Do You Need a Hand?
a Bimanual Robotic Dressing Assistance Scheme

Zhu, Jihong; Gienger, Michael; Franzese, Giovanni; Kober, Jens

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Do You Need a Hand? – A Bimanual Robotic Dressing Assistance Scheme

Jihong Zhu ⓘ *, Member, IEEE*, Michael Gienger ⓘ *, Member, IEEE*, Giovanni Franzese ⓘ *, Member, IEEE*,
and Jens Kober ⓘ *, Senior Member, IEEE*

Abstract—**Developing physically assistive robots capable of dressing assistance has the potential to significantly improve the lives of the elderly and disabled population. However, most robotics dressing strategies considered a single robot only, which greatly limited the performance of the dressing assistance. In fact, healthcare professionals perform the task bimanually. Inspired by them, we propose a bimanual cooperative scheme for robotic dressing assistance. In the scheme, an interactive robot joins hands with the human thus supporting/guiding the human in the dressing process while the dressing robot performs the dressing task. We identify a key feature: the elbow angle that affects the dressing action and propose an optimal strategy for the interactive robot using the feature. A dressing coordinate based on the posture of the arm is defined to better encode the dressing policy. We validate the interactive dressing scheme with extensive experiments and also an ablation study.**

Index Terms—**Dual arm manipulation, human-centered robotics, personalized dressing assistance, physically assistive devices.**

I. INTRODUCTION

THE shortage of caregivers is a pressing challenge worldwide due to a decrease in birth rates combined with an increase in life expectancy: According to the WHO, by 2030, 1 in 6 people in the world will be aged 60 years or over. By 2050, the world's population of people aged 60 years and older will double [1]. Assistive robots capable of physically aiding humans are promising to tackle this challenge. Among various tasks that the caregiver performs on a daily basis, the dressing was reported to be the greatest burden while the least automated [2]. In this article, we tackle the problem of dressing a human arm with a bimanual robot setup.

Robot-assisted dressing is challenging as it involves direct physical interaction with flexible clothes and humans. More specifically, it differs from a conventional human–robot interaction task such as comanipulation or cocarrying [3], [4], [5] where the object allows immediate force propagation. In the dressing assistance, the object i.e., the cloth, is soft and deformable; thus, the interaction force is hard to obtain. Due to the absence of direct force feedback and heavy occlusion during dressing, arm movement estimation during dressing is also difficult.

One common assumption to simplify the task is to consider static arm posture during dressing. Then, the main challenge becomes adapting the dressing policy to different static postures. Learning from demonstrations (LfD) is often used in obtaining a generalized policy [6], [7], [8].

However, a static arm posture is a very strong assumption. To make the proposed scheme work in practice, we need to consider dynamic poses during dressing. Since the task largely depends on postures, arm movement estimation is crucial.

Computer vision is a popular way to track arm postures. It is also used in the early work in assistive dressing [9] for tracking upper limb movement. However, the inevitable occlusions during dressing render most of the existing vision-based human posture tracking algorithms fail [10], [11]. Additional sensory modules such as proximity sensors may be employed for posture estimation [12].

To overcome the difficulties faced with conventional assistive dressing strategies, we draw inspiration from human-to-human dressing assistance from healthcare professionals.[1] Our proposed framework (hardware setup shown in Fig. 1) resembles the method adapted by health professionals. It requires two robots to perform the dressing assistance task. An interactive robot $\mathcal{I}$ that reaches out for holding hands with the human then supports/guides the human to facilitate the dressing process. The dressing robot $\mathcal{D}$ performs the main task, dressing. Hand holding not only makes the dressing task easier to execute by supporting/guiding the human arm movement but also results in the tracking of the arm posture with proprioceptive sensors only.

The main contribution of the article is the proposal of a bimanual dressing assistance framework that is inspired by caregivers doing the dressing assistance task. In the framework, instead of

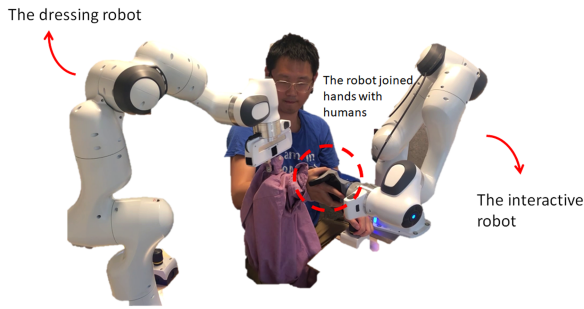[1][Online]. Available: https://youtu.be/Wn1OEfQt0ow?t=93

Fig. 1. Bimanual robot setup for the cooperative dressing. We employ two Franka Emika robots. The dressing robot grips the cloth and executes the dressing policy. The interactive robot is equipped with a SoftHand from qbrobotics that provides humanlike hand-holding. It guides the human in the dressing process.

considering interaction forces during dressing, which is often hard to obtain in practice [13], we offer a novel perspective by analyzing dressing geometrically. Benefiting from this new insight, we identify a key feature: the elbow angle in the human posture that affects dressing policy. The feature is then used for designing an optimal stretch policy for the interactive robot. For the dressing, in contrast to encoding the movement in Cartesian space, we define a task coordinate system for flexible policy encoding from demonstrations.

The rest of the article is organized as follows. In Section II, we survey the related work in three subcategories, namely: 1) robotic dressing assistance, 2) bimanual manipulation, and 3) arm poster estimation. Section III provides an overview of the overall framework and the underlying assumptions. Section IV introduces the feature that affects the dressing strategy and then designs an optimal strategy for the interactive robot $\mathcal{I}$ based on the feature. With the information on the hand position given by $\mathcal{I}$, we can solve the arm posture with proprioceptive sensors only. Later in Section V, we propose a dressing coordinate dependent on arm postures and then learn the dressing policy from expert demonstrations in the dressing coordinate. The overall framework is validated with robotic experiments in Section VI. Finally, Section VII concludes this article.

II. RELATED WORK

A. Robotic Dressing Assistance

Robotic dressing assistance is a recurrent topic that broadly covers dressing tops (t-shirts, jackets, etc.) [14], [15], trousers [16], and even shoes [17], [18]. Since in this article, our focus is dressing the upper body, we review recent advances, particularly in this area.

Dressing assistance is a skill mastered by humans. Therefore, LfD-based approaches are popular in obtaining motor skills in dressing. Pignat and Calinon [6] learned a hidden semi-Markov model from human demonstrations to encode the dressing skills. Hoyos et al. [7] implemented an incremental learning of task-parameterized Gaussian mixture models (TP-GMM) that allows the generalization of the dressing task. More recently, Zhu et al. [8] proposed a method to reduce the number of demonstrations needed for training the TP-GMM to encode a dressing policy in different static postures.

Recent research also explores the interaction between a robot and a human in the dressing process with reinforcement learning in a simulation environment [19], [20]. Yu et al. [21] learned an outcome classifier for dressing tasks with simulated haptic data. Instead of relying on simulations, data collected in the real dressing scenarios suggest that the interaction force between the cloth and the human allows for predicting dressing outcomes with high accuracy [22]. Later, Erickson et al. [23] conducted an investigation with physics-based simulation suggesting that the interaction force can be recovered with only end-effector measurements.

Further research in this area is concerned with cloth state modeling [24], safe motion planning [25], [26], grasping point selection [27], and novel gripper design for dressing assistance [28].

In the aforementioned research, the dressing tasks were mostly conducted with a single manipulator. The only papers considering bimanual dressing are [14], [15], and [20]. However, they consider the bimanual in a completely different setup where each robot is dressing one human arm. Therefore, they adopt the *one-robot-to-one-arm* setup just like the rest of the papers. In our proposal, we have instead a *two-robot-to-one-arm* setup, where a robot guides humans to facilitate the dressing task of the other robot. Zhang and Demiris [29] consider also a *two-robot-to-one-arm* setup while dressing the hospital gown with a mannequin lying on the bed. The task in [29] is in a completely different setting; thus, the employed framework is radically different from ours.

B. Bimanual Manipulation

In the proposed framework, we adopt implicit master–slave coordination between the interactive and the dressing robots. The dressing policy is dependent on the human arm posture, which is controlled by the interactive robot from the applied guidance force on the human hand. Such task setup is referred to as asymmetrical bimanual manipulation in the bimanual taxonomy literature [30].

The asymmetrical bimanual task that was investigated in the previous work includes vegetable peeling [31], cloth manipulation [32], assembly [33], and cooking [34]. These tasks usually do not require interaction with a human as in the dressing assistance.

More relevant bimanual assistive tasks: Xu et al. [35] designed an autonomous wheelchair with two robots to assist humans in daily tasks. Connan et al. [36] employed a humanoid robot to perform tasks, such as removing the lid, unscrewing a bottle, and pouring water. Edsinger and Kemp [37] discussed three key themes in bimanual manipulation design in assistive tasks: 1) cooperative manipulation, 2) tasks relevant features, and 3) let the body do the thinking and analyzing them in different assistive scenarios. These key themes remain very relevant to us in designing the framework for this article.

To date, robotics dressing assistance (specifically dressing clothes) has rarely been considered a bimanual task that requires both robots. However, as we argued in Section I, the additional robot can provide support and guidance to the arm being dressed,
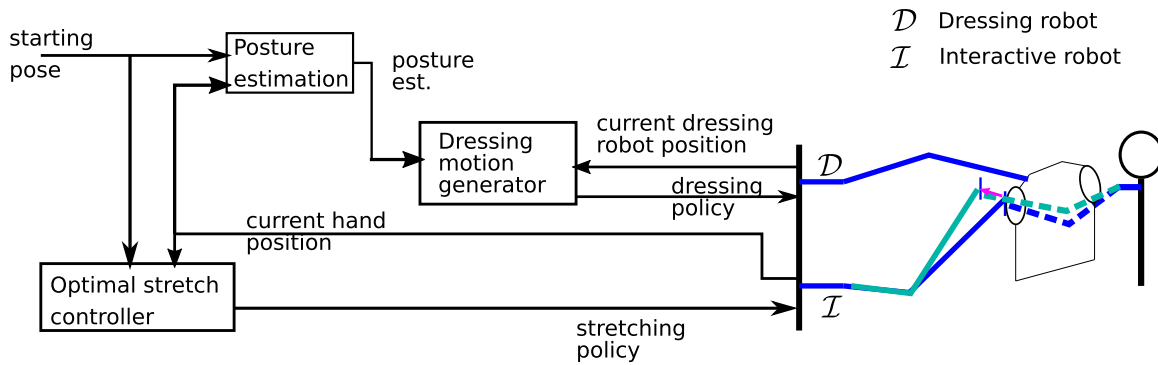
Fig. 2.   Schematic diagram of the cooperative dressing framework. The framework contains three blocks: 1) a posture estimation, 2) a dressing motion generator, and 3) an optimal stretch controller. The posture estimation takes in an initial posture of the human arm and the current hand position and outputs the estimated current arm posture. The posture is then fed into the dressing motion generator together with the current dressing robot position to derive the next step dressing policy. The optimal stretch controller takes the initial human arm posture and the current hand posture yields the stretching policy for the interactive robotics arm.

which makes the dressing easier and more comfortable. As the scheme is employed by caregiving professionals, humans might feel more contented dressing in such a composition. By adopting the bimanual setup in dressing, our work is the first of its kind and represents a paradigm shift in the thinking of the dressing assistance task.

C. Arm Posture Estimation

The dressing policy is highly dependent on human arm posture. As the guidance force at hand will change the human arm posture in our framework, robust arm posture tracking/estimation is required for dressing policy generation.

Due to heavy occlusion during dressing, the task is particularly challenging for popular vision-based deep learning posture tracking algorithms [38], [39], [40], [41].

Human modeling is employed for tracking the human posture under occlusions. A preliminary study in dressing assistance considers human modeling combined with visual tracking in personalized dressing: Gao et al. [9] adopted a mixture of Gaussians for modeling, and later a Gaussian process latent variable model was employed for posture estimation in [10]. Recently, Chance et al. [11] tackled the occlusion problem by employing a recurrent neural network (RNN) trained on human–human interaction data to predict the elbow position during dressing. Rather than relying on vision alone, previous works also explored capacitive proximity sensing [12] for posture tracking.

In this article, we will demonstrate that tracking is made easy by hand-holding. With the hand position always known, arm posture tracking can be formulated as an inverse kinematic problem that has analytical solutions and is solvable in real time with proprioceptive sensors.

III. FRAMEWORK OVERVIEW

We envision a bimanual dressing assistance system that holds the following:
1) provides support/guidance to the human arm so as to facilitate dressing;
2) tracks the arm posture without additional sensors (vision/force/proximity);

3) can be programmed easily without expert knowledge of robotics;
4) flexible to arms of different lengths.

Fig. 2 presents the overall cooperative dressing scheme that fulfills the earlier goals. The framework contains three parts: 1) an optimal stretch controller, 2) a posture estimation, and 3) a dressing motion generator.

The optimal stretch controller is designed by analyzing dressing behaviors under different arm postures to facilitate dressing. The optimal stretch controller takes in the initial human arm posture together with the current hand position and yields the stretching policy for the interactive robot (see Section IV-B).

Benefiting from the hand holding, the real-time posture tracking is done by the posture estimation module, which takes in the same inputs as the stretch controller and outputs the estimated current arm posture (see Section IV-C). It relies solely on proprioceptive sensors.

Using the estimated posture, we define a posture-dependent dressing coordinate and transform the robot movement in the Cartesian coordinate to the dressing coordinate for learning the dressing motion generator from human demonstrations. One of the benefits of employing LfD is that the motion can be programmed by nonexpert users [42], [43], [44]. The defined dressing coordinate enables flexible encoding of the dressing strategy to adapt to different arm lengths. Inputs to the dressing motion generator are the estimated posture together with the current dressing robot position. Using the inputs, it generates the next step dressing policy (see Section V).

The scheme described earlier is realized with a bimanual robot setup illustrated in Fig. 1. The dressing robot grips the cloth and executes the dressing policy. The interactive robot is equipped with a SoftHand from qbrobotics that provides humanlike hand-holding. It supports and guides the human in the dressing process.

In addition, we employ the following assumptions.
1) The human hand is already in the sleeve opening.
2) The starting arm posture is known.
3) The shoulder remains static during dressing.
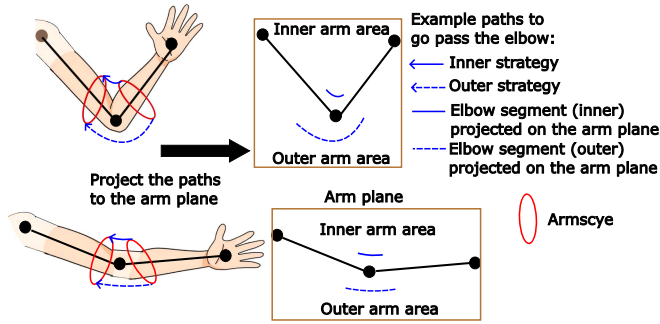4) The framework considers dressing one arm only.

Fig. 3. Illustration of the inner and outer path of the armscye. We consider the arm plane defined by the hand, elbow, and shoulder. The arm plane is divided into two parts: 1) the inner arm area with an angle less than π and 2) the outer area with an angle more than π.

IV. INTERACTIVE ROBOT—OPTIMAL STRETCH

In this section, we motivate a key feature in the human arm posture that affects the dressing policy. Based on this feature, we design an optimal stretch controller for the interactive robot. Finally, we present a scheme for the estimation of human arm posture during dressing.

A. Feature Affects Dressing—Elbow Angle

The dressing gets complicated around the elbow. If we simplify the forearm and the upper arm as two straight lines, then the discontinuity between these two lines is the linking point (elbow), which connects both lines. The degree of discontinuity can be represented by the elbow angle $\psi \leq \pi$. The angle ψ is a pure geometrical feature that influences the dressing policy.

If we neglect force and only consider geometry, dressing resembles a hotwire game. The cloth can be simplified into a ring that represents the armhole (more formally referred to as the armscye in the textile literature), and the arm is the wire that the ring needs to pass through. The goal of dressing is to transport the ring to a position above the shoulder.

However, a major difference that complicates the dressing task is the nature of the ring. In a hotwire game, the ring is rigid. All points on the ring move simultaneously with the grasping/control point on the ring. In dressing, due to the deformation nature of the armscye, we observe the property of diminishing rigidity [45] in the armscye—i.e., the effect of gripper motion along the armscye diminishes as the distance from the gripper increases. This property will affect our dressing strategy when the elbow angle is small. We will explain the reason in Fig. 3 in the following paragraph.

In Fig. 3, we define the arm plane: the plane given by shoulder, elbow, and hand positions. The plane is divided by the arm into two parts: 1) inner arm area (angle between the forearm and upper arm less than π) and 2) outer arm area (angle between the forearm and upper arm larger than π). Then, we project the dressing path onto the arm plane. The path can be divided into three segments. Two are segments that go past the forearm and upper arm. Another segment is the elbow segment that goes past the elbow and connects the forearm and upper-arm paths. We are interested in the elbow segment of the path, which is shown

in normal and dashed blue lines in Fig. 3. Using the projected paths, we distinguish two different strategies.[2]

1) *Inner strategy:* The projection of the elbow segment on the arm plane is within the inner arm area. This strategy offers a shorter path to go around the elbow.
2) *Outer strategy:* The projection of the elbow segment on the arm plane is within the outer arm area. This strategy offers a longer path to go around the elbow.

When ψ is small, taking the inner strategy as shown in blue lines in Fig. 3, the far end of the armscye where we have very little control needs to travel much longer to go pass the elbow (see Fig. 3, the dashed blue lines). There is a high possibility that it fails and results in the armscye getting stuck. It is more plausible in this case to take the outer strategy, which travels in the outer arm area, although travels longer in distance, it makes sure that the part with the least control is required to move as little as possible to minimize the chance of getting stuck. When ψ is larger, the distance difference between an inner and outer strategy diminishes and becomes zero when the arm is fully stretched: $\psi = \pi$.

To conclude, elbow angle ψ affects the dressing strategy. When ψ is large, the degree of discontinuity becomes small (the arm resembles a straight line), and we are more likely to take the inner strategy to have a smaller path length. When ψ is small, the degree of discontinuity gets large (the arm is more bent), and we will be more likely to take the outer strategy to avoid getting stuck. We will show this effect with expert demonstration data on different static arm postures (with different elbow angles) in Section VI-A.

B. Optimal Stretch Based on the Elbow Angle

The outer strategy requires a longer dressing trajectory, and when ψ is small, there is always the risk of getting stuck even taking the outer strategy. Thus, the objective of the interactive robot is to guide the human arm so that ψ increases as much as possible to avoid getting stuck and facilitate dressing.

The interactive robot is holding the human hand. Besides the singular configuration where the arm is fully stretched, the robot is able to guide the hand to move in all directions in 3-D, which is represented by the blue sphere in Fig. 4. In the figure, the arm posture is given by the shoulder, elbow, and hand joint positions: $\boldsymbol{P} = \{\boldsymbol{p}_s, \boldsymbol{p}_e, \boldsymbol{p}_h\}$. The movement direction $\delta \boldsymbol{d}$ that maximizes the increase of the elbow angle, here referred to as $\delta \boldsymbol{d}^*$, is aligned with the direction of $\overrightarrow{\boldsymbol{p}_s \boldsymbol{p}_h}$. Proving the optimal direction is straightforward, since $\psi \propto |\boldsymbol{d} + \delta \boldsymbol{d}|$, finding $\delta \boldsymbol{d}^*$ that maximizes ψ is equivalent to

$$\delta \boldsymbol{d}^* = \max_{\delta \boldsymbol{d}} |\boldsymbol{p}_s \boldsymbol{p}_h + \delta \boldsymbol{d}|, \text{ for } \delta \to 0, \text{ and } \boldsymbol{d} \in \mathrm{O}(3).$$

With the triangle inequality theorem, we have

$$|\boldsymbol{p}_s \boldsymbol{p}_h + \delta \boldsymbol{d}| \leq |\boldsymbol{p}_s \boldsymbol{p}_h| + |\delta \boldsymbol{d}|.$$

The maximum value for $|\boldsymbol{p}_s \boldsymbol{p}_h + \delta \boldsymbol{d}|$ is when the equality holds. And it only holds when $\delta \boldsymbol{d}$ and $\boldsymbol{p}_s \boldsymbol{p}_h$ are in the same direction.

---

[2]The trivial case of the shoulder, elbow, and hand is in a line: We cannot define a plane anymore, yet there is also no difference between an inner and outer arm area.
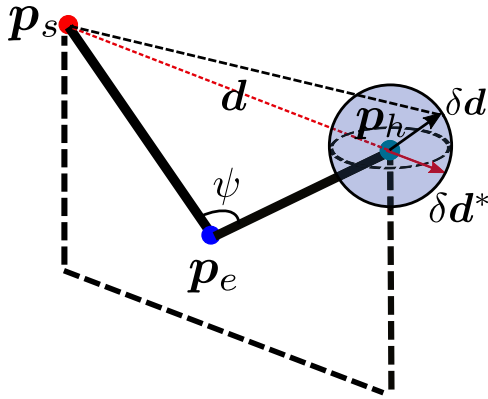
Fig. 4. Optimal direction that maximizes the increase of the elbow angle ψ. The arm posture is given by the shoulder, elbow, and hand joint positions: $\boldsymbol{P} = \{\boldsymbol{p}_s, \boldsymbol{p}_e, \boldsymbol{p}_h\}$. The optimal direction $\delta\boldsymbol{d}^*$ is aligned with the direction that connects the hand and the shoulder and marked with a red solid arrow. The triangle inequality theorem is demonstrated with another random direction $\delta\boldsymbol{d}$ and the resulting shoulder-to-hand distance is marked with a black dash line.

Therefore, to increase the elbow angle as much as possible, the guidance force direction from the interactive robot should always be aligned with the direction that connects the hand and the shoulder.

Knowing the optimal direction to move, we further implement the optimal stretch controller with Cartesian impedance control. The dynamic equation of the robot can be modeled according to

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q}, \dot{\boldsymbol{q}}) + G(\boldsymbol{q}) = \boldsymbol{\tau}_{\text{task}} + \boldsymbol{\tau}_{\text{ext}} \qquad (1)$$

where, in order from left to right, there are the mass, the Coriolis, and the gravitational term that depends on the joint configuration $\boldsymbol{q}$ and, on the right, the torque for the Cartesian (or task) control and the externally applied torque.

The input to the low-level controller is the desired Cartesian position (and orientation) obtained from the dressing policy. A PD control rule is used to generate the desired control force in the end effector simulating the impedance behavior. This is then converted into a control joint torque using the equivalent principle $\boldsymbol{\tau} = \boldsymbol{J}^{\top} \boldsymbol{F}_{\text{cart}}$.

Using the PD control rule and the force/torque equivalence principle, the task space torque is defined as

$$\boldsymbol{\tau}_{\text{task}} = \boldsymbol{J}^{\top} \left[\boldsymbol{K}(\boldsymbol{x}_d - \boldsymbol{x}) + \boldsymbol{D}(\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}) \right] + C(\boldsymbol{q}, \dot{\boldsymbol{q}}) + G(\boldsymbol{q})$$

where $\boldsymbol{J}$ is the geometric Jacobian, and the stiffness $\boldsymbol{K}$ and the damping $\boldsymbol{D}$ give the compliant behavior with a critically damped response; $\boldsymbol{x}_d$ and $\dot{\boldsymbol{x}}_d$ are the desired robot Cartesian position and velocity, respectively; $\boldsymbol{x}$ and $\dot{\boldsymbol{x}}$ are the current Cartesian position and velocity of the manipulator, respectively. Notice that the gravitational and the Coriolis terms are compensated to avoid having undesired behaviors on the simulated impedance, such as having a constant offset in the gravity direction. In our implementation, we set the desired velocity to zero.

Given a rotation matrix $\boldsymbol{\Gamma}$ of a local stiffness matrix with respect to the global coordinate, the equivalent stiffness matrix in global coordinates is

$$\boldsymbol{K}_{\text{global}} = \boldsymbol{\Gamma}^T \boldsymbol{K}_{\text{local}} \boldsymbol{\Gamma} \qquad (2)$$

where, in our framework, the rotation matrix $\boldsymbol{\Gamma}$ transforms the x-axis of the global coordinates in the direction of end effector movements that maximizes the increase of the human elbow angle ψ, i.e., $\overrightarrow{\boldsymbol{p}_s \boldsymbol{p}_h}$.

Additionally, we choose

$$\boldsymbol{K}_{\text{local}} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

in order to have a nonzero stiffness only in the direction of movement and complete compliance in the other two perpendicular directions: Such a controller will provide a guiding force in the optimal direction of hand movement that maximizes the increase of the elbow angle ψ. At the same time, the human can disagree and move easily in the plane perpendicular to the force direction.

C. Human Arm Posture Estimation

With the guidance force upon the hand, the human arm posture changes during dressing. For successfully dressing the human arm, we propose a real-time vision-free estimation of the human arm posture.

Consider two coordinate frames $\mathcal{O}^{\mathcal{D}}, \mathcal{O}^{\mathcal{I}}$ located at the base of each robot respectively. Any point $\boldsymbol{x}^{\mathcal{I}}$ in $\mathcal{O}^{\mathcal{I}}$ can be transformed into $\mathcal{O}^{\mathcal{D}}$ by

$$\boldsymbol{x}^D = \boldsymbol{R}\boldsymbol{x}^I + \boldsymbol{t} \qquad (3)$$

where $\boldsymbol{R}$ is a 3×3 rotation matrix and $\boldsymbol{t}$ is the translation vector. We can obtain $\boldsymbol{R}$ and $\boldsymbol{t}$ between two coordinates by calibration.

The human hand is joined by the interactive robot $\mathcal{I}$. Thus, its position can be regarded as the same as the end-effector position of the interactive robot $\boldsymbol{x}^I_{\text{end-eff}}$. Using (3), we can track the human hand position in $\mathcal{O}^{\mathcal{D}}$ given $\boldsymbol{R}$ and $\boldsymbol{t}$

$$\boldsymbol{p}^{\mathcal{D}}_h = \boldsymbol{R}\boldsymbol{x}^I_{\text{end-eff}} + \boldsymbol{t}. \qquad (4)$$

Assuming the initial arm posture in $\mathcal{O}^{\mathcal{D}}$ is known and the shoulder does not move during the dressing, we can move the origin of $\mathcal{O}^{\mathcal{D}}$ to the shoulder by translation, we denote this frame as shoulder frame. From now on, for simplicity, unless explicitly denoted with superscript $^{\mathcal{I}, \mathcal{D}}$, all positions are with regard to the shoulder frame with its origin at the shoulder.

Since the shoulder is assumed static during dressing, the posture estimation problem boils down to recovering the elbow position $\boldsymbol{p}_e$ given the hand position $\boldsymbol{p}_h$. A schematic diagram is shown in Fig. 5.

The human arm has seven degrees of freedom. Only four out of seven contribute to the displacement of the hand $\boldsymbol{p}_h$ [46]. The four joint angles $\boldsymbol{q} = [\alpha \, \beta \, \phi \, \gamma]^T$ are depicted in Fig. 6. A spherical joint located at the shoulder with two degrees of freedom (α, β), and another spherical joint at the elbow with two degrees of freedom (ϕ, γ). The shoulder, elbow, and hand representation in the shoulder frame can be transformed into the angle representation and vice versa (given forearm and upper-arm lengths). The hand position can be expressed as

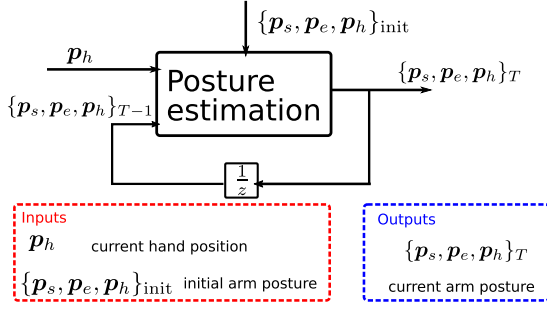$$\boldsymbol{p}_h = \boldsymbol{f}(\boldsymbol{q}). \qquad (5)$$

Fig. 5. Schematic diagram showing the inputs and outputs of the human posture estimation.
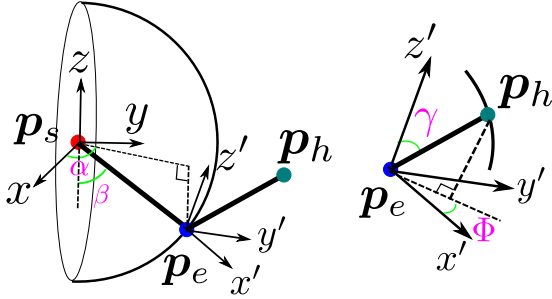

Fig. 6. Schematic representation of the human arm from the shoulder to the hand joint with four joint angles. A spherical joint located at the shoulder with two degrees of freedom (α, β), and another spherical joint at the elbow with two degrees of freedom (ϕ, γ).

The hand movement

$$\boldsymbol{p}_h + \Delta \boldsymbol{p}_h = \boldsymbol{f}(\boldsymbol{q} + \Delta \boldsymbol{q}) = \boldsymbol{f}(\boldsymbol{q}) + \boldsymbol{J}(\boldsymbol{q})\Delta \boldsymbol{q} + \text{h.o.t.} \quad (6)$$

where

$$\boldsymbol{J}(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}}(\boldsymbol{q}) \in \mathbb{R}^{3 \times 4}. \quad (7)$$

This then leads to

$$\Delta \boldsymbol{p}_h = \boldsymbol{J}(\boldsymbol{q})\Delta \boldsymbol{q}. \quad (8)$$

For a given $\Delta \boldsymbol{p}_h$, (8) has infinite solutions of $\Delta \boldsymbol{q}$. We aim to find $\Delta \boldsymbol{q}$ from (8) with good accuracy as compared to the human movement while enabling fast computation to be implemented in real time. Thus, we define a quadratic optimization with an analytical solution

$$\min_{\Delta \boldsymbol{q}} \boldsymbol{L} = \Delta \boldsymbol{q}^T \boldsymbol{Q} \Delta \boldsymbol{q}$$

$$\text{s.t. } \Delta \boldsymbol{p}_h = \boldsymbol{J}(\boldsymbol{q})\Delta \boldsymbol{q} \quad (9)$$

where $\boldsymbol{Q}$ is a diagonal weighting matrix with nonzero diagonal elements for solving the optimal $\Delta \boldsymbol{q}$. Physically, $\boldsymbol{Q}$ indicates how the hand velocity is distributed on each joint, a larger element on the diagonal indicates the corresponding joint is less likely to move during arm movements. We find $\boldsymbol{Q}$ using the arm movement data recorded during the stretch and the detail is presented in Section VI-B.

The solution to the optimization problem (9) is (detailed derivation in the Appendix)

$$\Delta \boldsymbol{q}^* = \boldsymbol{J}^+ \Delta \boldsymbol{p}_h - \left(\boldsymbol{\mu}^T \boldsymbol{Q} \Delta \boldsymbol{q}_h / \boldsymbol{\mu}^T \boldsymbol{Q} \boldsymbol{\mu} \right) \boldsymbol{\mu}$$

where $\boldsymbol{J}^+$ is Moore–Penrose inverse of $\boldsymbol{J}$ and $\boldsymbol{\mu}$ is the null space vector of $\boldsymbol{J}$.[3] Current postures in angle representation are then

$$\hat{\boldsymbol{q}}_T = \hat{\boldsymbol{q}}_{T-1} + \Delta \boldsymbol{q}^*. \quad (10)$$

Note that this is a recursive estimation as depicted in Fig. 5. Only the initial value of $\boldsymbol{q}$ is known such that $\hat{\boldsymbol{q}}_0 = \boldsymbol{q}_0$. The analytical solution can be computed fast online, which allows us to use the estimation scheme in real time during dressing.

Finally, the angle representation $\hat{\boldsymbol{q}}_T$ is transformed into current positions of the shoulder, elbow, and hand $\{\hat{\boldsymbol{p}}_s, \hat{\boldsymbol{p}}_e, \hat{\boldsymbol{p}}_h\}_T$ for deriving the dressing policy. We evaluate the accuracy of the estimation scheme in Section VI-B.

V. DRESSING ROBOT—DRESSING COORDINATE-BASED LEARNING

The dressing is highly dependent on the human arm posture. Previous research explored such dependence in learning the dressing policy. In [47], Clegg et al. placed in simulation haptic sensor on each segment of the arm as observation in deep reinforcement learning. Others define task parameters as local coordinates at the shoulder, elbow, and hand and then encode the dressing policy with TP-GMM [6], [7], [8]. Similarly, we explore such dependencies and define an arm-posture-dependent coordinate system for encoding the dressing policy with LfD. In the following sections, we first present the dressing coordinate then the LfD in the new coordinate system.

A. Dressing Coordinate

The dressing motion follows the arm from the hand to the elbow and finally reaches the shoulder. The monotonic motion naturally leads us to introduce a scalar quantity to model the dressing progress. We denote the progress scalar as s.

The dressing motion is additionally constrained by the size of the armscye. Thus, we use a polar coordinate around the forearm and the upper arm for encoding the constraint: l, the distance to the arm, and θ, the angle around the arm. The formulation yields two cylinder coordinates on the forearm and the upper arm. However, the complexity arises from the elbow where we require a smooth transition from the forearm coordinate to the upper-arm coordinate [see Fig. 7(a)].

To enable a smooth transition, we use a part of a horn torus (A torus where the distance from the center of the tube to the center of the torus equals the radius of the tube) for connecting the two cylinder coordinates. As a result of the torus defined around the elbow, the progress curve is divided into three pieces: 1) a forearm segment, 2) an elbow arc for a smooth transition to the upper arm, and 3) an upper-arm segment, as shown in Fig. 7(b).

To convert a point on the dressing path $\boldsymbol{x} \in \mathbb{R}^3$ in Cartesian coordinate into the dressing coordinate $[s, l, \theta]$, we first project $\boldsymbol{x}$ onto the arm plane given by the shoulder $\boldsymbol{p}_s$, elbow $\boldsymbol{p}_e$, and hand positions $\boldsymbol{p}_h$

$$\boldsymbol{x}_{\text{arm}} = \boldsymbol{x} + \frac{\boldsymbol{v}(\boldsymbol{p}_h \cdot \boldsymbol{v} - \boldsymbol{x} \cdot \boldsymbol{v})}{|\boldsymbol{v}|^2} \quad (11)$$

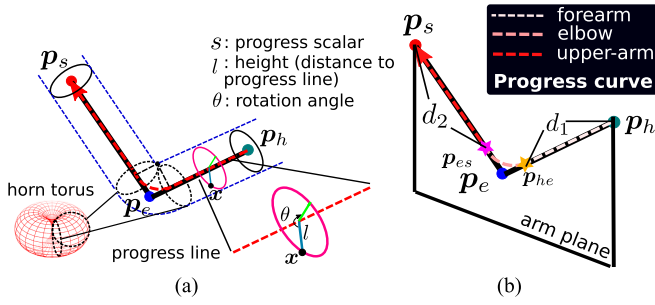[3]Detail solution can be found in the Appendix.

Fig. 7. Dressing coordinate is a cylinder coordinate defined around the arm, with a smoothing transitional corner around the elbow. (a) Full definition of the coordinate. (b) Progress curve in three segments.
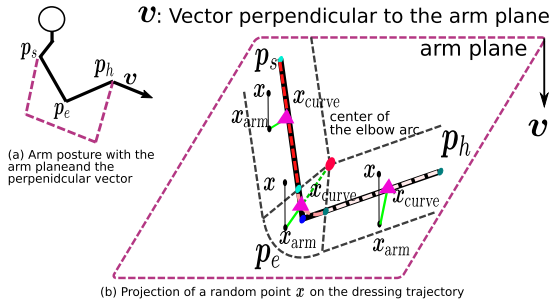

Fig. 8. Graphic illustration of finding the corresponding point on the progress curve x_{curve} given a random point x on the dressing path in Cartesian coordinate. The point x is first projected onto the arm plane to get x_{arm}. Then we determine which segment x_{arm} belongs to on the progress curve. Finally, we find the corresponding x_{curve}.

where v is the vector perpendicular to the arm plane.

We further find the progress scalar of x_{arm} on the progress curve. The progress curve is divided into three segments by two transitional points: p_{he} (transition from the forearm to elbow) and p_{es} (transition from elbow to upper arm) as shown by stars in Fig. 7(b). The distance from p_h to p_{he} is denoted $|d_1|$, and from p_s to p_{es} is $|d_2|$.

We need to determine which segments x_{arm} belongs to. To do that, we compute the distance x_{arm} project onto the line defined by the forearm and upper-arm segments, respectively

$$d_{\text{forearm}} = \frac{(x_{\text{arm}} - p_h) \cdot (p_e - p_h)}{|(p_e - p_h)|}$$

$$d_{\text{upperarm}} = \frac{(x_{\text{arm}} - p_s) \cdot (p_e - p_s)}{|(p_e - p_s)|}. \tag{12}$$

The value of d_{forearm} and d_{upperarm} is then compared against d_1 and d_2 so that

$$x_{\text{arm}} \text{ on the} \begin{cases} \text{forearm} & \text{if } d_{\text{forearm}} < |d_1| \text{ and } d_{\text{upperarm}} > |d_2| \\ \text{upper arm} & \text{if } d_{\text{upperarm}} < |d_2| \text{ and } d_{\text{forearm}} > |d_1| \\ \text{elbow,} & \text{otherwise.} \end{cases}$$

Depending on which segment x_{arm} belongs to, we find the corresponding x_{curve} (pink triangles in Fig. 8) on the progress curve for x_{arm}. For the forearm and upper-arm segment, calculating x_{curve} is straightforward. For the elbow segment, we find the line connecting the center of the elbow arc to x_{arm}. The

intersection point between the line and the progress curve is x_{curve} (see Fig. 8).

We then define $s = 0$ for $x_{\text{curve}} = p_h$, $s = 1$ for $x_{\text{curve}} = p_s$, for any x_{curve}, we have

$$s = \begin{cases} \frac{d_{\text{forearm}}}{|d_{\text{curve}}|} & \text{if } x_{\text{curve}} \text{ is on the forearm} \\ \frac{|d_{\text{curve}}| - d_{\text{upperarm}}}{|d_{\text{curve}}|} & \text{if } x_{\text{curve}} \text{ is on the upper arm} \\ \frac{d_1 + \omega r}{|d_{\text{curve}}|} & \text{if } x_{\text{curve}} \text{ is on the upper arm} \end{cases}$$

where $|d_{\text{curve}}|$ is the full length of the progress curve from p_h to p_s, ω is the angle of x_{curve} on the elbow arc and r is the radius of the elbow segment on the progress curve.

Once we found the projection of the dressing position on the progress curve x_{curve} and the progress scalar s, we can calculate the distance to the arm l and the angle around the arm θ as follows.

The length l of x is

$$l = |x - x_{\text{curve}}|. \tag{13}$$

We define v as the reference $0°$ for the angle θ computation. θ is defined as

$$\theta = \angle(x_{\text{curve}} - x, v). \tag{14}$$

Note that v has two directions, one pointing toward the human body and the other pointing outward. We select the former to avoid flipping angles from 0 to 2π during the dressing motion.

By converting the Cartesian coordinate to the dressing coordinate and obtaining the motion generation model in the dressing coordinate, we can conclude the following:
1) limit the policy space to be around the arm by imposing the maximum value for the length l;
2) ensure the motion convergence to the vicinity of the shoulder by progress scalar s;
3) transfer the trained motion generation model on one arm to another as the data are relative to the arm posture.

Note one prerequisite for the last statement is that the ratio between the forearm and upper arm is similar among people.

B. LfD in Dressing Coordinate

For LfD in dressing coordinate, we employ Gaussian mixture models (GMM) and Gaussian mixture regression (GMR) for learning and generating the dressing policy [48]. The GMM/GMR-based imitation learning offers fast movement retrieval and versatile inputs/outputs arrangement, which is desirable for our task. Note that the LfD algorithm is not a part of our contribution and it is possible to use different methods/formulations for learning the policy in the dressing coordinate.

We first briefly describe GMM/GMR-based LfD then show how to apply it to learning the dressing policy.

GMM models the joint distribution of the demonstration data. We denote the demonstrated data ξ

$$\xi = \left[\xi^{\mathcal{I}}, \xi^{\mathcal{O}}\right]^T \tag{15}$$

with $\boldsymbol{\xi}^{\mathcal{I}}$ is the input and $\boldsymbol{\xi}^{\mathcal{O}}$ the output. The probability density function $p(\boldsymbol{\xi}_i)$ is estimated with K Gaussian distributions

$$p(\boldsymbol{\xi}_i) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\xi}_i | \boldsymbol{\mu}_k, \Sigma_k)$$

where $\boldsymbol{\mu}_k$ and Σ_k are mean and variance of the kth Gaussian, respectively. To determine an optimal number of Gaussian K for GMM, we make use of the Bayesian information criterion [49] for balancing the model complexity and representation quality.

Once K is selected, we can initialize the GMM with K-means clustering and then employ the expectation–maximization algorithm to iterative compute the model parameters $\{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$.

The resulting K Gaussian parameters can be decomposed as

$$\boldsymbol{\mu}_k = \begin{bmatrix} \boldsymbol{\mu}_k^{\mathcal{I}} \\ \boldsymbol{\mu}_k^{\mathcal{O}} \end{bmatrix}, \quad \boldsymbol{\Sigma}_k = \begin{bmatrix} \boldsymbol{\Sigma}_k^{\mathcal{I}} & \hat{\boldsymbol{\Sigma}}_k^{\mathcal{IO}} \\ \boldsymbol{\Sigma}_k^{\mathcal{OI}} & \hat{\boldsymbol{\Sigma}}_k^{\mathcal{O}} \end{bmatrix}. \quad (16)$$

Given input data, the output is retrieved using GMR by conditioning the input data [50].

For encoding dressing with GMM/GMR in the dressing coordinate, we demonstrate the dressing task with static arm postures, the arm posture, and the dressing path is recorded for converting the dressing path into the dressing coordinate. We use the progress scalar s and the elbow angle ϕ as inputs. The outputs are the differences in the distance to the arm δl, and the angle difference around the arm $\delta\theta$. The differences are computed with regard to the starting position in the demonstration. For all demonstrations, we start from above the hand of the human.

VI. EXPERIMENTS

We conduct three different experiments to investigate various aspects of the proposed framework.

In Section VI-A, we investigate the effect of the elbow angle on the dressing strategy using expert demonstration data of static arm postures with different elbow angles. Then, in Section VI-B, we evaluate the arm posture estimation scheme to show that it is accurate enough for the dressing task.

Finally, in Section VI-C, we showcase our interactive dressing framework with real robot experiments. To demonstrate the flexibility of the defined dressing coordinate, we train the policy from demonstrations on a mannequin and apply the policy to human arms with different lengths. We compared our trajectory encoding method (using dressing coordinates) with TP-GMM-based methods [6], [7], [8] in terms of dressing success rate.

Furthermore, we conducted an ablation study where we disable the stretch movement and dress with a static human arm. Additionally, we test the framework under cases where the human is not fully compliant with the interactive robot and show that the overall framework still works well.

In these experiments, the robot dresses the human in short- and long-sleeved shirts with different materials. Note that the automatic dressing of long-sleeved shirts is unprecedented in previous research.
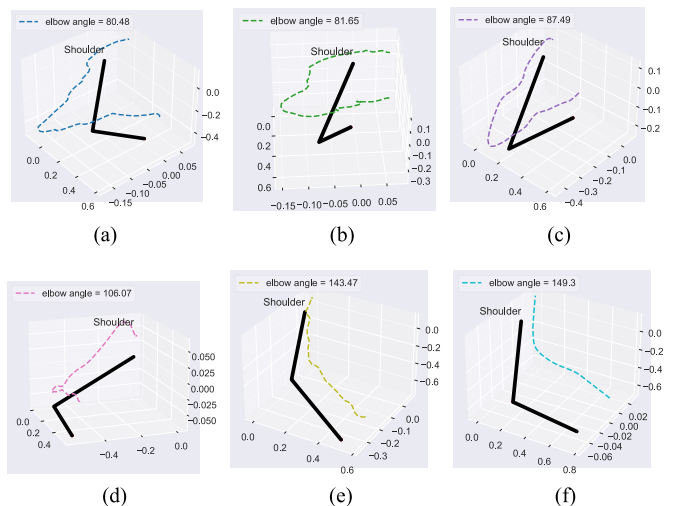

Fig. 9. Static arm postures with different elbow angles and the corresponding expert demonstrations of dressing with these postures.
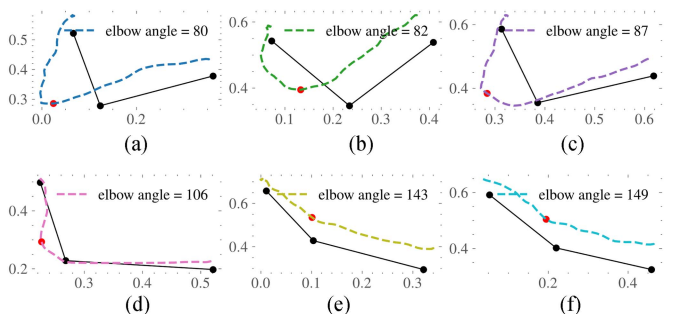

Fig. 10. Projection of the arm postures and the corresponding demonstrated dressing path on the arm plane. The red dot is a transition point corresponding to the middle of the elbow arc on the progress curve.

A. Effect of the Elbow Angle on Dressing Strategies

In Section IV-A, we discussed the effect of the elbow angle on the dressing strategy with diminished rigidity. In this section, we aim to show this effect with expert demonstration data.

We collect demonstration data through kinesthetic teaching with a Franka robot under different arm postures. The dressing paths and arm postures are recorded in the process. In Fig. 9, we present six postures with different elbow angles. We draw each posture and dressing path individually. In each figure, the solid black line is the arm posture where the shoulder positions are at the top and labeled with "Shoulder." The dashed line in each figure is the dressing path that corresponds to the posture.

To validate whether the elbow angle affects the dressing strategies, we project the dressing paths onto the arm plane together with the arm postures, as shown in Fig. 10. In the figure, each subfigure corresponds to a 2-D projection of the posture and dressing path in Fig. 9. The red dot on each subfigure is a transition point corresponding to the middle of the elbow arc in Fig. 7(b) on the progress curve.

In addition, we provide a quantitative result summarized in Table I. We calculate the distance between the transition point and the elbow. When the transition point is on the inner arm plane, the distance is positive, which indicates that the dressing

Elbow angle (°)	80.47	81.65	87.49	106.07	143.46	149.30
Distance (m)	-0.099	-0.11	-0.10	-0.07	0.107	0.105

employs an inner strategy. When the transition point is on the outer arm plane, we set the distance negative, which indicates that the dressing employs an outer strategy (see Section IV-B and Fig. 3 for the definition of inner/outer strategy).

We can observe both from Fig. 10 and Table I that the outer dressing strategy is employed when the elbow angle is small, which justifies our analysis in Section IV-A.

B. Evaluation of the Posture Estimation Scheme

As the dressing path is highly dependent on the arm posture, posture estimation is crucial to the success of dressing. In this section, we provide a quantitative evaluation of the posture estimation scheme. We consider the following.

1) The shoulder is static during dressing.
2) The initial arm posture (in the dressing robot frame $\mathcal{O}^D$) is known.

The first assumption is a common assumption in assistive dressing literature [6], [10]. The initial arm posture in our case is estimated by setting the dressing arm $\mathcal{D}$ in the gravity compensation mode, then putting the end-effector close to the hand, elbow, and shoulder, and using the reading from the end-effector positions as the initial hand, elbow, and shoulder positions. In assistive dressing, the starting posture is easy to obtain. It is often estimated via a (depth) camera in the previous literature [6], [10], [51], [52].

With the previous assumptions and real-time hand position recovery from the interactive robot (see Section IV-C), the only estimated quantity in the full posture description $\boldsymbol{P} = \{\boldsymbol{p}_s, \boldsymbol{p}_e, \boldsymbol{p}_h\}$ is the elbow position $\boldsymbol{p}_e$.

Thus, in the evaluation, for a series of arm postures during movements $\{\boldsymbol{P}_0, \boldsymbol{P}_1, \ldots, \boldsymbol{P}_T\}$, we use the maximum error of the elbow as the performance indicator of our posture estimation scheme

$$\text{error}_{\max} = \max_{i=1}^{T} |\boldsymbol{p}_e(i) - \hat{\boldsymbol{p}}_e(i)|$$

where $\hat{\boldsymbol{p}}_e(i)$ is the estimated elbow position at the ith instance.

One tunable parameter that affects the estimation performance is the diagonal weight matrix $\boldsymbol{Q}$ in (9). We compute $\boldsymbol{Q}$ using the movement data $\{\boldsymbol{P}_0, \boldsymbol{P}_1, \ldots, \boldsymbol{P}_T\}$. The data are first transformed to angle representation $\{\boldsymbol{q}_0, \boldsymbol{q}_1, \ldots, \boldsymbol{q}_T\}$ (refer to Section IV-C and Fig. 5 for the definition of angle representation and 4 angles $\boldsymbol{q} = [\alpha\,\beta\,\phi\,\gamma]^T$ associated). We then compute the difference between adjacent angles and yield

$$\{\delta\boldsymbol{q}_1, \delta\boldsymbol{q}_2, \ldots, \delta\boldsymbol{q}_T\} \tag{17}$$

where

$$\delta\boldsymbol{q}_i = [\delta\alpha_i\,\delta\beta_i\,\delta\phi_i\,\delta\gamma_i]^T = \boldsymbol{q}_i - \boldsymbol{q}_{i-1}. \tag{18}$$

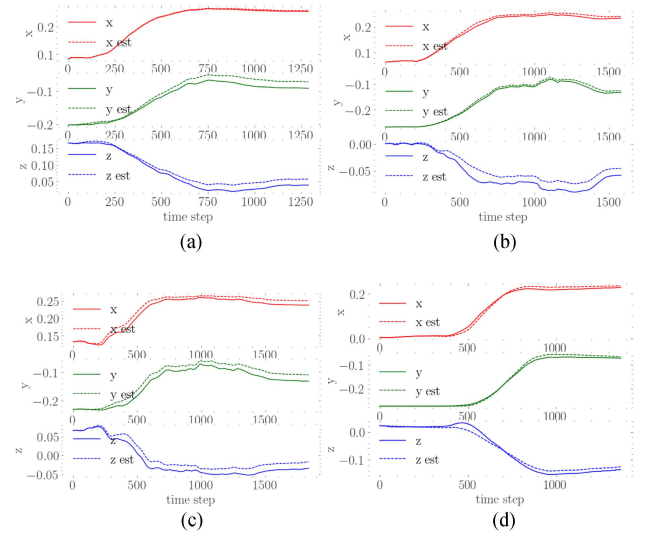|	Our method				RNN [11]	
	Case 1	Case 2	Case 3	Case 4	Position only	With features
$\text{error}_{\max}$ (mm)	28	26	32	28	44	25
$\text{error}_{\text{average}}$ (mm)	17	14	23	15	41	24

Fig. 11. Ground truth and the estimated elbow position (x, y, z) in solid and dash line, respectively, in test cases. (a) case 1. (b) case 2. (c) case 3. (d) case 4.

The weighting diagonal matrix $\boldsymbol{Q}$ in (9) is calculated as

$$\boldsymbol{Q} = \text{diag}\left\{\frac{1}{\sum_{i=1}^T \delta\alpha_i^2}, \frac{1}{\sum_{i=1}^T \delta\beta^2}, \frac{1}{\sum_{i=1}^T \delta\phi_i^2}, \frac{1}{\sum_{i=1}^T \delta\gamma_i^2}\right\}.$$

We conducted experiments to collect arm movement data with an XSens motion capture suit. During the data collection phase, one person is asked to stretch the arm of the other. The posture data were collected using the XSense suit and regarded as ground truth. Using one stretch data, we calculate the weight matrix as (this is the $\boldsymbol{Q}$ value used for the interactive dressing experiments in Section VI-C)

$$\boldsymbol{Q} = \text{diag}\{207.89, 654.28, 99.89, 184.65\}. \tag{19}$$

For the evaluation, we collected four new stretch data from different starting poses using the XSense suit. The maximum and average errors with regard to the ground truth in all four cases are presented in Table II. In the same table, we also include estimation errors from the work in [11] for comparison.

In [11], Chance et al. employed an RNN that trained on posture data during dressing for elbow position estimation under occlusion. In Table II, we present estimation errors in [11], which trains on position data only and additionally with engineered features. From the table, we observe that our method performs better than RNN trained with position data only and is comparable to RNN trained on position and feature data.

Fig. 11 shows the ground truth and the estimated elbow position (x, y, z) in solid and dash line, respectively, in all four cases.
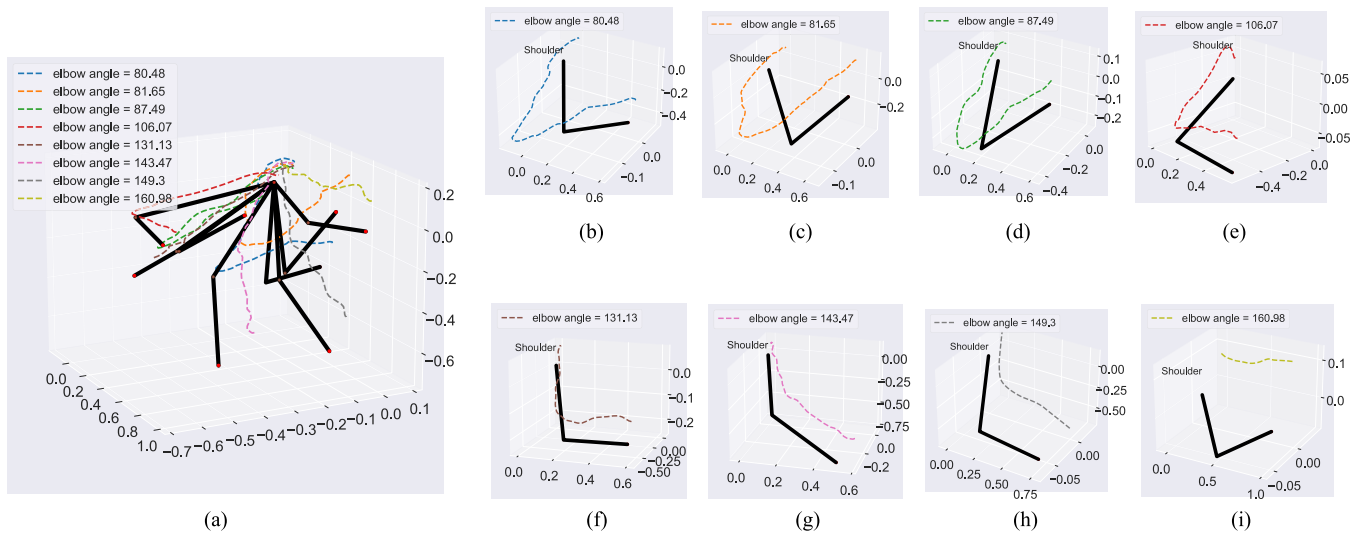
Fig. 12. Expert demonstrations on different postures used for (a) training dressing policy and (b)–(i) individual trajectory. (a) Expert demonstrations on different postures. (b) Elbow angle = 80:48. (c) Elbow angle = 81:65. (d) Elbow angle = 87:49. (e) Elbow angle = 106:07. (f) Elbow angle = 131:13. (g) Elbow angle = 143:47. (h) Elbow angle = 149:3. (i) Elbow angle = 160:98.

TABLE III
ARM LENGTHS OF THE HUMAN AND THE MANNEQUIN

	Hand−elbow (cm)	Elbow−shoulder (cm)	ratio
Mannequin	26.4	25.3	1.04
Human 1	30.5	29.6	1.03
Human 2	27.5	26.3	1.04

We observe that, generally, the deviation from ground truth is larger as the time step increases, this is probably due to the recursive nature of the estimation scheme. The error may accumulate with time. Despite this, the maximum error is around 3 cm, which is much smaller than the size of the armscye. The results are comparable with the results from the work in [11].

The arm posture estimation is an adapted solution to allow real-time tracking of human postures during dressing with our proposed bimanual scheme. We are not expecting it to outperform the state of the art (SOTA) but a compatible performance indicates that the framework individually is able to work properly. Later in Section VI-C and also the video (see Supplementary material available online), we will show that the integrated framework can dress different humans and clothes, which in return validates our choice of such an estimation scheme.

C. Interactive Dressing

In this section, we show the experimental results of our interactive dressing scheme. We present experiments with three types of clothes: 1) a short and rigid sleeve shirt, 2) a long and rigid sleeve shirt, and 3) a long and soft sleeve shirt.

To show the flexibility of our framework, our demonstrations were collected on the mannequin and the learned dressing policy was executed on the human arm. In demonstrations, we always start around above the hand and with the same orientation.[4] The arm length of the humans and the mannequin is listed in Table III, although having different lengths, the ratio between

[4]The orientation can be seen in the start configuration in dressing in Fig. 15.
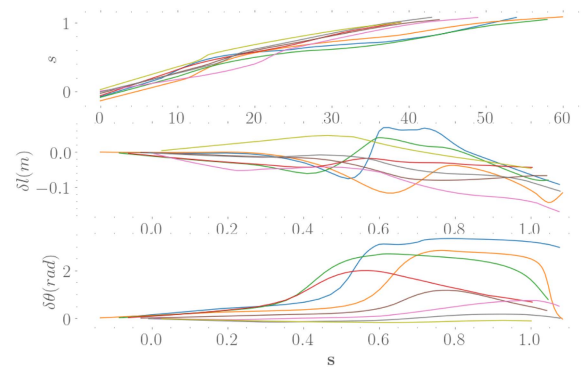
Fig. 13. Expert demonstrations for training the dressing policy in Fig. 12(a) transformed to dressing coordinate.
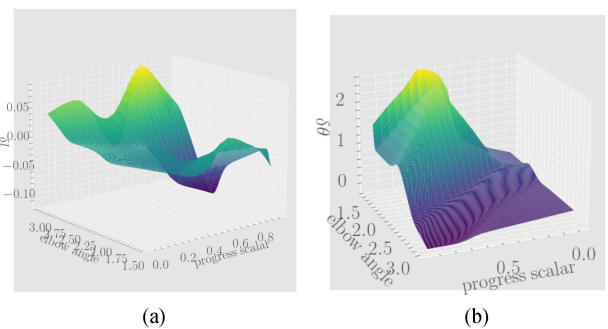
Fig. 14. Surface plot for GMM with elbow angles ψ and progress scalar s as inputs and outputs changes in the distance to the arm δl and changes in angles around the arm $\delta\theta$, respectively. (a) The 3D surface plot of GMM for encoding δl. (b) The 3D surface plot of GMM for encoding $\delta\theta$.

hand–elbow and elbow–shoulder is similar. Encoding the policy in the dressing coordinate allows it to be adaptive to different arm lengths.

Fig. 15. Starting and/or end configurations for three different experiments: 1) human passive/compliant, 2) human not fully compliant, and 3) the ablation study where the human remains static. For experiments where the human is not fully compliant, we also display with high transparency a nominal end configuration when the human is compliant. The deviation from the nominal end configuration is indicated by a blue arrow.

Fig. 12 shows the demonstration and transformed data in the dressing coordinate. For the angle around the arm θ and the distance to the arm l, we present the changes during dressing $\delta\theta$ and δl instead of the absolute values. The difference is computed with regard to the starting position, which is around above the hand. For postures with smaller elbow angles, we can observe a change in the angles around the arm θ since it needs to move to the back of the arm to execute an outer strategy. From Fig. 13, we observe that the progress scalar s is a monotonic increasing quantity. Consider a target s_{target}, we model s with a dynamical system with a constant increment $c > 0$

$$s_{T+1} = s_T + \min(c, (s_{\text{target}} - s_T)). \tag{20}$$

We, respectively, trained two GMM for the changes in angles around the arm $\delta\theta$ and changes in the distance to the arm δl with inputs progress scalar s and the elbow angle ϕ.[5] We use eight Gaussians for both δl and $\delta\theta$. The resulting mixture models are presented in 3-D surf plots in Fig. 14. We can observe a smaller elbow angle displays larger changes in θ, which is a clear indication of taking the outer strategy. For a large elbow angle, the θ change is almost zero, which resembles an inner strategy. Due to the diminishing rigidity, the end effector rotation has a very limited effect during dressing. In the experiment, we consider only the z-axis orientation of the end-effector of the dressing robot. The z-axis rotation is applied to avoid collision with the arm during dressing. We can use a constant incremental rotation with regard to s or a separate GMM with the same inputs ($[s, \phi]$) for encoding the difference dz in the z-axis rotation (the training procedure is the same as $\delta\theta$ and δl).

We implement the policy with our optimal stretch and posture estimation scheme. Three types of experiments are conducted, which are as follows.

1) Human passively follows the interactive robot.
2) Human is not fully compliant with the guiding force and moves in other directions.
3) Ablation study: Human arm remains static.

We tested in different starting postures and different clothes, as shown in Fig. 15. In the figure, we present only starting and end configurations, the full dressing sequences are recorded in the video (see Supplementary material available online). We tested the dressing while the human passively followed the guiding force and also moved in other directions (not fully compliant). When the human is not fully compliant, in the figure, we also display with high transparency a nominal end configuration when the human is compliant. The deviation from the nominal end configuration is indicated by a blue arrow. With these results, we prove that humans can move in other directions if desired and the framework is robust against noncompliant behavior and still succeed in dressing.

In the ablation study, we demonstrate cases where the elbow angle is small, and the cloth tends to get stuck around the elbow, thus reiterating the importance of having a bimanual setup for dressing. However, the result also suggests that if the human starts in these postures and refuses to move along with the guiding force, the dressing could fail.

Finally, we compare our dressing coordinate encoding with TPGMM that was commonly used to encode dressing behavior from demonstration [6], [7], [8] (while the rest of the framework stays the same), and present, respectively, the success rates in dressing while the human is compliant for both methods in

---

[5]Note that other formulations may be also possible, such as to learn the joint distribution of $[s, \phi, l, \theta]$.

TABLE IV
COMPARISON BETWEEN DRESSING COORDINATE-BASED APPROACH AND
TPGMM

Methods	Number of trials	Success trials	Success rate	Success rate excl. slipping
Dressing Coord	12	9	75%	90%
TPGMM	6	3	50%	50%

Table IV. There are three main reasons for failure, which are as follows.

1) The clothes slip from the gripper.
2) The trajectory collides with humans.
3) Robot stops before reaching the shoulder.

As clothes grasping is not the focus of this article, in Table IV, we show the overall success rate in the fourth column and the success rate excluding gripper slip in the fifth column. Our dressing coordinate approach achieves a higher success rate.

The reasons why TPGMM is more prone to fail compared with our dressing coordinate approach are as follows.

1) Due to covariance shifts, traditional GMM-based learning cannot guarantee convergence. Although there exist methods that combine dynamical systems with GMM to guarantee convergence [53], such methods do not work for TPGMM. While using our method to learn in dressing coordinates, there is one progress scalar that directly encodes the progress of the dressing. The scalar is defined as monotonically increasing, thus the trajectory will converge around the shoulder.
2) TPGMM requires the elbow position to determine the reference frames for calculating the final motion generation model; therefore, the error in the elbow estimation will be directly reflected in the motion generation model.

VII. CONCLUSION

Traditional assistive dressing considers a *one-robot-to-one-arm* setup, which holds the following:

1) either assumes static arm posture or requires additional sensors and algorithms for tracking the arm posture;
2) renders the human arm hanging in the air during the dressing process, which is often tiring;
3) may fail/get stuck at the elbow for some initial arm postures.

In the light of the following limitations, and inspired by strategies taken by caregiving experts, we designed a bimanual assistive dressing framework with a *two-robot-to-one-arm* setup that holds the following:

1) allows tracking of arm postures without additional sensors;
2) provides support and guiding force for the arm being dressed;
3) stretches the human arm while compliant to obtain postures for the ease of dressing.

We approached the design by first analyzing the effect of elbow angle in dressing. Based on this effect, we proposed an optimal stretch controller for the interactive robot. As the interactive robot holds the human hand, the arm posture can be estimated in real time with inverse kinematics. For the dressing robot, we utilized the dependence of the dressing policy on the arm posture and introduced a dressing coordinate defined by the arm posture for easy encoding of the dressing policy from demonstrations.

The adoption of the dressing coordinate greatly enhances the flexibility of the LfD. The trained policy can adapt to different arm lengths as long as the forearm-to-upper-arm ratio is similar. The framework allows dressing different types of clothes, even long sleeves, which is unprecedented in previous research. The dressing remains robust even though human is not fully compliant with the guidance from the interactive robot.

Since this is the first time such a bimanual setup was employed in dressing a human, there are two major limitations of the proposed framework.

1) The assumption that the shoulder remains static is not always true, since the interactive robot is stretching the arm, the shoulder is more likely to move forward. A forward-moving shoulder will subsequently increase the length of the arm (as now the hand can reach further when fully stretched). The increased length may render the arm posture estimation scheme unsolvable; thus, the overall scheme will fail.
2) The elbow angle is identified as the crucial parameter that affects dressing; however, there is no explicit coordination between the dressing and interactive robot to make sure that when the dressing robot is going through the elbow, the interactive robot already makes the elbow angle sufficiently large to avoid getting stuck.

It is difficult to solve the first limitation purely from the algorithm side, one possibility is to combine inverse kinematics with vision-based tracking for more accurate posture estimation. The second limitation is even trickier, to make the coordination explicit, a thorough analysis of dressing on different elbow angles needs to be conducted to determine a boundary condition of the elbow angle, then additionally, an assumption has to be placed on the human side (passive for instance) to make sure that the interactive robot will always succeed in bringing the human arm into favorable postures before dressing arm reach the elbow.

Another possible direction for future work is safety analysis. We did not explicitly analyze the safety of the overall system. However, for assistive robots, it is crucial to ensure human safety during robotic assistance. Inspirations for conducting system-level safety analysis can be drawn from the work in [54].

Despite the aforementioned limitations, our proposal is the first that considers interactive dressing assistance with a bimanual setup. The setup is inspired by caregiving experts conducting the task, which makes the scheme receptive to humans. It represents a paradigm shift in the thinking of the dressing task from a *one-robot-to-one-arm* setup to a *two-robot-to-one-arm* setup.

APPENDIX

In the Appendix, we provide the analytical solution to (9) in Section IV-C

$$\min_{\Delta q} L = \Delta q^T Q \Delta q$$

$$\text{s.t. } \Delta p_h = J \Delta q.$$

Since $\boldsymbol{J} \in \mathbb{R}^{3\times4}$, $\Delta\boldsymbol{p}_h = \boldsymbol{J}\Delta\boldsymbol{q}$ has a infinite number of solutions

$$\Delta\boldsymbol{q} = \boldsymbol{J}^+\Delta\boldsymbol{p}_h + \lambda\boldsymbol{\mu} = \Delta\boldsymbol{q}_h + \lambda\boldsymbol{\mu} \tag{21}$$

where $\boldsymbol{J}^+$ is the Moore–Penrose inverse of $\boldsymbol{J}$ and $\boldsymbol{\mu}$ is the vector in the null space of $\boldsymbol{J}$ (e.g., $\boldsymbol{J}\boldsymbol{\mu} = \boldsymbol{0}$), and $\lambda \in \mathbb{R}$ is a gain. Taking (22) into $\boldsymbol{L}$

$$\boldsymbol{L} = (\Delta\boldsymbol{q}_h + \lambda\boldsymbol{\mu})^T\boldsymbol{Q}(\Delta\boldsymbol{q}_h + \lambda\boldsymbol{\mu})$$
$$= (\boldsymbol{\mu}^T\boldsymbol{Q}\boldsymbol{\mu})\lambda^2 + 2\boldsymbol{\mu}^T\boldsymbol{Q}\Delta\boldsymbol{q}_h\lambda + \Delta\boldsymbol{q}_h^T\boldsymbol{Q}\Delta\boldsymbol{q}_h \tag{22}$$

which is quadratic in λ.

The original optimization problem can be transformed into

$$\min_\lambda \boldsymbol{L} = (\boldsymbol{\mu}^T\boldsymbol{Q}\boldsymbol{\mu})\lambda^2 + 2\boldsymbol{\mu}^T\boldsymbol{Q}\Delta\boldsymbol{q}_h\lambda + \Delta\boldsymbol{q}_h^T\boldsymbol{Q}\Delta\boldsymbol{q}_h. \tag{23}$$

Let us denote $\boldsymbol{\mu}^T\boldsymbol{Q}\boldsymbol{\mu} = a$, $\boldsymbol{\mu}^T\boldsymbol{Q}\Delta\boldsymbol{q}_h = b$, and $\Delta\boldsymbol{q}_h^T\boldsymbol{Q}\Delta\boldsymbol{q}_h = c$

$$\boldsymbol{L} = a\lambda^2 + 2b\lambda + c$$
$$= a\left(\lambda^2 + 2\frac{b}{a}\lambda + c\right)$$
$$= a\left(\left(\lambda + \frac{b}{a}\right)^2 + c - \frac{b^2}{a^2}\right). \tag{24}$$

The solution that minimize $\boldsymbol{L}$ is

$$\lambda = -\frac{b}{a} = -\boldsymbol{\mu}^T\boldsymbol{Q}\Delta\boldsymbol{q}_h / \boldsymbol{\mu}^T\boldsymbol{Q}\boldsymbol{\mu}. \tag{25}$$

Taking (25) into (22), we can obtain the optimal solution

$$\Delta\boldsymbol{q}^* = \boldsymbol{J}^+\Delta\boldsymbol{p}_h - \left(\boldsymbol{\mu}^T\boldsymbol{Q}\Delta\boldsymbol{q}_h / \boldsymbol{\mu}^T\boldsymbol{Q}\boldsymbol{\mu}\right)\boldsymbol{\mu}. \tag{26}$$

The first term ensures that the constraint is respected; on the other hand, the second term generates a null-space transition that minimizes the total weighted joint displacement.

REFERENCES

[1] "Ageing and health," Oct. 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ageing-and-health

[2] B. J. Dudgeon, J. M. Hoffman, M. A. Ciol, A. Shumway-Cook, K. M. Yorkston, and L. Chan, "Managing activity difficulties at home: A survey of medicare beneficiaries," *Arch. Phys. Med. Rehabil.*, vol. 89, no. 7, pp. 1256–1261, 2008.

[3] D. J. Agravante, A. Cherubini, A. Sherikov, P.-B. Wieber, and A. Kheddar, "Human-humanoid collaborative carrying," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 833–846, Aug. 2019.

[4] S. Tarbouriech, B. Navarro, P. Fraisse, A. Crosnier, A. Cherubini, and D. Sallé, "Admittance control for collaborative dual-arm manipulation," in *Proc. IEEE Int. Conf. Adv. Robot.*, 2019, pp. 198–204.

[5] L. van der Spaa, M. Gienger, T. Bates, and J. Kober, "Predicting and optimizing ergonomics in physical human-robot cooperation tasks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 1799–1805.

[6] E. Pignat and S. Calinon, "Learning adaptive dressing assistance from human demonstration," *Robot. Auton. Syst.*, vol. 93, pp. 61–75, 2017.

[7] J. Hoyos, F. Prieto, G. Alenyá, and C. Torras, "Incremental learning of skills in a task-parameterized Gaussian mixture model," *J. Intell. Robot. Syst.*, vol. 82, no. 1, pp. 81–99, 2016.

[8] J. Zhu, M. Gienger, and J. Kober, "Learning task-parameterized skills from few demonstrations," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4063–4070, Apr. 2022.

[9] Y. Gao, H. J. Chang, and Y. Demiris, "User modelling for personalised dressing assistance by humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 1840–1845.

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

[11] G. Chance, A. Jevtić, P. Caleb-Solly, G. Alenya, C. Torras, and S. Dogramadzi, ""Elbows out"—Predictive tracking of partially occluded pose for robot-assisted dressing," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3598–3605, Oct. 2018.

[12] Z. Erickson, M. Collier, A. Kapusta, and C. C. Kemp, "Tracking human pose during robot-assisted dressing using single-axis capacitive proximity sensing," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2245–2252, Jul. 2018.

[13] Y. Wang, D. Held, and Z. Erickson, "Visual haptic reasoning: Estimating contact forces by observing deformable object interactions," in *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 11426–11433, Oct. 2022.

[14] T. Tamei, T. Matsubara, A. Rai, and T. Shibata, "Reinforcement learning of clothing assistance with a dual-arm robot," in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, 2011, pp. 733–738.

[15] R. P. Joshi, N. Koganti, and T. Shibata, "A framework for robotic clothing assistance by imitation learning," *Adv. Robot.*, vol. 33, no. 22, pp. 1156–1174, 2019.

[16] K. Yamazaki, R. Oya, K. Nagahama, K. Okada, and M. Inaba, "Bottom dressing by a dual-arm robot using a clothing state estimation based on dynamic shape changes," *Int. J. Adv. Robot. Syst.*, vol. 13, no. 1, 2016. [Online]. Available: https://journals.sagepub.com/doi/full/10.5772/61930

[17] G. Canal, E. Pignat, G. Alenyá, S. Calinon, and C. Torras, "Joining high-level symbolic planning with low-level motion primitives in adaptive HRI: Application to dressing assistance," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 3273–3278.

[18] A. Jevtić et al., "Personalized robot assistant for support in dressing," *IEEE Trans. Cogn. Develop. Syst.*, vol. 11, no. 3, pp. 363–374, Sep. 2019.

[19] A. Kapusta et al., "Personalized collaborative plans for robot-assisted dressing via optimization and simulation," *Auton. Robots*, vol. 43, no. 8, pp. 2183–2207, 2019.

[20] A. Clegg, Z. Erickson, P. Grady, G. Turk, C. C. Kemp, and C. K. Liu, "Learning to collaborate from simulation for robot-assisted dressing," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2746–2753, Apr. 2020.

[21] W. Yu, A. Kapusta, J. Tan, C. C. Kemp, G. Turk, and C. K. Liu, "Haptic simulation for robot-assisted dressing," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 6044–6051.

[22] A. Kapusta, W. Yu, T. Bhattacharjee, C. K. Liu, G. Turk, and C. C. Kemp, "Data-driven haptic perception for robot-assisted dressing," in *Proc. 25th IEEE Int. Symp. Robot Hum. Interactive Commun.*, 2016, pp. 451–458.

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

[24] L. Twardon and H. Ritter, "Active boundary component models for robotic dressing assistance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 2811–2818.

[25] S. Li, N. Figueroa, A. J. Shah, and J. A. Shah, "Provably safe and efficient motion planning with uncertain human dynamics," in *Proc. Robot.: Sci. Syst.*, 2021.

[26] S. Li, T. Stouraitis, M. Gienger, S. Vijayakumar, and J. A. Shah, "Set-based state estimation with probabilistic consistency guarantee under epistemic uncertainty," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 5958–5965, Jul. 2022.

[27] F. Zhang and Y. Demiris, "Learning grasping points for garment manipulation in robot-assisted dressing," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 9114–9120.

[28] M. Dragusanu et al., "The DressGripper: A collaborative gripper with electromagnetic fingertips for dressing assistance," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 7479–7486, Jul. 2022.

[29] F. Zhang and Y. Demiris, "Learning garment manipulation policies toward robot-assisted dressing," *Sci. Robot.*, vol. 7, no. 65, 2022, Art. no. eabm6010.

[30] F. Krebs and T. Asfour, "A bimanual manipulation taxonomy," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 11031–11038, Oct. 2022.

[31] L. P. Ureche and A. Billard, "Constraints extraction from asymmetrical bimanual tasks and their use in coordinated behavior," *Robot. Auton. Syst.*, vol. 103, pp. 222–235, 2018.

[32] A. Colomé and C. Torras, "Dimensionality reduction for dynamic movement primitives and application to bimanual manipulation of clothes," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 602–615, Jun. 2018.

[33] F. Suárez-Ruiz, X. Zhou, and Q.-C. Pham, "Can robots assemble an IKEA chair?," *Sci. Robot.*, vol. 3, no. 17, 2018, Art. no. eaat6385.

[34] J. Liu et al., "Robot cooking with stir-fry: Bimanual non-prehensile manipulation of semi-fluid objects," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5159–5166, Apr. 2022.

[35] J. Xu et al., "Enhanced bimanual manipulation assistance with the personal mobility and manipulation appliance (PerMMA)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 5042–5047.

[36] M. Connan et al., "Learning to teleoperate an upper-limb assistive humanoid robot for bimanual daily-living tasks," *Biomed. Phys. Eng. Exp.*, vol. 8, no. 1, 2021, Art. no. 015022.

[37] A. Edsinger and C. C. Kemp, "Two arms are better than one: A behavior based control system for assistive bimanual manipulation," in *Recent Progress in Robotics: Viable Robotic Service to Human*. Berlin, Germany: Springer, 2007, pp. 345–355.

[38] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1653–1660.

[39] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7291–7299.

[40] R. A. Güler, N. Neverova, and I. Kokkinos, "DensePose: Dense human pose estimation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7297–7306.

[41] U. Iqbal, A. Doering, H. Yasin, B. Krüger, A. Weber, and J. Gall, "A dual-source approach for 3D human pose estimation from single images," *Comput. Vis. Image Understanding*, vol. 172, pp. 37–49, 2018.

[42] A. M. Schmidts, D. Lee, and A. Peer, "Imitation learning of human grasping skills from motion and force data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2011, pp. 1002–1007.

[43] S. Scherzinger, A. Roennau, and R. Dillmann, "Contact skill imitation learning for robot-independent assembly programming," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4309–4316.

[44] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 3, pp. 297–330, 2020.

[45] D. Berenson, "Manipulation of deformable objects without modeling and simulating deformation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 4525–4532.

[46] M. Desmurget and C. Prablanc, "Postural control of three-dimensional prehension movements," *J. Neuriophysiol.*, vol. 77, no. 1, pp. 452–464, 1997.

[47] A. Clegg, W. Yu, J. Tan, C. K. Liu, and G. Turk, "Learning to dress: Synthesizing human dressing motion via deep reinforcement learning," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–10, 2018.

[48] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Trans. Syst., Man, Cyber., Part B.*, vol. 37, no. 2, pp. 286–298, Apr. 2007.

[49] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, 1978.

[50] M. Muhlig, M. Gienger, S. Hellbach, J. J. Steil, and C. Goerick, "Task-level imitation learning using variance-based movement optimization," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2009, pp. 1177–1184.

[51] Y. Gao, H. J. Chang, and Y. Demiris, "Iterative path optimisation for personalised dressing assistance using vision and force information," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4398–4403.

[52] G. Chance, A. Jevtić, P. Caleb-Solly, and S. Dogramadzi, "A quantitative analysis of dressing dynamics for robotic dressing assistance," *Front. Robot. AI*, vol. 4, no. 13, pp. 1–14, 2017.

[53] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with Gaussian mixture models," *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 943–957, Oct. 2011.

[54] J. A. McDermid, Y. Jia, and I. Habli, "Towards a framework for safety assurance of autonomous systems," in *Proc. Artif. Intell. Saf. CEUR Workshop Proc.*, 2019, pp. 1–7.

Jihong Zhu (Member, IEEE) received the Ph.D. degree in robotics from the University of Montpellier, Montpellier, France, in 2020.

He conducted his research with LIRMM, France. He was a Postdoc jointly with Cognitive Robotics, TU Delft, Delft, The Netherlands, and Honda Research Institute Europe. He is currently an Assistant Professor in robotics with the School of Physics, Engineering, and Technology and an affiliated Researcher with the Institute for Safe Automony, University of York, York, U.K. He is also a Visiting Researcher with Cognitive Robotics, TU Delft.

Zhu is an Associate Editor for IEEE ROBOTICS AND AUTOMATION LETTERS.

Michael Gienger (Member, IEEE) received the Diploma degree in mechanical engineering and the Ph.D. degree in dissertation on "Design and Realization of a Biped Walking Robot" from the from the Technical University of Munich (TUM), Munich, Germany, in 1998.

From 1998 to 2003, he was a Research Assistant with the Institute of Applied Mechanics, TUM. After this, he joined Honda Research Institute Europe, Germany, in 2003. He is currently a Chief Scientist and Competence Group Leader in the field of robotics. His research interests include mechatronics, robotics, whole-body control, imitation learning, and human–robot interaction.
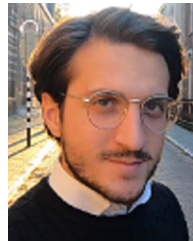
Giovanni Franzese (Member, IEEE) received the B.Sc. degree in mechanical engineering and the M.Sc. degree in mechatronics and robotics from Politecnico di Milano, Milan, Italy, in 2016 and 2018, respectively. He has been working toward the Ph.D. degree in robotics with the Department of Cognitive Robotics, TU Delft, Delft, The Netherlands, since 2019.

Since 2022, he has been an ELLIS member for artificial intelligence. His research focuses on interactive imitation learning applied to robot manipulation.

Jens Kober (Senior Member, IEEE) received the Ph.D. degree in engineering from TU Darmstadt, Darmstadt, Germany, in 2012.

He is currently an Associate Professor with TU Delft, Delft, The Netherlands. He was a Postdoctoral Scholar jointly with CoR-Lab, Bielefeld University, Bielefeld, Germany, and with Honda Research Institute Europe, Offenbach, Germany.

Dr. Kober was a recipient of the annually awarded Georges Giralt Ph.D. Award for the best Ph.D. dissertation in robotics in Europe, the 2018 IEEE RAS Early Academic Career Award, the 2022 RSS Early Career Award, and an ERC Starting grant.