RESEARCH

Open Access

Exploiting upper-limb functional principal components for human-like motion generation of anthropomorphic robots

Giuseppe Averta[1,2,3]*, Cosimo Della Santina[4], Gaetano Valenza[1,3], Antonio Bicchi[1,2,3] and Matteo Bianchi[1,3]

Abstract

Background: Human-likeliness of robot movements is a key component to enable a safe and effective human-robot interaction, since it contributes to increase acceptance and motion predictability of robots that have to closely interact with people, e.g. for assistance and rehabilitation purposes. Several parameters have been used to quantify how much a robot behaves like a human, which encompass aspects related to both the robot appearance and motion. The latter point is fundamental to allow the operator to interpret robotic actions, and plan a meaningful reactions. While different approaches have been presented in literature, which aim at devising bio-aware control guidelines, a direct implementation of human actions for robot planning is not straightforward, still representing an open issue in robotics.

Methods: We propose to embed a synergistic representation of human movements for robot motion generation. To do this, we recorded human upper-limb motions during daily living activities. We used functional Principal Component Analysis (fPCA) to extract principal motion patterns. We then formulated the planning problem by optimizing the weights of a reduced set of these components. For free-motions, our planning method results into a closed form solution which uses only one principal component. In case of obstacles, a numerical routine is proposed, incrementally enrolling principal components until the problem is solved with a suitable precision.

Results: Results of fPCA show that more than 80% of the observed variance can be explained by only three functional components. The application of our method to different meaningful movements, with and without obstacles, show that our approach is able to generate complex motions with a very reduced number of functional components. We show that the first synergy alone accounts for the 96% of cost reduction and that three components are able to achieve a satisfactory motion reconstruction in all the considered cases.

Conclusions: In this work we moved from the analysis of human movements via fPCA characterization to the design of a novel human-like motion generation algorithm able to generate, efficiently and with a reduced set of basis elements, several complex movements in free space, both in free motion and in case of obstacle avoidance tasks.

Keywords: Functional principal components, Human-robot interaction, Rehabilitation robotics, Assistive robotics, Companion robots, Exoskeletons

*Correspondence: giuseppe.averta@ing.unipi.it
[1] Research Center "Enrico Piaggio", University of Pisa, Largo Lucio Lazzarino 1,
56126 Pisa, Italy
[2] Soft Robotics for Human Cooperation and Rehabilitation, Fondazione Istituto
Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy
Full list of author information is available at the end of the article

Introduction

There are many examples in literature that have highlighted the importance of human-likeness (HL) to ensure a safe and effective Human-Robot Interaction (HRI) [1, 2]. This aspect has gained increasing attention, since it could open interesting perspectives for the control of artificial systems that closely interact with humans, as is the case of assistive, companion and rehabilitative robots. For the latter category, for example, human-inspired movement profiles - which are characterized by i) low jerk values at the Cartesian or joint level and ii) bell-shaped velocity profiles (see [3–5]) - could be used as reference trajectories for rehabilitation exoskeletons ([6–9], see also [10, 11] for review), as an alternative to, and/or in association with, classic rehabilitation procedures [12–14].

Indeed, the motion of a robot that shares its environment with humans can be more easily *predicted*, and hence accepted, by the user, if its movements are designed taking inspiration from actual human movements [1, 15], leading to a general enhancement in terms of system usability and effectiveness, especially in assistive robotics applications [16–18]. However, the design of control laws, which effectively ensure human-like behavior in robotic systems, is not straightforward, representing an important topic within the general framework of robot motion planning.

Usually, HL is achieved leveraging on a vast neuroscientific literature to devise cost functions (see [3]), whose optimization introduces HL characteristics in the motion. For example, in [26] human-like artificial motions were generated through jerk minimization, while in [33] the Authors exploited the minimization of joint torques, and in [27] the Virtual Spring-Damper Hypothesis was proposed. Neural Networks have also been used for human-like character animation [29–31]. However, optimization-based methods usually come with hypotheses on motion generation that may limit the variability of the planned movement and, sometimes also lack experimental support [34]. On the other hand, learning methods typically require a large dataset whose dimensionality dramatically increases with task complexity.

To the best of authors' knowledge, a direct exploitation of human observations for robotic arm motion generation has not been applied yet. This approach would come with several advantages, since human-likeness would be intrinsically guaranteed. However, a mere copy-cutting from nature would be unfeasible, and clearly a daunting task. What we propose instead is to use neuromechanistic data, intended here in terms of time-modulation of joint angular values, and model them with a mathematical language, which can be easily understood and effectively implemented in an artificial body. A notable example of this approach is represented by the concept of hand postural synergies, which was mathematically modelled in [35],
and then successfully exploited for the design and control of robotic end-effectors and for grasp planning [36–38].

In this work, we propose to directly embed human upper limb principal motion modes for the planning of anthropomorphic manipulators. To this end, we recorded and organized the joint trajectories of the arm of human subjects performing a set of Daily Living Activities (ADLs) to build a comprehensive dataset. We then applied statistical analysis [39] (namely functional Principal Component Analysis, fPCA) to extract a reduced number of basis functions, or *functional Principal Components*, which explain, for each joint, most of the trajectory variability. As reported later, our results show that a weighted sum of only three functional components takes into account more than 80% of the total variance at joint level.

Capitalizing on these results, we then formulate the planning problem - for a given anthropomorphic manipulator - as an optimization problem. More specifically, the final motion of the manipulator is obtained by solving an optimization problem in a latent space defined by the weights of the functional Principal Components. The core idea of the proposed approach is to use the functional Principal Components extracted from the observations of human movements as basis elements, whose combination is used to optimize the generation of any point-to-point trajectory of the arm in a dummy human. For free-motions, our method results into a closed form solution, which uses only one functional Principal Component.

This methodology comes with a significant perspective shift: from the search for optimal paths to the identification of a reduced number of scalars weighting the functional components. This could enable to rapidly achieve a solution for the planner, which is intrinsically human-like. To further increase the cost-effectiveness of our method, we propose an incremental enrollment of the functional components, as suggested in [40]. In this manner, the number of functional Principal Components needed to perform the task is tailored on task complexity, avoiding the useless inclusion of higher order functional Principal Components.

We demonstrate in simulations that our techniques can generate human-like motion using a reduced number of functional components. The human-likeliness of the generated movements is evaluated according to the indexes reported in [3, Appendix] observing the velocity profiles and jerk values [4, 41].

To conclude, this paper contributes with: i) an extensive study on human upper limb functional Principal Components, which we have pursued by applying our functional analysis approach reported in [39] to a dataset obtained by enrolling a substantially increased number of participants (33) in the experiments; ii) a new methodology for human-like motion generation, which we have applied to the case of point-to-point motions with and without

obstacle avoidance, which intrinsically embeds the principal modes of human upper limb motions; iii) simulation results, that show the effectiveness of our method.

Methods

Functional principal components of upper limb

In a previous work, we presented a functional characterization of upper-limb movements during Activities of Daily Living (ADLs) organized in a dataset that comprised seven subjects. We showed that a reduced-dimensionality set of functional Principal Components (fPCs) accounts for a large part of upper-limb movement variability at joint level [39].

To expand and confirm the validity of our analysis, in this work we built a completely new and extended dataset of ADLs, which includes movements recorded from 33 healthy subjects (17 women, 26.6 y.o. on average). The experimental protocol consisted of 30 activities of daily living as in [39]. Each activity was repeated three times for a grand total of 90 tasks per person.

All experimental procedures were approved by the local Ethical Committee of the University of Pisa. The movements were grouped into three categories, characterized by a different kind of interaction with external objects: *Intransitive*, i.e. gestures; *Transitive*, i.e. movements that involve the interaction with one object; *Tool-Mediated*, i.e. movements in which one object is used to interact with another one. Please refer to the Appendix of this paper for a textual description of the tasks, and to Fig. 1 for their graphical representation. Note that each subject was instructed about the task to accomplish but not on the specific motion to perform for achieving the task goal.

Motion identification

Movements were recorded using a 3D motion tracking system with active markers (Phase Space). Ten stereo-cameras working at 100Hz tracked the 3D position of active markers, which were placed on rigid supports and attached to upper limb links (see Fig. 2).
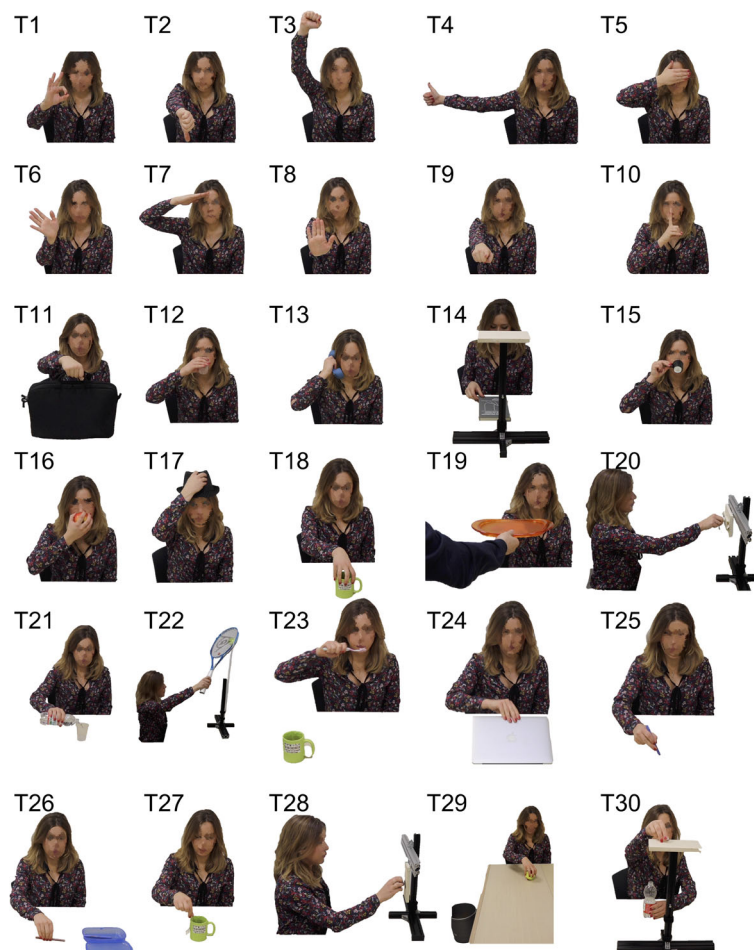
Fig. 1 From top to bottom, and from left to right, pictures of the tasks considered in this study. Verbal description of the tasks is reported in Table 5
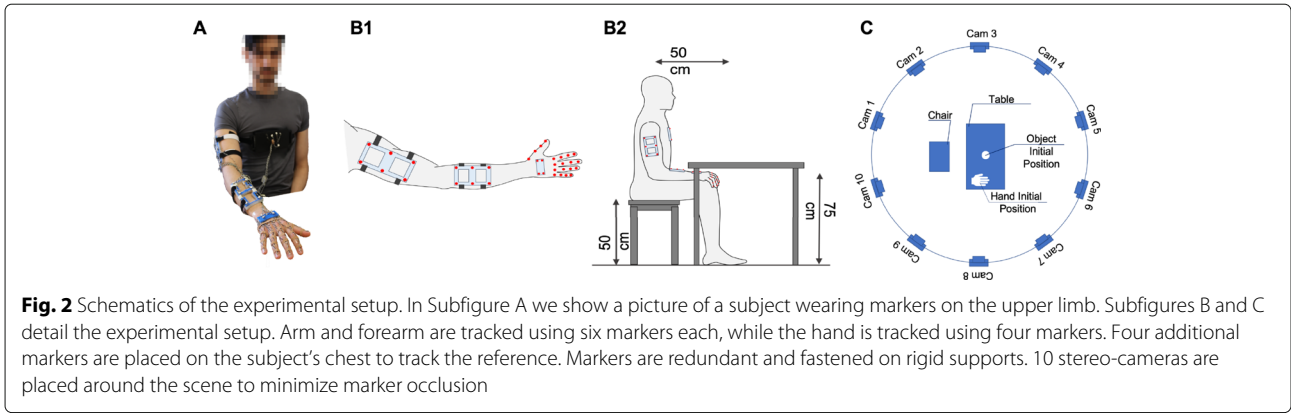
Fig. 2 Schematics of the experimental setup. In Subfigure A we show a picture of a subject wearing markers on the upper limb. Subfigures B and C detail the experimental setup. Arm and forearm are tracked using six markers each, while the hand is tracked using four markers. Four additional markers are placed on the subject's chest to track the reference. Markers are redundant and fastened on rigid supports. 10 stereo-cameras are placed around the scene to minimize marker occlusion

To map the recorded movements on an analytic kinematic model, we developed a two-phase procedure, which consisted of a preliminary calibration of the model, followed by a Kalman-based identification. In this work we used the same kinematic model discussed in [39], i.e. a serial manipulator with three rigid links connected by seven revolute joints (see Fig. 3). This model can be completely defined using 14 parameters: 2 parameters for bone length and 12 parameters to identify the position of the markers with respect to the kinematic chain (more technical details can be found in [39]). This set of parameters (p_G) was calibrated, for each subject, by solving the following optimization problem:

$$(q^*, p_G^*) = \arg \min_{q_k \in D_q, p_G \in D_p} \frac{1}{2} \sum_{k=1}^{N_p} r_k^T r_k, \tag{1}$$

where r_k is the residual function:

$$r_k(q_k, p_G) = y_k - f(q_k, p_G) \tag{2}$$

at time frame k, calculated as the difference between the measured position of markers y_k and their estimation via Forward Kinematics (FK) $f(q_k, p_G)$. The FK is function of q_k, i.e. the current estimation of joint angles, and p_G, i.e. the parameters of the kinematic model. The optimization

problem is constrained by D_q, i.e. the range of motion of the joints, and D_p, i.e. the maximum variation of parameters with respect to a preliminary manual estimation. N_p refers to the number of samples used for the calibration – 10 equally time-spaced samples in our case, as in [39, 42]. Given the calibrated vector p_G for the specific subject, the upper limb pose is completely described by 7 joints angles $[q_1, \ldots, q_7]^T$.

The calibrated model was then used as a component for a Kalman-based identification procedure. More specifically, if we consider, at time frame k, the joints angle vector q_k as the state of a stochastic process with w_k and v_k process and observation zero mean Gaussian noises, with covariance Q_k and R_k respectively, we have that the system can be written as

$$\begin{cases} q_k = q_{k-1} + w_k \\ y_k = f(q_k) + v_k \end{cases} \tag{3}$$

Recursively, the state q_k can be estimated from q_{k-1} with the following implementation of a Kalman filter [43]: *Prediction* of the future state $\hat{q}_{k|k-1} = \hat{q}_{k-1}$; *Update* of the state estimation as $\hat{q}_{k|k} = \hat{q}_{k|k-1} + K_k \tilde{r}_k$. The prediction correction is the product between the residual values vector $\tilde{r}_k = y_k - f(\hat{q}_{k|k-1})$ and the Kalman Gain K_k, defined as product between the covariance matrix estimation of the
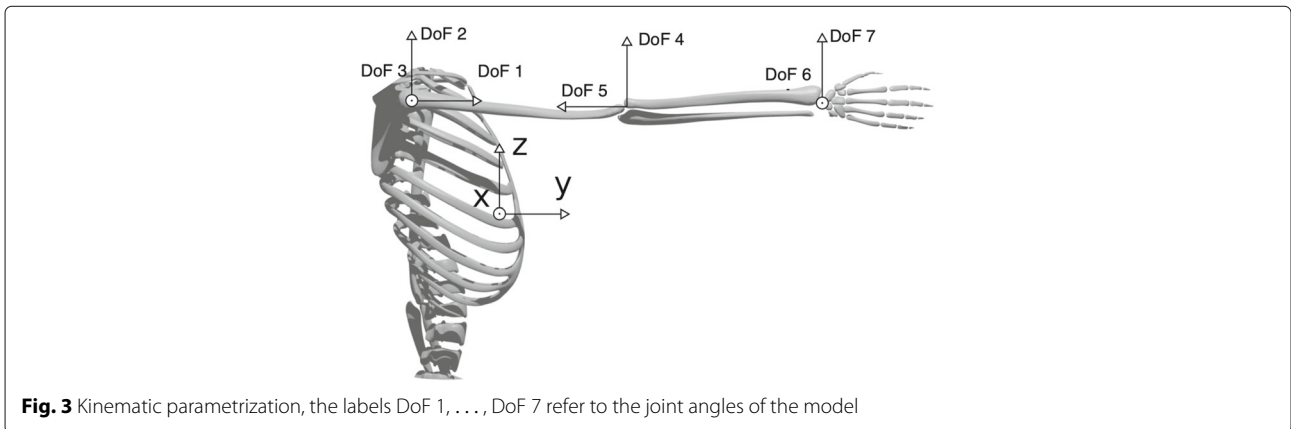
Fig. 3 Kinematic parametrization, the labels DoF 1, . . . , DoF 7 refer to the joint angles of the model

predicted state $P_{k|k-1}$, the Jacobian matrix $H_k = \frac{\partial(f(q))}{\partial q}$ and the inverse of the residual covariance.

Finally, to effectively and jointly process different acquisitions with different temporal lengths, a time normalization is required. To this end, we used Dynamic Time Warping. More specifically, given a reference signal extracted from the dataset $q_{ref}(t)$, all the trajectories were normalized in time with respect to (w.r.t.) the reference one, by solving the following minimization problem

$$(S_i, T_i) = \arg \min_{S_i>0,T_i} (||q_{ref}(t) - q_i(S_i t - T_i)||), \quad (4)$$

where S_i and T_i are the time-stretch and the translation parameter respectively.

Functional analysis

In the following, we will briefly summarize the main theoretical concepts behind functional Principal Component Analysis (fPCA), and discuss their application to the investigation of human upper limb motions. Let us consider a generic upper-limb movement $q(t) : \mathbb{R} \to \mathbb{R}^7$, where $t \in [0, t_{fin}]$ is the time variable, and 7 is the number of the upper limb degrees of freedom (DoFs) of the kinematic model we considered for our analyses. The goal of fPCA targets the identification of a suitable reduced functional representation, which can closely approximate $q(t)$ (joint trajectories). For these reasons, fPCA can be regarded as a functional extension in the temporal domain of Principal Component Analysis (PCA). Indeed, as Principal Components identify inter-joint couplings that account for most kinematic pose variability, functional PCs (fPCs) are functions that allow to describe most of the movement variability over time, at joint level. Using the fPCA framework, a generic upper limb motion $q(t)$ can be described as a weighted sum of a set of base functions $S_i(t)$, or functional Principal Components, that is:

$$q(t) \simeq \bar{q} + S_0(t) + \sum_{i=1}^{s_{max}} \alpha_i \circ S_i(t) , \quad (5)$$

where $\alpha_i \in \mathbb{R}^n$ is a vector of weights. n is the dimension of $q(t)$, in our case equal to 7. $S_i(t) \in \mathbb{R}^n$ is the ith basis element and s_{max} is the number of basis elements. The operator $\circ$ is the Hadamard product, i.e. the element-wise product [44]. $\bar{q} \in \mathbb{R}^7$ is the average posture of q evaluated as

$$\bar{q} = \int_0^{t_{fin}} q(\tau)d\tau , \quad (6)$$

while $S_0 : \mathbb{R} \to \mathbb{R}^7$ is the *zero-order* synergy, i.e. the average trajectory across all the trajectories q in the dataset, for all the tasks and subjects.

fPCA is used to identify a basis of functions $\{S_1, \ldots, S_{s_{max}}\}$ that maximizes the explained variance of the movements in the collected dataset. The first fPC $S_1(t)$ is the function that solves the following problem

$$\min_{S_1} \sum_{j=1}^{R} \left(\int S_1(t)q_j(t)dt \right)^2$$

$$\text{subject to} \quad ||S_1(t)||_2^2 = \int_0^{t_{fin}} S_1^2(t)dt = 1 , \quad (7)$$

where R is the number of trajectories that compose our dataset. The other fPCs $S_i(t)$ are the functions that solve the following constrained optimization problems

$$\min_{S_i} \sum_{j=1}^{R} \left(\int S_1(t)q_j(t)dt \right)^2$$

$$\text{subject to} \quad ||S_i(t)||_2^2 = 1$$

$$\int_0^{t_{fin}} S_i(t)S_k(t)dt = 0 , \ \forall k \in \{1, \ldots, i-1\} . \quad (8)$$

This results in an ordered set of functions that - given a number of basis elements s_{max} - maximizes the explained variability of the joint trajectories of the dataset. This opens the possibility to use these functions as a *smart basis* to generate HL movements in a latent space. In the remaining of this work, we present an algorithm that

Table 1 Initial (q_0) and final (q_{fin}) configurations for the four tasks considered

Task		dof 1	dof 2	dof 3	dof 4	dof 5	dof 6	dof 7
T.1	q_0	−0.31	−0.58	−0.8	0.47	−1.23	0.22	0.65
	q_{fin}	−0.33	0.34	−0.39	0.44	−0.69	0.36	0.15
T.2	q_0	0.09	−0.79	−0.62	0.65	−1.50	0.06	0.76
	q_{fin}	0.65	−0.40	0.07	0.79	−1.32	0.87	0.63
T.3	q_0	−0.48	−0.07	−0.71	0.03	−0.86	0.52	0.47
	q_{fin}	−1.67	−0.68	−0.33	−0.21	−0.09	−0.26	0.52
T.4	q_0	−0.28	−0.86	−0.92	0.86	−1.52	−0.12	0.77
	q_{fin}	0.65	−0.78	−0.12	2.18	−1.52	0.57	0.59

Angles are expressed in radians

exploits functional Principal Components extracted from the analysis of human upper limb movements for the generation of point-to-point robotic/artificial motions, with and without the presence of obstacles along the trajectory. These movements naturally embed HL characteristics, (see next section for the proposed method).

To test the effectiveness of this approach, we performed simulations of both free-motion and obstacle-avoidance cases. We used a robotic arm simulator with the same kinematic description we used during the experiments with human subjects. Hereinafter we refer to this simulator as *dummy human*. We considered four motions that are representative of the upper limb movement workspace (see the Results section): T.1) accounts for right-to-left side motions of the trunk, which typically sub-serve the execution of actions for maneuvering objects on a table; T.2) is representative of movements towards a position in front of the subject. T.3) relates to arm elevation towards a position upon the subject's head; T.4) deals with reaching movements toward the subject's own face, which are usually executed for self-care tasks, e.g. self-feeding etc. Numerical values of the 7 joint angles in the initial and desired final conditions, q_0 and q_{fin}, respectively, for each of the four tasks are reported in Table 1.

In our simulations, we considered three scenarios for each task: no obstacle (i.e. free motion), one obstacle, and two obstacles. In the latter two scenarios, we decided to place the obstacles along the optimal trajectories computed for the free-motion case. In this manner, we forced the algorithm to modify the initial guess. Details on the obstacle location – with respect to the Inertial System of Reference placed as in Fig. 3 – and dimension (expressed in terms of the center and radius of a sphere surrounding the obstacle, respectively) are reported in Table 2.

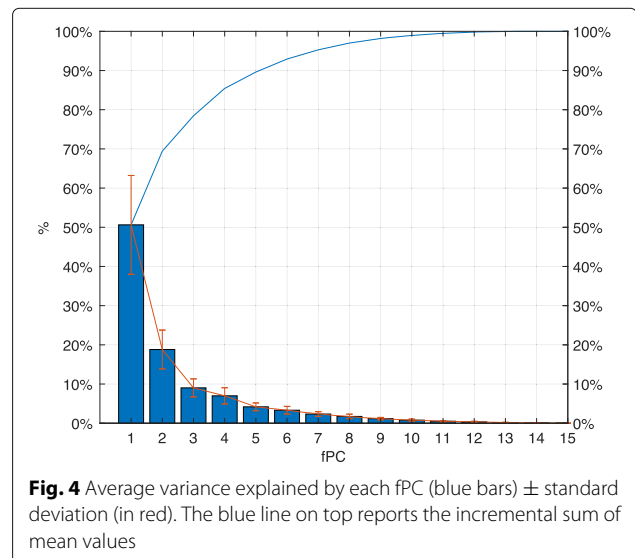## Motion generation via functional principal components

As discussed in the Introduction, typical approaches used in literature to achieve human likeness [26] in robotic motions rely on the strong assumption that human movements are generated by optimizing a known cost function $J_{hl}(q) : C_7^1[0, t'_{fin}) \rightarrow \mathbb{R}^+$, where $C_7^1[0, t'_{fin})$ is the space of smooth functions going from $[0, t'_{fin})$ to the joint space $\mathbb{R}^7$, and t'_{fin} is the final time, which in general will be different from t_{fin} as defined in the previous section. The function J_{hl} is used to produce artificial natural motions by solving the problem

$$\min_{q \in C_7^1[0, t'_{fin})} J_{hl}(q) . \tag{9}$$

How to choose J_{hl} is not obvious, and it is indeed a very debated topic in literature. However only achieving human likeness is meaningless without specifying also a task to be accomplished.

For this reason also a model of the task should be added to (9). We formulate the latter point in terms of the minimization of an additional cost function $J_{task} : C_7^1 \rightarrow \mathbb{R}^+$. As soon as the need for minimizing J_{task} is introduced, (9) becomes a multi-objective optimization, which is of very difficult formulation and solution, except for very simple cases [26].

In this work, we propose an approach that allows to by-pass this issue. Instead of using data to guess a reasonable $J_{hl}(\cdot)$, and then explicitly optimize it, we propose to directly embed human likeness in the choice of the functional subspace where the optimization occurs. More specifically, we move from the infinite dimensional functional space $C_7^1[0, t'_{fin})$, to its finite dimensional subspace

Table 2 Details on the obstacle location *Position* – with respect to the Inertial System of Reference placed as in Fig. 3 – and dimension *Radius* (expressed in terms of the center and the radius of a sphere surrounding the obstacle, respectively) considered during the simulations, for the one obstacle (*1 Obs*) and two obstacle (*2 Obs*) case, for the different tasks (*T.1, T.2, T.3, T.4*)

	1 Obs		2 Obs	
	Position	Radius	Position	Radius
T.1	[-167 200 323]	90	[-167 200 323]	40
			[-89 96 436]	40
T.2	[0 220 350]	50	[0 220 350]	50
			[70 170 135]	75
T.3	[313 480 321]	70	[313 480 321]	70
			[-2 311 353]	70
T.4	[-35 202 310]	40	[-35 202 310]	40
			[-168 158 322]	40

These quantities are expressed in [mm]

Fig. 4 Average variance explained by each fPC (blue bars) ± standard deviation (in red). The blue line on top reports the incremental sum of mean values
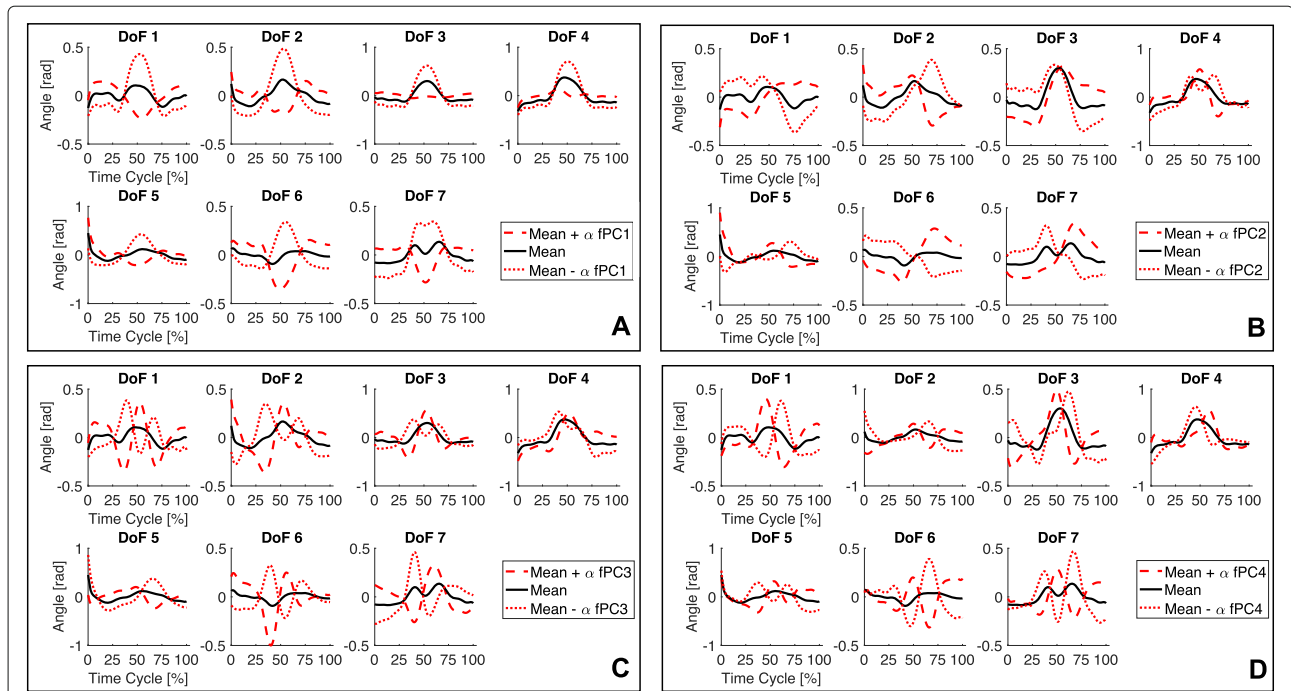
Fig. 5 From figure A to D, mean trajectory (in black) and the contribution (signed) of each functional Principal Component, from fPC1 to fPC4 respectively. Components ordering follows the indexing of Fig. 3

containing all the functions so constructed:

$$q(t) = \bar{q} + S_0(t'(t)) + \sum_{i=1}^{M} \alpha_i \circ S_i(t'(t)) \qquad (10)$$

with $\bar{q}, S_i, \alpha_i$ defined as in the previous section and t' is a linear warping w.r.t. the definition of time used in the

previous section, i.e. $t'(t) = \frac{t'_{\text{fin}}}{t_{\text{fin}}} t$. In this way the principal components can be used to generate motions happening within any time horizon $[0, t'_{\text{fin}})$.

$M \leq s_{\max}$ is the number of functional Principal Components considered in the optimization (with $s_{\max}$ as in (5)). According to the results preliminary presented in
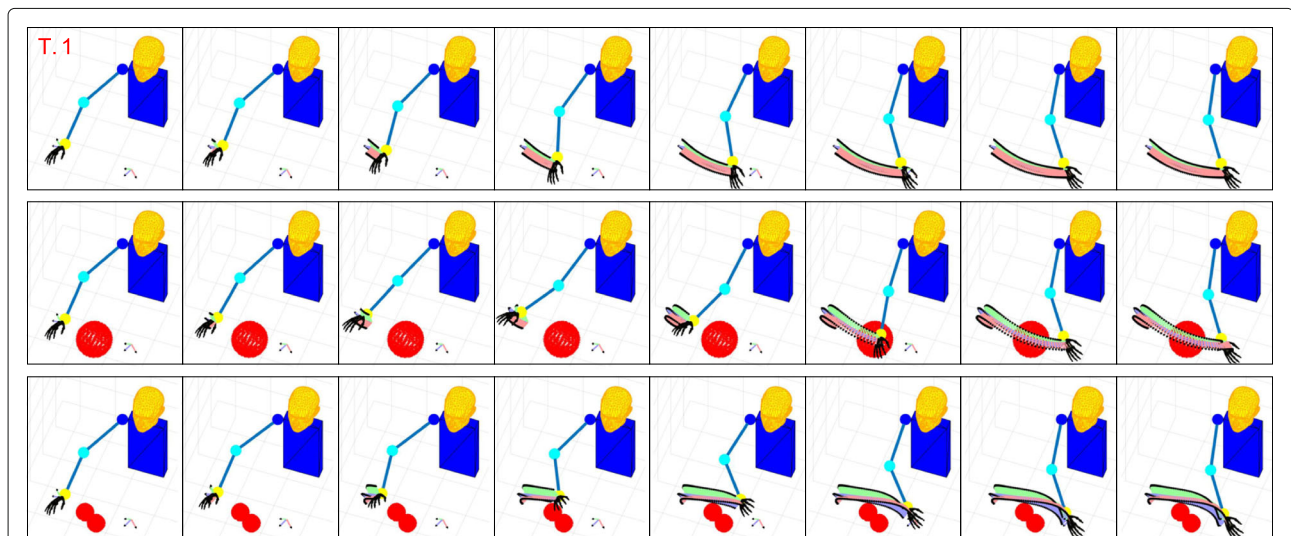
Fig. 6 From top to bottom, the reconstruction for task 1 is reported for the free motion case, the single obstacle avoidance and the double obstacle avoidance case respectively. For each row, motion execution is intended from left to right and time-frames are evenly spaced. Reference system at wrist level is finally plotted along the whole trajectory

Fig. 7 From top to bottom, the reconstruction for task 2 is reported for the free motion case, the single obstacle avoidance and the double obstacle avoidance case respectively. For each row, motion execution is intended from left to right and time-frames are evenly spaced. Reference system at wrist level is finally plotted along the whole trajectory
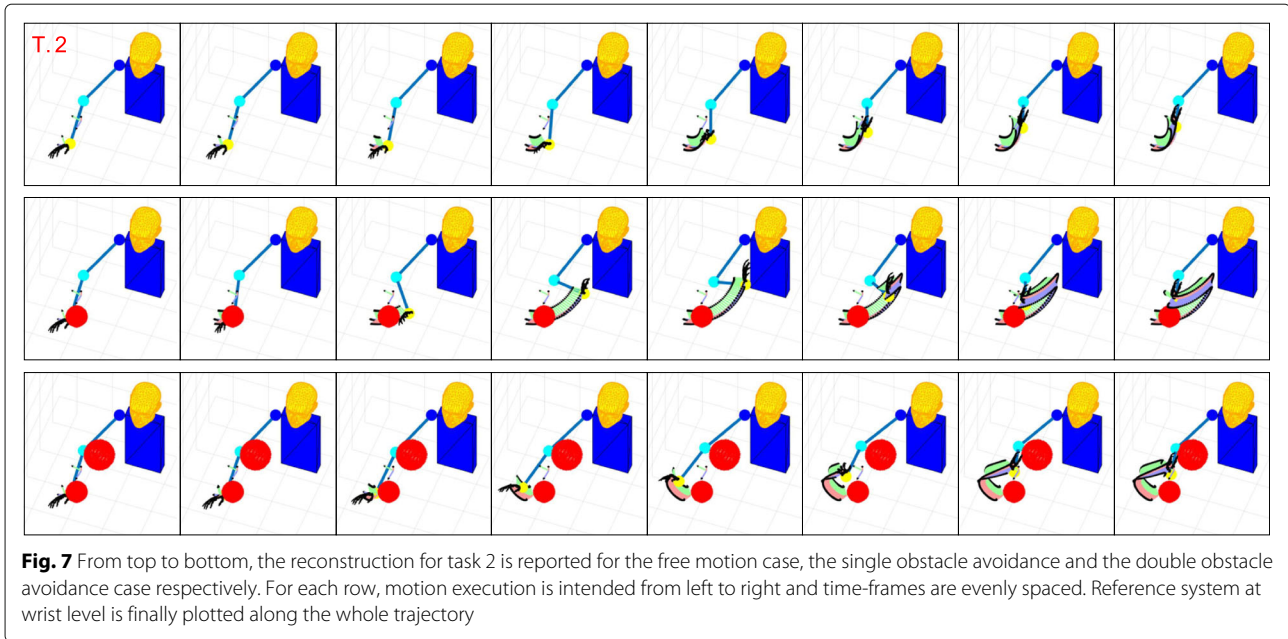
[39] and further extended and confirmed in this paper, it is plausible to expect that a low number of functional Principal Components should be sufficient to implement most of the human-like motions at the joint level. Therefore the multi-object and unconstrained optimization can be formulated as the following constrained optimization problem:

$$\min_{\bar{q},\alpha_1,\ldots,\alpha_M} \quad J_{\text{task}}(q)$$

$$\text{subject to} \quad q(t') = \bar{q} + S_0(t') + \sum_{i=1}^{M} \alpha_i \circ S_i(t') \,. \tag{11}$$

In this manner, we can narrow the search space, with the twofold purpose of ensuring human likeness, and strongly simplifying the control problem (indeed, the search space is now of dimension $M + 1$). In the following subsections we present the application of this approach, tailoring J_{task} on the generation of simple point-to-point free movements, as well as more complex motions with obstacle avoidance.

Point-to-Point free motions
We propose here to generate a human-like point-to-point motion by solving the following optimization problem, instance of the more general formulation (11)

$$\min_{\bar{q},\alpha_1,\ldots,\alpha_M} \quad ||q(0) - q_0||_2^2 + ||q(t'_{\text{fin}}) - q_{\text{fin}}||_2^2$$

$$\text{subject to} \quad q(t') = \bar{q} + S_0(t') + \sum_{i=1}^{M} \alpha_i \circ S_i(t') \,, \tag{12}$$

where $q(0)$ and $q(t'_{\text{fin}})$ are the initial and final poses of the calculated trajectory, while q_0 and q_{fin} are the desired initial and final poses respectively. In this simple case, a single functional Principal Component (i.e. $M = 1$) is already sufficient to solve (12) with zero error.

We start by imposing the zero error conditions $q_0 = q(0)$, $q_{\text{fin}} = q(t'_{\text{fin}})$, which can be rewritten by using (10)

$$q_0 = \bar{q} + S_0(0) + \alpha_1 \circ S_1(0)$$
$$q_{\text{fin}} = \bar{q} + S_0(t'_{\text{fin}}) + \alpha_1 \circ S_1(t'_{\text{fin}}) \,. \tag{13}$$

We subtract the second equation from the first, and exploit the associativeness of Hadamard product, to obtain

$$q_{\text{fin}} - q_0 = S_0(t'_{\text{fin}}) - S_0(0) + \alpha_1 \circ (S_1(t'_{\text{fin}}) - S_1(0)), \tag{14}$$

which implies

$$\alpha_1 = [\,(q_{\text{fin}} - q_0) - (S_0(t'_{\text{fin}}) - S_0(0))\,] \circ (S_1(t'_{\text{fin}}) - S_1(0))^{\circ-1} \tag{15}$$

where $\circ^{-1}$ is the Hadamard inverse, as defined in [44]. Substituting (15) back into (13) yields

$$
\begin{aligned}
\bar{q} &= q_0 - \alpha_1 \circ S_1(0) \\
&= q_0 - [\,[\,(q_{\mathrm{fin}} - q_0) - (S_0(t'_{\mathrm{fin}}) - S_0(0))] \\
&\quad \circ (S_1(t'_{\mathrm{fin}}) - S_1(0))^{\circ-1}] \circ S_1(0)\,,
\end{aligned}
\tag{16}
$$

which fully specifies the trajectory that performs the point to point motion while being human-like.

Obstacle avoidance

Let us consider the case in which we also need to avoid one or more obstacles, while performing the point-to-point motion. We can generalize the generation of the human-like trajectory as

$$
\min_{\bar{q}, \alpha_1, \dots, \alpha_M} \left\| \begin{bmatrix} q(0) - q_0 \\ q(t'_{\mathrm{fin}}) - q_{\mathrm{fin}} \end{bmatrix} \right\|_2^2 + \rho P(q, P_O)
$$

$$
\text{subject to} \qquad q(t') = \bar{q} + S_0(t') + \sum_{i=1}^{M} \alpha_i \circ S_i(t')\,.
\tag{17}
$$

Two terms can be distinguished in this cost function. The first contribution guarantees that the desired initial and final poses are achieved, as for the free motion case (12). The second term takes into account the distance w.r.t. obstacles. For the sake of conciseness, and without any loss of generality, we assume here N_O spherical obstacles. We call $P_O = \{P_{O_1}, \dots, P_{O_{N_O}}\}$ the set containing the Cartesian coordinates of all the centers of these obstacles.

$P(q, P_O)$ is a potential-based function [45] that sums up, for each obstacle, a term inversely proportional to the minimum distance between the obstacle and the closest joint trajectory, i.e.

$$
P(q, P_O) = \sum_{i=1}^{N_O} \frac{1}{m_i(q([\,0, t'_{\mathrm{fin}}\,]), P_{O_i})^2}
\tag{18}
$$

where m_i is the distance between the arm and the $i - th$ obstacle, defined as

$$
m_i(q([\,0, t'_{\mathrm{fin}}\,]), P_{O_i}) = \min_k \{d(h_k(q([\,0, t'_{\mathrm{fin}}\,])), P_{O_i})\}\,.
$$

The distance between the $k - th$ point of contact with forward kinematics h_k, and the $i - th$ sphere is

$$
d(h_k(q([\,0, t'_{\mathrm{fin}}\,])), P_{O_i}) = \max \left\{ \min_{x \in h_k(q([0, t'_{\mathrm{fin}}]))} \|P_{O_i} - x\|_2, R_{O_i} \right\},
\tag{19}
$$

with R_{O_i} radius of the sphere.

Incremental optimization procedure

The problem of motion generation with obstacle avoidance does not have a closed-form solution, hence the optimal trajectory is calculated via numerical optimization. To do this, we took inspiration by the ordering of fPCs basis, according to a descending amount of the associated explained variance, and implemented an incremental procedure reported in Algorithm 1.

The proposed approach calculates for each step the optimal trajectory that minimizes the error in starting and final position while maximizing the distance from the obstacles. We consider as initial condition the one specified by (15) and (16). If the corresponding solution is sufficiently far from the obstacles, this choice already defines the globally optimal solution. If the obstacles are
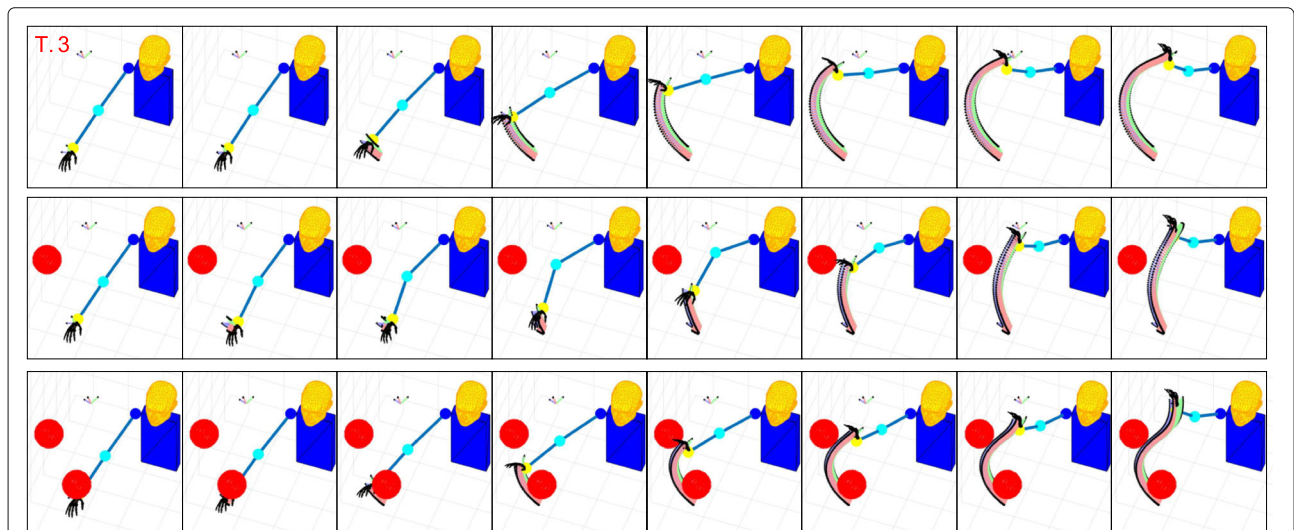
Fig. 8 From top to bottom, the reconstruction for task 3 is reported for the free motion case, the single obstacle avoidance and the double obstacle avoidance case respectively. For each row, motion execution is intended from left to right and time-frames are evenly spaced. Reference system at wrist level is finally plotted along the whole trajectory

Algorithm 1 Procedure of Motion Generation

1: **procedure** INITIALIZATION
2:     Load fPCs basis set;
3:     Load Obstacles set;
4:     Set initial and final pose;
5:     $M \leftarrow 1$;
6:     $r_0 \leftarrow \infty$;
7:     $i \leftarrow 0$;
8:     Initialize α_{in} as in (15);
9:     Initialize $\bar{q}_{\text{in}}$ as in (16);
10: **procedure** OPTIMIZATION
11:     $i \leftarrow i + 1$;
12:     $[\alpha_{\text{opt}}, \bar{q}_{\text{opt}}] \leftarrow \text{InteriorPoint}(\alpha_{\text{in}}, \bar{q}_{\text{in}})$;
13:     Calculate residual cost r_i via Eq. (17);
14:     **if** $r_i - r_{i-1} \leq R_{\text{tol}}$ **or** $M = S_{\text{max}}$ **then**
15:         Break;
16:     **else**
17:         $\alpha_{\text{in}} \leftarrow \alpha_{\text{opt}}$;
18:         $\bar{q}_{\text{in}} \leftarrow \bar{q}_{\text{opt}}$;
19:         $M \leftarrow M + 1$;
20:         Go to Step 10;
    end
end

---

not very close to the aforementioned trajectory, then solving (17) with $M = 1$ would fine tune the initial guess, achieving good results.

In case of obstacles very close to or even intercepting the free-motion trajectory, at least one more fPC should be enrolled to suitably solve the problem. The more are the basis elements enrolled, the more complex are the final trajectories that can be implemented. However, a higher cardinality of enrolled elements usually comes with a larger computational cost and dramatically increases the number of local minima, thus boosting the complexity of the problem.

Results

Functional principal components of upper limb motion

In Fig. 4, we report, for each joint, the functional Principal Components over the time. As shown in Fig. 5, the first four fPCs together (i.e. a weighted sum of them) account for around the 85% of the total variance of the dataset.

What is also noticeable is that lower order functional Principal Components (i.e. fPC1 and fPC2) introduce low-frequency contributions to the mean function, while higher order functional Principal Components add higher-frequency terms to the signal. In other terms, this behavior suggests that lower order functional Principal Components are mainly devoted to a gross definition of the whole trajectory, while higher order components deal with the fine tuning of the specific movement. Note that these findings are totally coherent and, hence, further confirm the outcomes of [39].

Human-Like motion generation: validation

Figures 6, 7, 8, and 9 present graphical representations of the human-like trajectories generated by the proposed method, from T.1 to T.4. The first row of each figure depicts the free motion case, the second row the one-obstacle case, and the third the two-obstacle one. In all the considered cases the result is a smooth trajectory connecting the two postures without interacting with the
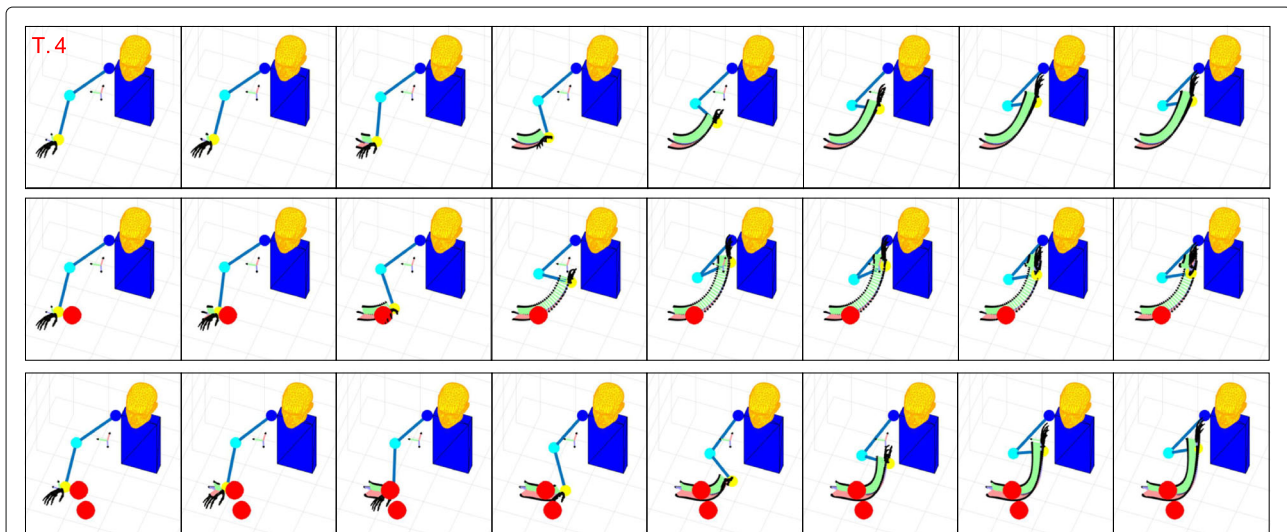
Fig. 9 From top to bottom, the reconstruction for task 4 is reported for the free motion case, the single obstacle avoidance and the double obstacle avoidance case respectively. For each row, motion execution is intended from left to right and time-frames are evenly spaced. Reference system at wrist level is finally plotted along the whole trajectory

Fig. 10 From top to bottom, the reconstruction for task 1 and 2 are reported in case of obstacles on the elbow. For each row, motion execution is intended from left to right and time-frames are evenly spaced. Reference system at wrist level is finally plotted along the whole trajectory
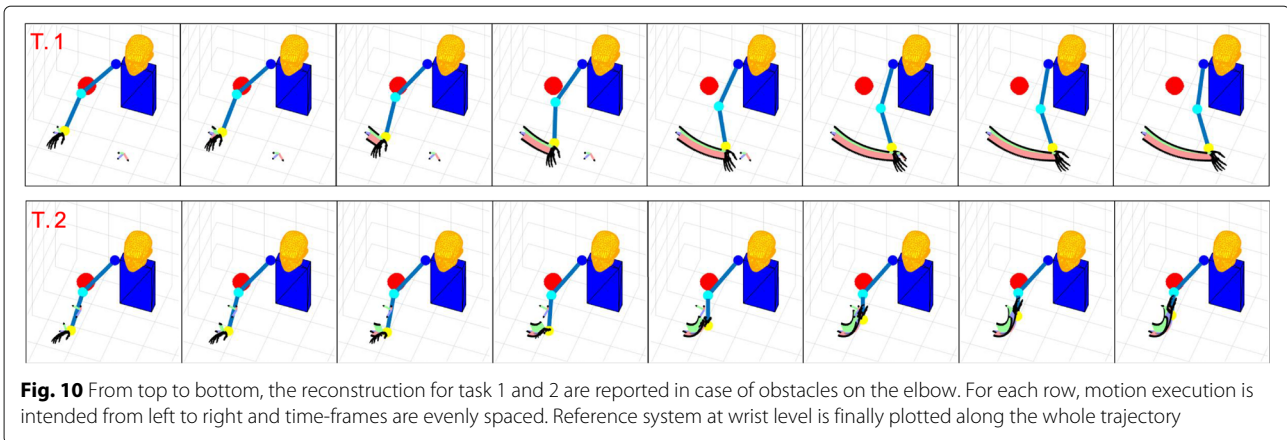
obstacles. This is also evident from Fig. 10, where we present the profiles of angles and angular velocities over time.

Table 3 reports, for each task and each environment, the number of functional Principal Components retrieved by Algorithm 1. As expected, in the zero-obstacle case, all tasks can be executed using only one fPC. When obstacles are introduced, the number M of functional Principal Components increases. However, the complex actions that we can observe in Figs. 6, 7, 8, and 9 can be performed with a maximum of three functional Principal Components. Numerical values of the minimum distance (expressed as Euclidean norm) between the manipulator and the obstacles in the simulations, considering both the free motion case and the Obstacle Avoidance part, are reported in Table 4.

It is worth noticing that, considering our optimization routine with the Obstacle Avoidance (OA) part, the minimum distance between the dummy manipulator and the obstacle is always greater than the radius of the obstacles (whose dimensions and position with respect to the Inertial System of Reference – see Fig. 3 – are reported in Table 2). This does not apply when the OA part is not implemented (i.e. free motion case).

In the considered simulations, the first fPC alone accounts for the 96% of the cost reduction, while - with only the first three functional Principal Components - 100% reduction of the cost is achieved. This is coherent

with the results in Table 3, and confirms the effectiveness of the proposed strategies for selecting the number of functional Principal Components to be used. Additional experiments have been performed to show how the algorithm works in case of obstacles close to the elbow trajectory. We reported in Fig. 11 results for tasks 1 and 2. The interested reader is invited to refer to the attached video from different points of view [1].

Human-Likeliness of trajectories

The method proposed in this work embeds human-like profiles for upper-limb movements without explicitly constraint this behavior in the optimization problem. As discussed previously, it has been observed that human movements are typically characterized by i) bell-shapes of velocity profiles [3, 4]; ii) minimization of jerk values [3]. For this reason, in this work we rely on this observation to verify the human-likeliness of the trajectories generated though our method.

Regarding the velocity profiles, in Fig. 10, we report the angular profiles and the corresponding velocities, for all the considered simulations. The bell-shaped distributions are effectively embedded in the considered evolutions, not only in the case of point-to-point free motion but also

Table 3 # of functional Principal Components (*M*) enrolled by Algorithm 1

Number of Obstacles	Task 1	Task 2	Task 3	Task 4
0	1	1	1	1
1	2	2	2	2
2	3	2	2	2

Table 4 Minimum distance (in [mm]) between the manipulator and the obstacles with (*With OA*) and without (*Without OA*) the Obstacle Avoidance (OA) part, for *1 Obs* and *2 Obs* case and the different tasks *T.1, T.2, T.3, T.4*

	1 Obs		2 Obs	
	Without OA	With OA	Without OA	With OA
T.1	9.5	164	(9.5, 129.5)	(164, 147)
T.2	18	86.5	(18, 75)	(170, 92)
T.3	33.5	157	(33.5, 121)	(117.5, 107.5)
T.4	48	88.5	(48, 119.5)	(88.5, 122.5)

while avoiding contact with one or more obstacles. More-over, the norm of the trajectories jerk is always lower than $5 \; 10^{-7}$rad ($7.92 \; 10^{-8} \pm 8.53 \; 10^{-8}$rad), in line with the observations of [5]. Finally, all the evolutions of angular velocity present a marked bell-shaped profile, as observed in [4]. Few exceptions can be evidenced mostly for the elbow joint (dof 4), which however can be seen as a sequence of two bell shaped trajectories, one for the opening and one for the closing.

Discussions

In this paper, we presented a novel method to endow robotic manipulators with human-like movements by directly using human principal motion modes identified through functional analysis. In this way, human-likeness is intrinsically guaranteed.

This leaves room to optimize other aspects related to motion generation, such as obstacle avoidance - as done in this work - energetic consumption, joint torques, time efficiency and so on. It is worth noticing that the aim of our work is not to solve planning problems but instead to propose a new method for intrinsically embedding human-likeness in motion generation. Future work will encompass the evaluation with sophisticated planning problems and state of the art planning benchmarks.

In the general case, the resulting optimization problem deals with a non-linear cost function which may present local minima. The current implementation relies on the `MATLAB` optimization routine `fmincon`, which implements a gradient-based local optimization. This does not necessarily guarantee the convergence to the global optimum, for which the use of global optimization algorithms would be needed and will be considered as future work. Moreover, any comparison with other existing algorithms in terms of execution time is beyond the scope of this work. Nevertheless, we do believe that the proposed
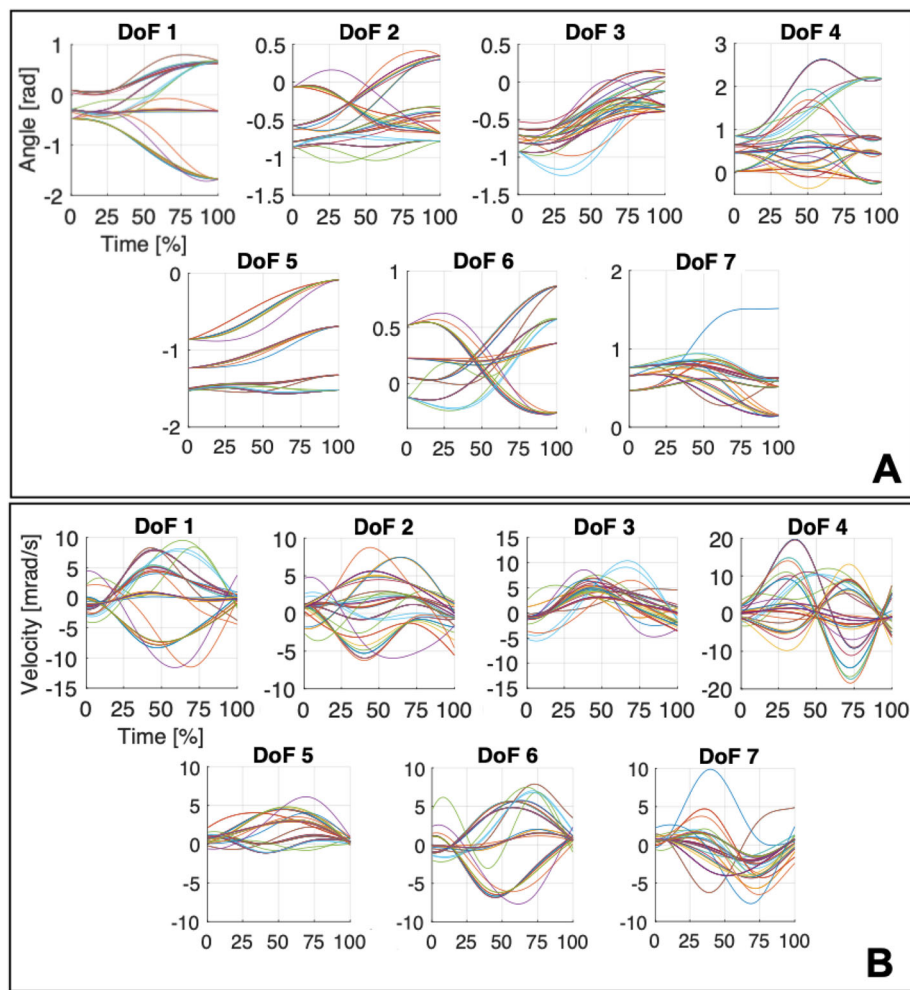
Fig. 11 Angular positions and velocities for a set of simulations performed on movements T.1, T.2, T.3 and T.4 with different obstacles. In Subfigure A, we report the angular positions for the seven DoFs, arranged following the order of Fig. 4, while in Subfigure B we show the corresponding angular velocities

Table 5 List of Movements

#	# Cut	Class	Description
1		Int	Ok gesture (lifting hand from the table)
2		Int	Thumb down (lifting hand from the table)
3		Int	Exultation (extending the arm up in the air with closed fist)
4		Int	Hitchhiking (extending the arm along the frontal plane, laterally, parallel to the floor, with extended elbow, closed fist, extended thumb)
5		Int	Block out sun from own face (touching the face with the palm and covering the eyes)
6		Int	Greet (with open hand, moving wrist) (3 times)
7		Int	Military salute (with lifted elbow)
8		Int	Stop gesture (extending the arm along the sagittal plane, parallel to the floor, open palm)
9		Int	Pointing (with index finger) of something straight ahead (with outstretched arm)
10		Int	Silence gesture (bringing the index finger, with the remainder of the hand closed, on the lips)
11	2	Tr	Reach and grasp a small suitcase from the handle, lift it and place it on the floor (close to own chair, along own sagittal plane)
12	3	Tr	Reach and grasp a glass, drink for 3 seconds and place it in the initial position
13	4	Tr	Reach and grasp a phone receiver, carry it to own ear for 3 seconds and place it in the initial position
14	6	Tr	Reach and grasp a book (placed overhead on a shelf), put in on the table and open it (from right side to left side)
15	8	Tr	Reach and grasp a small cup from the handle (2 fingers + thumb), drink for 3 seconds and place it in the initial position
16	11	Tr	Reach and grasp an apple, mimic biting and put it in the initial position
17	12,13	Tr	Reach and grasp a hat from its top and place it on own head
18	12	Tr	Reach and grasp a cup from its top, lift it and put it on the left side of the table
19	15	Tr	Receive a tray (straight ahead, with open hand) and put it in the middle of the table
20	16	Tr	Reach and grasp a key in a lock (vertical axis), extract it from the lock and put it on the left side of the table
21	1	T-M	Reach and grasp a bottle, pour water into a glass and put the bottle in the initial position
22	2,3,4	T-M	Reach and grasp a tennis racket (placed along own frontal plane) and play a forehand (the subject is still seated)
23	5	T-M	Reach and grasp a toothbrush, brush teeth (horizontal axis, one time left-right) and put it inside a holder (on the right side of the table)
24	6	T-M	Reach and grasp a laptop, open it (without changing its position) (4 fingers + thumb)
25	7,8,9	T-M	Reach and grasp a pen (placed on the right side of the table) and draw a vertical line on the table (from the top to the bottom)
26	7	T-M	Reach and grasp a pencil (placed along own frontal plane) (3 fingers + thumb) and put it inside a squared pencil holder (placed on the left side of the table)
27	9	T-M	Reach and grasp a tea bag in a cup (1 finger + thumb), remove it from the cup and place it on the table on the right side of the table
28	10	T-M	Reach and grasp a doorknob, turn it clockwise and counterclockwise and open the door
29	13	T-M	Reach and grasp a tennis ball (with fingertips) and place it in a basket on the floor (right)
30	14	T-M	Reach and grasp a cap (2 fingers + thumb) of a bottle (held by left hand), unscrew it and place it overhead on a shelf

method could also enable a strong reduction of the computational costs w.r.t. state of the art algorithms, since the optimization space is constrained to a reduced set of parameters (i.e. the principal function weights) instead of the whole evolution of the robot pose. Future work will focus on more efficient implementation of the proposed algorithm.

Note that the method is specifically tailored on systems having the same kinematic structure of the human upper limb i.e the kinematic model used to describe human upper limb. The major limitation of this specific implementation is that the direct generalization to other kinematics is not straightforward and would require to solve a mapping problem. A solution could be to select

a mapping policy as done in [46]. An approach based on Impedance Control [47] could also be used to define a framework for which the trajectories generated with our approach can be used as a reference for the controller of a generic anthropomorphic manipulator (see [48].)

Conclusions

The methodology presented in this paper is designed to easily and uniquely generate human-like trajectories. We do believe that this work may have a major impact for the control of active devices during the interaction with humans. This is particularly relevant for the applications in which the robot is used as an instrument for human assistance or rehabilitation. Indeed, human-like robotic platforms that interact with humans, for example for self-feeding or elderly care, could increase their effectiveness, relying on the acceptability and predictability of the generated movements. The potential translational impact of our method in the field of companion robots could be also extremely significant. Indeed, the implementation of human-inspired point-to-point trajectories fully meets one of the three main envisioned requirements that companion robots are expected to fulfil in the future, which is related to the need for smooth robotic motion patterns (see [49] for further details).

Applications for rehabilitation devices are finally envisioned. Indeed, our strategy could be used to suggest preferred movements for the rehabilitation exoskeletons, in conjunction with an analogous modulation of the robot impedance. In our future work we will investigate the extension of this approach to the modulation of the impedance parameters. We believe that this could provide more therapeutic opportunities and enable novel approaches in rehabilitation.

List of actions

Table 5 lists the 30 actions considered in the protocol of this study. For each row, the first element counts the task number, the second links to the grasp taxonomy propose by Cutkosky in [50], the third indicates the class of movement, and the fourth reports a brief description of the task. The considered actions are also pictured in Fig. 1. These movements are clustered in three classes, depending on the interaction with the external environment: intransitive, which collects gestures and actions that do not require the use of an object; transitive, in which an object is used to complete the task; tool-mediated, in which the actions require an object to interact with another object.

Abbreviations
ADLs: Activities of daily living; fPCA: Functional principal component analysis; fPCs: Functional principal components; HL: Human-likeliness; HRI: Human-robot interaction; FK: Forward kinematics; OA: Obstacle avoidance

Author details
[1] Research Center "Enrico Piaggio", University of Pisa, Largo Lucio Lazzarino 1, 56126 Pisa, Italy. [2] Soft Robotics for Human Cooperation and Rehabilitation, Fondazione Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy. [3] Dipartimento di Ingegneria dell'Informazione, University of Pisa, Via G. Caruso, 16, 56122 Pisa, Italy. [4] Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 32 Vassar st, 02139 Cambridge, MA, USA.

References
1. Bartneck C, Kulić D, Croft E, Zoghbi S. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. Int J Soc Robot. 2009;1(1):71–81.
2. Fink J. Anthropomorphism and human likeness in the design of robots and human-robot interaction. In: International Conference on Social Robotics. Springer; 2012. p. 199–208. https://doi.org/10.1007/978-3-642-34103-8_20.
3. von Zitzewitz J, Boesch PM, Wolf P, Riener R. Quantifying the human likeness of a humanoid robot. Int J Soc Robot. 2013;5(2):263–76.
4. Morasso P. Spatial control of arm movements. Exp Brain Res. 1981;42(2): 223–7.
5. Flash T, Hogan N. The coordination of arm movements: an experimentally confirmed mathematical model. J Neurosci. 1985;5(7):1688–703.
6. Lo AC, Guarino PD, Richards LG, Haselkorn JK, Wittenberg GF, Federman DG, Ringer RJ, Wagner TH, Krebs HI, Volpe BT, et al. Robot-assisted therapy for long-term upper-limb impairment after stroke. N Engl J Med. 2010;362(19):1772–83.
7. Frisoli A, Salsedo F, Bergamasco M, Rossi B, Carbonini MC. A force-feedback exoskeleton for upper-limb rehabilitation in virtual reality. Appl Bionics Biomech. 2009;6(2):115–26.
8. Klamroth-Marganska V, Blanco J, Campen K, Curt A, Dietz V, Ettlin T, Felder M, Fellinghauer B, Guidali M, Kollmar A, et al. Three-dimensional, task-specific robot therapy of the arm after stroke: a multicentre, parallel-group randomised trial. Lancet Neurol. 2014;13(2):159–66.
9. Huber M, Rabin B, Docan C, Burdea GC, AbdelBaky M, Golomb MR. Feasibility of modified remotely monitored in-home gaming technology

for improving hand function in adolescents with cerebral palsy. IEEE Trans Inf Technol Biomed. 2010;14(2):526–34.

10. Maciejasz P, Eschweiler J, Gerlach-Hahn K, Jansen-Troy A, Leonhardt S. A survey on robotic devices for upper limb rehabilitation. J Neuroeng Rehabil. 2014;11(1):3.

11. Lo HS, Xie SQ. Exoskeleton robots for upper-limb rehabilitation: State of the art and future prospects. Med Eng Phys. 2012;34(3):261–8.

12. Merians AS, Fluet G, Tunik E, Qiu Q, Saleh S, Adamovich S. Movement rehabilitation in virtual reality from then to now: how are we doing?. Int J Disabil Hum Dev. 2014;13(3):311–7.

13. Crocher V, Sahbani A, Robertson J, Roby-Brami A, Morel G. Constraining upper limb synergies of hemiparetic patients using a robotic exoskeleton in the perspective of neuro-rehabilitation. IEEE Trans Neural Syst Rehabil Eng. 2012;20(3):247–57.

14. Krebs HI, Hogan N. Am J Phys Med Rehabil/Assoc Acad Physiatrists. 2012;91(11 0 3):290.

15. Riek LD, Rabinowitch T-C, Chakrabarti B, Robinson P. How anthropomorphism affects empathy toward robots. In: Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction. ACM; 2009. p. 245–6. https://doi.org/10.1145/1514095.1514158.

16. Edsinger A, Kemp CC. 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. Springer; 2007. p. 345–55. https://doi.org/10.1007/978-3-540-76729-9_27.

17. Mukai T, Hirano S, Nakashima H, Sakaida Y, Guo S. Realization and safety measures of patient transfer by nursing-care assistant robot riba with tactile sensors. J Robot Mech. 2011;23(3):360–9.

18. Duffy BR. Anthropomorphism and the social robot. Robot Auton Syst. 2003;42(3):177–90.

19. Elbanhawi M, Simic M. Sampling-based robot motion planning: A review. IEEE Access. 2014;2:56–77.

20. Kuffner JJ, LaValle SM. Rrt-connect: An efficient approach to single-query path planning. In: Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference On. IEEE; 2000. p. 995–1001. https://doi.org/10.1109/robot.2000.844730.

21. Karaman S, Frazzoli E. Sampling-based algorithms for optimal motion planning. Int J Robot Res. 2011;30(7):846–94.

22. Stilman M. Global manipulation planning in robot joint space with task constraints. IEEE Trans Robot. 2010;26(3):576–84.

23. García N, Suárez R, Rosell J. Hg-rrt*: Human-guided optimal random trees for motion planning. In: Emerging Technologies & Factory Automation (ETFA), 2015 IEEE 20th Conference On. IEEE; 2015. p. 1–7. https://doi.org/10.1109/robot.2007.363983.

24. Schulman J, Ho J, Lee AX, Awwal I, Bradlow H, Abbeel P. Finding locally optimal, collision-free trajectories with sequential convex optimization. In: Robotics: Science and Systems. Citeseer; 2013. p. 1–10. https://doi.org/10.15607/rss.2013.ix.031.

25. Zucker M, Ratliff N, Dragan AD, Pivtoraiko M, Klingensmith M, Dellin CM, Bagnell JA, Srinivasa SS. Chomp: Covariant hamiltonian optimization for motion planning. Int J Robot Res. 2013;32(9–10):1164–93.

26. Piazzi A, Visioli A. Global minimum-jerk trajectory planning of robot manipulators. IEEE Trans Ind Electron. 2000;47(1):140–9.

27. Arimoto S, Sekimoto M, Hashiguchi H, Ozawa R. Natural resolution of ill-posedness of inverse kinematics for redundant robots: A challenge to bernstein's degrees-of-freedom problem. Adv Robot. 2005;19(4):401–34.

28. Mordatch I, Wang JM, Todorov E, Koltun V. Animating human lower limbs using contact-invariant optimization. ACM Trans Graph (TOG). 2013;32(6):203.

29. Mordatch I, Lowrey K, Andrew G, Popovic Z, Todorov EV. Interactive control of diverse complex characters with neural networks. In: Advances in Neural Information Processing Systems; 2015. p. 3132–40.

30. Peng XB, Berseth G, Van de Panne M. Terrain-adaptive locomotion skills using deep reinforcement learning. ACM Trans Graph (TOG). 2016;35(4):81.

31. Zhang Z, Li Z, Zhang Y, Luo Y, Li Y. Neural-dynamic-method-based dual-arm cmg scheme with time-varying constraints applied to humanoid robots. IEEE Trans Neural Netw Learn Syst. 2015;26(12):3251–62.

32. Figueroa Fernandez NB, Billard A. Modeling compositions of impedance-based primitives via dynamical systems. In: In Proceedings of the Workshop on Cognitive Whole-Body Control for Compliant Robot Manipulation (COWB-COMP); 2018.

33. Breteler MDK, Gielen SC, Meulenbroek RG. End-point constraints in aiming movements: effects of approach angle and speed. Biol Cybern. 2001;85(1):65–75.

34. Miossec S, Kheddar A. Human motion in cooperative tasks: Moving object case study. In: Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference On. IEEE; 2009. p. 1509–14. https://doi.org/10.1109/robio.2009.4913224.

35. Santello M, Flanders M, Soechting JF. Postural hand synergies for tool use. J Neurosci. 1998;18(23):10105–15.

36. Ciocarlie MT, Allen PK. Hand posture subspaces for dexterous robotic grasping. Int J Robot Res. 2009;28(7):851–67.

37. Ficuciello F. Synergy-based control of underactuated anthropomorphic hands. IEEE Trans Ind Inform. 2018. https://doi.org/10.1109/tii.2018.2841043.

38. Santello M, Bianchi M, Gabiccini M, Ricciardi E, Salvietti G, Prattichizzo D, Ernst M, Moscatelli A, Jörntell H, Kappers AM, et al. Hand synergies: integration of robotics and neuroscience for understanding the control of biological and artificial hands. Phys life Rev. 2016. https://doi.org/10.1016/j.plrev.2016.02.001.

39. Averta G, Della Santina C, Battaglia E, Felici F, Bianchi M, Bicchi A. Unvealing the principal modes of human upper limb movements through functional analysis. Front Robot AI. 2017;4:37.

40. Averta G, Angelini F, Bonilla M, Bianchi M, Bicchi A. Incrementality and hierarchies in the enrollment of multiple synergies for grasp planning. IEEE Robot Autom Lett. 2018;3(3):2686–93.

41. Flash T. The control of hand equilibrium trajectories in multi-joint arm movements. Biol Cybern. 1987;57(4–5):257–74.

42. Gabiccini M, Stillfried G, Marino H, Bianchi M. A data-driven kinematic model of the human hand with soft-tissue artifact compensation mechanism for grasp synergy analysis. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE; 2013. p. 3738–45. https://doi.org/10.1109/iros.2013.6696890.

43. Welch G, Bishop G, et al. An introduction to the kalman filter. Technical Report. Chapel Hill: University of North Carolina; 1995.

44. Reams R. Hadamard inverses, square roots and products of almost semidefinite matrices. Linear Algebra Appl. 1999;288:35–43.

45. Hwang YK, Ahuja N. A potential field approach to path planning. IEEE Trans Robot Autom. 1992;8(1):23–32.

46. Gioioso G, Salvietti G, Malvezzi M, Prattichizzo D. Mapping synergies from human to robotic hands with dissimilar kinematics: an approach in the object domain. IEEE Trans Robot. 2013;29(4):825–37.

47. Hogan N. Impedance control: An approach to manipulation: Part i—theory; 1985. https://doi.org/10.1115/1.3140702.

48. Averta G, Caporale D, Della Santina C, Bicchi A, Bianchi M. A technical framework for human-like motion generation with autonomous anthropomorphic redundant manipulators. In: Robotics and Automation (ICRA), 2020 IEEE International Conference On. IEEE; 2020.

49. Dario P, Verschure PF, Prescott T, Cheng G, Sandini G, Cingolani R, Dillmann R, Floreano D, Leroux C, MacNeil S, et al. Robot companions for citizens. Procedia Comput Sci. 2011;7:47–51.

50. Cutkosky MR. On grasp choice, grasp models, and the design of hands for manufacturing tasks. IEEE Trans Robot Autom. 1989;5(3):269–79.

Publisher's Note