR for robot navigation (Ridgeback navigation stack (*76*)), while three waypoints are annotated near the garment rail and two sides of the bed respectively. In stage A, the robot navigates to the first waypoint, captures a RGB image,

segments out the gown using Mask R-CNN (*66*), and randomly selects a grasping point (Fig. 1A) near the hanging point on the segmented garment (between 70 and 80 pixels away from the hanging point).

Stage B of fully unfolding the garment

The second stage is to fully unfold the gown in the air. The robot navigates to the second waypoint on the map, estimates the manipulation actions (Fig. 1B) on the left and right side of the garment in sequence to unfold the garment. The two manipulation actions (3D positions, orientations and action primitives) are learned using our proposed sim-to-real active pre-grasp manipulation approach. These points are localized approximately at two corners of the garment collar respectively, which facilitates full unfolding. The unfolding reduces the number of possible configurations when the gown is picked up randomly from the rail. The robot then recognizes the correct side of the garment using a binary classification convolutional neural network, moves closer to the manikin and places the garment roughly flat on the hospital bed with the correct side facing up. Here the mobile robot moves closer to the bed without changing its orientation to place the gown flatly on the bed, which guarantees that the grasping points in stage C (Fig. 1C) are visible to the camera in most cases. The robot stops when the manikin's head is detected in the camera view to ensure the whole upper limb is within the robot manipulation workspace. The manikin's postures are real-time tracked using HRNet library in real time (*77*).

Stage C of dressing the user's arms

In this stage, the robot localizes the grasping point (Fig. 1C-1), grasps the garment with its right gripper, lifts up the user's arm with its left gripper, and pulls the hospital gown along the arm to finish dressing. Then the robot moves back to the second waypoint, navigates to the third waypoint in the map, slides closer to the user, detects the grasping point (Fig. 1C-2) and

performs the user's left arm dressing assistance in the same procedure.

Three key motions in this stage are garment grasping, user's arm lifting and dressing: 1) As the garment has been fully unfolded into a spread-out configuration, a grasping-only policy is robust enough in this stage for dressing the user. We use supervised pixel-wise CNN Networks to estimate the grasping points. The ground truth of the grasping points is approximately localized at the middle of the sleeve opening edge for a robust dressing. 2) Imitating a human nurse who offers external support to hold the bedridden person's paralyzed/contracted arm, the robot also lifts up the manikin's arm during dressing. The robot gripper grasps the manikin's upper arm near the elbow position with 4Dof top-down movements and lifts it up. The grasping yaw orientation is calculated according to the upper arm posture. 3) Then the robot pulls the gown along the real-tracked arm using the hierarchical controller proposed in our previous work (*19*).

Stage D of spreading the garment to cover the upper body

Lastly, the robot spreads the gown to cover the user's upper body. The garment is grasped at a random location on the segmented garment near the collar (Fig. 1D, localized within 150 pixels away from the user's right shoulder position) and pulled to the user's shoulder. This process can be repeated on the contralateral of the garment to fully cover the upper body. Here garment segmentation is learned in the same sim-to-real manner as in stage A.

Contrastive learning garment physics framework

Fig. 3-top illustrates the framework of the proposed learning algorithm in two phases: 1) learning garment physical similarity in phase one, inspired by (*70*) which has successfully measured physical properties for cloth in the wind; 2) learning physics residual from embedded representations in a supervised manner in phase two, inspired by (*71*) which estimates the parameter difference between two models using a single observation.

Garment in simulation

In this work, the Blender simulation is used to model garment dynamics using its in-built solver with Bullet Physics Engine. Visual observations of garment dropping behaviors are presented as depth video clips $\boldsymbol{x}$ (i. e., a sequence of depth images for their invariance to color and texture). The following manipulations are executed to observe the garment dynamic physical behaviors: 1) the garment is initially hung by two grasping vertices; 2) one grasping vertex is released and the garment falls based on the simulated gravity, as shown in Fig. 4B. Details of the simulation environment are presented in the supplementary materials.

To define the parameter values search base to generate input video clips, the default material 'Cotton' in Blender is selected as the base material as it closely resembles the real hospital gown in this work. The range of the parameters are restricted by multiplying the base material parameters by 10^{-1} and 5 to obtain the most flexible and stiffest material respectively, as presented in Table 2. Given each set of parameters $\boldsymbol{\theta}_s^i$, the corresponding depth video clips of garment falling $\boldsymbol{x}_s \in \mathbb{R}^{N\times H\times W}$ are generated from simulation. The camera frame rate in Blender is set as same as the real camera (30Hz). To match the dynamics among different video clips, we extract same 15 frames in each video clip using the timestamp of the first image in the series.

Physical similarity learning

We use simulation data $\boldsymbol{x}_s$ to train the embedding network. The contrastive loss is described as:

$$L(Y, \boldsymbol{X}_1, \boldsymbol{X}_2) = (1-Y)\frac{1}{2}D^2 + Y\frac{1}{2}\left\{\max(0, m-D)\right\}^2$$

where $\boldsymbol{X}_1, \boldsymbol{X}_2$ are the pair of inputs, Y is a binary label assigned to this pair ($Y = 1$, if negative pair; $Y = 0$, if positive pair), m is the margin that is usually set as 1, D denotes the Euclidean distance between embedded representations of $\boldsymbol{X}_1, \boldsymbol{X}_2$:

$$D = \left\| f_e(\boldsymbol{X}^1), f_e(\boldsymbol{X}^2) \right\|$$

where $\boldsymbol{f}_e(\boldsymbol{x})$ denotes the embedding network.

A Siamese network structure is used as the embedding network to map the generated video clip, instead of single frames, onto the embedding space. The Siamese network adopts two symmetrical embedding neural networks sharing the same weights for the input pairs (*78*). Modified from the structure of LRCN (*79*), the embedding function $\boldsymbol{f}_e(\boldsymbol{x})$ is formatted as a deep neural network combining ImageNet-pretrained Densenet-121 with LSTM (*80*).

Parameter residual learning

We estimate the parameter error directly based on the two embedded representations, using a supervised fully-connected network f_p. Parameter residual estimation performs better at estimating small deltas due to its nature of difference transformation. The input of the parameter residual estimation network is the concatenation of the two embedded representations $(f_e(\boldsymbol{x}_s^i), f_e(\boldsymbol{x}_s^j))$, and the output is the parameter $\Delta\boldsymbol{\theta}_s$. The ground truth of the output is calculated as $\Delta\boldsymbol{\theta}_s = \boldsymbol{\theta}_s^i - \boldsymbol{\theta}_s^j$.

Physics identification of the real garment

The last step is to identify the parameters of the real garment. Following the same procedure in simulation, a real video clip $\boldsymbol{x}_r$ is collected using a depth camera. The clip $\boldsymbol{x}_r$ is then mapped onto the same embedded space using embedding network $f_e(\boldsymbol{x}_r)$. Its nearest neighbour in the embedded space is found $f_e(\boldsymbol{x}_r^n)$, where $\boldsymbol{x}_r^n$ is the corresponding simulated video clip. $f_e(\boldsymbol{x}_r)$ and $f_e(\boldsymbol{x}_r^n)$ are treated as the input of the parameter residual estimation network. The parameters of the real garment $\boldsymbol{\theta}_r$ are obtained as:

$$\boldsymbol{\theta}_r - \boldsymbol{\theta}_r^n = f_p(f_e(\boldsymbol{x}_r), f_e(\boldsymbol{x}_r^n))$$

where $\boldsymbol{\theta}_r^n$ is the corresponding simulation parameters of the video clip $\boldsymbol{x}_r^n$.

Active pre-grasp manipulation learning framework

The proposed method is formatted in a simplified deep Q-learning formulation to learn a state-independent policy, inspired by (*36*) which has successfully learned object push-and-grasp policies. The agent makes only one attempt per episode to myopically maximize immediate reward (discount factor = 0), as shown in Fig. 3-bottom.

State representations

On pre-grasp attempt t, multi-view RGB-D image representations of the gown are observed to serve as the state $\boldsymbol{s}_t$. Specifically, six RGB-D images covering 180 degree around the garment (different multiples of 30 degree) are captured on the x-y plane. In the simulation environment, 6 cameras are placed around the garment to capture RGB-D images from various points of view. In real-world experiments, we set the robot to rotate the garment six times to capture multi-view images from one on-board camera. In our case, each pixel spatially represents approximately a 3^2mm vertical column of 3D space in the agent's workspace.

Primitive actions

We parameterize each action $\boldsymbol{a}_t$ as a pre-grasp manipulation primitive behavior ψ (i. e., move-left-grasp ψ_l, direct-grasp ψ_g and move-right-grasp ψ_r) executed at the 3D location $\boldsymbol{q}$ of pixel $\boldsymbol{p}$ in one of $k = 6$ orientations. We account for different primitive orientations using the orientations of camera views. We have six camera views $\boldsymbol{\gamma}_c^i|_{i=1,2,...,6}$ covering 180 degree around the garment on the x-y plane. Thus, we define six discretized primitive action orientations on the x-y plane as $\boldsymbol{\gamma}_p^i = \boldsymbol{\gamma}_c^i + \boldsymbol{\Delta}\gamma$, where $\boldsymbol{\Delta}\gamma$ is manually designed as 15 degree in our case. Therefore, each action is designed as 4Dof motion (x, y, z, and x-y orientation):

$$\boldsymbol{a} = (\psi, \boldsymbol{q}, \boldsymbol{\gamma})|\psi \in \{\psi_l, \psi_g, \psi_r\}, \boldsymbol{q}, \boldsymbol{\gamma} \twoheadrightarrow \boldsymbol{p} \in \boldsymbol{s}_t$$

The pre-grasp manipulation primitive behaviors are defined as:

- move-left-grasp: $\boldsymbol{q}$ denotes the starting position of a 5cm moving left prior to grasping in one of k = 6 orientations. $\boldsymbol{q}$ physically corresponds to the middle position of the fingertips of an open two-finger gripper. The trajectory of moving left is orthogonal to the primitive orientation $\boldsymbol{\gamma}_p^i$ on the x-y plane, while $\boldsymbol{\gamma}_p^i$ is decided by the view of the image that pixel $\boldsymbol{p}$ belongs to.

- direct-grasp: $\boldsymbol{q}$ denotes the middle position of the fingertips of an open two-finger gripper in one of k = 6 orientations.

- move-right-grasp: The definition is similar to move-left-grasp behavior except that the gripper moves right 5cm to isolate the grasping area.

Learning pre-grasp affordance functions

The robot plans a pre-grasp action $\boldsymbol{a}_t$ using a policy $\pi(\boldsymbol{s}_t)$. The pre-grasp affordance functions are formatted as three feed-forward fully convolutional networks (Densenet-121 (*81*)) ψ_l, ψ_g and ψ_r, one for each pre-grasp primitive behavior. Each fully convolutional network (FCN) takes the RGB-D image representation of the state $\boldsymbol{s}_t$ as input and outputs a dense pixel-wise map of affordance, where each individual affordance value prediction at pixel $\boldsymbol{p}$ represents the expected reward of executing primitive ψ at 3D location $\boldsymbol{q}$ and the corresponding orientation $\boldsymbol{\gamma}$ where $\boldsymbol{q}, \boldsymbol{\gamma} \twoheadrightarrow \boldsymbol{p} \in \boldsymbol{s}_t$.

Therefore, in total k = 6 sets of RGB-D images are fed into each FCN in sequence, the output is 18 pixel-wise affordance maps (as in Fig. 3). The action that maximizes the affordance is the pixel with the highest affordance across all 18 pixel-wise maps: $\mathrm{argmax}_{(\psi,\boldsymbol{q},\boldsymbol{\gamma})}(\psi_l(\boldsymbol{s}_t), \psi_g(\boldsymbol{s}_t), \psi_r(\boldsymbol{s}_t))$. The action $\boldsymbol{a}_t$ is then selected as the pixel with highest estimated affordance with ϵ-greedy exploration.

Policy learning from demonstration

Training policies on complex tasks with sparse rewards is usually challenging. Inspired by the method of deep Q-learning from demonstration (*82*), we use demonstration data to guide and accelerate the agent towards good behaviors. On pre-grasp attempt t, multi-view RGB-D images are captured. A human participant is asked to annotate the pixel p. Then the robot executes at the 3D location $\boldsymbol{q}$ of pixel $\boldsymbol{p}$ in its corresponding orientation. A binary reward is given by the participant, depending on whether there is a clear single-layer garment grasping. Note here that a set of RGB-D images containing the state of the gripper and gown after the action are also collected for the reward learning detailed in the next section. Five people are involved to collect both positive and negative data, to make these demonstrations a more effective data source. These demonstration data, including a set of states, actions and rewards (S, A, R), are stored in an additional buffer R_D for pre-training the network $\pi(\boldsymbol{s}_t)$. Then at active learning time, we draw N_D samples from this demonstration buffer along with environment interaction data from main buffer in batches for policy training.

FCNs are trained at each iteration i using the classic Huber loss function:

$$L_f = L_h(F_\pi(\boldsymbol{s}_i, \boldsymbol{a}_i) - r_{\boldsymbol{a}_i}(\boldsymbol{s}_i))$$

where F_π is the policy function to estimate affordance, and $r_{\boldsymbol{a}_i}(\boldsymbol{s}_i)$ is the received award and the action executed. In this loss, we pass gradients only through the single pixel p from which the value predictions of the executed action a_i was computed. All other pixels at iteration i backpropagate with 0 loss.

Similar to (*82*), an additional supervised loss L_s, in a Mean Squared Error function, is used as an auxiliary loss to train the network on the positive demonstration samples in the data and defined as,

$$L_s = L_{mse}(F_\pi(\boldsymbol{s}_i, \boldsymbol{a}'_i) - H_\pi(\boldsymbol{s}_i, \boldsymbol{a}_D))$$

where $\boldsymbol{a}_i'$ represents the set of all available actions, $\boldsymbol{a}_D$ is the positive action the expert demonstrator took in state $\boldsymbol{s}_i$, and $H_\pi(\boldsymbol{s}_i, \boldsymbol{a}_D)$ is a heat map affordance of demonstration data with a 2D Gaussian blob centered on the pixel of demonstrated action. This loss forces the affordance values of the other actions to be at least a margin lower than the value of the positive demonstration action. Adding this loss makes the greedy policy induced by the value function to imitate the demonstrator. Therefore, we define a hybrid loss for network training combining the supervised loss and affordance loss with weighting parameter λ:

$$L = L_f + \lambda L_s$$

Rewards

Upon executing $\boldsymbol{a}_t$, the robot receives a reward $r_{\boldsymbol{a}_t}(\boldsymbol{s}_t) = 1$ if it successfully grasps the gown with a clear single layer and $r_{\boldsymbol{a}_t}(\boldsymbol{s}_t) = 0$ otherwise. The reward is described as:

$$r_{\boldsymbol{a}_t}(\boldsymbol{s}_t) = \begin{cases} 1, & \text{if single-layer garment grasping} \\ 0, & \text{if otherwise} \end{cases}$$

Formally, our learning objective is to iteratively minimize the error δ_t of estimated affordance $F_\pi(\boldsymbol{s}_t, \boldsymbol{a}_t)$ to the actual received reward $r_{\boldsymbol{a}_t}(\boldsymbol{s}_t)$.

Engineering reward functions is generally challenging when image observations are involved (*83*). Inspired by (*84*), we train a binary reward classifier without manual engineering. The RGB-D images captured during the data collection phase with the labeled reward, which contains the states of the gripper and gown after the action executed, are used to train a binary reward classifier.

Robot-garment interaction in simulation and real-world environment

In the simulation environment, we use blendtorch open-source library (*85*) which integrates Blender simulation engine with OpenAI Gym (*86*) and PyTorch. We place a Robotiq 2F-85

gripper model in simulation to interact with the environment. Camera parameters in the simulation (e. g., horizontal and vertical viewing angles, focal length, the distance to the garment) are set to be the same as real ones. The garment is initially hung by two grasping points (one approximately localized at the collar, one randomly positioned on the garment). Then we release the latter point so that the garment could fall naturally based on simulated gravity to obtain different garment configurations at each state. While in real-world experiments, various garment configurations at each state are achieved by either human interfere or robot arm shaking.

List of Supplementary Materials

- Supplementary Results, Materials and Methods
- Fig. S1. Snapshots of the complete dressing pipeline.
- Fig. S2. Baseline models.
- Fig. S3. Simulation and real-world environments of robot-assisted dressing.
- Fig. S4. Failure cases of the dressing pipeline.
- Table S1. Summary of network and data collection for policy learning of each gown.

References

1. T. L. Mitzner, T. L. Chen, C. C. Kemp, W. A. Rogers, Identifying the potential for robotics to assist older adults in different living environments, *International journal of social robotics*. **6**, 213–227 (2014).

2. H. Robinson, B. MacDonald, E. Broadbent, The role of healthcare robots for older people at home: A review, *International Journal of Social Robotics*. **6**, 575–591 (2014).

3. B. J. Dudgeon, J. M. Hoffman, M. A. Ciol, A. Shumway-Cook, K. M. Yorkston, Managing activity difficulties at home: a survey of medicare beneficiaries, *Archives of physical medicine and rehabilitation*. **89**, 1256–1261 (2008).

4. A. Colomé, A. Planells, C. Torras, A friction-model-based framework for reinforcement learning of robotic tasks in non-rigid environments, *2015 IEEE international conference on robotics and automation (ICRA)* (IEEE, 2015), pp. 5649–5654.

5. T. Tamei, T. Matsubara, A. Rai, T. Shibata, Reinforcement learning of clothing assistance with a dual-arm robot, *2011 11th IEEE-RAS International Conference on Humanoid Robots* (IEEE, 2011), pp. 733–738.

6. T. Matsubara, D. Shinohara, M. Kidode, Reinforcement learning of a motor skill for wearing a t-shirt using topology coordinates, *Advanced Robotics*. **27**, 513–524 (2013).

7. N. Koganti, T. Tamei, T. Matsubara, T. Shibata, Estimation of human cloth topological relationship using depth sensor for robotic clothing assistance, *Proceedings of Conference on Advances In Robotics* (2013), pp. 1–6.

8. N. Koganti, T. Tamei, T. Matsubara, T. Shibata, Real-time estimation of human-cloth topological relationship using depth sensor for robotic clothing assistance, *The 23rd IEEE in-*

ternational symposium on robot and human interactive communication (IEEE, 2014), pp. 124–129.

9. N. Koganti, J. G. Ngeo, T. Tomoya, K. Ikeda, T. Shibata, Cloth dynamics modeling in latent spaces and its application to robotic clothing assistance, *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2015), pp. 3464–3469.

10. Z. Erickson, A. Clegg, W. Yu, G. Turk, C. K. Liu, C. C. Kemp, What does the person feel? learning to infer applied forces during robot-assisted dressing, *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 6058–6065.

11. F. Zhang, A. Cully, Y. Demiris, Personalized robot-assisted dressing using user modeling in latent spaces, *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 3603–3610.

12. Z. Erickson, M. Collier, A. Kapusta, C. C. Kemp, Tracking human pose during robot-assisted dressing using single-axis capacitive proximity sensing, *IEEE Robotics and Automation Letters*. **3**, 2245–2252 (2018).

13. E. Pignat, S. Calinon, Learning adaptive dressing assistance from human demonstration, *Robotics and Autonomous Systems*. **93**, 61–75 (2017).

14. N. Koganti, T. Tamei, K. Ikeda, T. Shibata, Bayesian nonparametric learning of cloth models for real-time state estimation, *IEEE Transactions on Robotics*. **33**, 916–931 (2017).

15. Z. Erickson, H. M. Clever, V. Gangaram, G. Turk, C. K. Liu, C. C. Kemp, Multidimensional capacitive sensing for robot-assisted dressing and bathing, *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)* (IEEE, 2019), pp. 224–231.

16. A. Kapusta, Z. Erickson, H. M. Clever, W. Yu, C. K. Liu, G. Turk, C. C. Kemp, Personalized collaborative plans for robot-assisted dressing via optimization and simulation, *Autonomous Robots*. **43**, 2183–2207 (2019).

17. Y. Gao, H. J. Chang, Y. Demiris, Iterative path optimisation for personalised dressing assistance using vision and force information, *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (IEEE, 2016), pp. 4398–4403.

18. Y. Gao, H. J. Chang, Y. Demiris, User modelling using multimodal information for personalised dressing assistance, *IEEE Access*. **8**, 45700–45714 (2020).

19. F. Zhang, A. Cully, Y. Demiris, Probabilistic real-time user posture tracking for personalized robot-assisted dressing, *IEEE Transactions on Robotics*. **35**, 873–888 (2019).

20. E. Hernandez-Medina, S. Eaton, D. Hurd, A. White, *Training programs for certified nursing assistants* (AARP Public Policy Instutute, 2006).

21. CNA Skill 15 Dress the resident with a paralyzed / contracted arm (2021). [Online]. Available: https://www.youtube.com/watch?v=-IkJ5ev3edM&t=263s.

22. Dress a Resident with a Weak Arm CNA Skill (2021). [Online] Available: https://www.youtube.com/watch?v=eI4wBjavIBk.

23. H. Yin, A. Varava, D. Kragic, Modeling, learning, perception, and control methods for deformable object manipulation, *Science Robotics*. **6** (2021).

24. J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, Y. Mezouar, Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey, *The International Journal of Robotics Research*. **37**, 688–716 (2018).

25. V. E. Arriola-Rios, P. Guler, F. Ficuciello, D. Kragic, B. Siciliano, J. L. Wyatt, Modeling of deformable objects for robotic manipulation: A tutorial and review, *Frontiers in Robotics and AI*. **7**, 82 (2020).

26. A. Ganapathi, P. Sundaresan, B. Thananjeyan, A. Balakrishna, D. Seita, J. Grannen, M. Hwang, R. Hoque, J. E. Gonzalez, N. Jamali, K. Yamane, S. Iba, K. Goldberg, Learning to smooth and fold real fabric using dense object descriptors trained on synthetic color images, *arXiv preprint arXiv:2003.12698*. (2020).

27. E. Corona, G. Alenya, A. Gabas, C. Torras, Active garment recognition and target grasping point detection using deep learning, *Pattern Recognition*. **74**, 629–641 (2018).

28. Z. Erickson, V. Gangaram, A. Kapusta, C. K. Liu, C. C. Kemp, Assistive gym: A physics simulation framework for assistive robotics, *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 10169–10176.

29. F. Zhang, Y. Demiris, Learning grasping points for garment manipulation in robot-assisted dressing, *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 9114–9120.

30. A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrik, A. Kargakos, L. Wagner, V. Hlaváč, T.-K. Kim, S. Malassiotis, Folding clothes autonomously: A complete pipeline, *IEEE Transactions on Robotics*. **32**, 1461–1478 (2016).

31. D. Seita, N. Jamali, M. Laskey, A. K. Tanwani, R. Berenstein, P. Baskaran, S. Iba, J. Canny, K. Goldberg, Deep transfer learning of pick points on fabric for robot bed-making, *International Symposium on Robotics Research (ISRR)* (2019).

32. K. Saxena, T. Shibata, Garment recognition and grasping point detection for clothing assistance task using deep learning, *2019 IEEE/SICE International Symposium on System Integration (SII)* (IEEE, 2019), pp. 632–637.

33. M. Cusumano-Towner, A. Singh, S. Miller, J. F. O'Brien, P. Abbeel, Bringing clothing into desired configurations with limited perception, *2011 IEEE international conference on robotics and automation* (IEEE, 2011), pp. 3893–3900.

34. R. Jangir, G. Alenyà, C. Torras, Dynamic cloth manipulation with deep reinforcement learning, *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 4630–4636.

35. Y. Tsurumine, Y. Cui, E. Uchibe, T. Matsubara, Deep reinforcement learning with smooth policy update: Application to robotic cloth manipulation, *Robotics and Autonomous Systems*. **112**, 72–83 (2019).

36. A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, T. Funkhouser, Learning synergies between pushing and grasping with self-supervised deep reinforcement learning, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 4238–4245.

37. H. Liang, X. Lou, Y. Yang, C. Choi, Learning visual affordances with target-orientated deep q-network to grasp objects by harnessing environmental fixtures, *2021 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2021), pp. 2562–2568.

38. H. Ha, S. Song, Flingbot: The unreasonable effectiveness of dynamic manipulation for cloth unfolding, *Conference on Robotic Learning (CoRL)* (2021).

39. K. S. Sahari, H. Seki, Y. Kamiya, M. Hikizu, Edge tracing manipulation of clothes based on different gripper types, *Journal of Computer Science*. **6**, 872–879 (2010).

40. I. Garcia-Camacho, M. Lippi, M. C. Welle, H. Yin, R. Antonova, A. Varava, J. Borras, C. Torras, A. Marino, G. Alenya, D. Kragic, Benchmarking bimanual cloth manipulation, *IEEE Robotics and Automation Letters*. **5**, 1111–1118 (2020).

41. J. Matas, S. James, A. J. Davison, Sim-to-real reinforcement learning for deformable object manipulation, *Conference on Robot Learning (CoRL)* (PMLR, 2018), pp. 734–743.

42. K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, V. Vanhoucke, Using simulation and domain adaptation to improve efficiency of deep robotic grasping, *2018 IEEE international conference on robotics and automation (ICRA)* (IEEE, 2018), pp. 4243–4250.

43. P. D. Nguyen, T. Fischer, H. J. Chang, U. Pattacini, G. Metta, Y. Demiris, Transferring visuomotor learning from simulation to the real world for robotics manipulation tasks, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 6667–6674.

44. F. Sadeghi, S. Levine, Cad2rl: Real single-image flight without a single real image, *Proceedings of Robotics: Science and Systems* (2017).

45. K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, M. Khansari, Rl-cyclegan: Reinforcement learning aware simulation-to-real, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 11157–11166.

46. J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, M. Hutter, Learning agile and dynamic motor skills for legged robots, *Science Robotics*. **4** (2019).

47. F. Golemo, A. A. Taiga, A. Courville, P.-Y. Oudeyer, Sim-to-real transfer with neural-augmented robot simulation, *Conference on Robot Learning (CoRL)* (2018), pp. 817–828.

48. P. Christiano, Z. Shah, I. Mordatch, J. Schneider, T. Blackwell, J. Tobin, P. Abbeel, W. Zaremba, Transfer from simulation to real world through learning deep inverse dynamics model, *arXiv preprint arXiv:1610.03518*. (2016).

49. S. Kolev, E. Todorov, Physically consistent state estimation and system identification for contacts, *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)* (IEEE, 2015), pp. 1036–1043.

50. Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, S. Song, Densephysnet: Learning dense physical object representations via multi-step dynamic interactions, *Proceedings of Robotics: Science and Systems* (2019).

51. J. K. Li, W. S. Lee, D. Hsu, Push-net: Deep planar pushing for objects with unknown physical properties, *Proceedings of Robotics: Science and Systems* (2018), vol. 14, pp. 1–9.

52. A. Zeng, S. Song, J. Lee, A. Rodriguez, T. Funkhouser, Tossingbot: Learning to throw arbitrary objects with residual physics, *IEEE Transactions on Robotics*. (2020).

53. A. Ajay, J. Wu, N. Fazeli, M. Bauza, L. P. Kaelbling, J. B. Tenenbaum, A. Rodriguez, Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing, *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 3066–3073.

54. A. Kloss, S. Schaal, J. Bohg, Combining learned and analytical models for predicting action effects from sensory data, *The International Journal of Robotics Research*. p. 0278364920954896 (2020).

55. P. Chang, T. Padif, Sim2real2sim: Bridging the gap between simulation and real-world in flexible object manipulation, *2020 Fourth IEEE International Conference on Robotic Computing (IRC)* (IEEE, 2020), pp. 56–62.

56. E. Miguel, D. Bradley, B. Thomaszewski, B. Bickel, W. Matusik, M. A. Otaduy, S. Marschner, Data-driven estimation of cloth simulation models, *Computer Graphics Forum* (Wiley Online Library, 2012), pp. 519–528.

57. H. Wang, J. F. O'Brien, R. Ramamoorthi, Data-driven elastic models for cloth: modeling and measurement, *ACM transactions on graphics (TOG)*. **30**, 1–12 (2011).

58. K. L. Bouman, B. Xiao, P. Battaglia, W. T. Freeman, Estimating the material properties of fabric from video, *Proceedings of the IEEE international conference on computer vision* (2013), pp. 1984–1991.

59. Y. Li, Y. Wang, M. Case, S.-F. Chang, P. K. Allen, Real-time pose estimation of deformable objects using a volumetric approach, *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2014), pp. 1046–1052.

60. S. Yang, J. Liang, M. C. Lin, Learning-based cloth material recovery from video, *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 4383–4393.

61. Y.-L. Qiao, J. Liang, V. Koltun, M. C. Lin, Scalable differentiable physics for learning and control, *International Conference on Machine Learning (ICML)* (2020).

62. J. K. Murthy, M. Macklin, F. Golemo, V. Voleti, L. Petrini, M. Weiss, B. Considine, J. Parent-Lévesque, K. Xie, K. Erleben, L. Paull, F. Shkurti, D. Nowrouzezahrai, S. Fidler, gradsim: Differentiable simulation for system identification and visuomotor control, *International Conference on Learning Representations* (2020).

63. X. Lin, Y. Wang, J. Olkin, D. Held, Softgym: Benchmarking deep reinforcement learning for deformable object manipulation, *Conference on Robot Learning (CoRL)* (2020).

64. Y. Wu, W. Yan, T. Kurutach, L. Pinto, P. Abbeel, Learning to manipulate deformable objects without demonstrations, *Proceedings of Robotics: Science and Systems* (2020).

65. R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (IEEE, 2006), vol. 2, pp. 1735–1742.

66. K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, *Proceedings of the IEEE international conference on computer vision* (2017), pp. 2961–2969.

67. P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, D. Krishnan, Supervised contrastive learning, *Advances in Neural Information Processing Systems* (2020), vol. 33, pp. 18661–18673.

68. L. v. d. Maaten, G. Hinton, Visualizing data using t-sne, *Journal of machine learning research*. **9**, 2579–2605 (2008).

69. J. Cardona, M. Howland, J. Dabiri, Seeing the wind: Visual wind speed prediction with a coupled convolutional and recurrent neural network, *Advances in Neural Information Processing Systems* (2019), pp. 8735–8745.

70. T. F. Runia, K. Gavrilyuk, C. G. Snoek, A. W. Smeulders, Cloth in the wind: A case study of physical measurement through simulation, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 10498–10507.

71. A. Allevato, E. S. Short, M. Pryor, A. Thomaz, Tunenet: One-shot residual tuning for system identification and sim-to-real robot task transfer, *Conference on Robot Learning (CoRL)* (2020), pp. 445–455.

72. A. Medela, A. Picon, Constellation loss: Improving the efficiency of deep metric learning loss functions for the optimal embedding of histopathological images, *Journal of Pathology Informatics*. **11** (2020).

73. M. Lippi, P. Poklukar, M. C. Welle, A. Varava, H. Yin, A. Marino, D. Kragic, Latent space roadmap for visual action planning of deformable and rigid object manipulation, *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2020), pp. 5619–5626.

74. W. Yan, A. Vangipuram, P. Abbeel, L. Pinto, Learning predictive representations for deformable objects using contrastive estimation, *Conference on Robot Learning (CoRL)* (2020).

75. IEC60601 (2021). [Online] Available: https://en.wikipedia.org/wiki/IEC_60601.

76. Ridgeback Navigation Stack (2021). [Online] Available: https://github.com/ridgeback/ridgeback.

77. K. Sun, B. Xiao, D. Liu, J. Wang, Deep high-resolution representation learning for human pose estimation, *Proceedings of the IEEE conference on computer vision and pattern recognition* (2019), pp. 5693–5703.

78. A. Veit, S. Belongie, T. Karaletsos, Conditional similarity networks, *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 830–838.

79. J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell, Long-term recurrent convolutional networks for visual recognition and description, *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 2625–2634.

80. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation*. **9**, 1735–1780 (1997).

81. G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), pp. 4700–4708.

82. T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband, J. Agapiou, J. Z. Leibo, A. Gruslys, Deep q-learning from demonstrations, *Proceedings of the AAAI Conference on Artificial Intelligence* (2018).

83. A. Singh, L. Yang, K. Hartikainen, C. Finn, S. Levine, End-to-end robotic reinforcement learning without reward engineering, *Proceedings of Robotics: Science and Systems* (2019).

84. A. Xie, A. Singh, S. Levine, C. Finn, Few-shot goal inference for visuomotor learning and planning, *Conference on Robot Learning (CoRL)* (2018), pp. 40–52.

85. C. Heindl, S. Zambal, J. Scharinger, Learning to predict robot keypoints using artificially generated images, *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)* (IEEE, 2019), pp. 1536–1539.

86. G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, W. Zaremba, Openai gym, *arXiv preprint arXiv:1606.01540*. (2016).

Acknowledgments

We thank all members from Personal Robotics Laboratory, Imperial College London for fruitful discussions. **Funding:** This research is financially supported in part by a Royal Academy of Engineering Chair in Emerging Technologies to Professor Yiannis Demiris, and in part by UKRI Grant EP/V026682/1. **Author contributions:** Dr Fan Zhang developed algorithms, performed experiments, and wrote the manuscript. Professor Yiannis Demiris conceived the assistive dressing scenarios, contributed in results analysis, and revised the manuscript. **Competing interests:** Authors declare that they have no competing interests. **Data and materials availability:** Data and code needed to evaluate the conclusions in the paper can be found at `https://github.com/ImperialCollegeLondon/robot-assisted-dressing-pipeline`.

Movie 1. Overview of the dressing pipeline intended for bedridden people. Following the Certified Nursing Assistant Practice Test guidance, we use a professional training manikin to simulate a bedridden person, and design the pipeline as following stages: (**A**) The robot would navigate to the rail and grasp a hospital gown is naturally hung on a rail. (**B**) The robot would fully unfold the garment in the air. (**C**) The robot would navigate around the hospital bed, lift up and dress the user's both arms; (**D**) The final operation spreads the gown to cover the upper body.

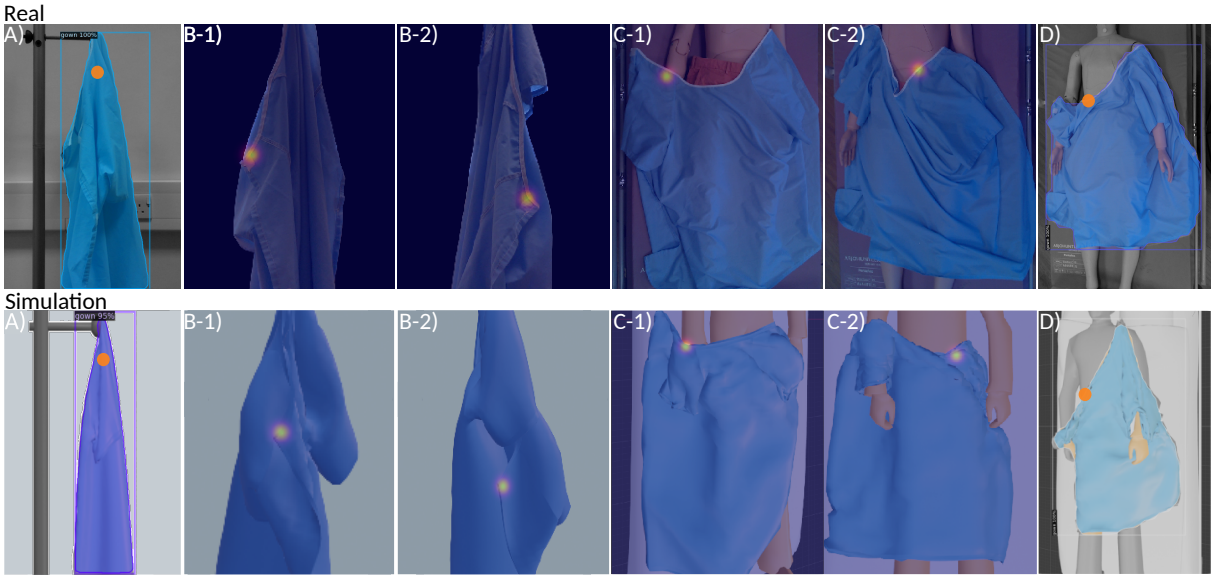

Figure 1: Illustration of six grasping/manipulation points (orange dot/heat map) on the garment to achieve the dressing pipeline in real-world (Top) and simulation environments (Bottom). (**A**) The grasping point in stage A for picking up the garment on the rail, chosen randomly near the hanging point on the segmented garment. (**B**) Two manipulation points in stage B for fully unfolding the garment in the air, localized by our proposed active pre-grasp manipulation learner along with their manipulation orientations and motion primitives. (**C**) Two grasping points in stage C for upper-body dressing, learned by pixel-wise supervised neural networks. (**D**) The last grasping point in stage D for spreading the gown to cover upper body, chosen randomly near the collar on the segmented garment.

Table 1: Performance of the complete dressing pipeline and its each stage.

	Stage A: Garment Grasping	Stage B: Garment Unfolding	Stage C: Robotic-dressing	Stage D: Garment Spreading	Overall
Success Rate (Independent Trial)	98%	94.5%	91.5%	97.5%	–
Success Rate (Overall Trial)	99%	96.9%	95.3%	98.9%	90.5%

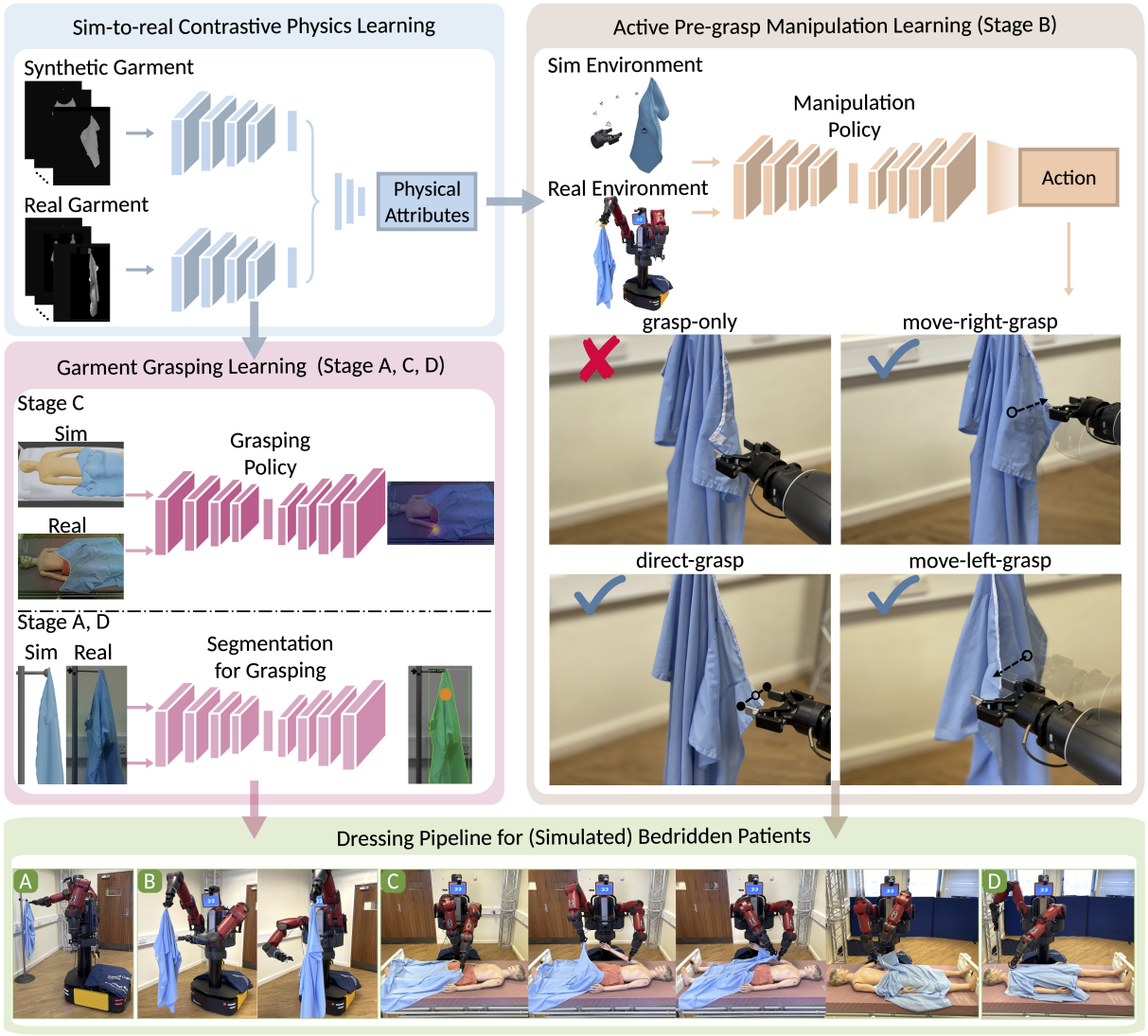

Figure 2: The framework of the dressing pipeline with sim-to-real garment grasping and manipulation. For each grasping/manipulation policy learning, simulation with learned garment physics using the proposed contrastive learning approach is leveraged to either generate cost-effective labeled data for neural network training (stage A, C, D), or learn the proposed pre-grasp manipulation policy directly in simulation before transferring to real systems (stage B).

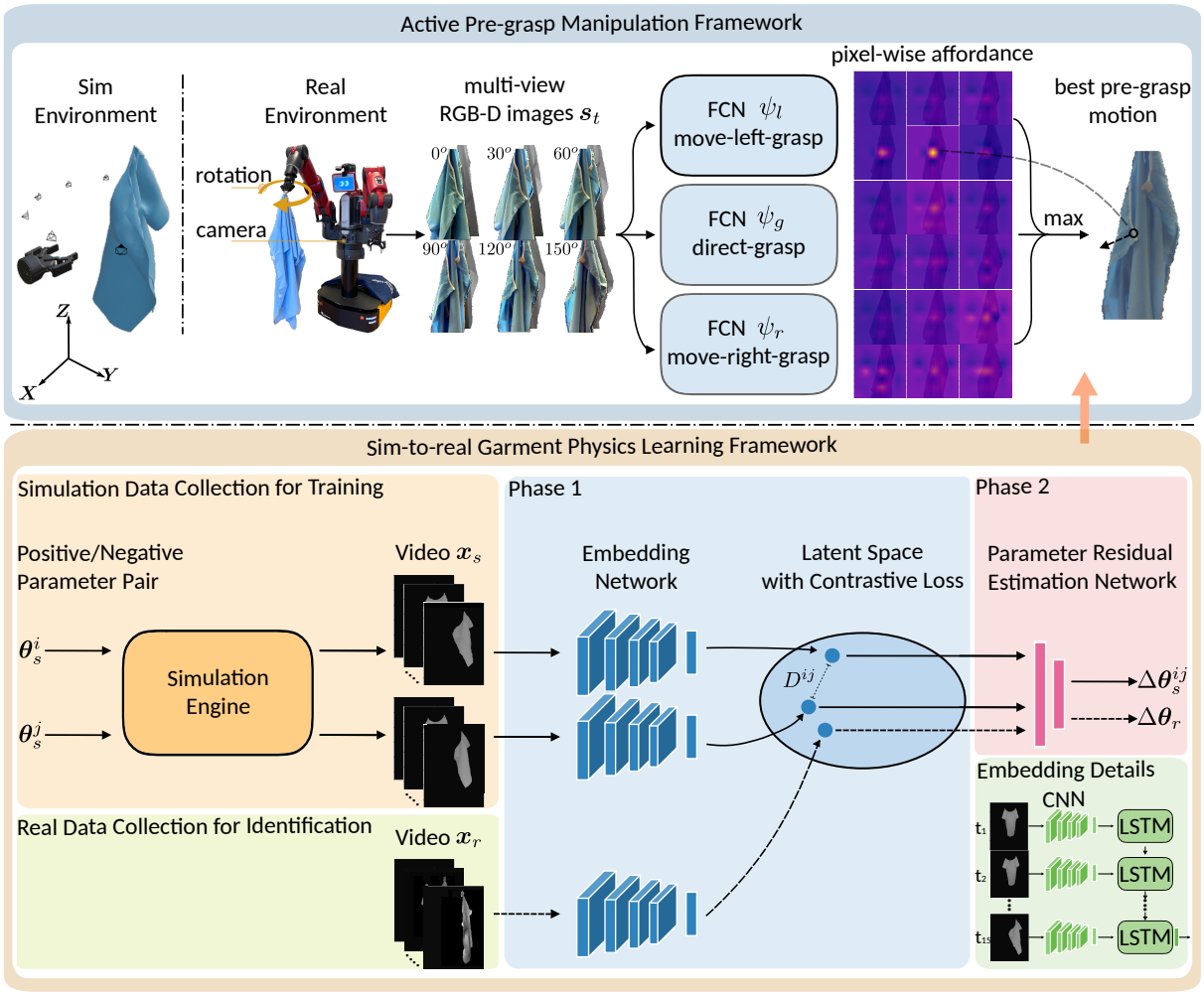

Figure 3: Proposed framework of active pre-grasp manipulation policy learned in a sim-to-real manner. (**Top**) Sim-to-real garment physics learning. From simulation only, garment physical similarity is learned in the embedding space with contrastive loss, and the parameter error is learned from the embedded representations in a supervised manner. Then the learned embedding function is used to measure physical parameters from real-world garment behavior by comparison to its simulated counterpart. (**Bottom**) Active pre-grasp manipulation. Multi-view RGB-D images of the hospital gown are fed into three fully convolutional networks to respectively infer pixel-wise affordance for three motion primitive behaviors. The chosen pixel is decoded into pre-grasp actions.

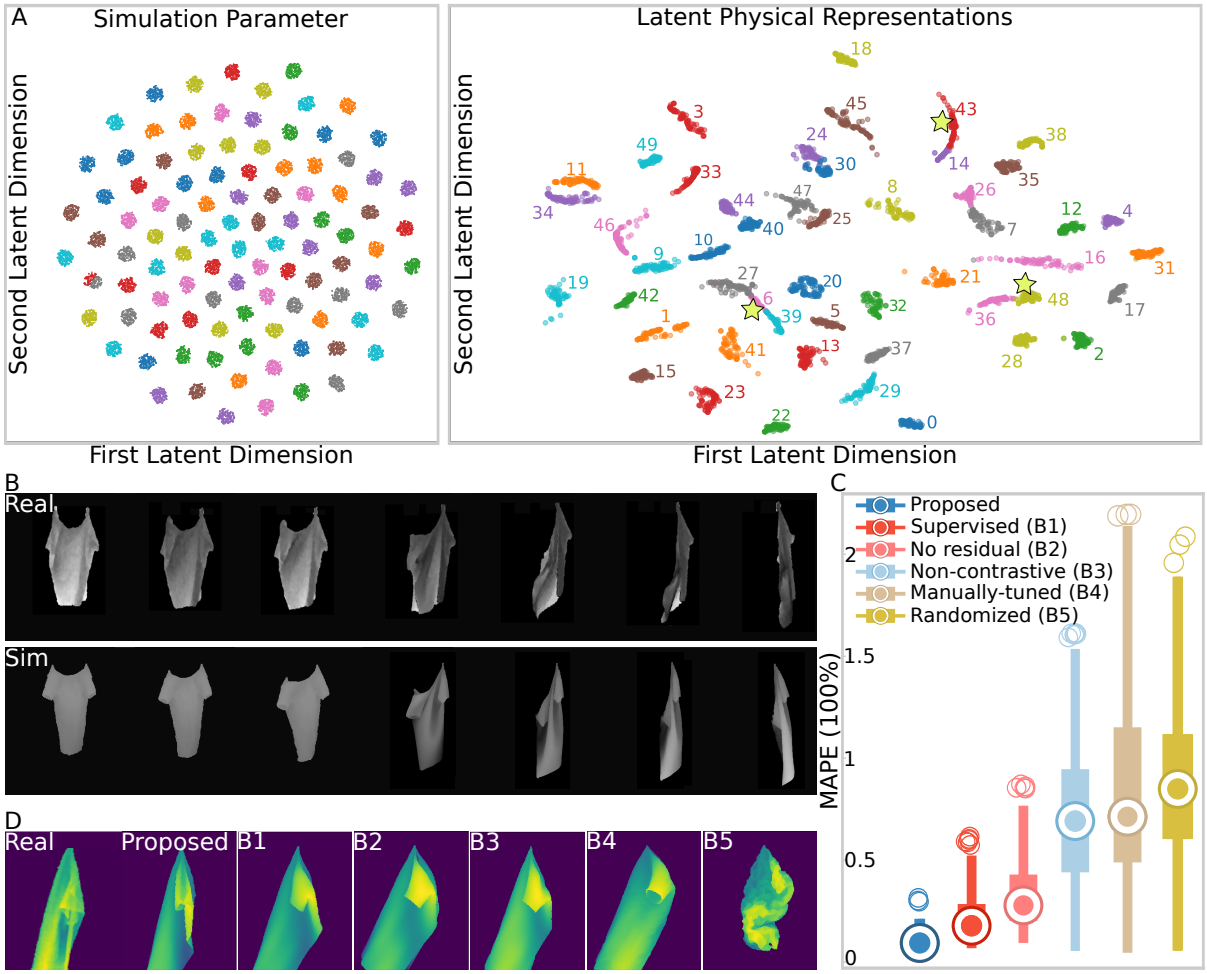

Figure 4: Results of contrastive learning garment physics. (**A**) Results of ablation studies. The left figure shows the 2D t-SNE visualization of the generated simulation properties clustered into 100 classes. The right figure shows the 2D t-SNE visualization of a subset of the corresponding learned embedding space. Number annotations represent different classes. Embedding points from the same class are depicted with the same color. For clarity, first 50 classes are shown in this figure and 10 colors are used iteratively. Stars (near class 6, 43, 48) here represent the embeddings of the three real gowns. (**B**) This panel shows the snapshots of the simulated and real garments dropping. The simulated garment is animated using the physics estimated by our method. (**C**) Results of comparisons with baselines. The boxplot pictures MAPE of estimated garment physical properties using the proposed method and five baselines. The central dot corresponds to the median value of the errors, while the sides of the box refer to the first and third quartiles of the data. (**D**) Visual examples of simulated garment from one point of view, generated using the estimated real garment parameters achieved by the proposed method and five baselines respectively.

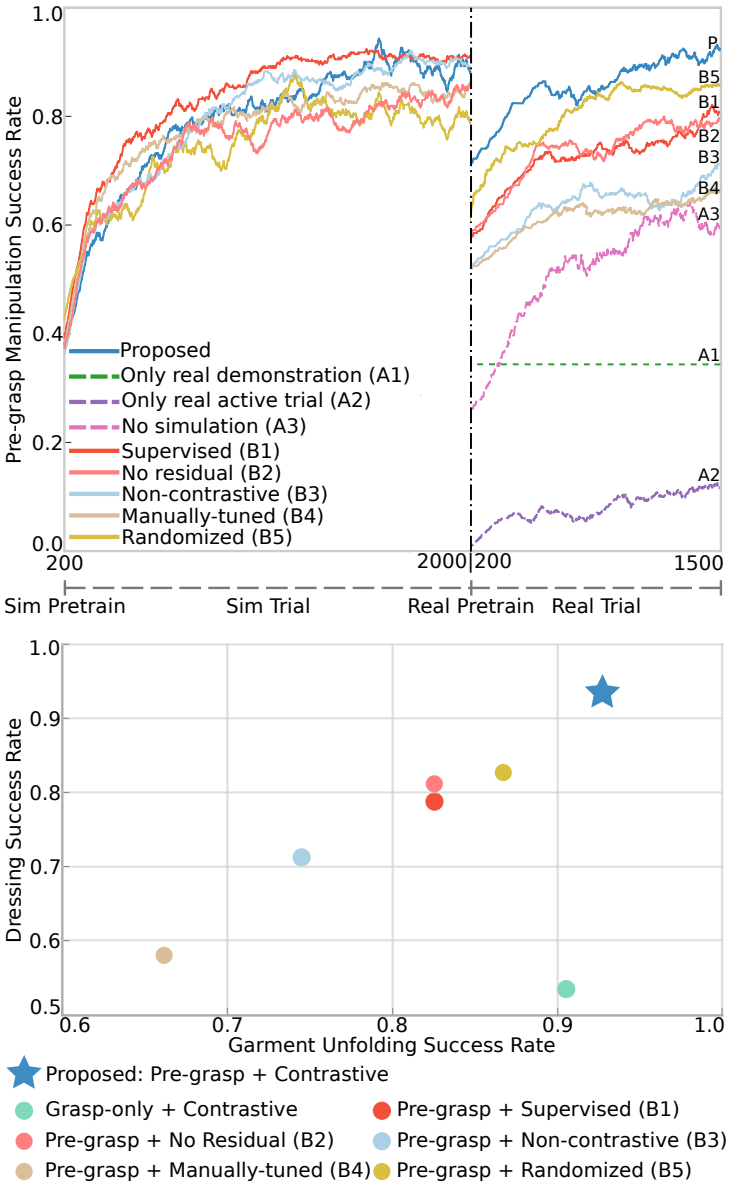

Figure 5: Results of sim-to-real garment pre-grasp manipulation evaluation against ablations and state-of-the-art baselines. (**Top**) Training process of pre-grasp manipulation policy learning on one hospital gown. This panel shows the comparison results against ablations of active learning and baselines of sim-to-real garment manipulation policy transfer. (**Bottom**) Effects of pre-grasp manipulation and sim-to-real physics learning on garment unfolding (stage B) and dressing (stage C) performance.

Table 2: Blender Garment Simulation Parameters

Parameter	Explanation	Search Space
Cloth Mass (kg)	–	$[0.03, 1.5]$
Tension Stiffness	stiffness of tension springs	$[1.50, 75]$
Compression Stiffness	stiffness of compression springs	$[0.15, 2.5]$
Shear Stiffness	stiffness of shear springs	$[0.15, 2.5]$
Bending Stiffness	stiffness of bending springs	$[0.15, 2.5]$
Friction	friction with self-contact	$[0.50, 25]$