HUMAN-ROBOT INTERACTION

Learning garment manipulation policies toward robot-assisted dressing

Fan Zhang* and Yiannis Demiris

Assistive robots have the potential to support people with disabilities in a variety of activities of daily living, such as dressing. People who have completely lost their upper limb movement functionality may benefit from robot-assisted dressing, which involves complex deformable garment manipulation. Here, we report a dressing pipeline intended for these people and experimentally validate it on a medical training manikin. The pipeline is composed of the robot grasping a hospital gown hung on a rail, fully unfolding the gown, navigating around a bed, and lifting up the user's arms in sequence to finally dress the user. To automate this pipeline, we address two fundamental challenges: first, learning manipulation policies to bring the garment from an uncertain state into a configuration that facilitates robust dressing; second, transferring the deformable object manipulation policies learned in simulation to real world to leverage cost-effective data generation. We tackle the first challenge by proposing an active pre-grasp manipulation approach that learns to isolate the garment grasping area before grasping. The approach combines prehensile and nonprehensile actions and thus alleviates grasping-only behavioral uncertainties. For the second challenge, 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 dual-arm robot to put back-opening hospital gowns onto a medical manikin with a success rate of more than 90%.

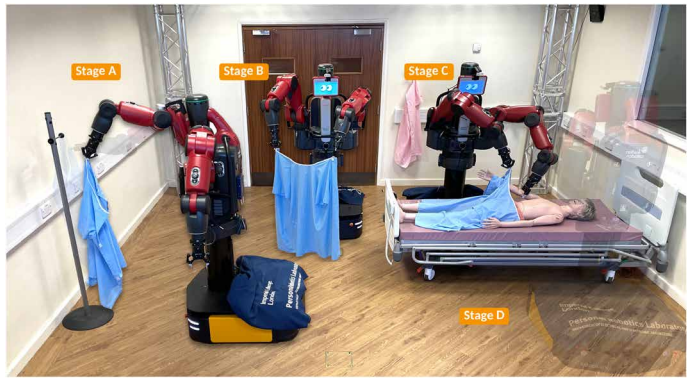
INTRODUCTION

Dressing assistance is a basic assistive activity in the daily life of elderly people and people who suffer from impairments. Studies indicated that although more than 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 by Dudgeon *et al.* (*3*) of 14,500 Medicare beneficiaries showed that 8.2% of those beneficiaries have reported difficulties with dressing. Dudgeon *et al.*'s study also reported that of all activities of daily living, dressing has shown the highest burden on caregiving staff and the lowest use of assistive technologies (*3*).

Recent work has made great progress toward addressing the challenges of using a robot to perform dressing (*4*–*7*). Some work tackled the problem of dressing healthy users who remain stationary during laboratory experiments (*8*–*13*). Other research contributed to providing dressing assistance for (simulated) impaired users who can move their arms with limited motion capability (*14*–*19*). This work focuses on developing a pipeline toward dressing people who have completely lost the ability to move their limbs and validating the approach on a medical training manikin.

Dressing a person with weak/paralyzed arms 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 skills in CNA tests can be found online (*21*, *22*). In this work, we endeavor to equip a robot with an ability to assist in putting back-opening hospital gowns on people in a manner that emulates a CNA. We used 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 person. The patient care manikin (weighing 18 kg), equipped with natural movement of the joints, arms, and legs for realistic positioning, represents a life-sized adult (174 cm). Movie 1 shows the complete dressing sequence. Note that tying the hospital gown on the back is not included in this pipeline because most robot arms do not have the required payload to lift the manikin's upper body.

Several state-of-the-art surveys have identified open challenges in perceiving and handling deformable objects (*23*–*25*), two of which were faced in this work as well to endow robots with the ability to autonomously execute such a dressing pipeline: first, 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



Movie 1. Overview of the dressing pipeline. Following the CNA Practice Test guidance, we used a professional training manikin to simulate a person who has completely lost upper limb movement functionality and designed the pipeline as the following stages: stage A, the robot would navigate to the rail and grasp a hospital gown that is naturally hung on a rail; **s**tage B, the robot would fully unfold the garment in the air; stage C, the robot would navigate around the hospital bed and lift up and dress the user's both arms; stage D, the final operation spreads the gown to cover the upper body.

Personal Robotics Laboratory, Department of Electrical and Electronic Engineering, Imperial College London, London, UK.
*Corresponding author. Email: f.zhang16@imperial.ac.uk

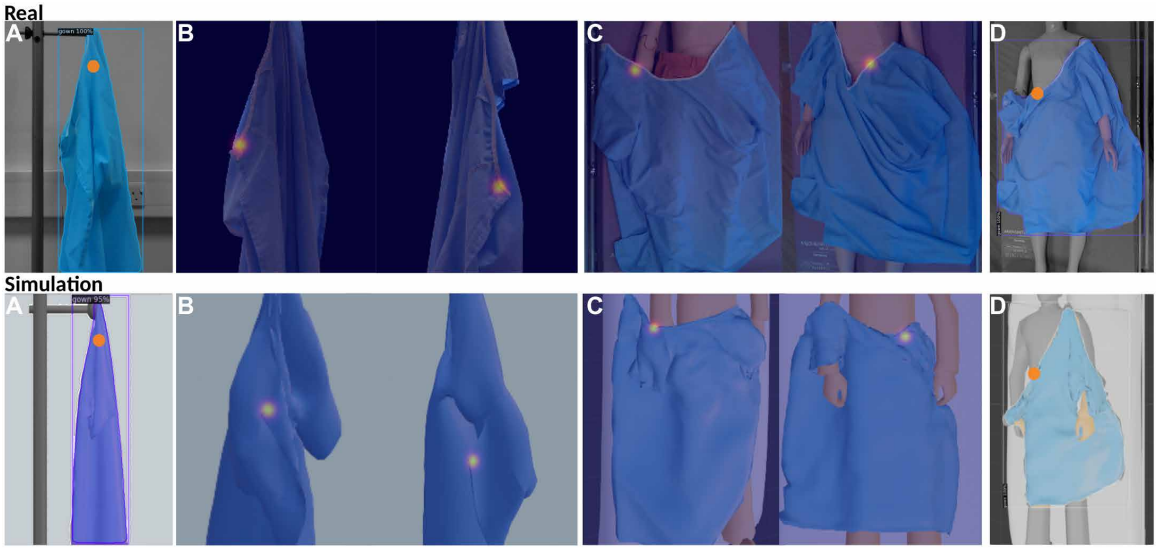


Fig. 1. Illustration of six grasping/manipulation points (orange dot/heat map) on the garment to achieve the dressing pipeline in real-world and simulation environments. (**A**) Grasping point in stage A for picking up the garment on the rail, chosen randomly near the hanging point on the segmented garment. (**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**) Last grasping point in stage D for spreading the gown to cover the upper body, chosen randomly near the collar on the segmented garment.

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: 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 a large dimensional configuration space, most studies on robot-assisted dressing set up 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 before any garment manipulations, which could lead to issues including multiple-layer garment grasping. Unlike the successful garment hanging or folding tasks in the aforementioned research, such issues could induce the occlusions of the sleeve opening and consequently cause garment dressing failures. Multistep manipulation that combines both prehensile and nonprehensile (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 (*37*) actions. 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*).

The second challenge concerning 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*, *43*, *44*). However, visual appeal does not equate to physical realism because a visually realistic garment could behave in a physically unrealistic manner due to occasional deletion or addition of object details (*45*). Therefore, Yin *et al.* (*23*) have 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 the underlying physics is pertinent to deformable object manipulation because their deformations are sensitive to the physical properties of the object itself (*46*–*55*). Recent work focuses on inferring garment physical attributes either by comparing handcrafted garment features (*56*–*59*) or directly from visual observations of object dynamic behaviors using a deep learning method (*60*–*63*). However, both methods remain challenging because 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.

To overcome the first challenge, we propose an active pre-grasp manipulation learning framework, which aims to isolate the garment grasping area before 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 nonprehensile 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 multiview visual observations of the garment to affordance-based pre-grasp manipulation actions through robot exploration and exploitation. Our proposed approach alleviates uncertainties of the grasping-only behavior (e.g., multiple-layer garment grasping and sleeve opening occlusion) and thus facilitates more robust dressing.

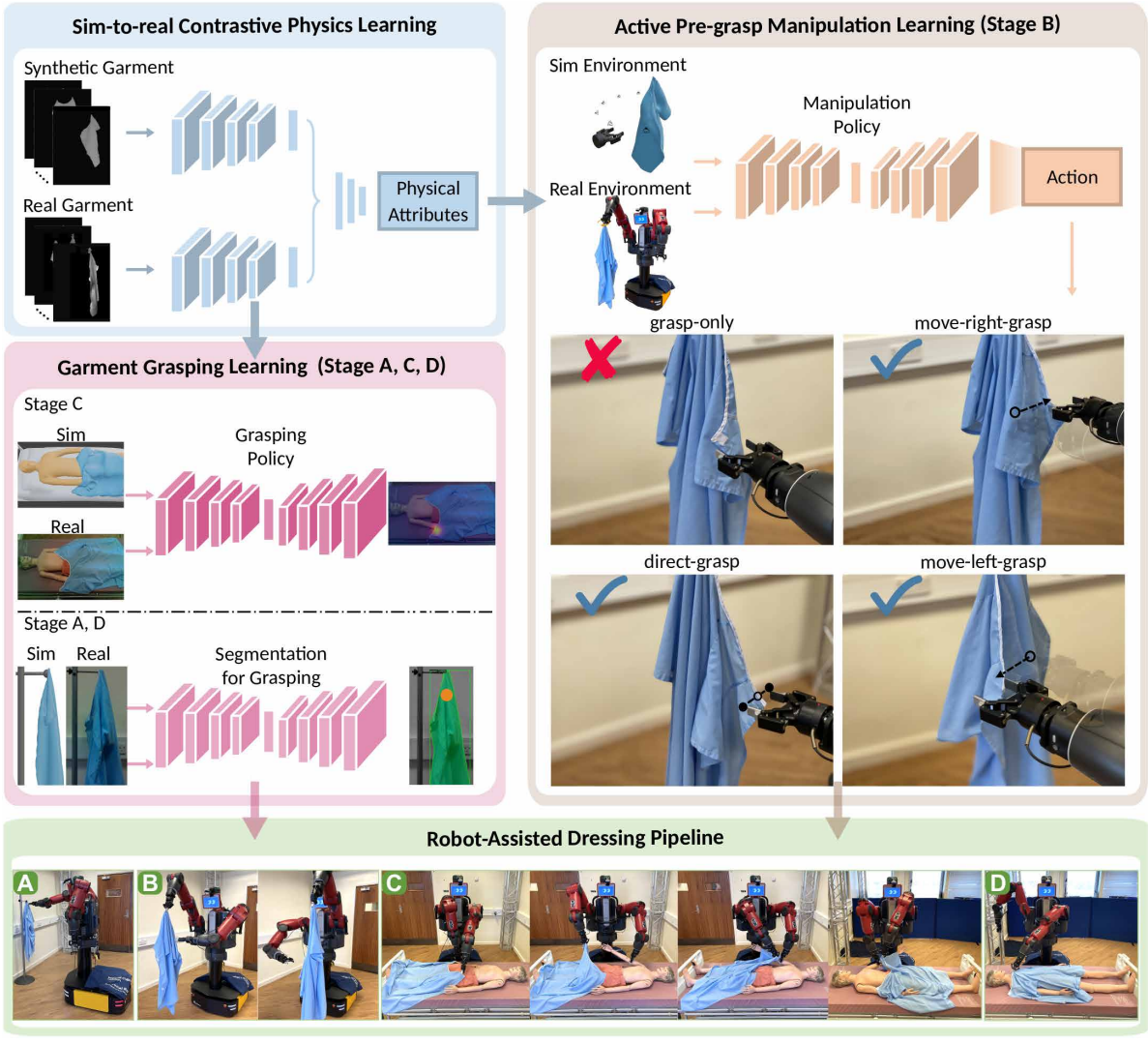


Fig. 2. The framework of the dressing pipeline with sim-to-real garment grasping and manipulation. For each grasping/manipulation policy learning, simulation with learned garment physics using the proposed contrastive learning approach is leveraged to either generate cost-effective labeled data for neural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

In regard to the second challenge, we 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 using contrastive loss to learn the garment physical similarity metric (*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 toward maximizing the physical similarity between the real-world behavior and its simulated counterpart. Our experimental results 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 aforementioned proposed methods, we introduce a dressing pipeline for people who have completely lost the ability to move their limbs. 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. In the image domain, classic domain randomization is used 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: 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*); 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; stage C, the robot would navigate around the bed to grasp the garment (Fig. 1C) and 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); 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: (i) a robot-assisted dressing pipeline intended for people who have completely lost the ability to move their limbs, which is composed of a series of garment grasping and manipulation, robot navigation, and user upper-body dressing, and validated on a medical training manikin; (ii) an active pre-grasp manipulation approach that learns to isolate the garment grasping area before grasping to facilitate a more robust dressing; and (iii) 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 performing statistical comparisons of the proposed garment contrastive physics learning and pre-grasp manipulation methods against several ablations and state-of-the-art baselines.

Experiments on dressing a medical training manikin

In this experiment, we evaluated the performance of the complete dressing pipeline on a medical manikin. Three back-opening hospital gowns were used in our experiments. For each gown, its physical attributes were estimated using the proposed contrastive learning methods, and the pre-grasp manipulation policy (stage B) as well as other grasping policies (stages A, C, and D) were trained in the sim-to-real manner.

First, we carried out ablation studies of each stage in the pipeline. To evaluate the performance of each stage independently, for each stage, we assumed that 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. Two hundred 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 have been 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 failed, then stage C would not be executed. The experimental results show that 181 trials have been successful, yielding an 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 included but were not limited to (i) 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 (ii) stage C, the sleeve getting caught by the manikin's inflexible hand. Details of failure analysis are presented in the Supplementary Materials.

Patient safety was one of the priorities during the design of our dressing pipeline. We have taken into account four aspects of user's safety. First, the robot is operated in a moderate speed, as a trade-off between the safety consideration and the user's comfort. Thus, most of the experimental time is spent by robot physical movements, but not by perception or reasoning. Second, we designed 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 s to ensure that no large force is applied to the user's arm and shoulder joint. Third, a moderate maximum grip force of the gripper was set to ensure that the gripper is firm enough to hold and lift the user's arm without applying large forces on the user. Fourth, the robot dressing motion planning was 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

Figure 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 investigated the performance of physical similarity learning in phase one. It has been observed in our preliminary experiments that some properties have larger influences on garment deformation in this particular physical phenomenon of garment dropping. Therefore, six physical properties were selected as the input of the

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%

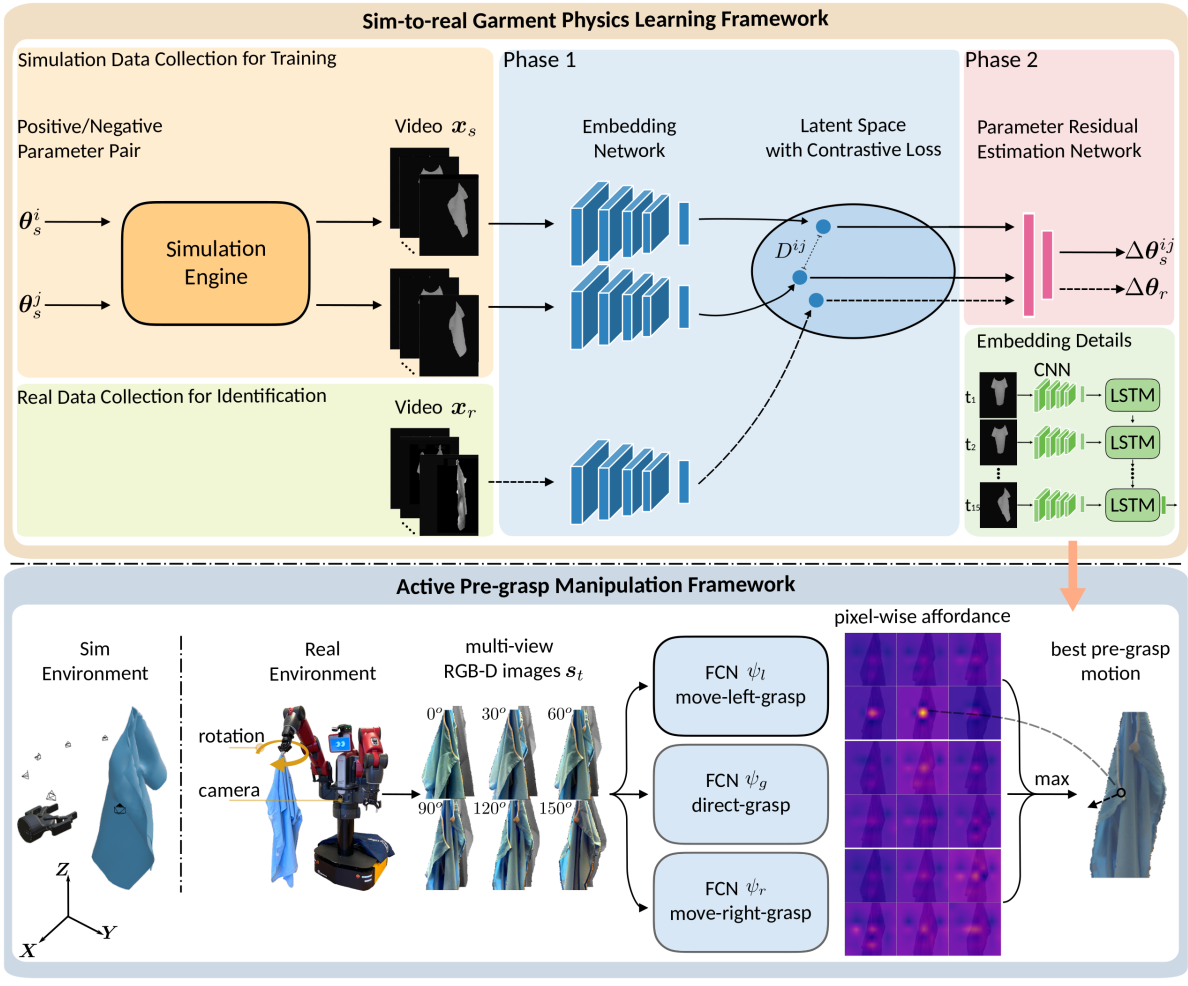


Fig. 3. Proposed framework of active pre-grasp manipulation policy learned in a sim-to-real manner. (**Top**) Sim-to-real garment physics learning. From simulation only, garment physical similarity is learned in the embedding space with contrastive loss, and the parameter error is learned from the embedded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

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]

simulation in our experiments, as presented in Table 2. The simulation properties dataset is described as

$$\Theta_s \equiv \left\{ \theta_s^i = \left[P_1^i, \ldots, P_6^i \right] \right\}^{i=1,2,\ldots,N}$$

where $[P_1, \ldots, P_6]$ denotes six physical properties, θ_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*). Because similar properties always lead to similar garment behaviors in the simulation, we defined that video clips generated from similar physical properties are considered as positive pairs, whereas negative pairs have dissimilar physical parameters. In total, 100 classes of simulation properties have been 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 used these

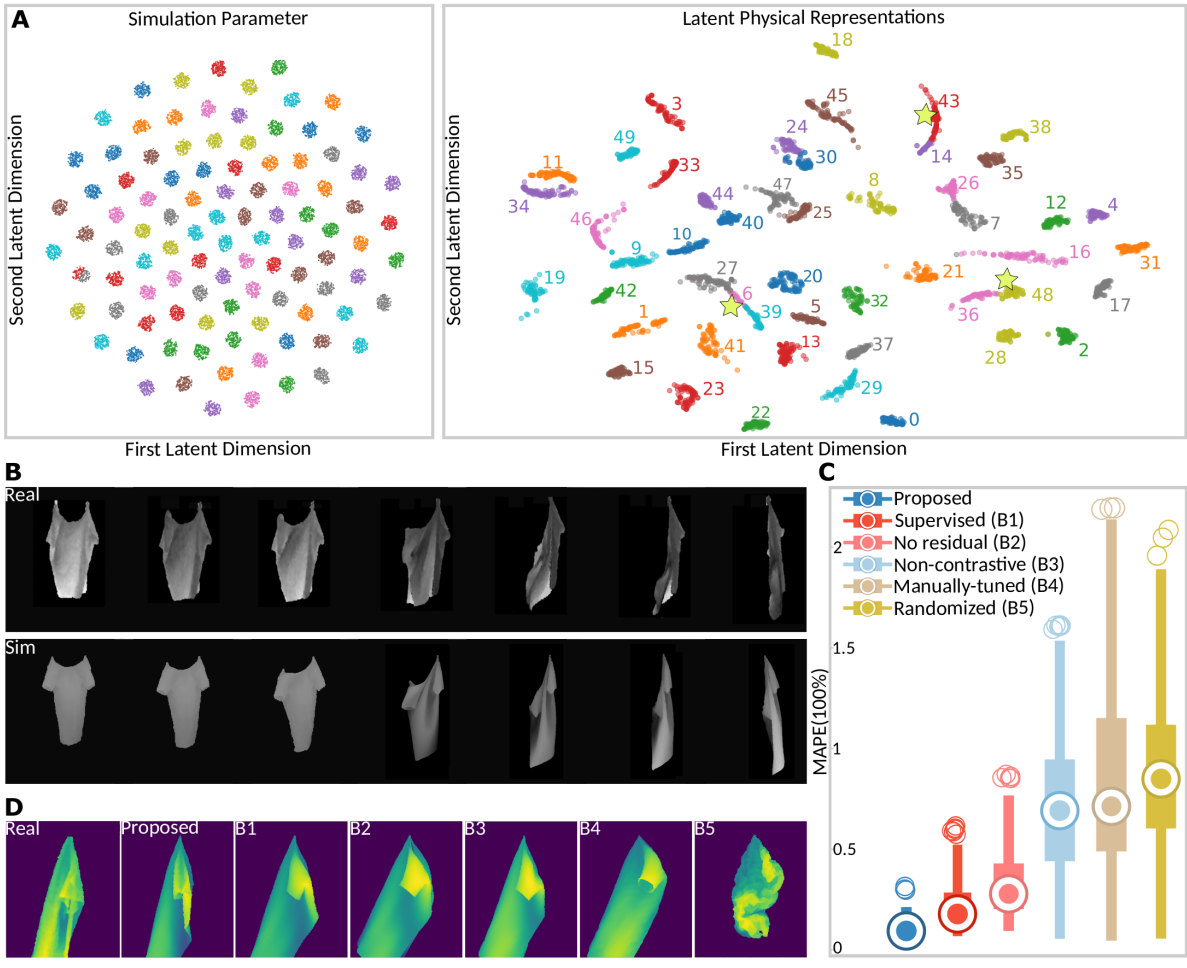


Fig. 4. Results of contrastive learning garment physics. (**A**) Results of ablation studies. The left graph shows the 2D *t*-SNE visualization of the generated simulation properties clustered into 100 classes. The right graph shows the 2D *t*-SNE visualization of a subset of the corresponding learned embedding space. Number annotations represent different classes. Embedding points from the same class are depicted with the same color. For clarity, the first 50 classes are shown in this figure and 10 colors are used iteratively. Stars (near classes 6, 43, and 48) here represent the embeddings of the three real gowns. (**B**) 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 the MAPEs of estimated garment physical properties using the proposed method and five baselines. The central dot corresponds to the median value of the errors, whereas 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.

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 have been generated between [0,1] and mapped to the actual value afterward. The SD of the clusters is set as 0.005 for each set of parameters to make sure that parameters in the same class are closer to each other than to other classes. Figure 4A (left) visualizes the generated simulation properties using two-dimensional (2D) *t*-distributed stochastic neighbor embedding (*t*-SNE) (*68*).

Figure 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) tend 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 each other, with 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

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

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 6000 synthetic testing dataset, which shows a high accuracy of physics inference.

Comparisons with baselines

Next, we benchmarked the proposed method against five baselines, including baseline 1, parameter identification using a classic supervised neural network (*69*); baseline 2, parameter identification with

contrastive loss (*70*); baseline 3, parameter residual estimation without contrastive loss (*71*); baseline 4, manually tuned parameters (*26*); and baseline 5, randomized parameters (*41*, *64*). For comparison purposes, the embedding functions are all formatted as the same network structures in the above baselines.

We first used only synthetic data to evaluate the performance of physics learning in this experiment. Figure 4C presents the results of estimated physical attributes. Three conclusions can be drawn from this figure: (i) The proposed method achieves the smallest MAPE (1.57%). (ii) 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. (iii) Baseline 2 obtains 13.88% median value of MAPE on the 6000 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 state transitions of deformable objects 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 drop has been collected, as shown in Fig. 4B. The garment physical parameters have been predicted using the proposed method and the five baselines described in the last section. Figure 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. Figure 4B shows the garment drop snapshots from the real garment and the synthetic garment with our estimated physics, which also indicates identical deformation. Because 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

Figure 3 (bottom) illustrates the framework of the proposed active pre-grasp manipulation learner. Multiview RGB-D images of the hospital gown are fed into three fully convolutional networks (FCNs) to 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 toward good behaviors. The policy is first pretrained using synthetic demonstration data and fine-tuned with trial and error in simulation. Then, on the basis of the training in simulation, the policy is transferred to the 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 compared our method with three ablations: ablation 1, training using only real demonstration data; ablation 2, training using only real active trial; and ablation 3, training using only real demonstration data and active trial. For our method, 3000 sets of demonstration states (2000 data in simulation, 1000 in real world, 3000 * 6 RGB-D images in total) have been collected by five human participants. For the ϵ-greedy exploration strategy, ϵ was initialized at 0.5 in simulation and annealed over time, whereas it was initialized at 0.1 in real world to highlight the immediate performance of sim-to-real policy transfer.

Figure 5 (top) presents the training process of learning a manipulation action (Fig. 1B) 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 to A3 ablations and the proposed method in this figure, we could draw four conclusions. First, the proposed method is capable of learning effective pre-grasping policies, achieving a success rate of more than 92% in real-world experiments. Second, 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 about 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. Third, through comparisons between ablation 2 and ablation 3, we can see that the demonstration data help the algorithm learn not only faster (higher performance in the early training stage) but also better (higher performance after active trial). Fourth, the comparisons between ablation 3 and the proposed method show that the learned policy from simulation has been transferred to the real world, especially when real-world data are insufficient.

Comparisons with a grasping-only policy

We also investigated whether the proposed pre-grasp manipulation leads to more robust dressing than the grasping-only policy. For the latter method, we trained a supervised CNN to estimate the grasping point directly from each image with the same amount of demonstration data as in our method. This method was also adopted in (*29*, *27*). We tested both methods on the physical robot performing 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 that our proposed pre-grasp manipulation (represented as a star) outperforms the grasping-only policy (green dot) in terms of dressing success rate. This result is expected because the grasping-only policy may successfully grasp the garment but could cause multilayer cloth grasping that occludes sleeve openings and thus yield dressing failures.

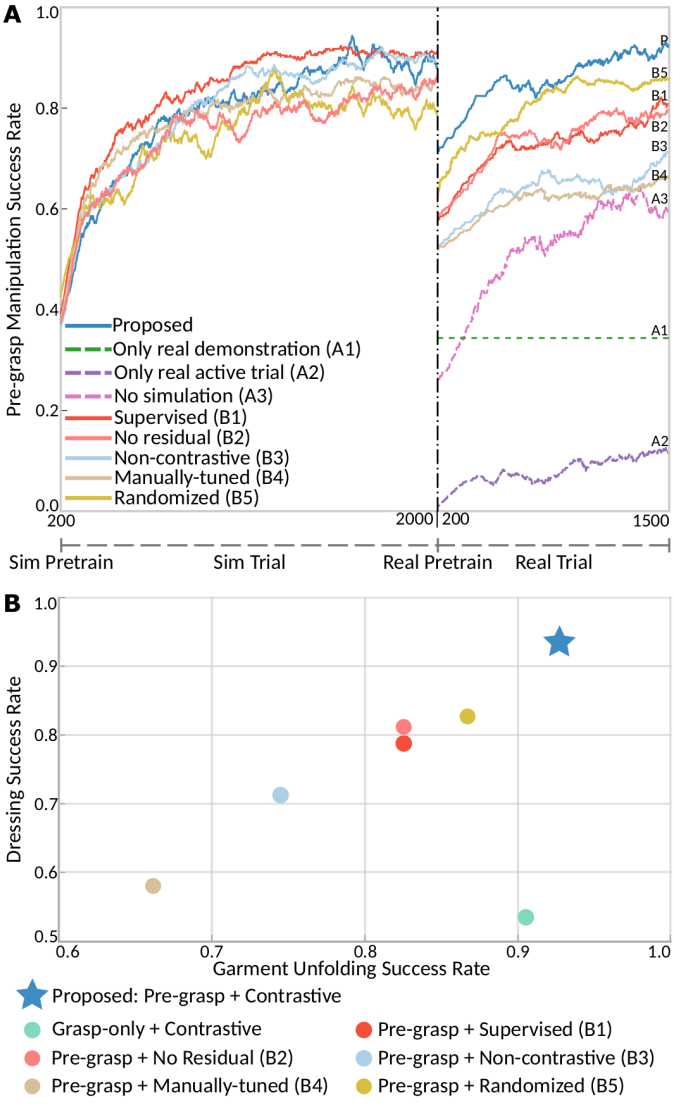


Fig. 5. Results of sim-to-real garment pre-grasp manipulation evaluation against ablations and state-of-the-art baselines. (**A**) Training process of pre-grasp manipulation policy learning on one hospital gown. This panel shows the comparison results against ablations of active learning and baselines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

Sim-to-real pre-grasp manipulation analysis

In this experiment, we investigated how the estimated real garment physics affects garment unfolding performance. We deployed the estimated physical properties from our method and five baselines to train the garment pre-grasp manipulation policies following the same procedure. From the lines of B1 to 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 because 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 those of the other five baselines, which shows that more gained knowledge from simulation has been transferred to the real world.

Sim-to-real robotic-dressing analysis

Last, we deployed the aforementioned 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 styles but different physical attributes have been used in our experiments. We estimated their physical properties using our method and five baselines and trained 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. Figure 5 (bottom) shows that 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 whether the grasping point is accurately estimated in stage C.

DISCUSSION

This work addressed the necessary subtasks for a complete dressing pipeline intended for people with weak or contracted arms and validated on a medical manikin. The high success rate, with various hospital gowns, is notable given the complexity of the pipeline. Because 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. This model-free pixel-wise parameterization also allows the proposed method to be easily generalized to different manipulation tasks with deformable objects 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 solution tractability in mind. Thus, our 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 dragging and folding garments, pouring fluids, or gathering granular materials.

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 pretraining manipulation policies than using only trial-and-error data from the robot because the demonstration data contain more diverse and successful pre-grasping examples. An interesting question for future work is to investigate how to 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 trained the pre-grasp manipulation policy for each gown in simulation with its estimated physics and efficiently transferred 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.

Because the procedures of the dressing pipeline were 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 investigations that relax 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 weighing 18 kg to simulate people with weak or contracted arms, as in the 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 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 that considers potentially bent elbows of real patients could be explored.

MATERIALS AND METHODS

Dressing pipeline learning details

We used 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 was prebuilt using the LiDAR for robot navigation [Ridgeback navigation stack (*76*)], whereas three waypoints were annotated near the garment rail and two sides of the bed, respectively. In stage A, the robot navigates to the first waypoint, captures an 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 and estimates the manipulation actions (Fig. 1B) on the left and right sides of the garment in sequence to unfold the garment. The two manipulation actions (3D positions, orientations, and action primitives) were learned using our proposed sim-to-real active pre-grasp manipulation approach. These points were approximately localized at two corners of the garment collar, which facilitates full unfolding. The unfolding reduces the number of possible configurations when the gown was picked up randomly from the rail. The robot then recognizes the correct side of the garment using a binary classification CNN, 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 ensures 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 that the whole upper limb is within the robot manipulation workspace. The manikin's postures were tracked using HRNet library in real time (*77*).

Stage C of dressing the manikin's arms

In this stage, the robot localizes the grasping point (Fig. 1C), grasps the garment with its right gripper, lifts up the manikin'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, and performs the manikin's left arm dressing assistance in the same procedure.

Three key motions in this stage are garment grasping, manikin's arm lifting, and dressing: (i) Because the garment has been fully unfolded into a spread-out configuration, a grasping-only policy is robust enough in this stage for dressing the manikin. We used supervised pixel-wise CNNs 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. (ii) Imitating a human nurse who offers external support to hold the user'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 4-df top-down movements and lifts it up. The grasping yaw orientation is calculated according to the upper arm posture. (iii) 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

Last, the robot spreads the gown to cover the manikin'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 manikin's right shoulder position) and pulled to the manikin's shoulder. This process can be repeated on the contralateral of the garment to fully cover the upper body. Here,

garment segmentation was learned in the same sim-to-real manner as in stage A.

Contrastive learning garment physics framework

Figure 3 (top) illustrates the framework of the proposed learning algorithm in two phases: (i) learning garment physical similarity in phase one, inspired by (*70*), which has successfully measured physical properties for cloth in the wind, and (ii) 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 $\mathbf{x}$ (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: (i) the garment is initially hung by two grasping vertices and (ii) 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 because it closely resembles the real hospital gown in this work. The range of the parameters is 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 $\mathbf{x}_s \in \mathbb{R}^{N\times H\times W}$ are generated from simulation. The camera frame rate in Blender is set the same as the real camera (30 Hz). To match the dynamics among different video clips, we extract the same 15 frames in each video clip using the timestamp of the first image in the series.

Physical similarity learning

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

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

where $\mathbf{X}_1, \mathbf{X}_2$ 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, and D denotes the Euclidean distance between embedded representations of $\mathbf{X}_1, \mathbf{X}_2$

$$D = \|f_e(\mathbf{X}^1), f_e(\mathbf{X}^2)\|$$

where $\mathbf{f}_e(\mathbf{x})$ 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 long-term recurrent convolutional networks (LRCNs) (*79*), the embedding function $\mathbf{f}_e(\mathbf{x})$ is formatted as a deep neural network combining ImageNet-pretrained DenseNet-121 with long short-term memory (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 $(f_e(\mathbf{x}_s^i), f_e(\mathbf{x}_s^j))$, 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 $\mathbf{x}_r$ is collected using a depth camera. The clip $\mathbf{x}_r$ is then mapped onto the same embedded space using embedding network $f_e(\mathbf{x}_r)$. Its nearest neighbor in the embedded space is found $f_e(\mathbf{x}_r^n)$, where $\mathbf{x}_r^n$ is the corresponding simulated video clip. $f_e(\mathbf{x}_r)$ and $f_e(\mathbf{x}_r^n)$ are treated as the input of the parameter residual estimation network. The parameters of the real garment $\boldsymbol{\theta}_r$ are obtained as

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

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

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*) that 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, multiview RGB-D image representations of the gown are observed to serve as the state $\mathbf{s}_t$. Specifically, six RGB-D images covering 180° around the garment (different multiples of 30°) are captured on the x-y plane. In the simulation environment, six 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 multiview images from one on-board camera. In our case, each pixel spatially represents approximately a vertical column of 3D space with 32 mm^2 base area in the agent's workspace.

Primitive actions

We parameterize each action $\mathbf{a}_t$ 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 $\mathbf{q}$ of pixel $\mathbf{p}$ in one of $k = 6$ orientations. We account for different primitive orientations using the orientations of camera views. We have six camera views $\boldsymbol{\gamma}_c^i \mid_{i=1,2,\ldots,6}$ covering 180° around the garment on the x-y plane. Thus, we define six discretized primitive action orientations on the x-y plane as $\boldsymbol{\gamma}_p^i = \boldsymbol{\gamma}_c^i + \Delta\boldsymbol{\gamma}$, where $\Delta\boldsymbol{\gamma}$ is manually designed as 15° in our case. Therefore, each action is designed as 4-df motion (x, y, z, and x-y orientation)

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

The pre-grasp manipulation primitive behaviors are defined as follows:

1) Move-left-grasp: $\mathbf{q}$ denotes the starting position of a 5 cm moving left before grasping in one of $k = 6$ orientations. $\mathbf{q}$ 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 γ_p^i on the x-y plane, whereas γ_p^i is decided by the view of the image that pixel $\mathbf{p}$ belongs to.

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

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

Learning pre-grasp affordance functions

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

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: $\text{argmax}_{(\psi,\mathbf{q},\gamma)}(\psi_l(\mathbf{s}_t), \psi_g(\mathbf{s}_t), \psi_r(\mathbf{s}_t))$. The action $\mathbf{a}_t$ 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 reward are usually challenging. Inspired by the method of deep Q-learning from demonstration (*82*), we use demonstration data to guide and accelerate the agent toward good behaviors. On pre-grasp attempt t, multiview RGB-D images are captured. A human participant is asked to annotate the pixel p. Then, the robot executes at the 3D location $\mathbf{q}$ of pixel $\mathbf{p}$ 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 is 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 pretraining the network $\pi(\mathbf{s}_t)$. 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

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

where F_π is the policy function to estimate affordance and $r_{\mathbf{a}i}(\mathbf{s}_i)$ 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 were 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

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

where $\mathbf{a}_{i'}$ represents the set of all available actions, $\mathbf{a}_D$ is the positive action the expert demonstrator took in state $\mathbf{s}_i$, and $H_\pi(\mathbf{s}_i, \mathbf{a}_D)$ 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 $\mathbf{a}_t$, the robot receives a reward $r_{\mathbf{a}t}(\mathbf{s}_t) = 1$ if it successfully grasps the gown with a clear single layer and $r_{\mathbf{a}t}(\mathbf{s}_t) = 0$ otherwise. The reward is described as

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

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

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, and 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 and 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. In real-world experiments, various garment configurations at each state are achieved by either human interference or robot arm shaking.

SUPPLEMENTARY MATERIALS

www.science.org/doi/10.1126/scirobotics.abm6010
Sections S1 to S5
Figs. S1 to S4
Table S1

REFERENCES AND NOTES

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. *Int. J. Soc. Robot.* **6**, 213–227 (2014).

2. H. Robinson, B. MacDonald, E. Broadbent, The role of healthcare robots for older people at home: A review. *Int. J. Soc. Robot.* **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. *Arch. Phys. Med. Rehabil.* **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, in *Proceedings of the 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, in *Proceedings of the 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. *Adv. Robot.* **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, in *Proceedings of Conference on Advances In Robotics* (ACM, 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 International 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, in *Proceedings of the 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, in *Proceedings of the 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, in *Proceedings of the 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 Robot. Autom. Lett.* **3**, 2245–2252 (2018).
13. E. Pignat, S. Calinon, Learning adaptive dressing assistance from human demonstration. *Robot. Auton. Syst.* **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 Trans. Robot.* **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, in *Proceedings os the 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. *Auton. Robot.* **43**, 2183–2207 (2019).
17. Y. Gao, H. J. Chang, Y. Demiris, Iterative path optimisation for personalised dressing assistance using vision and force information, in *Proceedings of the 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 Trans. Robot.* **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); www.youtube.com/watch?v=-lkJ5ev3edM&t=263s.
22. Dress a Resident with a Weak Arm CNA Skill (2021); www.youtube.com/watch?v=eI4wBjavIBk.
23. H. Yin, A. Varava, D. Kragic, Modeling, learning, perception, and control methods for deformable object manipulation. *Sci. Robot.* **6**, eabd8803 (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. *Int. J. Robot. Res.* **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. *Front. Robot. 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:2003.12698 (2020).
27. E. Corona, G. Alenya, A. Gabas, C. Torras, Active garment recognition and target grasping point detection using deep learning. *Pattern Recogn.* **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, in *Proceedings of the 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, in *Proceedings of the 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 Trans. Robot.* **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)* (Springer, 2019).
32. K. Saxena, T. Shibata, Garment recognition and grasping point detection for clothing assistance task using deep learning, in *Proceedings of the 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, in *Proceedings of the 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, in *Proceedings of the 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. *Robot. Auton. Syst.* **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, in *Proceedings of the 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)* (PMLR, 2021), pp. 24–33.
39. K. S. Sahari, H. Seki, Y. Kamiya, M. Hikizu, Edge tracing manipulation of clothes based on different gripper types. *J. Comput. Sci.* **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 Robot. Autom. Lett.* **5**, 1111–1118 (2020).
41. J. Matas, S. James, A. J. Davison, Sim-to-real reinforcement learning for deformable object manipulation, in *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, in *Proceedings of the 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, in *Proceedings of the 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* (IEEE, 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. *Sci. Robot.* **4**, eaau5872 (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)* (PMLR, 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:1610.03518 [cs.RO] (11 October 2016).
49. S. Kolev, E. Todorov, Physically consistent state estimation and system identification for contacts, in *Proceedings of the 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. *Proc. Robot.* **14**, 1–9 (2018).
52. A. Zeng, S. Song, J. Lee, A. Rodriguez, T. Funkhouser, Tossingbot: Learning to throw arbitrary objects with residual physics, in *Proceedings of the IEEE Transactions on Robotics* (IEEE, 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, in *Proceedings of the 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. *Int. J. Robot. Res.* , 0278364920954896 (2020).
55. P. Chang, T. Padif, Sim2real2sim: Bridging the gap between simulation and real-world in flexible object manipulation, in *Proceedings of the 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, in *Computer Graphics Forum* (Wiley Online Library, 2012), vol. 31, pp. 519–528.
57. H. Wang, J. F. O'Brien, R. Ramamoorthi, Data-driven elastic models for cloth: Modeling and measurement. *ACM Trans. Graphics* **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* (IEEE, 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, in *Proceedings of the 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, in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 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)* (PMLR, 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)* (PMLR, 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, in *Proceedings of the 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, in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 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. *Adv. Neural Inf. Proces. Syst.* **33**, 18661–18673 (2020).
68. L. van der Maaten, G. Hinton, Visualizing data using t-SNE. *J. Mach. Learn. Res.* **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, in *Advances in Neural Information Processing Systems* (PMLR, 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, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 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, in *Conference on Robot Learning (CoRL)* (PMLR, 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. *J. Pathol. Informatics* **11**, 38 (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)* (PMLR, 2020).
75. IEC60601 (2021); https://en.wikipedia.org/wiki/IEC_60601.
76. Ridgeback Navigation Stack (2021); https://github.com/ridgeback/ridgeback.
77. K. Sun, B. Xiao, D. Liu, J. Wang, Deep high-resolution representation learning for human pose estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2019), pp. 5693–5703.
78. A. Veit, S. Belongie, T. Karaletsos, Conditional similarity networks, in *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, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2015), pp. 2625–2634.
80. S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
81. G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 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, in *Proceedings of the AAAI Conference on Artificial Intelligence* (PMLR, 2018).
83. A. Singh, L. Yang, K. Hartikainen, C. Finn, S. Levine, End-to-end robotic reinforcement learning without reward engineering, in *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)* (PMLR, 2018), pp. 40–52.
85. C. Heindl, S. Zambal, J. Scharinger, Learning to predict robot keypoints using artificially generated images, in *Proceedings of the 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:1606.01540 [cs.LG] (5 June 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 Y.D. and, in part, by UKRI grant EP/V026682/1. **Author contributions:** F.Z. developed algorithms, performed experiments, and wrote the manuscript. Y.D. conceived the assistive dressing scenarios, contributed in results analysis, and revised the manuscript. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** All data and code needed to evaluate the conclusions in the paper can be found at https://imperialcollegelondon.box.com/s/oc1spqys3u3nrgtjgj2ntzr037jmrqge.

Submitted 28 September 2021
Accepted 10 March 2022
Published 6 April 2022
10.1126/scirobotics.abm6010

Science Robotics

Learning garment manipulation policies toward robot-assisted dressing

Fan Zhang and Yiannis Demiris

Sci. Robot. **7** (65), eabm6010. DOI: 10.1126/scirobotics.abm6010

Science Robotics (ISSN 2470-9476) is published by the American Association for the Advancement of Science. 1200 New York Avenue NW, Washington, DC 20005. The title *Science Robotics* is a registered trademark of AAAS.