Generalization of Human Grasping for Multi-Fingered Robot Hands

Heni Ben Amor, Oliver Kroemer, Ulrich Hillenbrand, Gerhard Neumann, and Jan Peters

Abstract— **Multi-fingered robot grasping is a challenging problem that is difficult to tackle using hand-coded programs. In this paper we present an imitation learning approach for learning and generalizing grasping skills based on human demonstrations. To this end, we split the task of synthesizing a grasping motion into three parts: (1) learning efficient grasp representations from human demonstrations, (2) warping contact points onto new objects, and (3) optimizing and executing the reach-and-grasp movements. We learn low-dimensional latent grasp spaces for different grasp types, which form the basis for a novel extension to dynamic motor primitives. These latent-space dynamic motor primitives are used to synthesize entire reach-and-grasp movements. We evaluated our method on a real humanoid robot. The results of the experiment demonstrate the robustness and versatility of our approach.**

I. Introduction

The ability to grasp is a fundamental motor skill for humans and a prerequisite for performing a wide range of object manipulations. Therefore, grasping is also a fundamental requirement for robot assistants, if they are to perform meaningful tasks in human environments. Although there have been many advances in robot grasping, determining how to perform grasps on novel objects using multi-fingered hands still remains an open and challenging problem.

A lot of research has been conducted on robot grippers with few degrees of freedom (DoF) which may not be particularly versatile. However, the number of robot hands developed with multiple fingers has been steadily increasing in recent years. This progress comes at the cost of a much higher dimensionality of the control problem and, therefore, more challenges for movement generation. Hard coded grasping strategies will typically result in unreliable robot controllers that can not sufficiently adapt to changes in the environment, such as the object's shape or pose. Such hard coded strategies will also often lead to unnatural 'robotic looking' grasps, that do not account for the increased sophistication of the hardware. Alternative approaches, such as the optimization of grasps using stochastic optimization techniques, are computationally expensive and require the specification of a grasp quality metric [27]. Defining an adequate grasp metric is often hard to do, as it requires specifying intuitive concepts in a mathematical form. Additionally, such approaches typically do not consider the whole reach-and-grasp movement but exclusively concentrate on the hand configuration at the goal.

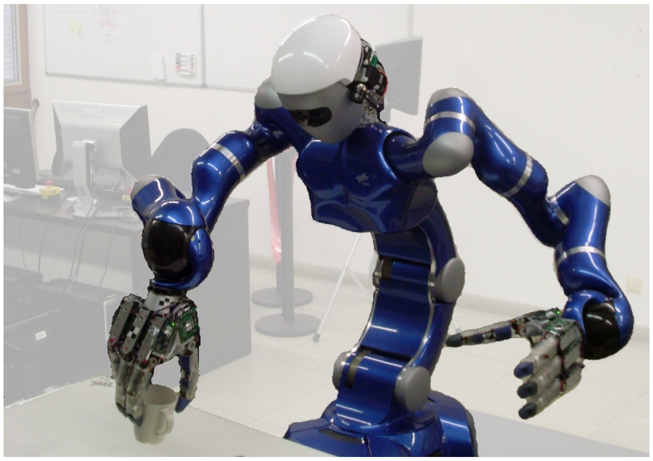

Fig. 1. The Justin robot learns to grasp and lift-up a mug by imitation. The reach-and-grasp movement is learned from human demonstrations. Latent-space dynamic motor primitives generalize the learned movement to new situations.

In this paper, we present an imitation learning approach for grasp synthesis. Imitation learning allows a human to easily program a humanoid robot [3], and also to transfer implicit knowledge to the robot. Instead of programming elaborate grasping strategies, we use machine learning techniques to successfully synthesize new grasps from human demonstration. The benefits of this approach are threefold. First, the computational complexity of the task is significantly reduced by using the human demonstrations along with compact low-dimensional representations thereof. Second, the approach allows us to imitate human behavior throughout the entire reach-and-grasp movement, resulting in seamless, natural-looking motions. Typical transitions between a discrete set of hand shapes, as can be found in traditional approaches, are thus avoided. Finally, this approach also allows the user to have control over the type of grasp that is executed. By providing demonstrations of only one particular grasp type, the synthesis algorithm can be used to generate distinct grasps e.g., only lateral, surrounding, or tripod grasp. The use of assorted grasps can considerably improve the robustness of the grasping strategy as the robot can choose a grasp type which is appropriate for the current task.

A. Related Work

In order to generalize human grasping movements, we need to understand how humans perform grasps. Human grasping motions consist of two components: the reaching motion of the arm for transporting the hand, and the motions

Heni Ben Amor, Oliver Kroemer, Gerhard Neumann and Jan Peters are with the Technische Universitaet Darmstadt, Intelligent Autonomous Systems, Darmstadt, Germany. {amor, kroemer, neumann, peters}@ias.tu-darmstadt.de

Ulrich Hillenbrand is with the German Aerospace Center - DLR, Institute of Robotics and Mechatronics, Oberpfaffenhofen, Germany. Ulrich.Hillenbrand@dlr.de

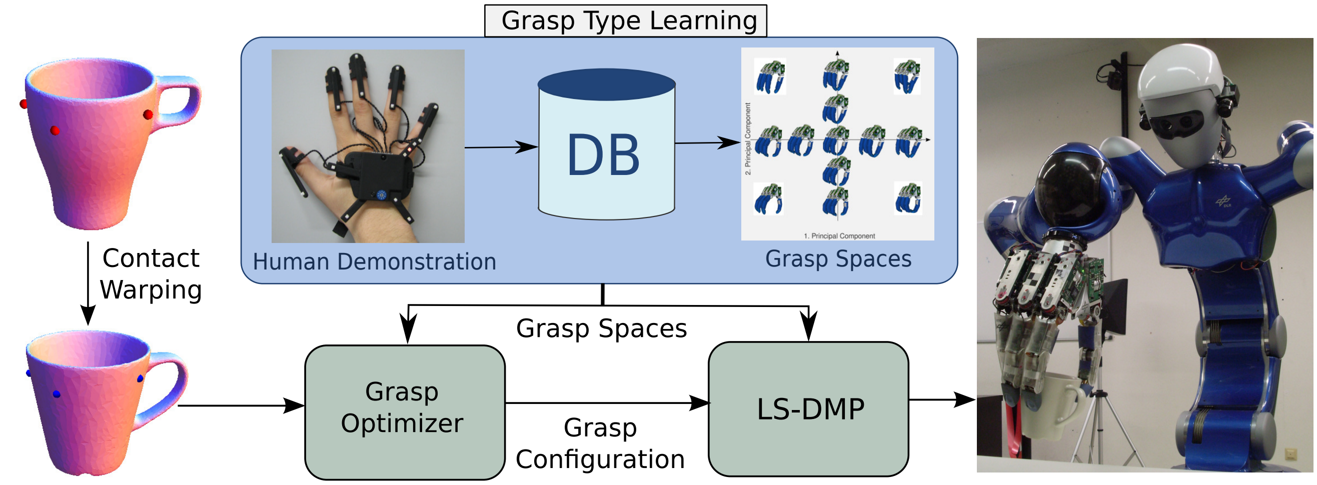

Fig. 2. An overview of the proposed approach. The contact points of a known object are warped on the current object. Using the resulting positions, an optimizer finds the ideal configuration of the hand during the grasp. The optimizer uses low-dimensional grasp spaces learned from human demonstrations. Finally, a latent space dynamic motor primitive robustly executes the optimized reach-and-grasp motion. The approach is data-driven and can be used to train and execute different types grasps.

of the fingers for shaping the hand [16], [17]. These two components are synchronized during the grasping movement [7]. For example, at around 75% of the movement duration, the hand reaches its preshape posture and the fingers begin to close [15]. At this point in time, the reaching motion of the hand shifts into a low velocity movement phase.

Early studies into human hand control assumed muscles and joints as being controlled individually by the central nervous system [26], [19]. However, more recent studies have found evidence suggesting that the fingers are controlled using hand synergies [23], [2] — i.e., the controlled movements of the fingers are synchronized.

According to this view, fingers are moved “synergistically” thereby reducing the number of DoF needed for controlling the hand. Such hand synergies can be modeled as projections of the hand configuration space into lower-dimensional subspaces [20] such as the principal components. Movements along the first principal component of this subspace result in a basic opening and closing behavior of the hand. The second and higher-order principal components refine this motion and allow for more precise shaping of the hand [20], [24], see Fig. 3. Although the majority of the variation in the finger configurations is within the first two principal components, higher-order principal components also contain important information for accurately executing grasps [23]. The gain in grasp accuracy does, however, plateau at around five dimensions [22], [20]. Therefore, the space of human hand synergies during grasping can be well represented by a five-dimensional subspace.

Following this idea, various researchers have used dimensionality reduction techniques to find finger synergies in recorded human grasps [4], [8]. Once a low-dimensional representation of finger synergies is found, it can be used to synthesize new grasps in a generate-and-test fashion. For example, the authors of [8] use Simulated Annealing to find an optimal grasp on a new object while taking into account the finger synergies. Common to such approaches is the use of a grasp metric [27] that estimates the quality of a potential solution candidate. However, such metrics can be computationally demanding and rely on having an accurate model of the objects. In general, it is difficult to define a grasp metric that includes both, physical aspects of the grasps (such as the stability) as well as functional aspects that depend upon the following manipulations.

Alternative approaches to grasp synthesis predict the success probability of grasps for different parts of the object. For example, good grasping regions are estimated from recorded 2D images of the object in [25]. A labeled training set of objects including the grasping region is subsequently produced by using a ray-tracing algorithm. The resulting dataset is then used to train a probabilistic model of the ideal grasping region. The learned model, in turn, allows a robot to automatically identify suitable grasping regions based on visual features. In a similar vein, Boularias et al. [6] use a combination of local features and Markov Random Fields to infer good grasping regions from recorded point clouds. Given an inferred grasping region, the reach-and-grasp motion still needs to be generated using a set of heuristics. Additionally, this approach does not address the problems of how to shape the hand and where to place the finger contacts.

Tegin et. al. [28] also used imitation learning from human demonstration to extract different grasp types. However, they do not model the whole reach-and-grasp movement and circumvent the high-dimensionality problem by using simpler manipulators.

II. Our Approach

In our approach, we address the challenges of robot grasping by decomposing the task into three different stages: (1)

learning efficient grasp representations from human demonstrations, (2) warping contact points onto new objects, and (3) optimizing and executing the synchronized reach-and-grasp movements.

An overview of the proposed approach can be seen in Fig. 2. The contact points of a known object are first warped onto the current object using the techniques in Sec. II-B. The warped contact points are then used by the optimizer to identify all parameters needed for executing the grasp, i.e., the configuration of the fingers and the position and orientation of the hand. The optimization is performed in low-dimensional grasp spaces which are learned from human demonstrations. Finally, the reach-and-grasp movement is executed using a novel extension to dynamic motor primitive [14] called latent-space dynamical systems motor primitive (LS-DMP).

A. *Learning Grasp Types from Human Demonstration*

Using human demonstrations as reference when synthesizing robot grasps can help to narrow down the set of solutions and increase the visual appeal of the generated grasp. At the same time, a discrete set of example grasps can also heavily limit the power of such an approach. To overcome this problem, we use dimensionality reduction techniques on the set of human demonstrations in order to infer the low-dimensional grasp space. To this end, we recorded the movements of nine test-subjects, where each test subject was asked to perform reach-and-grasp actions on a set of provided objects. We subsequently performed Principal Component Analysis (PCA) on the dataset, projecting it onto five principal components. This choice of dimensionality is based on research on the physiology of the human hand [22], [20] which suggested that five principle components are sufficient for accurately modeling the movements of the human hand.

The resulting *grasp space* is a compact representation of the recorded grasps as it models the synergies between the different fingers and finger segments. The first principal component, for example, encodes the opening and closing of the hand. Fig. 3 shows grasps from the space spanned by the first two principal components.

The above approach yields general grasp spaces that do not give the user control over the grasp type to be executed by the robot. However, for many tasks it is important to favor a particular grasp type over another when synthesizing the robot movements. For example, for carrying a pen one can use a tip grasp, while for writing with the pen an extension grasp is better suited. Hence, in a second experiment with the same test subjects we learned grasp spaces for specific grasp types, such as lateral grasps or tripod grasps. To determine the grasp space, we devised a grasp taxonomy [10] consisting of twelve grasp types and recorded specific datasets for each of these types. The datasets were subsequently used to learn grasp spaces for the specific grasp type.

Due to the differences in kinematics of the human and robot hand, there are multiple ways to map the hand state to the robot state, also known as the *correspondence problem* in the robotics literature [9]. In this paper, we solve the correspondence problem by dividing the generalization of grasps into two parts, i.e., the reproduction of the hand shape and the adaptation of Cartesian contact points. The reproduction of the hand shape is realized by directly mapping the human joint angles to the robot hand. For the index, middle and ring fingers this results in an accurate mapping with robot hand configurations similar to the demonstrated human hand shapes. In order to map the thumb, an additional offset needed to be added to the carpometacarpal joint. Using this type of mapping, the reproduced hand shapes will be similar to those of the human. The generalization of the Cartesian contact points is achieved by the contact warping algorithm described in Sec. II-B. The two generalizations in Cartesian space and in joint space are then reconciled through the optimization process explained in Sec. II-D.

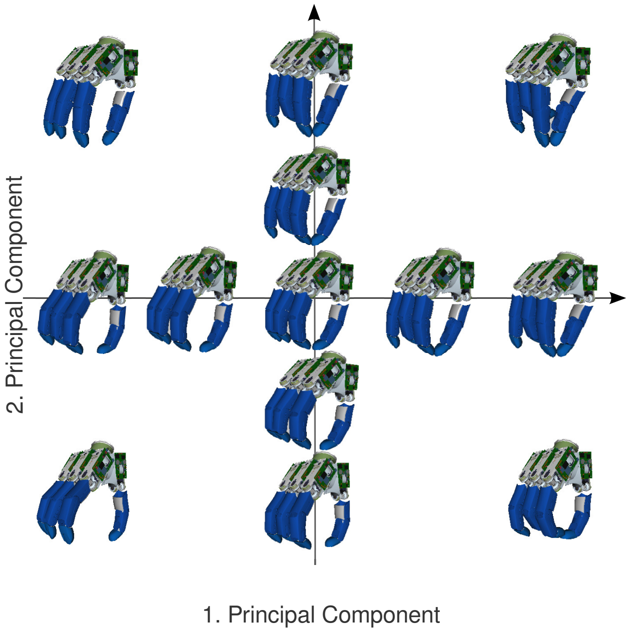

Fig. 3. The space spanned by the first two principal components of human recorded grasps applied to the robot hand. The first component describes the opening and closing of the hand. The second principal component modulates the shape of the grasp.

B. *Generalizing Grasps through Contact Warping*

In this section, we introduce the contact warping algorithm. This algorithm allows the robot to adapt given contact points from a known object to a novel object. As a result, we can generalize demonstrated contact points to new situations. Assume that we are given two 3D shapes from the same semantic/functional category through dense sets of range data points. In our approach, the process of shape warping, that is, computing a mapping from the source shape to the target shape, has been broken down into three steps.

1) Rigid alignment of source and target shapes, such that semantically/functionally corresponding points get close to each other.

2) Assignment of correspondences between points from the source shape and points on the target shape.
3) Interpolation of correspondences to a continuous (but possibly non-smooth) mapping.

The alignment step involves sampling and aligning many surflet pairs, i.e., pairs of surface points and their local normals, from source and target shapes. The estimation of relative clusters of the pose parameters is obtained from the surflet-pair alignments [11], [12].

Since the alignment of source and target shapes has brought corresponding parts close to each other, we can again rely on the local surface description by surflets to find correspondences, based on proximity of points and alignment of normal vectors. The correspondence assignment that we have used here is an improved version of the method described in [11]. In this approach, correspondences were assigned for each source surflet independently into the set of target surflets. For strong shape variations or unfavorable alignment between source and target, such an approach could result in a confusion of similar parts.

In order to cope with larger shape variation, some interaction between assignments of neighboring points has to be introduced. We have, therefore, formulated correspondence search as an optimal assignment problem. In this formulation, interaction between assignments of different points is enforced through uniqueness constraints.

Let $\{x_1, \ldots, x_N\}$ be points from the source shape, transformed to align with the target shape; let $\{y_1, \ldots, y_N\}$ be points from the target shape.[1] Assignment of source point i to target point j is expressed as an assignment matrix,

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ is assigned to } j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Furthermore, let $d_{ij} = \|x_i - y_j\|$ be the Euclidean distances between source and target points and $c_{ij} = n_i \cdot m_j$ be the angle cosines between the unit normal vectors n_i and m_j at source point i and target point j, respectively.

The objective is to minimize the sum of distances between correspondences, i.e., mutually assigned points,

$$D(a_{11}, \ldots, a_{1N}, a_{21}, \ldots, a_{NN}) = \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} a_{ij} \ , \tag{2}$$

subject to the constraints

$$\sum_{i=1}^{N} a_{ij} = 1 \quad \forall j \in \{1, \ldots, N\} \ , \tag{3}$$

i.e., to assign every *target point* to exactly one source point,

$$\sum_{j=1}^{N} a_{ij} = 1 \quad \forall i \in \{1, \ldots, N\} \ , \tag{4}$$

i.e., to assign every *source point* to exactly one target point, and

$$c_{ij} a_{ij} \geq 0 \quad \forall i, j \in \{1, \ldots, N\} \ , \tag{5}$$

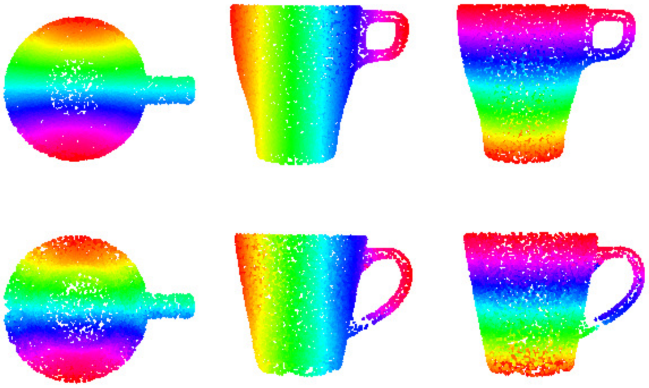

Fig. 4. Mug warping example. A dense set of surface points from the source mug (top row) and their mappings to the target mug (bottom row) are colored to code their three Cartesian source coordinates (three columns).

i.e., to assign only between points with inter-normal angle of $\leq$ 90 degrees. The two equality constraints (3) and (4) mediate the desired interaction between assignments of different points. The inequality constraint (5) can exclude points from being assigned and, therefore, the problem may become infeasible. Thus, we have to add imaginary source and target points x_0 and y_0 which have no position and no normal direction. They can be accommodated by appending large entries d_{0j} and d_{i0} to the distance matrix, which larger than all real distances in the data set, as well as zero entries $c_{0j} = c_{i0} = 0$ to the angle cosine matrix. These imaginary points can be assigned to all real points with a penalty, which is chosen such that only points without a compatible partner will receive this imaginary assignment. We subsequently minimize the cost function

$$\begin{aligned} & C(a_{01}, \ldots, a_{0N}, a_{10}, \ldots, a_{NN}) \\ &= D(a_{11}, \ldots, a_{1N}, a_{21}, \ldots, a_{NN}) \\ &\quad + \sum_{i=1}^{N} d_{i0} a_{i0} + \sum_{j=1}^{N} d_{0j} a_{0j} \ . \end{aligned} \tag{6}$$

For solving this constrained optimization problem, we use the interior-point algorithm, which is guaranteed to find an optimal solution in polynomial time [30].

Finally, point correspondences are interpolated to obtain a continuous (but possibly non-smooth) mapping of points from the source domain to the target domain. More theory and systematic evaluations of the procedure are given in [13].

Fig. 4 shows an example of a dense set of surface points warped between two mugs. A warp of the contact points of an actual grasp from the source to the target mug is shown on the left of Fig. 2.

C. Latent Space Dynamic Motor Primitives

In order to execute different grasps, the robot requires a suitable representation of the grasping actions. Ideally, the grasping action should be straightforward to learn from

[1] An equal number N of points from source and target shapes can always be re-sampled from the original data sets.

a couple of human demonstrations and easily adapted to various objects and changes in the object locations. The action representation should also ensure that the components of the grasping movement are synchronized. The dynamical systems motor primitives (DMPs) representation fulfills all of the above requirements [14]. DMPs have been widely adopted in the robotics community, and are well-known for their use in imitation learning [21], [18]. The DMP framework represents the movements of the robot as a set of dynamical systems

$$\ddot{y} = \alpha_z(\beta_z \tau^{-2}(g - y) - \tau^{-1}\dot{y}) + a\tau^{-2} f(x, \theta_{1:N})$$

where y is a state variable, g is the corresponding goal state, and τ is a time scale. The first set of terms represents a critically-damped linear system with constant coefficients α_z and β_z. The last term, with amplitude coefficient $a = g - y_0$, incorporates a shaping function

$$f(x, \theta_{1:N}) = \frac{\sum_{i=1}^{N} \psi_i(x)\theta_i x}{\sum_{j=1}^{N} \psi_j(x)},$$

where $\psi_i(x)$ are Gaussian basis functions, and the weight parameters $\theta_{1:N}$ define the general shape of the movements. The weight parameters $\theta_{1:N}$ are straightforward to learn from a single human demonstration of a goal directed movement. The variable x is the state of a canonical system shared by all DoFs. The canonical system acts as a timer to synchronize the different movement components. It has the form $\dot{x} = -\tau x$, where $x_0 = 1$ at the beginning of the motion and thereafter decays towards zero. The meta-parameters g, a, and τ can be used to generalize the learned DMP to new situations. For example, the goal state g of the reaching movement is defined by the position of the object and the desired grasp. We explain how the DMP goal meta-parameters are computed for new objects in Sec. II-D. However, we need first to define how the finger trajectories can be encoded as DMPs, such that they generalize to new situations in a human-like manner.

Representing and generalizing the motions of the fingers is a challenging task due to the high dimensionality of the finger-configuration space. A naive solution would be to assign one DMP to each joint [19]. However, as previously discussed in Sec. I-A, humans seem to generalize their movement trajectories within lower-dimensional spaces of the finger configurations, and not at the level of each joint independently [23], [20]. If the robot's generalization of the grasping action does not resemble the human's execution, implicit information contained within the human demonstrations is lost. Therefore, in order to facilitate behavioral cloning of human movements, the DMPs for multi-fingered hands should be realized in a lower dimensional space. In addition, overfitting is avoided by representing the movement in a lower-dimensional space.

In particular, the DMPs can be defined in the latent spaces learned in Sec. II-A. As such spaces are learned from complete trajectories of the grasping movements, they also include the finger configurations needed for representing the hand during the approach and preshaping phases of the action, as well as the final grasps [20]. We use a DMP for each of the latent space dimensions as well as DMPs for the wrist position and orientation. The weight parameters for these DMPs can be learned from human demonstrations by first projecting the tracked motions into the latent space and subsequently learning the weights from the resulting trajectory. Thus, the same data that is used to learn the latent space can be reused for learning the weight parameters. The resulting latent-space DMPs, as well as the reaching movement's DMPs, are linked to the same canonical system, thus, ensuring that they remain synchronized. The output of the latent-space DMPs is afterwards mapped back into the high-dimensional joint space by the PCA projection. In this manner, the grasping action can be executed seamlessly, and the robot can begin closing its fingers before the hand has reached its final position.

Thus, we have defined a human-like representation of the grasping movements that can be acquired by imitation learning. Given this DMP representation, the robot still needs to determine the meta-parameters for new situations. This process is described in the next section.

D. Estimating the Goal Parameters

In order to generalize the latent-space DMPs to new objects, we need to estimate the goal state g for each latent-space dimension, as well as the orientation of the hand for a new set of contact points which we have acquired from contact warping as discussed in Sec. II-B. We use one contact-point per finger, where the contact point is always located at the finger tip. Each point is specified in Cartesian coordinates. As we have four fingers, this results into a 12-dimensional task space vector $\mathbf{x}_C$. Additionally, we also want to estimate the position and orientation of the hand in the world coordinate frame. We therefore add six virtual joints $\mathbf{v}$, i.e., three translational and three rotational joints. We will denote the transformation matrix, which is defined by these six virtual joints, as $\mathbf{T}(\mathbf{v})$. We define the finger tip position vector $\mathbf{x}_{1:4}$ as the concatenation of all four finger tip positions. This vector is a function of the transformation matrix $\mathbf{T}(\mathbf{v})$ and the joint configurations of the fingers $\mathbf{q} = \mathbf{m} + \mathbf{K}\mathbf{g}$, i.e.,

$$\boldsymbol{\phi}_W(\mathbf{y}) = \mathbf{T}(\mathbf{v})\boldsymbol{\phi}_H(\mathbf{m} + \mathbf{K}\mathbf{g}).$$

The vector $\mathbf{m}$ represents the mean of the PCA transformation and $\mathbf{K}$ is given by the first five eigenvectors. The function $\boldsymbol{\phi}_H(\mathbf{q})$ calculates the finger tip-positions in the local hand coordinate frame. This setup is an inverse kinematics problem with the difference that we want to optimize the joint positions of the fingers in the latent space instead of directly optimizing the joint positions $\mathbf{q}$. Thus, the inverse kinematics problem is over-constrained as we have twelve task variables and only eleven degrees of freedom. Therefore, instead of the standard Jacobian pseudo-inverse solution, we need to employ a different approach.

Our task is to estimate the optimal configuration $\mathbf{y}^* = [\mathbf{v}^*, \mathbf{g}^*]$ of the hand, which consists of the orientation and the latent space coordinates, such that the squared distance between finger-tip positions $\mathbf{x}_{1:4}$ and the contact points $\mathbf{x}_C$ is minimal, i.e.,

$$\begin{aligned}\mathbf{y}^* &= [\mathbf{v}^*, \mathbf{g}^*] = \text{argmax}_{\mathbf{y}} L(\mathbf{y}), \\ L(\mathbf{y}) &= -(\boldsymbol{\phi}_W(\mathbf{y}) - \mathbf{x}_C)^T \mathbf{C}^{-1} (\boldsymbol{\phi}_W(\mathbf{y}) - \mathbf{x}_C) \\ &\quad + \mathbf{y}^T \mathbf{W} \mathbf{y}.\end{aligned} \quad (7)$$

The matrix $\mathbf{W} = \text{diag}(\mathbf{w})$ defines a damping or regularization term for the step-size of $\mathbf{y}$, and $\mathbf{C} = \text{diag}(\mathbf{c})$ defines the inverse precision for each task variable. The Jacobian $\mathbf{J} = \partial \boldsymbol{\phi}_W / \partial \mathbf{y}$ for this problem can be obtained straightforwardly, i.e., for the derivation w.r.t $\mathbf{v}$ it is given by the standard geometric Jacobian and for the derivation w.r.t the latent variable $\mathbf{g}$ it is given by $\mathbf{J}_l = \mathbf{J}_q \mathbf{K}$, where $\mathbf{J}_q$ denotes the geometric Jacobian.

We will solve the optimization problem given in Equation (7) by iteratively applying a least squares solution. Given the current hand configuration $\mathbf{y}_k$ and the desired finger-tip positions $\mathbf{x}_C$, the update step for the hand configuration is therefore given by

$$\Delta \mathbf{y}_k = (\mathbf{J}^T \mathbf{C} \mathbf{J} + \boldsymbol{W})^{-1} \mathbf{J}^T \mathbf{C} \left(\mathbf{x}_C - \boldsymbol{\phi}_W(\mathbf{y}_k) \right). \quad (8)$$

As we have to solve an *over*constrained inverse kinematics setting, in contrast to the more common *under*constrained inverse kinematics setting, we use the *left-pseudo inverse* in Equation (8). This update equation also corresponds to a Bayesian view on inverse kinematics [29]. We repeat the update until convergence in order to get the optimal hand configuration $\mathbf{y}^*$. We always start our optimization from an initial posture where the hand is pointing downwards.

III. Setup and Evaluations

To evaluate the proposed approach, we conducted a set of experiments using the *Rollin' Justin* robot platform [5]. Justin is a mobile humanoid robot system with an upper-body including 43 actuated degrees-of-freedom (DoF). In our experiments we controlled 22 DoF pertaining to the Torso (3 DoF), the right arm (7 DoF), and the four-fingered right hand (12 DoF). The experiments were performed both in simulation and on the physical robot.

A. Simulation Results

In the first experiment, we evaluated the performance and the results of our approach in a simulated environment for the Justin robot. As explained in Sec. II, we trained individual LS-DMPs for each of the principal components of the demonstrated reach-and-grasp movement. Fig. 5 shows the latent-space trajectories for three out of the five principal components of the hand shape. The example trajectories are depicted in blue, while the trajectory learned by the LS-DMP is depicted in red. This figure reveals an interesting insight into the nature of the recorded human reach-and-grasp movements: many of the example trajectories have a distinct sigmoid shape that has a bell-shaped velocity profile. This insight corresponds to the results in [1] , which showed that humans perform point-to-point reaching movements such that the velocity profile along the path can be characterized by a symmetric bell-shape. Our results indicate that a similar property holds for the latent space trajectories of the hand shape during a reach and grasp.

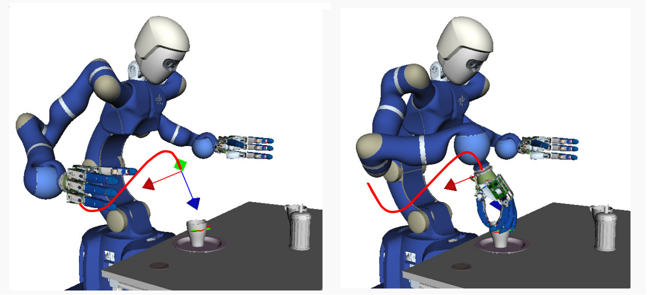

Fig. 6. The Justin robot executes a reach-and-grasp movement in simulation. Using the trained LS-DMP a new trajectory (red) to the target object is synthesized. The optimal hand position and orientation (shown as a coordinate system) is estimated along with the optimal hand shape in latent-space.

After learning, we first executed the LS-DMP in simulation. Fig. 6 shows the start and end configuration during one run of the algorithm. The red curve depicts the trajectory of the hand as generated by the LS-DMP, while the displayed coordinate system shows the estimated hand orientation of the robot. To evaluate the accuracy of the produced grasping motions, we repeatedly changed the position and orientation of the target object and measured the distance between the warped contact points on the object and the fingertip positions. Ideally, the fingertips should always coincide with the contact points. Tab. I shows the average distance of the fingers to the warped contact points after executing a reach-and-grasp movement. We also varied the grasp spaces in order to evaluate the effect of the grasp type on the the resulting hand shape. The grasp space indicated by *Multi* in Tab. I was learned using all available human demonstrations. This grasp space encompasses a wide range of variations of the human hand. As can be seen in the table, we achieved the most accurate results by using this grasp space. In this case, the average error is about 7mm.

It should be noted that the fingers of the robot are much larger than human fingers and have a width of about 3cm. Given the size of the robot's fingers, the produced error only corresponds to about a quarter of the finger width. The table clearly shows that changing the grasp type results in higher average error. This increased error is to be expected, as we constrained the space of possible solutions to a specific grasp type. At the same time, visual inspection of the resulting grasps shows that this error does not deteriorate the quality of the resulting grasps, as will be seen in the next section.

B. Real Robot Experiments

We also conducted experiments with the real *Rollin' Justin* robot. Three different types of mugs were used during the experiments. After placing a mug on a table in front of the robot, all information about the pose of the mug

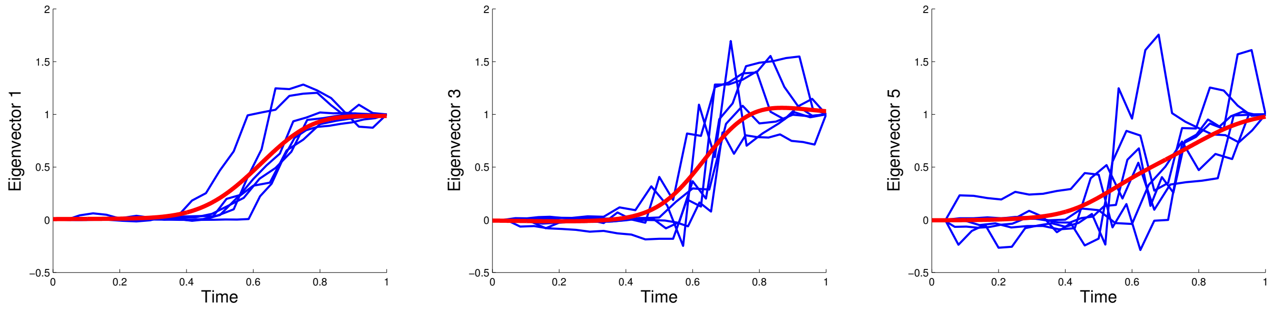

Fig. 5. The plots show example trajectories (blue), and the mean trajectory (red), for three (out of five) latent space dimensions during the closing of the hand. The trajectories have been shifted and scaled to start at zero and end at one, in order to allow for easier comparison of their shapes. As the scaled trajectories have similar shapes, they can be represented by individual DMPs and easily learned from human demonstrations.

TABLE I

AVERAGE DISTANCE BETWEEN WARPED CONTACT POINTS AND FINGERTIPS AFTER GRASPING.

Grasp Type	Multi	Tripod	Surrounding	Lateral
Avg. Error (m)	0.007	0.013	0.0157	0.014

was estimated using a Kinect camera and the techniques explained in [12]. Subsequently, using the contact warping techniques from Sec. II-B the contact points from a known mug were warped onto the currently seen mug. The resulting contact points were subsequently fed into the optimizer to estimate all parameters that are needed to execute the reach-and-grasp movement. The estimation of all parameters using the algorithm in Sec. II-D takes about one to five seconds. We performed about 20 repetitions of this experiment with the different mugs placed at various positions and heights. Additionally, we included a lifting-up motion to our movement, in order to evaluate whether the resulting grasp was stable or not. In all of the repetitions the robot was able to successfully grasp and lift-up the observed object.

Furthermore, we also executed the reach-and-grasp movements using grasp spaces belonging to different grasp types. No change was made to the structure or other parameters of the algorithm. The only difference between each execution run was the grasp space to be loaded. Fig. 7 shows three of the grasp types used in our taxonomy along with the result of applying them to the Justin robot. The figure clearly shows that changing the grasp type can have a significant effect on the appearance of the executed grasp. For example, we can see that the use of the *tripod*-grasp results in delicate grasps with little finger opposition, while surrounding grasps lead to more caging grasps with various finger oppositions. Our approach exploits the redundancy in hand configurations and allows desired grasp types to be set according to the requirements of the manipulation task that is going to be executed. Fig. 8 shows a sequence of pictures captured from one of the reach-and-grasp movements executed on the real robot. Reach-and-grasp movements for different grasp types and situations are shown in the video submitted as *supplemental material*.

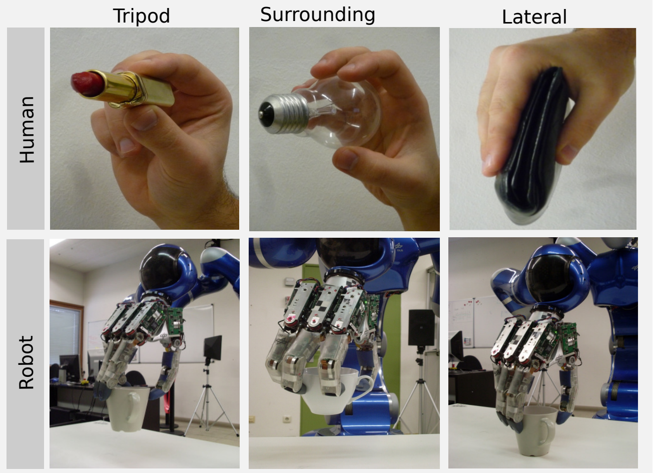

Fig. 7. The three grasp types lateral, surrounding, and tripod from our taxonomy are demonstrated by a human and later reproduced by the Justin robot. All parameters of the reach-and-grasp movement, such as the shape of the hand, its position, and orientation are automatically determined using latent space dynamic motor primitives.

IV. Conclusion

In this paper, we presented a new approach for imitation and generalization of human grasping skills for multi-fingered robots. The approach is fully data-driven and learns from human demonstrations. As a result, it can be used to easily program new grasp types into a robot – the user only needs to perform a set of example grasps. In addition to stable grasps on the object, this approach also leads to visually appealing hand configurations of the robot. Contact points from a known object are processed by a contact warping technique in order to estimate good contact points on a new object.

We, furthermore, presented latent-space dynamic motor primitives as an extension to dynamic motor primitives that explicitly models synergies between different body parts. This significantly reduces the number of parameters needed to control systems with many DoF such as the human hand. Additionally, we have presented a principled optimization scheme that exploits the low-dimensional grasp spaces to estimate all parameters of the reach and grasp movement.

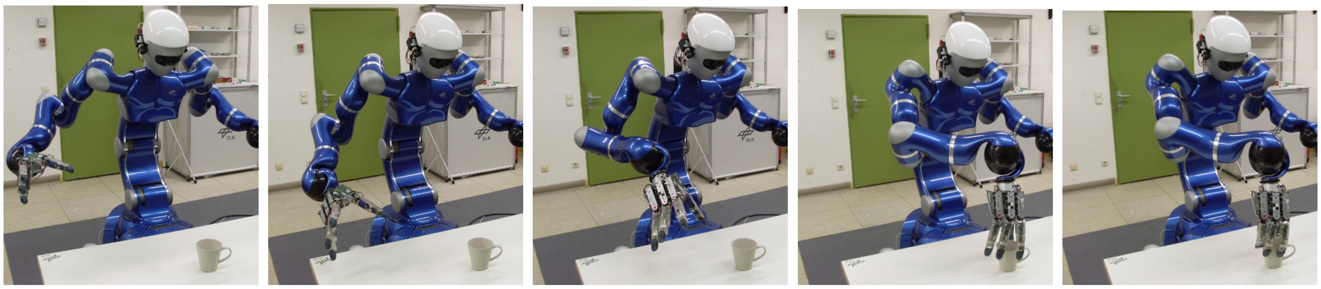

Fig. 8. A sequence of images showing the execution of a reach-and-grasp movement by the Justin humanoid robot. The executed *latent-space dynamic motor primitive* was learned by imitation. The type of the grasp to be executed can be varied according to the requirements of the task to subsequently executed. New grasp types can be trained within minutes by recording a new set of human demonstrations.

The proposed methods were evaluated both in simulation and on the real Justin robot. The experiments exhibited the robustness of the approach with respect to changes in the environment. In all of the experiments on the real, physical robot, the method successfully generated reach-and-grasp movements for lifting up the seen object.

ACKNOWLEDGMENT

We thank Florian Schmidt and Christoph Borst from the DLR - German Aerospace Center for their help with the Justin robot and for their valuable comments and suggestions. H. Ben Amor was supported by a grant from the Daimler-und-Benz Foundation. The project receives funding from the European Community's Seventh Framework Programme under grant agreement n ICT- 248273 GeRT.

REFERENCES

[1] W. Abend, E. Bizzi, and P. Morasso. Human arm trajectory formation. *Brain : a journal of neurology*, 105(Pt 2):331–348, jun 1982.

[2] M. Arbib, T. Iberall, and D. Lyons. Coordinated control programs for movements of the hand. *Experimental brain research*, pages 111–129, 1985.

[3] H. Ben Amor. *Imitation learning of motor skills for synthetic humanoids*. PhD Thesis, Technische Universitaet Bergakademie Freiberg, Freiberg, Germany, 2011.

[4] H. Ben Amor, G. Heumer, B. Jung, and A. Vitzthum. Grasp synthesis from low-dimensional probabilistic grasp models. *Comput. Animat. Virtual Worlds*, 19(3-4):445–454, sep 2008.

[5] C. Borst, T. Wimbock, F. Schmidt, M. Fuchs, B. Brunner, F. Zacharias, P. R. Giordano, R. Konietschke, W. Sepp, S. Fuchs, C. Rink, A. Albu-Schaffer, and G. Hirzinger. Rollin' justin - mobile platform with variable base. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 1597 –1598, may 2009.

[6] A. Boularias, O. Kroemer, and J. Peters. Learning robot grasping from 3-d images with markov random fields. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2011*, pages 1548–1553, 2011.

[7] S Chieffi and M Gentilucci. Coordination between the transport and the grasp components during prehension movements. *Experimental Brain Research*, pages 471–477, 1993.

[8] M. T. Ciocarlie and P. K. Allen. Hand posture subspaces for dexterous robotic grasping. *Int. J. Rob. Res.*, 28(7):851–867, July 2009.

[9] K. Dautenhahn and C. L. Nehaniv. *Imitation in Animals and Artifacts*. MIT Press, Campridge, 2002.

[10] G. Heumer. *Simulation, Erfassung und Analyse direkter Objektmanipulationen in virtuellen Umgebungen*. PhD Thesis, Technische Universitaet Bergakademie Freiberg, Freiberg, Germany, 2011.

[11] U. Hillenbrand. Non-parametric 3d shape warping. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2656 –2659, 2010.

[12] U. Hillenbrand and A. Fuchs. An experimental study of four variants of pose clustering from dense range data. *Computer Vision and Image Understanding*, 115(10):1427 – 1448, 2011.

[13] U. Hillenbrand and M. A. Roa. Transferring functional grasps through contact warping and local replanning. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2012.

[14] A.J. Ijspeert, J. Nakanishi, and S. Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 2, pages 1398 –1403, 2002.

[15] M Jeannerod. The timing of natural prehension movements. *Journal of Motor Behavior*, 16(3):235–254, 1984.

[16] M. Jeannerod. *Perspectives of Motor Behaviour and Its Neural Basis*, chapter Grasping Objects: The Hand as a Pattern Recognition Device. 1997.

[17] M. Jeannerod. *Sensorimotor Control of Grasping: Physiology and Pathophysiology*, chapter The study of hand movements during grasping. A historical perspective. Cambridge University Press, 2009.

[18] J. Kober, B. Mohler, and J. Peters. Learning perceptual coupling for motor primitives. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 834 –839, sept. 2008.

[19] R. N. Lemon. Neural control of dexterity: what has been achieved? *Exp Brain Res*, 128:6–12+, 1999.

[20] C. R. Mason, J. E. Gomez, and T. J. Ebner. Hand Synergies During Reach-to-Grasp. *Journal of Neurophysiology*, 86(6):2896–2910, December 2001.

[21] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. Learning from demonstration and adaptation of biped locomotion. *Robotics and Autonomous Systems*, 47:79?–91, 2004.

[22] M. Saleh, K. Takahashi, and N.G. Hatsopoulos. Encoding of coordinated reach and grasp trajectories in primary motor cortex. *J Neurosci*, 32(4):1220–32, 2012.

[23] M. Santello, M. Flanders, and J. F. Soechting. Postural Hand Synergies for Tool Use. *The Journal of Neuroscience*, 18(23):10105–10115, December 1998.

[24] M. Santello and J. F. Soechting. Gradual molding of the hand to object contours. *Journal of neurophysiology*, 79(3):1307–1320, March 1998.

[25] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic grasping of novel objects using vision. *Int. J. Rob. Res.*, 27(2):157–173, feb 2008.

[26] M H Schieber. How might the motor cortex individuate movements? *Trends Neurosci*, 13(11):440–5, 1990.

[27] R. Suárez, M. Roa, and J. Cornella. Grasp quality measures. Technical report, Technical University of Catalonia, 2006.

[28] Johan Tegin, Staffan Ekvall, Danica Kragic, Jan Wikander, and Boyko Iliev. Demonstration-based learning and control for automatic grasping. *Intelligent Service Robotics*, 2009.

[29] M. Toussaint and C. Goerick. A bayesian view on motor control and planning. In *From Motor Learning to Interaction Learning in Robots*, pages 227–252. 2010.

[30] R. J. Vanderbei. *Linear Programming: Foundations and Extensions*. Springer, 2001.