Autonomous Robots (2023) 47:453–464
https://doi.org/10.1007/s10514-023-10101-z

Robotic hand synergies for in-hand regrasping driven by object information

Dimitrios Dimou[1] · José Santos-Victor[1] · Plinio Moreno[1]

Received: 17 March 2022 / Accepted: 5 March 2023 / Published online: 11 April 2023

Abstract

We develop a conditional generative model to represent dexterous grasp postures of a robotic hand and use it to generate in-hand regrasp trajectories. Our model learns to encode the robotic grasp postures into a low-dimensional space, called Synergy Space, while taking into account additional information about the object such as its size and its shape category. We then generate regrasp trajectories through linear interpolation in this low-dimensional space. The result is that the hand configuration moves from one grasp type to another while keeping the object stable in the hand. We show that our model achieves higher success rate on in-hand regrasping compared to previous methods used for synergy extraction, by taking advantage of the grasp size conditional variable.

Keywords Robotics · Dexterous robotic grasping · In-hand manipulation · Regrasping

1 Introduction

The control of dexterous artificial hands with a high number of degrees of freedom has been a long-standing research problem in robotics. Researchers have tried mimicking the way humans control their hands in order to facilitate the control of robotic hands (Santello et al., 2016). Studies from the field of neuroscience have shown that the human brain leverages a synergistic framework that relies on coupled neural signals, muscles and kinematic constraints to perform dexterous manipulation skills like grasping (Santello et al., 1998, 2016). This type of organization allows a small number of parameters (synergies) to control a large number of degrees of freedom (hand joints, muscles etc.).

Roboticists have taken advantage of this concept in neuroscience by modelling the control of dexterous hands in lower dimensional spaces. This way they can effectively reduce the computational burden of dexterous control (Ciocarlie et al., 2007a), as control algorithms can operate directly on the low dimensional space, thus reducing the number of control parameters. The problem is formulated as finding a mapping from the high-dimensional configuration space (e.g. joint angle values) of an artificial hand to a lower-dimensional embedding, commonly referred to as Synergy Space (Salvietti, 2018).

The first studies (Ciocarlie et al., 2007a, b) to apply this principle, closely following the nominal neuroscience study (Santello et al., 1998) that established this concept, used the classical dimensionality reduction linear method, Principal Component Analysis (PCA), to extract a low-dimensional representation from a set of recorded grasp postures, and used this representation to search for grasp postures. This model though cannot be conditioned on additional information such as the object's properties.

Following works (Romero et al., 2013; Xu et al., 2016), have improved the reconstruction error of the grasp postures, when compared with PCA, by using a non-linear latent variable model based on Gaussian processes (GPLVM). Auto-Encoder (AE) models, which is a non-linear, deterministic dimensionality reduction method, based on neural networks, have also been shown to improve the performance in terms of reconstruction and are able to encode additional information such as the object size (Starke et al., 2018, 2020). However, these models produce irregular latent spaces that can be hard to generate trajectories in.

✉ Dimitrios Dimou
mijuomij@gmail.com

José Santos-Victor
jasv@isr.tecnico.ulisboa.pt

Plinio Moreno
plinio@isr.tecnico.ulisboa.pt

[1] Institute for Systems and Robotics, Instituto Superior Tecnico, Universidade de Lisboa, Lisboa, Portugal

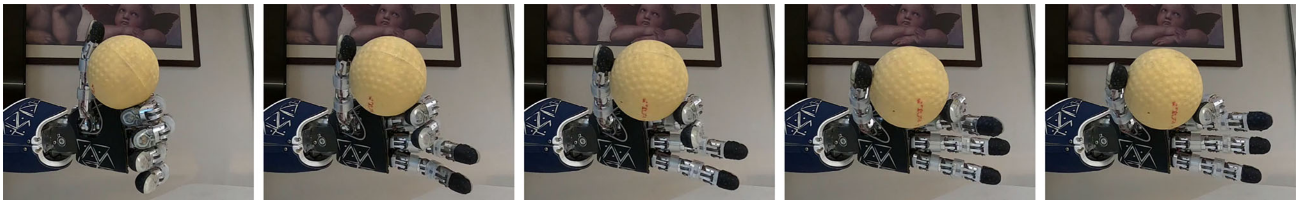


Fig. 1 Example regrasp trajectory generated by our method and executed using the iCub robot. The grasp posture changes from a lateral pinch to a tip pinch

In our previous work (Dimou et al., 2021), we used a Conditional Variational Auto-Encoder to learn a conditional low-dimensional latent space from a set of grasp postures executed on a robotic hand. This latent space was conditioned on the size of the object and its shape category. The smoothness of the latent space was introduced as a metric to evaluate its effectiveness. We showed that this model learns a smoother latent space, and we used it to generate successful trajectories for in-hand regrasping tasks in a simulated environment, using the Shadow Hand that resembles the human adult in size and degrees of freedom.

This work evaluates further the methods proposed in Dimou et al. (2021), applying the same learning architecture to the real world iCub hand, which resembles a child human size and has less degrees of freedom than the Shadow Hand (i.e. less dexterity). We use data collected for the iCub Robot hand to learn a grasp posture generation model and use it to generate trajectories for in-hand regrasping. We found that in the real world setting the generated trajectories became unstable due to unaccounted object properties such as the material and mass. To overcome this we reduce the conditional size variable of the model during testing and we improve the success rate in the regrasping tasks. An example of a regrasp trajectory executed with our model can be seen in Fig. 1. In our experiments we also present the results from Dimou et al. (2021) for the Shadow Robot Hand and we show that our results for the iCub Robot hand exhibit the same patterns.

In summary our contributions are:

- We apply the conditional model for the generation of grasp postures presented in Dimou et al. (2021), to a real-world in-hand regrasping task using the iCub robot.
- We propose a method for generating regrasp trajectories in the latent space of the model.
- We show the practical impact of the object size variable, in the regrasping performance of our model. By adapting the object size, our model avoids slippage in grasp execution.
- We perform real world experiments with the iCub robot demonstrating that this method greatly improves the performance of previous approaches, showing the versatility of our model that was applied to two dexterous hands with different dexterity and size.

2 Related work

Synergy extraction. In neuroscience, the study (Santello et al., 1998) established the concept of synergies as the method used by the human brain to ease the control of the human hand. By analyzing grasp postures of human subjects who where asked to grasp imaginary objects, they showed that the first two principal components, that were extracted using the PCA method, were responsible for 80% of the variation in the data, suggesting that using only two components they could represent the acquired data to a high degree.

In robotics, the data collected in Santello et al. (1998) were used to transfer the grasp postures of the human subjects to 4 robotics hands by Ciocarlie et al. (2007a). Then the PCA method was used to find a low-dimensional basis to express the recorded grasp postures. The recovered basis components were called *eigengrasps*, and were combined to generate new grasp postures. They showed that control algorithms that search in that space for stable grasps were more efficient. But PCA is a linear model that cannot model complex high dimensional data and also cannot be conditioned on additional variables.

Precision grasps were analysed in Bernardino et al. (2013), where the PCA method was used to extract the principal components from dataset of grasps, achieved by a human operator controlling two dexterous robotic hands. In our experiments we use the dataset collected in this work for the iCub robot to learn the conditional latent space for precision grasps.

Since the purpose of synergies is to find a low-dimensional space to represent the grasp postures, most classical dimensionality reduction methods can be applied to the problem. In Jenkins (2006); Tsoli Odest & Jenkins (2007) several of them were compared for encoding the control of a robotic hand in 2D subspaces. The methods were applied on recorded hand movements and evaluated based on the inconsistency and continuity of the representations, while a method for denoising graphs consisted of embedded grasp postures was proposed. In this work, we use the smoothness of the latent space to evaluate its effectiveness which is similar to the continuity presented in Tsoli Odest & Jenkins (2007).

An Auto-Encoder (AE) model (Kramer, 1991) was used in Starke et al. (2018, 2020) to extract postural synergies from human data. The relative object size compared to the palm, was also used as an additional input variable in the decoder for the grasp generation. The model was trained with a modified loss in order to disentangle the grasp types in the latent space. It achieved better reconstruction results when compared to the PCA, and was able to generate grasp postures for objects of different sizes. However, Auto-Encoder models tend to learn non smooth latent spaces that can result in unstable trajectories in manipulation tasks (Dimou et al., 2021).

The methods presented until now in this section were deterministic. In Romero et al. (2013); Xu et al. (2016), the Gaussian Process Latent Variable Model (GPLVM) (Lawrence, 2003), which is a non linear probabilistic model, was used to learn a grasp manifold from a dataset of recorded grasp postures. They showed that this model has lower reconstruction error when compared with the PCA.

In this work we use a Conditional Variational Auto-Encoder (CVAE) (Sohn et al., 2015), which is conditional probabilistic model based on the VAE framework (Kingma &Welling, 2013). This model learns to represent and generate grasp postures given additional input signals such as the object size and type (Dimou et al., 2021). In addition, it has been shown to learn smoother latent spaces that previous approaches which is instrumental in planning for in-hand regrasping.

Regrasping. Postural synergies have also been used to facilitate in-hand manipulation. In Palli et al. (2014), they used the PCA method to compute a Synergy Space from grasp postures achieved by human subjects. Then they created regrasp trajectories by linearly interpolating between the boundary configurations in the Synergy Space. They also computed an additional Synergy Space from demonstrations of manipulation tasks and showed that combining these two spaces improves the quality of the manipulations. In Katyara et al. (2021), they used the PCA to compute a Synergy Space from a set of demonstrations of in-hand manipulations. Then they parameterized the demonstrations in the Synergy Space using Kernelized movement primitives. Using this parameterization they were able to achieve four in-hand manipulation tasks.

In this work, we generate the trajectories in the Synergy Space by linearly interpolating between the initial and target grasp postures, similarly to Palli et al. (2014), but we do not use any demonstration data. Instead to improve the success rate of the generated trajectories we adjust the conditional variables of the model in order to perform firmer grips. We show that our method outperforms previous approaches in in-hand regrasping tasks in a real world setting.

3 Background

In the context of robotics, postural synergies are modeled with dimensionality reduction techniques. Given a hand posture $\mathbf{x} \in \mathbb{R}^{d_x}$, usually represented by a vector containing the joint angle values of the hand, we want to find a mapping $e(\mathbf{x})$ that encodes the grasp into a low-dimensional point $\mathbf{z} \in \mathbb{R}^{d_z}$, where $d_z << d_x$. We also need a mapping $d(\mathbf{z})$ that decodes the low-dimensional point back into the original space. These mappings are parameterized by vectors θ, ϕ. Given a dataset of observations $\mathbf{X}$, we want to compute the parameter vectors (θ, ϕ) that minimize an optimization criterion. The low-dimensional embeddings $\mathbf{z_i}$ are the synergistic components and the space is called a latent space.

PCA parameterizes this mapping as a linear function which can be written as $e(\mathbf{x}) = \mathbf{xW}$, where $\mathbf{W} \in \mathbb{R}^{d_x \times d_z}$ are the parameters to be found. Auto-encoders models, use neural networks to represent the encoding and decoding mappings. So the input has a non-linear relationship with the latent embedding. In both cases the optimization criterion used to find the optimal parameters is the mean squared error between the input and the reconstruction of the model. In the case of PCA, an analytic solution can be found, while for Auto-encoders gradient based optimization is usually performed.

Probabilistic approaches assume that the data are generated by unobserved latent variables following a probability distribution $p(\mathbf{x}, \mathbf{z}; \theta)$, where $\mathbf{x}$ are the observed data, i.e. the grasp postures, $\mathbf{z}$ are the latent variables, i.e. the synergistic components, and θ are the parameters of the model. The GPLVM approach uses Gaussian Processes to model the probability distribution p. In Xu et al. (2016); Romero et al. (2013), a variant of the GPLVM with back constraints (BCGPLVM) (Lawrence et al., 2006) is used for synergy extraction. This model enforces a constraint on the latent variables that ensures that points that are close in the original space remain close in the latent space.

In this work, we use a Conditional Variational Auto-Encoder to model the representation learning process. The CVAE consists of an encoder and a decoder network. The encoder takes as input a data point $\mathbf{x}$ and a corresponding conditional variable $\mathbf{c}$ and produces a latent point $\mathbf{z}$. The decoder takes as input a latent point $\mathbf{z}$ and a conditional variable $\mathbf{c}$ and generate a new data point $\mathbf{x}$. The encoder models the probability distribution $q(\mathbf{z} \mid \mathbf{x}, \mathbf{c})$, while the decoder the probability distribution $p(\mathbf{x} \mid \mathbf{z}, \mathbf{c})$. During training we maximize the evidence lower bound (ELBO):

$$\begin{aligned}\mathcal{L}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}) = {} & \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{c})}\left[\log p_\theta(\mathbf{x} \mid \mathbf{c}, \mathbf{z})\right] \\ & - D_{KL}\left(q_\phi(\mathbf{z} \mid \mathbf{x}, \mathbf{c}) \| p(\mathbf{z})\right)\end{aligned} \tag{1}$$

The first term corresponds to the mean squared error between the reconstruction and the input, while the second minimizes the Kullback–Leibler divergence between the true posterior distribution $p(\mathbf{z} \mid \mathbf{x})$ and a variational distribution $q(\mathbf{z} \mid \mathbf{x})$, which works as a regularization criterion for the latent space.

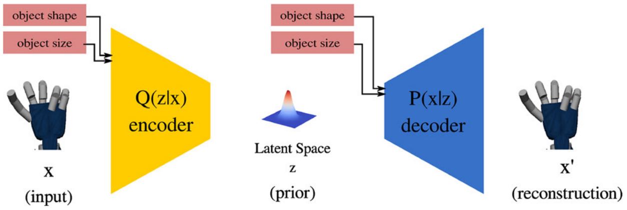


Fig. 2 Schematic representation of CVAE model

The loss is minimized using standard gradient descent methods like (Kingma &Ba, 2014).

4 Methods

Grasp posture representation. Given a set of recorded grasp postures, the goal of this work is to learn a low-dimensional Synergy Space that can be used to generate regrasping trajectories. By regrasping we mean changing the grasp type from an initial type of grasp to a target one, while holding the object in the hand. In previous works, the goal was to learn a model that accurately reproduces the recorded grasps. In Dimou et al. (2021) we showed that this criterion is not sufficient to evaluate the effectiveness of the learned latent space in regrasping tasks. Instead, we used the smoothness of the latent space as an alternative evaluation metric. Smoothness is defined using the distance between grasps decoded from latent points on a grid, similar to Chen et al. (2016). The distance is computed as the average difference of their joint angles. More precisely, given two latent points $\mathbf{z}_1$ and $\mathbf{z}_2$ and a decoding mapping $d(z)$ to the grasp posture space we define smoothness as:

$$S(\mathbf{z}_1, \mathbf{z}_2) = \|d(\mathbf{z}_1) - d(\mathbf{z}_2)\|_2$$

This gives us the average change in the joint angles of the robotic hand if we move from one point on the grid to another. The use of smoothness as a metric for evaluating the learned latent space for manipulation tasks, agrees with our intuition that if we are planning finger movements we want to avoid sudden changes in finger joints that might make our grasp of the object unstable. Instead, we want to perform smooth transitions between hand states that keep the object stable and balanced.

Following the work in Dimou et al. (2021) we train a Conditional Variational Auto-Encoder to generate grasp postures given as additional signals the size of the side of the object to be grasped and the type of the object. A schematic representation of our model can be seen in Fig. 2. We use a dataset of recorded grasps originally presented in (Bernardino et al., 2013). The dataset consists of *536 grasps* performed using the iCub robot by a human operator teleoperating the robot with a data glove. Each grasp, following the grasp taxonomy (Feix et al., 2009), belongs in one of the eight following precision-grasp categories: tripod, palmar pinch, lateral, writing tripod, parallel extension, adduction grip, tip pinch, lateral tripod. The objects that were grasped were three balls of different radius, three cylinders of different radius and height, and a box with three different sizes in each side. So for each grasp posture there is a corresponding label denoting the size of the side of the object that was grasped, i.e. if it was the large, the medium or the small side, and the shape category of the object. i.e. if it is a ball, a cylinder or a box. The size label is represented with a scalar value in (0.0, 1.0), where 0 corresponded to a grasp on the small side of the object, 0.5 to a grasp on the medium side of the object, and 1.0 to a grasp on the large side of the object. The shape category label is represented using one-hot encoding.

The model is trained by feeding the grasp postures $\mathbf{x_i}$ and the corresponding labels $\mathbf{c_i}$, which denote the size and the shape category of the object, into the encoder, which models a sampler of the probability distribution $q(\mathbf{z} \mid \mathbf{x}, \mathbf{c})$. So given a grasp posture $\mathbf{x_i}$ from the dataset and a label $\mathbf{c_i}$, we sample a latent point $\mathbf{z_i}$. The decoder models a sampler of the distribution $p(\mathbf{x} \mid \mathbf{z}, \mathbf{c})$, so given a latent point $\mathbf{z_i}$ and a label $\mathbf{c_i}$ it generates a grasp posture $\hat{\mathbf{x}}_i$ that has the properties of the given label. The parameters of the model are then optimized in order to minimize the mean squared error between the input $\mathbf{x_i}$ and the output $\hat{\mathbf{x}}_i$, and the KL divergence between the prior $p(\mathbf{z})$, which is a standard normal distribution, and the posterior $q(\mathbf{z} \mid \mathbf{x}, \mathbf{c})$. The minimization of the mean squared error forces the model to learn to reconstruct the input grasp postures, while the minimization of the KL divergence forces the latent space to follow the standard normal distribution.

In-hand regrasping. In the in-hand regrasping task we want to execute a regrasp trajectory that changes the grasp type executed on the object without changing the side of the grasp that the object is grasped from and without breaking contact with the object. To generate regrasp trajectories we encode the initial and the target grasp postures into the latent space, so we obtain two latent points $\mathbf{z}_{initial}$ and $\mathbf{z}_{target}$. We then linearly interpolate between these two points in the Euclidean space and sample N points, where N equals the number of steps in the trajectory. Finally we decode the new sampled points to obtain a trajectory in the configuration space. The complete trajectory generation procedure is outlined in Algorithm 1. A schematic representation of the trajectory generation process can be seen in Fig. 3. In essence, instead of employing a complex planning algorithm to find a path of states that can perform the required regrasp we rely on the structure of the latent space of the model. If the latent space is smooth we can generate successful trajectories by a simple procedure such as linear interpolation.

Algorithm 1 Trajectory generation in latent space

```
Initial $\mathbf{x}_{initial}$ and target $\mathbf{x}_{target}$ grasp postures
$\mathbf{z}_{initial} = encoder(\mathbf{x}_{initial})$
$\mathbf{z}_{target} = encoder(\mathbf{x}_{target})$
N : number of steps in trajectory
Initialize trajectory T = []
for i=0 to N do
    $\mathbf{z}_{new} = \mathbf{z}_{initial} * (1 - \frac{i}{N}) + \mathbf{z}_{target} * \frac{i}{N}$
    $\mathbf{x}_{new} = decoder(\mathbf{z}_{new})$
    T.append($\mathbf{x}_{new}$)
end for
```

Although this method has worked in simulation (Dimou et al., 2021), we found that in real world experiments the method was not outperforming the other approaches, due to issues during contact. More specifically, small objects with smooth surface and high mass were slipping. In order to overcome these problems, during testing we adjusted the conditional size variable used by the decoder to generate the regrasp trajectory. The initial model was trained with size values in (0.0, 1.0), while when testing the size values were reduced by 0.5. This way the model produced firmer grips which were more stable. We show in the experimental results that this process was crucial to increasing the performance of the model.

5 Experimental results

Dataset description. The dataset used to train our models was originally presented in (Bernardino et al., 2013). It was acquired by teleoperating two robotic hands: the Shadow Dexterous Hand, and the iCub robot hand. In this work we use the dataset recorded for the iCub robot.

For the teleoperation of the robot, a mapping was used that transformed the joint angles of the human operator's hand to the joint angles of the iCub hand. The mapping, was fixed for each user and it was generated by the acquisition software. So each recorded grasp is represented by the angle values in degrees of the 9 joints of the iCub robot hand. This way our model is trained directly on the robot angles and we do not need the human to robot mapping after the data collection process. Twelve objects were grasped, from five distinct categories: ball (three sizes), box (three sizes), cylinder (three sizes), pen and cube. The objects were grasped from different sides adding up to a total of 20 different object configurations.

The models were trained on a subset of this dataset, more specifically on the grasps performed on the balls, the cylinders and the box with three different size sides. We rep-

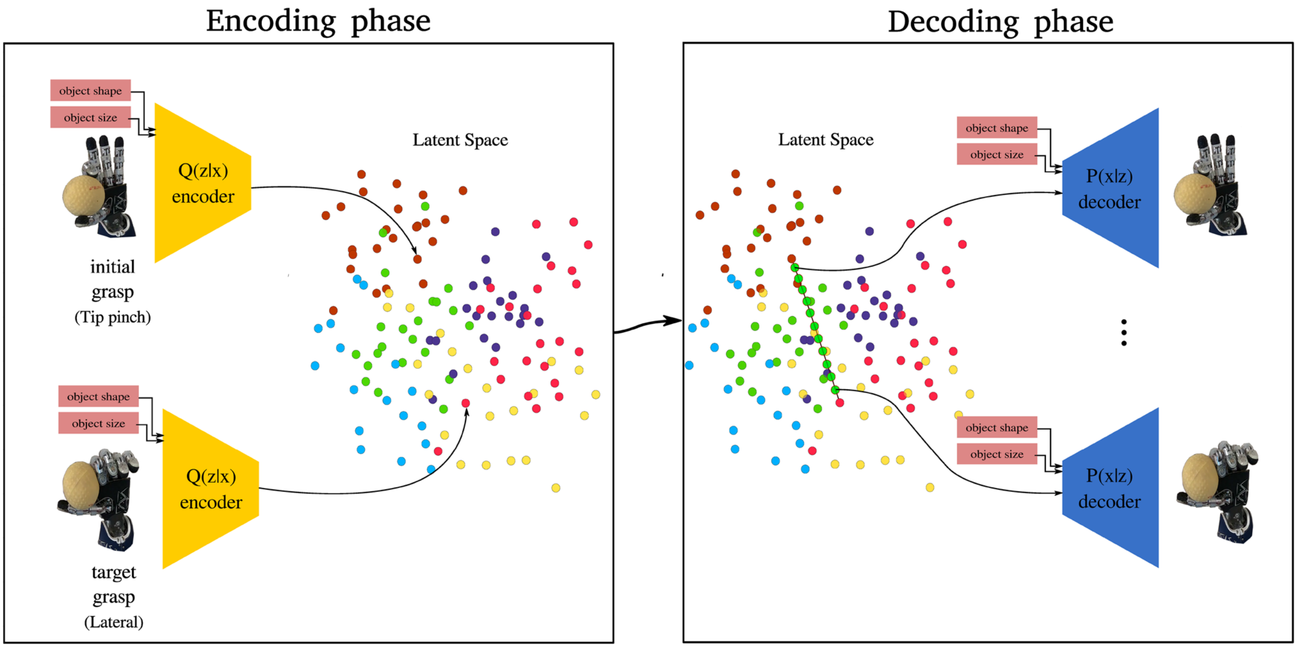


Fig. 3 Schematic representation of generating trajectory in latent space. In the encoding phase both the initial and the target grasp are encoded into the latent space. The different colors of the points in the latent space denote different grasp types. During the decoding phase, N points along the line connecting the initial and target grasps are decoded and a regrasp trajectory is produced

Table 1 Size value label for each object configuration

Object configuration	Size label
Medium green cylinder, top	0.0
Medium green cylinder, side	
Small wooden cylinder, side	
Small white ball	
Box, small side	
Box, medium side	
Medium yellow ball	0.5
Small wooden cylinder, top	
Big red cylinder, side	
Box, large side	
Big blue ball	1.0
Big red cylinder, top	

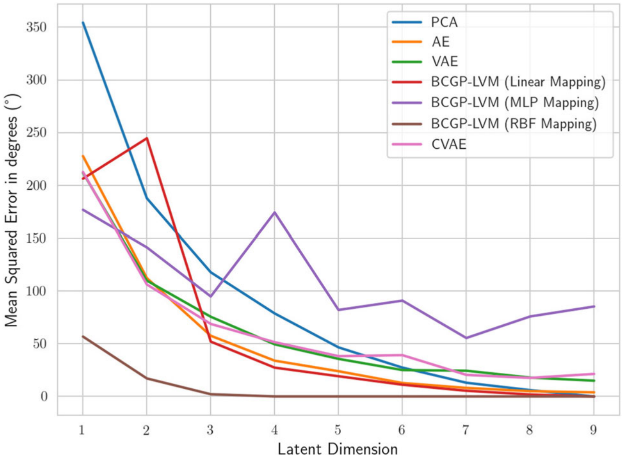


Fig. 4 Plot of the mean squared error of each model for latent dimensions from one to the number of degrees of freedom of the iCub hand

Table 2 Smoothness results for iCub Robot: The mean and standard deviation calculated from the latent space gradients of each model for three grid resolutions. Lower values suggest a smoother latent space

	(μ, σ)		
	N=5	N=15	N=25
PCA	(26.0, 5.0)	(7.4, 1.4)	(4.3, 0.8)
AE	(33.9, 8.9)	(10.8, 2.9)	(6.4, 1.7)
VAE	(29.1, 10.2)	(8.7, 3.2)	(5.1, 1.9)
CVAE	**(21.9, 6.7)**	**(6.3, 2.0)**	**(3.7, 1.2)**
BCGPLVM (Linear)	(28.2, 8.8)	(12.1, 6.0)	(7.3, 3.7)
BCGPLVM (MLP)	(27.1, 8.3)	(8.9, 3.5)	(5.3, 2.1)
BCGPLVM (RBF)	(46.2, 17.5)	(26.6, 14.5)	(17.1, 10.1)

The lowest mean values are highlighted in bold

resented the shape category of the object, meaning if it was a ball, a box or a cylinder using one-hot encoding. For the size we used a continuous scalar variable in (0.0, 1.0). In the original dataset grasps were labeled as: large, medium, and small, according to the size of the object. In our dataset, we labeled the grasps for each object configuration with the values 0.0, 0.5, 1.0. The object configurations were selected such that the size of the grasps were similar. In Table 1, you can see each object configuration and the corresponding size label. In our dataset we mapped these labels to the (0.0, 1.0) range. Large size grasps were represented by the value 1.0, medium size grasps by the value 0.5, and small size grasps by the value 0.0. But the value was given as a continuous variable and can take any real value, in contrast the shape variable that is discrete and it was one-hot encoded. We concatenated the shape and the size into a vector and used it as the conditional variable c_i in the CVAE model.

Models. We trained seven models on the iCub dataset. A PCA model, a standard AE architecture model, a standard VAE, the CVAE model described in Sect. 4 and three BCGPLVM models with the following back projection mappings: (1) a linear mapping, (2) a multi layer perceptron (MLP) and (3) a radial basis function (RBF). The choice of mapping has an effect on the models reconstruction error and the smoothness of its latent space.

Latent space analysis. Following (Dimou et al., 2021), we performed the same analysis of the latent space for each model. More specifically, for each model we computed its reconstruction error on the dataset, seen in the left Fig. 4. The BCGPLVM model with the RBF kernel as the back constraint has the lowest reconstruction error among all models. As we increase the latent dimensions to the number of degrees of freedom of the hand, the reconstruction error in most models goes to zero. That is not the case for the VAE and the CVAE models because in their loss function there is the additional term of the KL divergence between the prior and the posterior.

Following the nominal neuroscience study (Santello et al., 1998), that showed that 2 synergies are enough to represent most human grasps, and previous robotics works that also used 2 synergistic components, the dimensionality of the latent space for the following experiments was chosen to be 2, to be directly comparable with the other works. We then calculated the smoothness of each model's latent space as proposed in Dimou et al. (2021) for three grid resolutions $N = 5, 15, 25$, seen in Table 2. The CVAE model exhibits the lowest average change in joint angle values between neighboring grasps, indicating that movements between near states in the latent space will results in smooth transitions in the configuration space. On the other hand, the BCGPLVM model with the RBF kernel that had the lowest reconstruction error exhibits high variation in the latent space which results in sudden changes in the configuration space.

Finally, Fig. 5 shows the latent space traversals for the PCA, the VAE and the BCGPLVM with a linear kernel. We

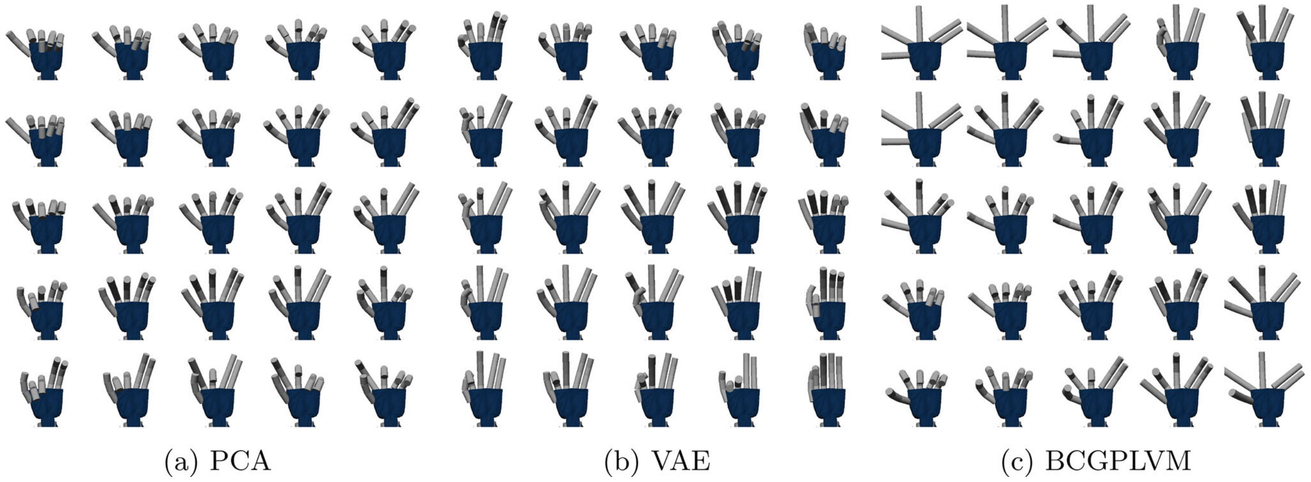


Fig. 5 Latent space traversals in the spaces learned by the PCA, the VAE, and the BCGPLVM with a linear kernel model

used the iCub simulator to decode the grasps from a square grid of size 5 in each model's latent space. We can notice that the latent space of the PCA and the VAE exhibit more variability but also smoother transitions between neighboring grasps. On the other hand, in the latent space of the BCGPLVM we see a lot of grasps that are not valid, and sudden changes between neighboring grasps. The smoothness of the latent space is a result of the KL divergence regularization in the loss function of the VAE and cVAE models as mentioned in Asperti &Trentin (2020); Oring et al. (2021). This term forces the prior distribution of the model to match the normal distribution. It brings the latent points towards the origin of the axes making the latent space more dense and evenly distributed, compared for example with the AE model that uses the same loss function without the regularization.

Real robot experiments. In Dimou et al. (2021), regrasp experiments were conducted in simulation and using only one of the objects of the dataset, the box. In this work, we performed real world experiments with the iCub robot, using all the objects that were used to execute the grasps from the training dataset, seen in Fig. 6.

For each object configuration, i.e. each side an object can be grasped from (e.g. a sphere has one object configuration, a cylinder two: top and side, a box with three different sized sides has three), 5 pairs of grasps, where each grasp had an associated grasp type, were chosen randomly from the dataset and for each pair a regrasp trajectory was generated by each model using Algorithm 1, totaling 60 regrasp trajectories. The number of steps in each trajectory was set to 10. Each trajectory was performed three times to account for variability in initial conditions. In Fig. 7, we see a chord diagram representing the connections of grasp types in the regrasp trajectories executed. The robot was restricted to perform the regrasp trajectory on the same side of the object. This was done using the object size conditional variable, that forced the model to produce tight grasps and the fingertips to always be in contact with the object. So since there was no breaking and creating contacts the object was always grasped from the initial side.

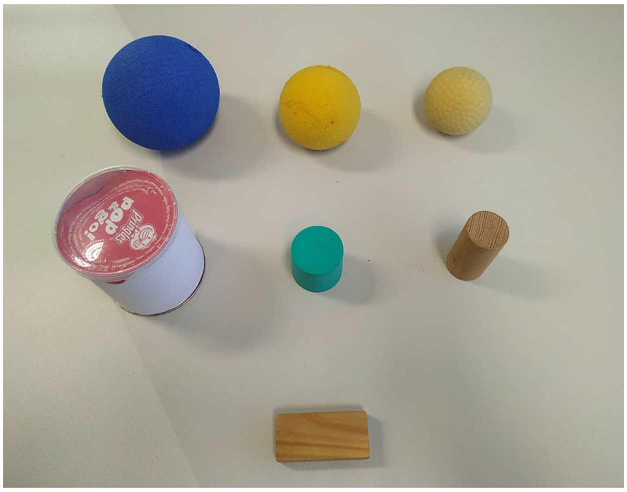


Fig. 6 Objects used to execute regrasp trajectories

During preliminary experiments we noticed that the regrasp trajectories generated by the CVAE model, when executed on the real robot, were not outperforming the other models as suggested in Dimou et al. (2021). The reason was that although the transitions between states were smooth, the unaccounted properties of the objects, such as their material and mass, were causing some grasp postures to be unstable mainly due to slippage. This phenomenon was most apparent in objects with small size and smooth surface. In order to overcome this, we took advantage of the conditional variables that the CVAE model encodes, i.e. the object size. When generating trajectories using the CVAE model we adjusted the size label to be lower than the corresponding original label.

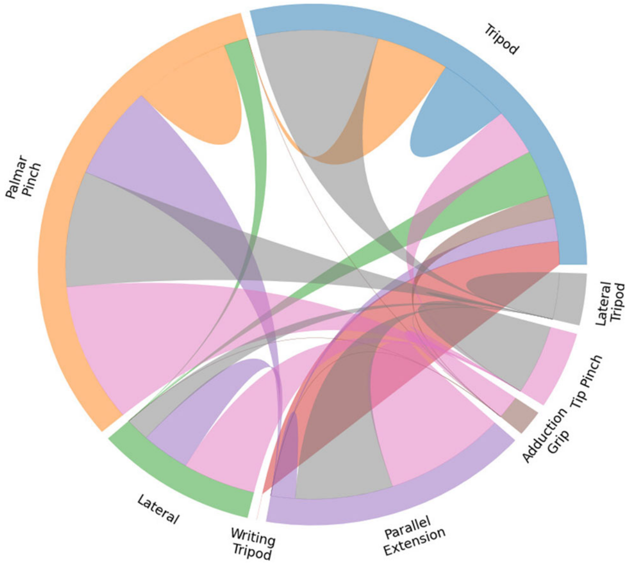


Fig. 7 Chord diagram representing connections of grasp types in executed regrasp trajectories. The arcs represent each grasp type in the dataset. The size of each arc is analogous to the number of occurrence of each grasp type in the trajectories. A connection (chord) between two arcs means that a regrasp from one grasp type to another was executed. The chords are colored according to the initial grasp type. Regrasps between grasps of the same type are represented as half-ellipses on arcs. These regrasps occur because the initial grasps in the dataset were recorded by multiple human operators, and since the robotic has a lot of DoFs some operators performed the same grasp type differently

More specifically, when decoding the latent points of each trajectory we reduced each point's size label by the value of 0.5. This way the model produced firmer grips that were more stable during execution. This adjustment is not possible to be applied to the other models as they cannot be conditioned on additional variables. The results of the execution of the trajectories can be seen in Fig. 8 in a box plot format. On the vertical axis is the percentage of successful regrasp trajectories generated by each model. Each box represents the interquartile range between the three runs of each trajectory, while the line in the middle of the box the median. We see that most models follow the same pattern with the results in Dimou et al. (2021), but the CVAE model using the original labels for the size of the object does not surpass the performance of the other models. On the other hand the trajectories generated with the CVAE model but with the adjusted size labels generates twice as many successful regrasp trajectories. In Fig. 9, we can see some of the regrasp trajectories generated by the CVAE model with the adjusted size labels.

Finally, we investigated the size interpolation and extrapolation capabilities of the proposed CVAE model. More specifically, as the size label during training is a scalar value in (0.0, 1.0), we wanted to see if the model is able to produce grasps for other values in that range, as well as values outside that range. To test this, we generated 100 different grasps, randomly choosing a latent point and an object type, and using 20 evenly spaced grasp size values in the range (−1.0, 2.0). We executed each grasp on the simulated model of the iCub hand and computed the Euclidean distance between the fingertip

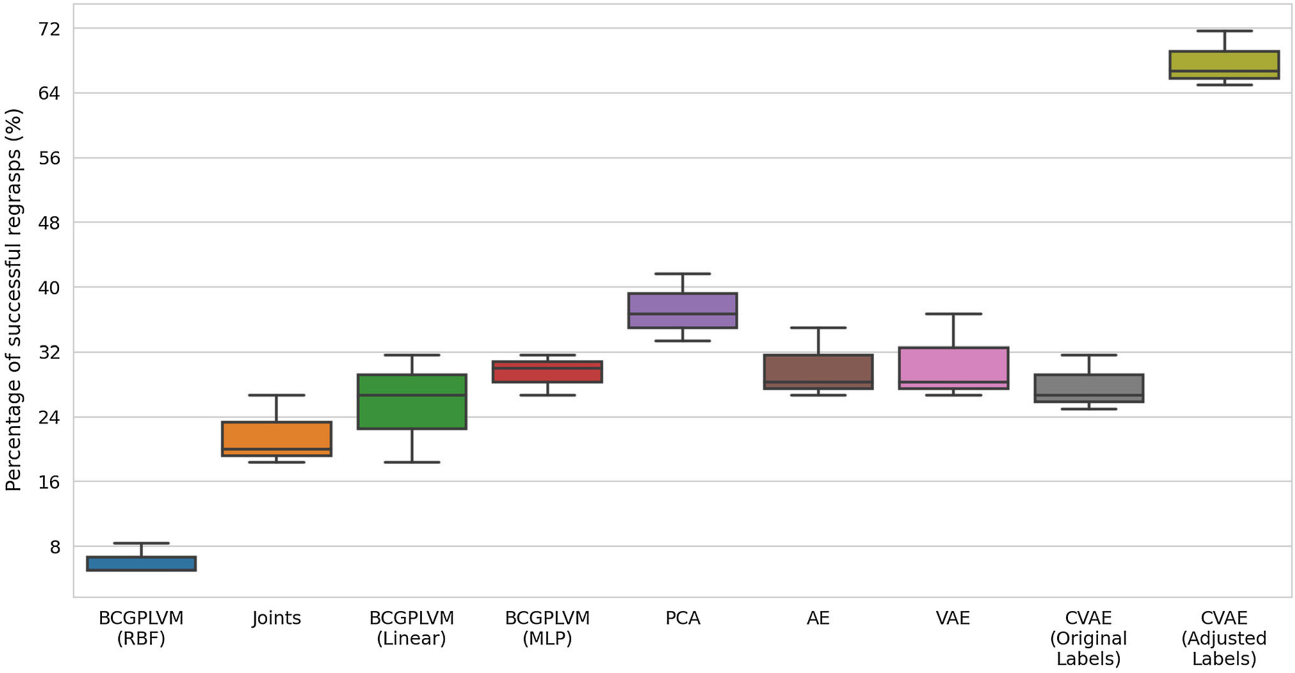


Fig. 8 Percentage of successful regrasp trajectories generated from each model

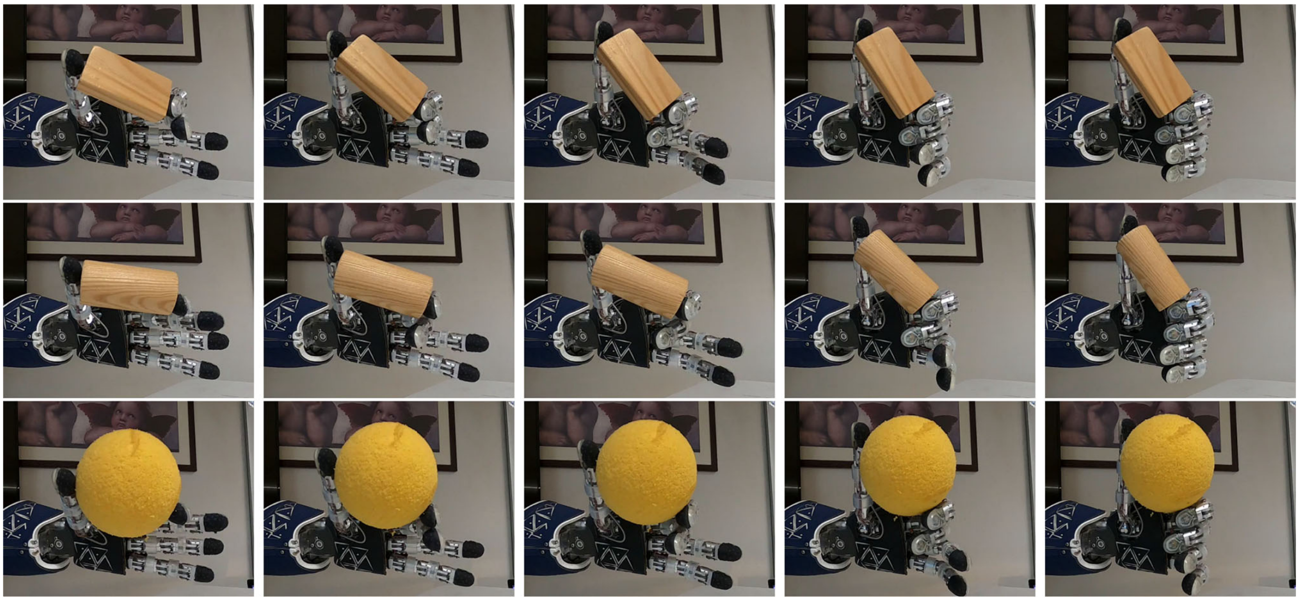


Fig. 9 Example regrasp trajectories executed on the iCub robot using four different objects. Each trajectory has ten steps. The pictures are taken every two steps. The first trajectory moves from a tripod grasp type to a lateral, the second from a tip pinch to a lateral, and the third from a tip pinch to a lateral

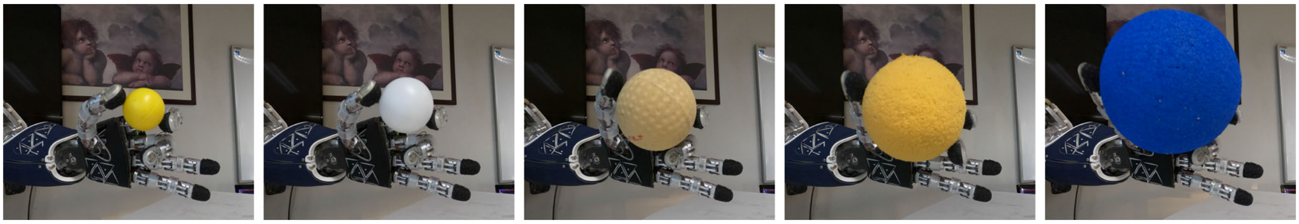


Fig. 10 Examples of size extrapolations. All grasps generated from the same latent point with conditional size values = {−2.0, −1.0, 0.0, 0.5, 1.0}

of the thumb and the index for each grasp. In Fig. 11, on the x axis we plot the value of the conditional size variable given to the network and on the y axis the distance between the fingertips of the generated grasp. The graph demonstrates that as the grasp size variable increases the distances between the fingertips also increases in an almost linear fashion. That indicates that the network learns to encode the relation between the grasp size variable and the fingertip distance. In Fig. 12, we show the average fingertip distance for each size value and the standard deviation. The variance present in the distances is a result of the differences between grasp types, for example in the tripod grasp the object is stabilised in the opposition created between the tips of the thumb and the index, while in the parallel extension grasps the object is stabilised in the opposition between the tip of the thumb and the distal link of the index. In addition, we tested this on a real-world experiment, where we chose a tip pinch grasp executed on a ball from the dataset, we encoded it into the latent space, and then decoded the produced latent point by varying the size label from −2.0 to 1.0. In Fig. 10, we can see that the model is able to generate grasps for very small objects without having seen this object size during training.

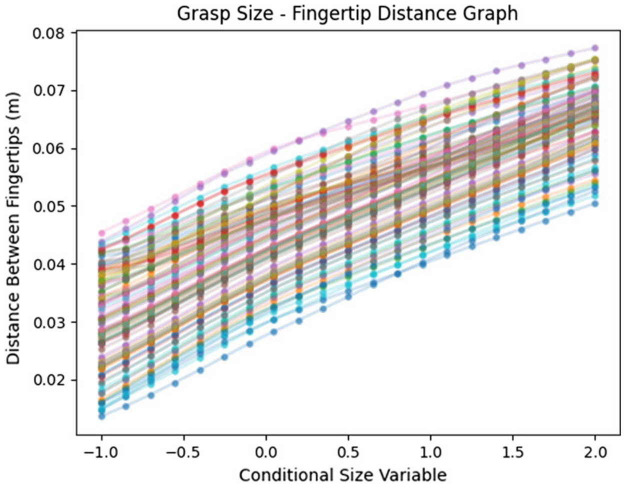


Fig. 11 Grasp size as a function of the conditional size variable

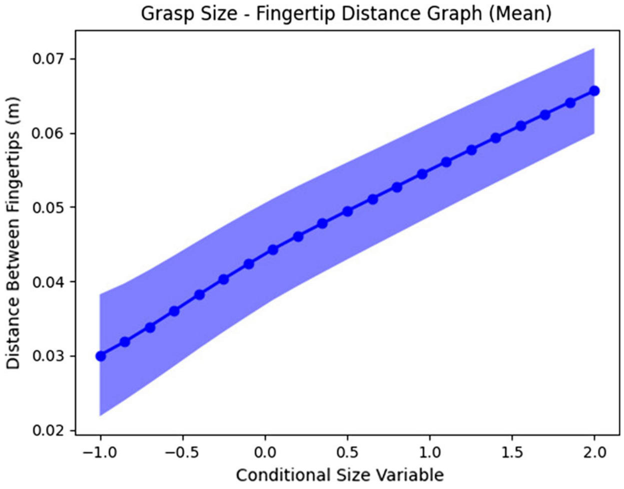


Fig. 12 Average grasp size as a function of the conditional size variable

6 Conclusions

In summary, we presented a conditional model based on the VAE framework for grasp generation and used it to generate trajectories directly in its latent space for in-hand regrasping. We also show that reducing the size labels during testing can avoid slippage during execution of the generated trajectories. We presented experiments that validate this approach as we were able to double the success rate of regrasp trajectories in a real world setting. Finally, we investigated the capabilities of the model to extrapolate on the size of the grasps that it generates.

Another line of research has explored the use reinforcement learning in in-hand manipulation tasks. In Chen et al. (2021) they train a policy using deep reinforcement learning to reorient objects to arbitrary orientations. They find that their system can perform the reorientation task and deal with novel objects without any visual information about the object's shape. In our work the problem is framed differently, as we want to regrasp the object using a specific grasp type but not explicitly reorient it or create new contacts with it, so it is not possible to directly compare both works because the objectives of each are different. In order for our model to be able to perform a similar behavior, we would need to acquire more data that have intermediate steps from the finger trajectories while reorienting the object and the object states at each step. In future work we plan to explore smarter ways to generate the trajectories in latent space, for example by taking advantage of the smoothness of the neighborhood of the latent space we can avoid regions that result in large changes in the hand configuration, and test the model on arbitrary objects with more complex shapes. In addition, we would like to explore generating trajectories that regrasp the object from different sides. This could be achieved by adding a continuous conditional variable that represents the side of the object that is being grasped from. This way we could smoothly interpolate from one side of the object to the other. It would also require a lot more training data points for the intermediate steps of the transition from one side to the other, with the finger gaits in each step. Finally another future research direction would be to add force feedback to the conditional model which could be used to automatically adjust the grasp size and generate hand configurations based on it.

Acknowledgements Work partially supported by the Robotics, Brain and Cognition Lab, part of the FCT - Portuguese Roadmap of Research Infrastructures [01/SAICT/2016 SAICT Proj. 22084], the LARSyS - FCT Project [UIDB/50009/2020], the H2020 FET-Open project Reconstructing the Past: Artificial Intelligence and Robotics Meet Cultural Heritage (RePAIR) under EU grant agreement 964854, the Lisbon Ellis Unit (LUMLIS), and the FCT PhD grant [PD/BD/09714/2020].

Funding Open access funding provided by FCT|FCCN (b-on).

Declarations

Compliance with Ethical Standards We have no conflicts of interest to disclose.

References

Asperti, A., & Trentin, M. (2020). Balancing reconstruction error and Kullback-Leibler divergence in variational autoencoders. *IEEE Access, 8*, 199440–199448.

Bernardino, A., Henriques, M., Hendrich, N., et al. (2013). Precision grasp synergies for dexterous robotic hands. In 2013 IEEE international conference on robotics and biomimetics (ROBIO), pp. 62–67, https://doi.org/10.1109/ROBIO.2013.6739436.

Chen, N., Karl, M., Smagt, PVD. (2016). Dynamic movement primitives in latent space of time-dependent variational autoencoders. In 2016 IEEE-RAS 16th international conference on humanoid robots (Humanoids) pp. 629–636.

Chen, T., Xu, J., Agrawal, P. (2021). A system for general in-hand object re-orientation. In: CoRL.

Ciocarlie, M., Goldfeder, C., Allen, P. (2007a). Dimensionality reduction for hand-independe.nt dexterous robotic grasping. In 2007 IEEE/RSJ international conference on intelligent robots and systems, pp 3270–3275, https://doi.org/10.1109/IROS.2007.4399227.

Ciocarlie, M.T., Goldfeder, C., Allen, P.K. (2007b). Dexterous grasping via eigengrasps : A low-dimensional approach to a high-complexity problem

Dimou, D., Santos-Victor, J., Moreno, P. (2021). Learning conditional postural synergies for dexterous hands: A generative approach based on variational auto-encoders and conditioned on object size and category. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 4710–4716, https://doi.org/10.1109/ICRA48506.2021.9560818.

Feix, T., bodo Schmiedmayer, H., Romero, J, et al. (2009). A comprehensive grasp taxonomy. In In robotics, science and systems conference: Workshop on understanding the human hand for advancing robotic manipulation.

Jenkins, OC. (2006). 2d subspaces for sparse control of high-dof robots. In 2006 international conference of the ieee engineering in medicine and biology society, pp. 2722–2725, https://doi.org/10.1109/IEMBS.2006.259857.

Katyara, S., Ficuciello, F., Caldwell, DG., et al. (2021). Leveraging kernelized synergies on shared subspace for precision grasp and dexterous manipulation. arXiv:2008.11574

Kingma, DP., Ba, J. (2014). Adam: A method for stochastic optimization. CoRR abs/1412.6980.

Kingma, DP., Welling, M. (2013). Auto-encoding variational bayes. CoRR abs/1312.6114.

Kramer, M. (1991). Nonlinear principal component analysis using autoassociative neural networks. *AIChE Journal, 37*, 233–243.

Lawrence, N. (2003). Gaussian process latent variable models for visualisation of high dimensional data. In: NIPS.

Lawrence, N., Candela, JQ. (2006). Local distance preservation in the GP-LVM through back constraints. In: ICML '06.

Oring, A., Yakhini, Z., Hel-Or, Y. (2021). Autoencoder image interpolation by shaping the latent space. In: ICML.

Palli, G., Ficuciello, F., Scarcia, U., et al. (2014). Experimental evaluation of synergy-based in-hand manipulation. *IFAC Proceedings Volumes, 47*, 299–304.

Romero, J., Feix, T., Ek, C. H., et al. (2013). Extracting postural synergies for robotic grasping. *IEEE Transactions on Robotics, 29*(6), 1342–1352. https://doi.org/10.1109/TRO.2013.2272249

Salvietti, G. (2018). Replicating human hand synergies onto robotic hands: A review on software and hardware strategies. *Frontiers in Neurorobotics, 12*, 27. https://doi.org/10.3389/fnbot.2018.00027

Santello, M., Flanders, M., & Soechting, J. F. (1998). Postural hand synergies for tool use. *Journal of Neuroscience, 18*(23), 10105–10115. https://doi.org/10.1523/JNEUROSCI.18-23-10105.1998

Santello, M., Bianchi, M., Gabiccini, M., et al. (2016). Hand synergies: Integration of robotics and neuroscience for understanding the control of biological and artificial hands. *Physics of Life Reviews, 17*, 1–23. https://doi.org/10.1016/j.plrev.2016.02.001

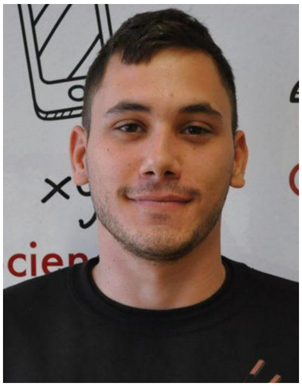
Sohn, K., Lee, H., & Yan, X., et al. (2015). Learning structured output representation using deep conditional generative models. In C. Cortes, N. D. Lawrence, & D. D. Lee (Eds.), *Advances in neural information processing systems* (pp. 3483–3491). Curran Associates Inc.

Starke, J., Eichmann, C., Ottenhaus, S., et al. (2018). Synergy-based, data-driven generation of object-specific grasps for anthropomorphic hands. 2018 IEEE-RAS 18th international conference on humanoid robots (Humanoids) pp. 327–333

Starke, J., Eichmann, C., Ottenhaus, S., et al. (2020). Human-inspired representation of object-specific grasps for anthropomorphic hands. *International Journal of Humanoid Robotics, 17*, 2050008.

Tsoli Odest, A., Jenkins, O. (2007). 2d subspaces for user-driven robot grasping. Robotics, Science and Systems Conference: Workshop on Robot Manipulation.

Xu, K., Liu, H., Du, Y., et al. (2016). A comparative study for postural synergy synthesis using linear and nonlinear methods. *International Journal of Humanoid Robotics, 13*(03), 1650009. https://doi.org/10.1142/S0219843616500092



Dimitrios Dimou is a Ph.D. candidate in Electrical and Computer Engineering at Instituto Superior Técnico, Universidade de Lisboa. He received his M.Sc in Electrical and Computer Engineering from the University of Patras, Greece in 2018. His research interests are in the areas of robotic grasping and manipulation using machine learning methods.

José Santos-Victor is a full Professor of Electrical and Computer Engineering, Instituto Superior Técnico, Universidade de Lisboa. He is the President of the Institute for Systems and Robotics|Lisboa and the Coordinator of LARSyS (Laboratory of Robotics and Engineering Systems) that includes ISR|LISBOA and three other research units (M-ITI, IN+ and MARETEC). He founded the Computer and Robot Vision Lab - VisLab at ISR|Lisboa. He graduated 21 Ph.D. students. José's research interests are in the areas of Computer and Robot Vision, particularly in the relationship between visual perception and the control of action, biologically inspired vision and robotics, cognitive vision and visual controlled (land, air and underwater) mobile robots. Recent research has a focus on biologically inspired models of human(oid) cognition, through the creation of artificial models of human(oid) cognition. He explored the neuroscientific findings of the Mirror Neurons to propose models that use motor information for visual action recognition; and the concept of Gibsonian Affordances drawn from psychology for building advanced cognitive skills in humanoid-robots. He was the scientific responsible for the participation of IST/ISR in 10+ European Projects in the areas of Computer Vision and Robotics, (e.g. MIRROR, CONTACT, ROBOSOM, ROBOTCUB, FIRST-MM, POETICON+), and the current EU-FET-Pathfinder project "REPAIR- AI-and-Robotics Meet Cultural-Heritage".

Plinio Moreno received a B.Sc. in Mechanical Engineering, B.Sc. in Computer Science and M.Sc. in Computer Science from the Universidad de los Andes (Bogotá, Colombia) in 1998, 2000 and 2002 respectively. He completed the Ph.D. degree in Electrical and Computers Engineering at the Instituto Superior Técnico (IST, Lisbon, Portugal) in 2008. During his Ph.D. studies, P. Moreno was holder of a Portuguese Science Foundation (Fundação para a Ciência e Tecnologia) grant (SFRH/BD/10753/2002). P. Moreno published a total of 61 papers in top journals and conferences in the areas of Computer Vision and Pattern Recognition (Pattern Recognition Letters), Artificial Intelligence (Neurocomputing), and Robotics (Autonomous Robots, Robotics and Autonomous Systems, IROS and ICRA). Of the 61 articles, 15 were published in international peer review journals and the remaining 46 at international peer review conferences. Since 2016, P. Moreno co-authored 10 articles in journals and 17 articles at conferences. P. Moreno's global citation indexes h-index and i10-index are currently 16 and 26 according to Google Scholar, and since 2017 h-index is 13 and i10-index is 20. According to Scopus his h-index is 10.