Mediterranean Conference on Control and Automation (MED 2024)

An Empirical Investigation on Variational Autoencoder-Based Dynamic Modeling of Deformable Objects from RGB Data

Tomás Coleman[1], Robert Babuška[1,2], Jens Kober[1], Cosimo Della Santina[1,3]

***Abstract*— Formulating the dynamics of continuously deformable objects and other mechanical systems analytically from first principles is an exceedingly challenging task, often impractical in real-world scenarios. What makes this challenge even harder to solve is that, usually, the object has not been observed previously, and the only information that we can get from it is a stream of RGB camera data. In this study, we explore the use of deep learning techniques to solve this non-linear identification problem. We specifically focus on extracting dynamic models of simple deformable objects from the high-dimensional sensor input coming from an RGB camera. We investigate a two-stage approach to achieve this goal. First, we train a variational autoencoder to extract an extremely low-dimensional representation of the object configuration. Then, we learn a dynamic model that predicts the evolution of these latent space variables. The proposed architecture can accurately predict the object's state up to one second into the future.**

I. Introduction

Agri-food, disaster response, and manufacturing industries are examples of sectors that involve physically demanding tasks with highly repetitive actions, often involving deformable objects. Nowadays, the robotics literature has almost exclusively focused on the manipulation of garments or other objects with low mass, low compressive stiffness, and high friction-to-inertia ratio [1]–[3]. These objects can be described using purely geometric descriptions developed under quasi-static assumptions [4]–[6]. Still, many objects that are commonly encountered in the above-mentioned scenarios do not fulfill these hypotheses. This paper will refer to failure to verify one or more of these characteristics as having *non-negligible physical response*. With these objects, a geometric representation alone becomes inadequate, necessitating the inclusion of dynamic response in the model.

Formulating accurate models that describe the behavior of these objects is extremely difficult as it relies on continuum mechanics [7]. Numerical techniques like FEM or discrete rod formulations can be used to generate approximate finite-dimensional models [8]–[12]. However, these methods rely on precise knowledge of the object's geometry and material properties, which are often unavailable in advance. Moreover, FEM comes with substantial computational costs.

Machine learning techniques present a promising alternative to first principle modeling as demonstrated by a rich and active body of literature in nonlinear identification [13]–[17]. Learning can also be used to directly regress manipulation policies for deformable objects [18]–[26]. Yet, all these works focus exclusively on objects with negligible physical response.

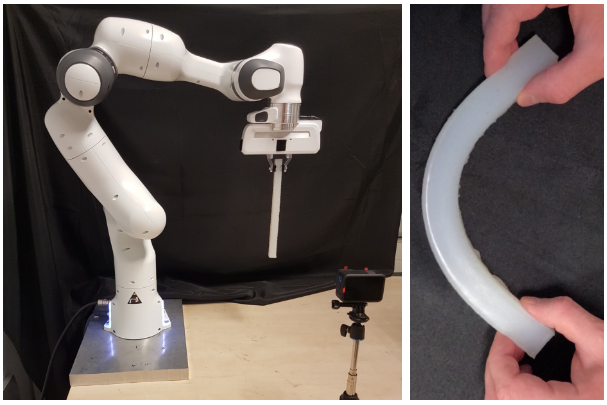

Fig. 1. Robotic systems need access to precise models to predict the dynamic response of the objects they manipulate. In this work, we investigate the use of deep learning to achieve this goal. We demonstrate the architecture on an experimental setup comprising soft silicone objects held by a Franka Emika Panda robot arm and a Go Pro camera as the only source of data.

Outside robotics, the deep learning and control communities have made notable strides in learning low-dimensional representations of high-dimensional data streams [27], [28], with applications spanning fluid dynamics [29], climate systems [30], aerospace [31], and bio-mechanics [32]. However, despite these advancements, limited work has been done to learn continuum mechanical systems as deformable objects. Among those, [33] specifically focuses on representing deformability in faces, hands, and clothes, while [34] tackles generic kinematic representations. None of these works focus on learning the dynamics. Recent publications still focus exclusively on garments [35], [36]. Works considering more general objects [37]–[40] assume an existing working model derived from first principles and only focus on compressing its dimensionality. Furthermore, none of these techniques have been experimentally validated, highlighting the need for empirical verification in this domain.

To summarize, to the authors' knowledge, no work in literature focuses on learning the dynamics of objects with non-negligible physical responses, nor do they experimentally test their results. This paper aims to make a first step towards filling both gaps. We focus on learning the dynamics of deformable objects like the one in Fig. 1 while moving passively as they are constrained in a robotic manipulator. Fig. 2 shows a sketch of the proposed neural architecture. We find that to learn the passive dynamics of these simple deformable objects in the latent space, we need a smooth, monotonic mapping of the configuration from the feature space to the

This work is supported by the EU EIC project EMERGE, grant number 101070918. [1]Department of Cognitive Robotics, Delft University of Technology, Building 34, Mekelweg 2, 2628 CD Delft, Netherlands. Email: {t.coleman, r.babuska, j.kober, c.dellasantina}@tudelft.nl. [2] Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University in Prague, 16000 Prague, Czech Republic [3]Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Wessling, Germany.

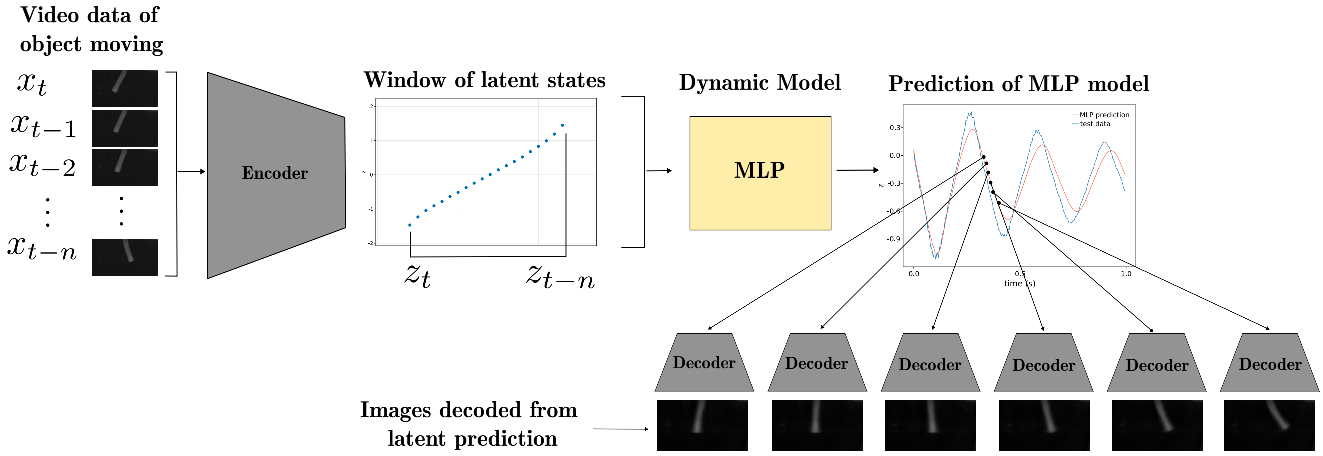

Fig. 2. A view of the overall neural architecture (from top left to bottom right). Thanks to the regularization imposed by the variational mechanism, the encoder can extract a one-dimensional smooth representation of the object configuration. Based on that, we show that a simple MLP architecture is sufficient to predict the evolution accurately in latent space from n past images. Finally, the prediction is passed through the decoder, yielding predicted images.

latent space. We show that such a representation can be achieved using the method of variational autoencoders [41]. Remarkably, with the proposed pipeline, we can describe the configuration of these albeit simple but theoretically infinite dimensional objects with just *one* configuration variable.

Note: All the code, model architectures, training parameters and more results can be found in the following GitHub repository https://github.com/moss-coleman/Learning-Low-Dimensional-Representations-for-Deformable-Objects

II. Proposed Learning Architecture

This section will discuss the proposed approach for learning a low-dimensional dynamic representation of infinite-dimensional deformable objects[1].

A. Assumptions on the setup

We assume to have a stream of high-dimensional sensor information of the deformable objects so as to capture their theoretically infinite dimensionality. We use grayscale images to this end, captured by a camera placed in front of the gripper holding the objects, as shown in Fig. 1. An example of some samples from these input data streams is shown in Fig. 10.

B. Architecture at a glance

The overall architecture is summarized in Fig. 2 and articulated in two steps. The first is about extracting a very low dimensional representation of the configuration space (Fig. 3). It is worth noting now that due to its low dimensionality, this encoding needs to be *smooth enough* to allow for learning of the dynamics. We will thoroughly test this claim later in the paper. In the second step, we show how we can train a neural network as a dynamic model in this latent space using the latent variables as the coordinates of the system.

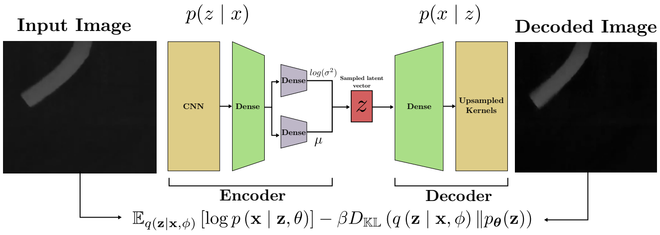
Input Image
$p(z \mid x)$
CNN
Dense
Dense
Dense
$log(\sigma^2)$
μ
Sampled latent vector
z
$p(x \mid z)$
Dense
Upsampled Kernels
Decoded Image
Encoder
Decoder
$\mathbb{E}_{q(\mathbf{z}|\mathbf{x},\phi)}\left[\log p\left(\mathbf{x} \mid \mathbf{z},\theta\right)\right] - \beta D_{\mathbb{KL}}\left(q\left(\mathbf{z} \mid \mathbf{x},\phi\right) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right)$

Fig. 3. Architecture of the Variational autoencoder framework, highlighting the unsupervised nature of the training process.

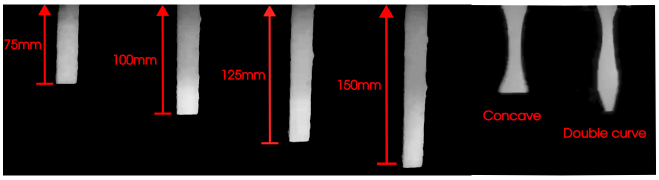

Fig. 4. The soft objects used in the experiments consisted of rectangular objects of width 20mm, thickness 15mm, and lengths of 75mm, 100mm, 125mm, and 150mm. Two more objects of varied geometry were tested, referred to as the "Concave" and "Double curve" objects.

C. Learning a latent representation

1) Autoencoder: We employ an autoencoder architecture [42] to map the grayscale image data to a lower dimensional latent space representation ($x \in \mathbb{R}^D \rightarrow z \in \mathbb{R}^L$). The aim is to retrieve a representation of the configuration of the object, thus replicating in an automatic and purely data-driven fashion the heuristic process performed based on expert intuition in the deformable mechanics control literature [43], [44].

The autoencoder is composed of an encoder function, $f_{\mathrm{e}}(x;\theta_{\mathrm{e}}) = z_n$ and decoder function $f_{\mathrm{d}}(z;\theta_{\mathrm{d}}) = \hat{x}$, where θ_{e} and θ_{d} are the parameters of the encoder and decoder functions respectively. The parameters of the autoencoder are trained by minimizing a reconstruction loss of the reconstruction function $r(x) = f_{\mathrm{d}}(f_{\mathrm{e}}(x))$ for example, $\mathcal{L}(\theta_{\mathrm{e}},\theta_{\mathrm{d}}) = ||r(x) - x||_2^2$ or $\mathcal{L}(\theta_{\mathrm{e}},\theta_{\mathrm{d}}) = -\mathrm{log}p(x|r(x))$.

2) Smooth mapping into latent space: To learn the dynamics of a deformable object in the latent space, the latent space is required to be smooth and have a strictly monotonic

[1] These are the soft pendula in Fig. 4.

evolution. This strictly monotonic condition implicitly gives a unique, invertible mapping of the object from the feature space to the latent space. While the network of the encoder is a continuous function, it does not guarantee that the latent representation of an image of the object in one state in the feature space, $f_{\mathrm{e}}(x_n) \rightarrow z_n$, will be close to and in the same order as the states $f_{\mathrm{e}}(x_{n+1}) \rightarrow z_{n+1}$ and $f_{\mathrm{e}}(x_{n-1}) \rightarrow z_{n-1}$. We do not assume to have an ordering of data in the training set that can be used to enforce monoticity. Instead, we favour smoothness to penalize abrupt changes in mapping. We propose to do that by using Variational autoencoders (VAE) [41]. Fig. 3 summarizes our implementation of the architecture. VAEs represent the latent space as random variables with a Gaussian probability distribution, using two encoder networks. One, the μ-encoder, estimates the mean of the random variable. The other encoder, the logvar-encoder, estimates the variance. Both encoders share the weights of the first number of layers. To train the VAE, the evidence lower bound is minimized through the following loss function

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

The first term weights the expected log-likelihood for the reconstruction, while the second term, known as the Kullback–Leibler divergence, weights the distance between the posterior distributions. The mechanism in the training process that allows an estimate of the posterior distribution is re-sampling the latent variable based on the estimated variance of that variable. We believe that this mechanism can serve as a regularisation term for learning the monotonic state representation of a dynamical system. This is because resampling around the estimated latent variable can give some sense of relative position and order to the configuration images taken for a training set.

In our examples, we will manually order the images to test whether the latent representation is monotonic. This information is, however, not available to the algorithm in the training phase. To investigate the effect the Kullback-Leibler divergence measure in the loss function has on discovering the order from feature space to latent space based on the configuration of the object, we use the β variation of the VAE [45]. The β-VAE adds a weighting parameter, β, proportional to the Kullback-Leibler divergence.

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}\right) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x},\phi)}\left[\log p\left(\mathbf{x} \mid \mathbf{z}, \theta\right)\right] \\ &- \beta D_{\mathbb{KL}}\left(q\left(\mathbf{z} \mid \mathbf{x}, \phi\right) \| p_{\boldsymbol{\theta}}(\mathbf{z})\right)\end{aligned} \tag{2}$$

As the data is in the form of a grid of pixels, we use convolutional layers at the beginning of the encoder and convolutional transpose layers to upsample to the dimension of the input image in the decoder.

3) Sizing of the latent space: With our experimental investigation, we want to test the ability of this architecture to compress the information from high-dimensional data to a latent space that has the minimum dimension that maintains a reasonably high reconstruction accuracy. This helps with generalizability, interpretability, and performing analysis of the dynamic system in this space - and it is coherent with the research in heuristic reduced order modeling discussed above. To examine the ability of the autoencoder to give a minimal representation of the state of the system, we investigate the extreme case of a latent space with dimension compressed down to *one*. Note that this extreme choice is motivated by results in expert-driven model compression in soft robotics [46].

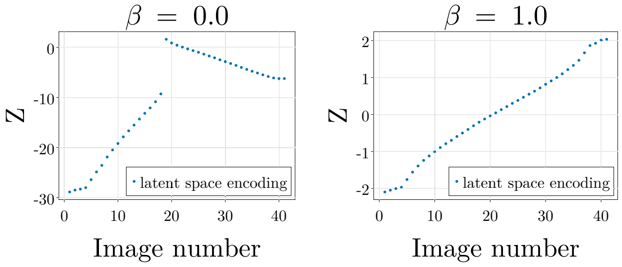

Fig. 5. Mapping of training images to latent space after training VAE on 41 images of the soft object at 100mm, with values of β in the loss function of 0.0, and 1.0.

D. Learning latent Dynamics

With the latent representation of the deformable object as it deforms through a range of states of interest, smooth and monotonic, the passive dynamics of the object can then be estimated from video footage of the object in motion. We use video data transformed to the latent space to generate the time series data to train on.

To learn the dynamics, we train a fully connected Multi-layer perceptron (MLP) on the time series data gathered from video of the soft object passively moving from random initial conditions. Note that mechanical systems are second-order systems, i.e., the state is composed of position and velocity. The latent space variable z is a representation of the configuration. To allow the model to estimate velocity $\dot{z}$, the input nodes of the MLP are given multiple state measurements. This includes the current state $z(t)$ with a window of multiple state measurements in the past, n, i.e., $z_t, z_{t-1}, z_{t-2}, \ldots, z_{t-n}$. The output of the network will predict the state of the system in the latent space one-time step into the future, z_{t+1}. This can be seen as a discrete dynamical system as there is a constant time difference, δt, between the frames of the video encoding to the latent space. The following predictive model results

$$z_{t+1} = f_{\mathrm{mlp}}(z_t, z_{t-1}, \ldots, z_{t-n}; \theta_{\mathrm{mlp}}) \tag{3}$$

To predict multiple time steps into the future, the predicted value of z_{t+1} can be used instead of z_t. The window of values shifts one step forward, where z_{t-n} is no longer input to the network, and z_{t-n+1} is the last input into the network. Then, the network predicts z_{t+2}, etcetera.

E. Hallucinating future images

Latent space predictions can then be mapped back into the space of images by applying the decoder f_{e} to z_t. This way, $f_{\mathrm{e}}(z_t), f_{\mathrm{e}}(z_{t+1}), f_{\mathrm{e}}(z_{t+2}), \ldots$ is effectively a stream of predicted images of the objects that will be recorded in the future. This process is pictorially represented in the bottom right of Fig. 2.

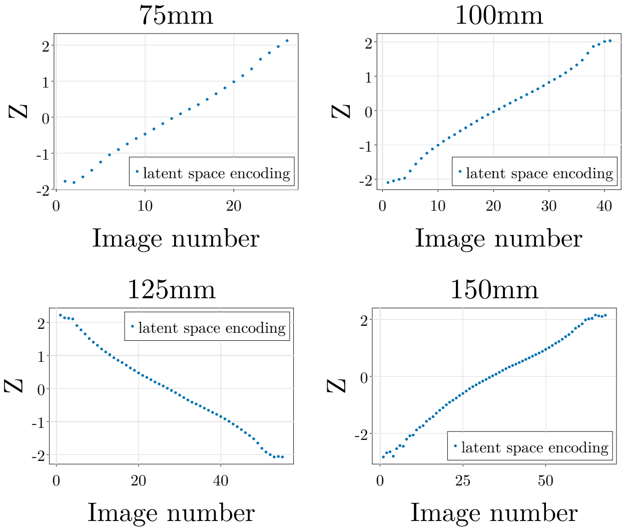

Fig. 6. Mapping of training images from objects of length 75mm, 100mm, 125mm, and 150mm to the latent space, showing desirable smooth monotonic properties.

III. Experimental Setup and Training Details

In this section, after discussing the experimental setup, we will show how the VAE was trained and then its performance in representing the state of the soft object in the latent space. Then, we will show how a neural network was trained to learn the dynamics of time series data that was mapped into the latent space from video data using the encoder of the VAE.

A. Experimental setup

The experimental setup is shown in Fig. 1. To perform the experiments, the soft object is grasped by the parallel gripper of a Franka Emika robot manipulator. To isolate the method from background noise in the images of the object, we place a black homogeneous curtain as a background. The experiments are conducted by initializing the object at random configurations and then allowing the object to swing under its own passive dynamics. The video data is gathered from a Go Pro Hero10 camera at 240 frames per second frame and a resolution of 360x640 pixels. These images are then compressed to a resolution of 36x64 and grey-scaled to reduce the size of the network needed. The objects used in the experiments are shown in Fig. 4. They are made from silicone and, given their geometry, have non-negligible inertia compared to their stiffness, which is seen in their significant passive dynamic response while clamped in the parallel gripper. For each deformable object of length 75mm, 100mm, 125mm, and 150mm, there were 26, 41, 55, and 68 images, respectively, for the data to train the VAE. For the Concave and Double curve objects, there were 55 and 21, respectively. The number of images sampled is due to the varying time it takes each object to swing through a similar bending angle. For the video data to train the dynamic model with latent space configuration coordinates, there were 5 trials of initializing the object and recording until the velocity was close to null at 240 fps. From the five trials taken, four were used for training, and one was used as the test data.

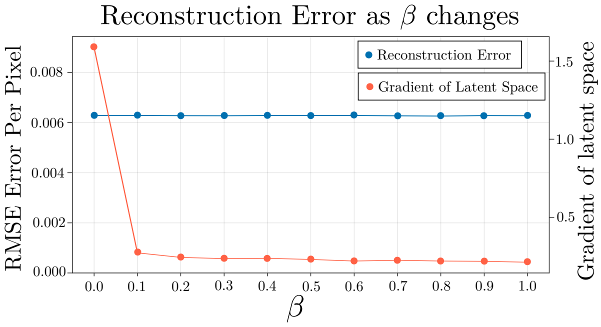

Fig. 7. Results of varying the β parameter on the reconstruction loss and on the level of smoothness. The latter is measured as the maximum absolute value of the gradient of the training-images-number vs z profiles. See Fig. 5 for examples.

B. Training the VAE

A sample of the images used to train the VAE is shown in the top row of Fig. 10 for a soft object of length 100mm. To gather these images in the states of interest, the object was given an initial condition and then allowed to passively swing through the range of motion. The series of images from the initial condition to the next time the object reaches zero velocity is then used as the training data for the VAE. The VAE model is composed of an Encoder and a Decoder. The Encoder consists of 3 convolutional layers *image size* $\rightarrow$ *32* $\rightarrow$ *32* $\rightarrow$ *32*, followed by a Dense fully connected layer of size *1024* $\rightarrow$ *256* and then two heads of size *256* $\rightarrow$ *1* for the μ and σ estimation. The convolutional layers all have a filter size of 4 and a stride of 2. The Decoder is the inverse of the Encoder, as it up samples the latent space value to the size of the original image through two Dense fully connected layers of size *1* $\rightarrow$ *256* $\rightarrow$ *1024*, then 3 up sampling layers of size *1024* $\rightarrow$ *32* $\rightarrow$ *32* $\rightarrow$ *image size*. A *tanh* activation function is used throughout. The hyperparameters used for training are as follows: learning rate is 10^{-4}, optimizer is ADAM, regularisation λ = 0.01, and $20,000$ epochs.

C. Training the dynamic model

With an encoder trained that can map the pixel data to the latent space, we use the μ-encoder to map video recordings of the soft object moving passively from random initial conditions. To construct the training set, we split the data collected in Sect. III-A up into a training and test split, using 4 of the experiments as the training data and one as the test set. This time series data is then used to train the MLP model discussed in Sect. II-D. The MLP consists of a fully connected Multi-Layer Perceptron, with 5 layers of size *20* $\rightarrow$ *20* $\rightarrow$ *20* $\rightarrow$ *20* $\rightarrow$ *20* $\rightarrow$ *1*. A *tanh* activation function is used. The hyperparameters used for training are as follows: the learning rate is 10^{-5}, the optimizer is ADAM, and the learning goes on for 30,000 epochs. For input data to the model, the time series data in the latent space was divided into the samples $z_t, z_{t-1}, \ldots, z_{t-n}$ and the corresponding z_{t+1} in the time series as the prediction output data. The data is then randomly shuffled in the training.

IV. Results and Discussion

In this subsection, we especially focus on the extreme case of latent space of dimension *one*. Still, we will also present

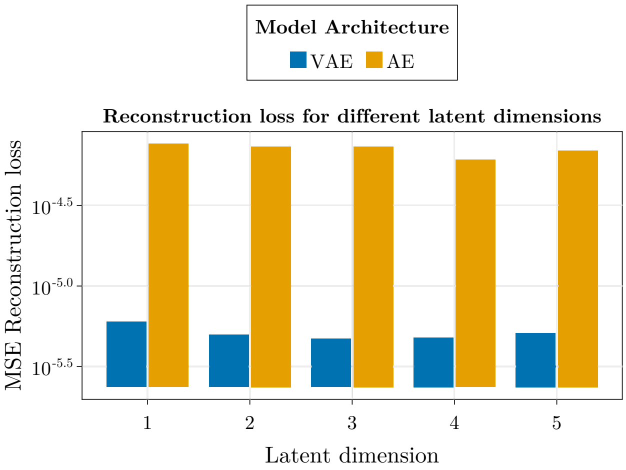

Fig. 8. The comparison of the per pixel MSE error of the reconstruction performance, between VAE and standard AE, as the dimension of the latent space increases from 1-5

results for higher dimensional latent spaces as ablation studies.

A. Auto-encoders reconstruction capabilities

1) Qualitative analysis of latent space smoothness: In the experiment, the μ-encoder maps selected training images of the swinging object in Fig.1 to latent space, as shown in Fig.5. The ordinate plots the latent coordinate z. Each panel represents a different β value in the loss function. The aim is to assess the smoothness of the latent space. Results show that for $\beta = 1.0$, the latent space is smooth and monotonic. In contrast, a non-variational architecture ($\beta = 0.0$) leads to a discontinuous, non-monotonic latent space. We will see later that this characteristic will result in poor performance when learning the dynamics.

Similarly, we report the results of a similar experiment when training the VAE with $\beta = 1.0$ on the data sets collected for the length of objects mentioned in Sect. III-A. For the object with lengths of 75mm, 100mm, 125mm, and 150mm, the resulting latent space of training the VAE to encode the representation from the state training data is shown in Fig. 6. In all cases, the characteristics are smooth and monotone, confirming the soundness of the approach.

2) Performance when varying β: To assess the impact of varying the β parameter on decoder reconstruction quality, we trained the VAE on a 100mm dataset with β ranging from 0.0 to 1.0 in 0.1 steps. We used per-pixel RMSE as our metric. Fig. 7 shows that β has no effect on reconstruction accuracy within this range. Thus, increasing β to smooth the latent space incurs no loss in accuracy. Furthermore, this analysis also aligns with our qualitative findings discussed in the previous subsection, where the maximum gradient sharply drops between $\beta = 0$ and 0.1 and continues to decrease marginally with higher β values.

3) Performance when varying the latent space dimension: We also tested what the effect of the dimension of the latent space had on the loss (1) during training. For this, we used the concave data set. As can be seen from the results in Fig. 8, increasing the size of the latent space does not significantly improve the reconstruction accuracy of either the VAE or AE.

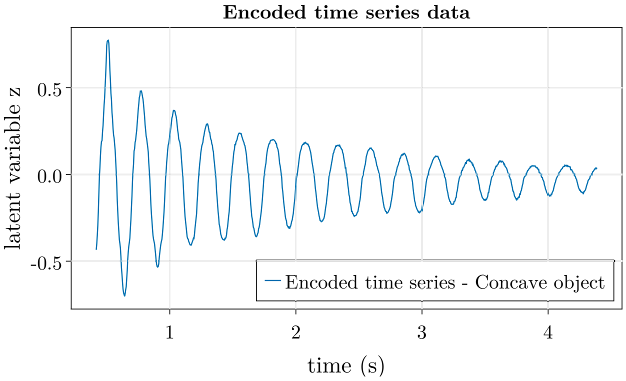

Fig. 9. Example of 4.5 seconds of video data encoded into the latent space via the AE with variational smoothing.

4) Qualitative remark on the latent space evolution: Using the μ-encoder, Fig. 9 shows latent space values of the training video data captured at 240 fps for 4.5 seconds. Remarkably, this evolution is the one we would expect from the unforced response of a one-dimensional mechanical system, i.e., damped oscillations converging quasi-exponentially to a steady state.

B. Dynamic model performance

1) Qualitative display of image prediction capabilities: Fig. 10 displays a qualitative comparison between the MLP-decoder predictions (bottom) and the test set images (top). Notably, the prediction horizon length doesn't seem to affect the decoder's image quality, sidestepping a common issue in learning dynamics within latent spaces [47].

2) Latent space prediction capabilities: In this evaluation, the MLP is initialized with the latent representation of 20 consecutive images and predicts the next 240 time steps (1 second). Fig. 11 shows the model's prediction for objects of varying lengths and shapes, compared to video images in the latent space at those times. The initial prediction is highly accurate due to the use of ground truth values. Subsequently, the model maintains reasonable accuracy and phase consistency, the latter being particularly important for a second-order mechanical system.

3) Effect of a non-smooth representation: To highlight the encoder smoothness's role in learning latent space dynamics, we conducted an ablation using a μ-encoder trained with $\beta = 0$, as depicted in Fig.5. We encoded 100mm object video data similarly to the smooth encoder (Fig.11). Test results (Fig. 12) show that this model, trained on discontinuous time series, performs poorly in predictions compared to the test data.

4) Performance when varying the latent space dimension: Fig. 13 compares the reconstruction accuracy of predicted images to real video test data for increasing the dimension of the latent space. We observe a marginal reduction of error for the VAE but not for the AE. Moreover, using an AE together with the MLP always leads to a significant rise in prediction error. This suggests that smooth latent mapping is not just beneficial for accuracy; it's essential for method feasibility.

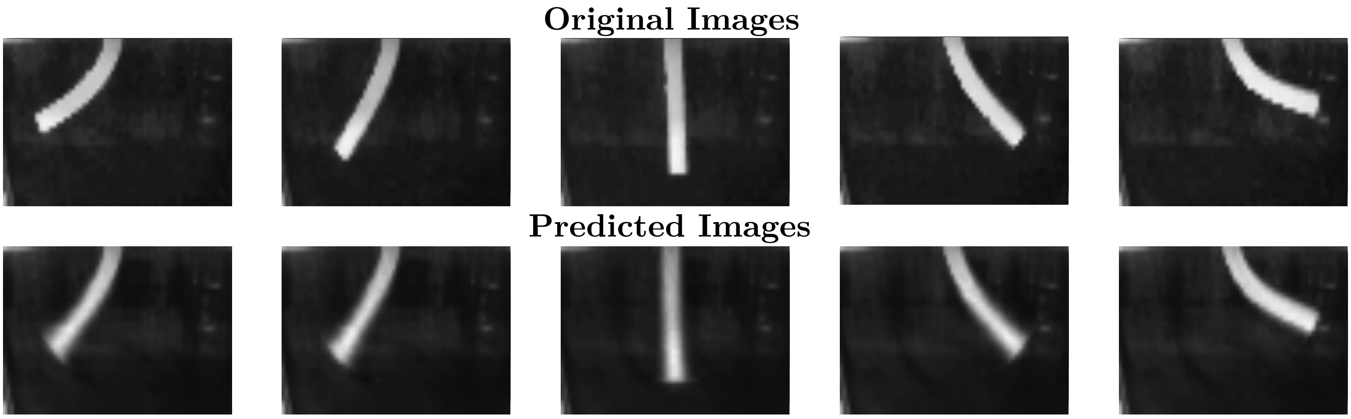

Fig. 10. Comparison between observed and predicted images. (Top) 5 grey scaled images of the soft object, sampled at 240 fps, from the test data. (Bottom) MLP model initiated at the previous time window from the start of the top images, and its predicted state at the same time step as the above image, is put through the decoder to give the image predictions shown.

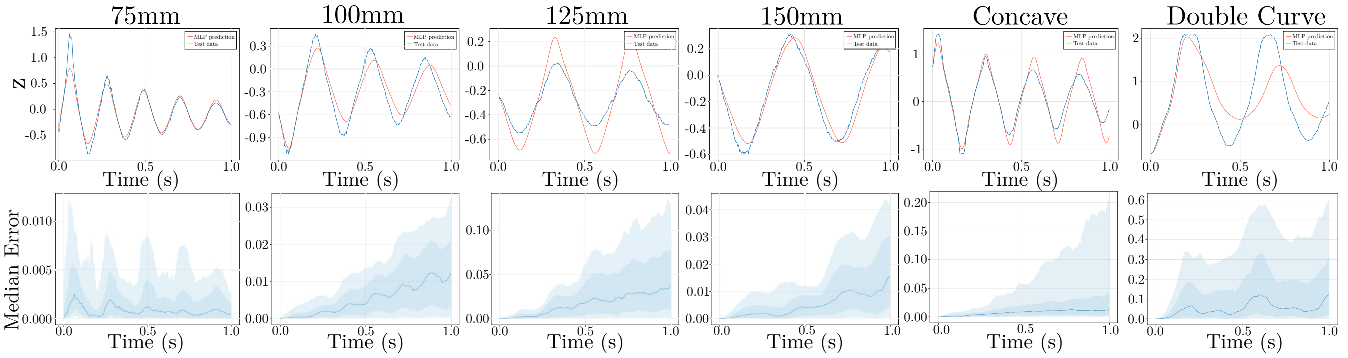

Fig. 11. (Top) Example predictions of the MLP compared to the test data over a one second horizon. (Bottom) Median prediction error of the MLP model measured against test data, for the rectangular object at 75mm, 100mm, 125mm, 150mm, concave and double curve respectively. The first band is the quantile range 25%-75%, second band 10%-90% range.

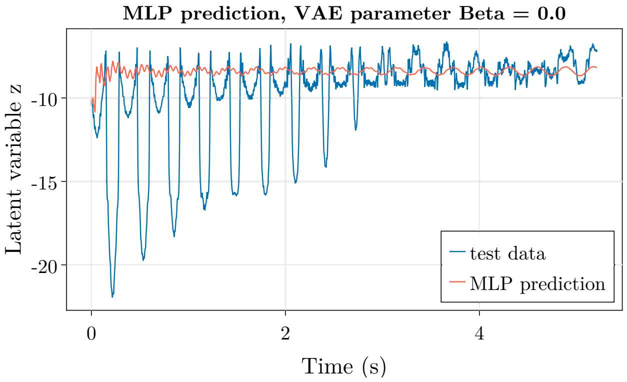

Fig. 12. Results of a dynamic model trained with time series data from a discontinuous encoder representation of the latent space.

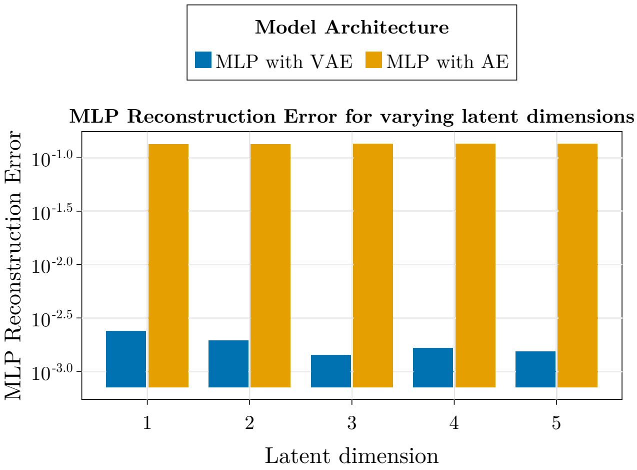

Fig. 13. A comparison of the performance of the learned dynamic predictive model, as the dimension of the latent space increases from 1-5, for data encoded with both the VAE and standard AE

V. Conclusion

This paper investigated learning extremely low-dimensional dynamical representations of theoretically infinite dimensional deformable objects. In opposition to the existing literature in robotics, we focus on objects with a non-negligible dynamic response. We show experimentally that a latent space of dimension *one* can be sufficient to describe the configuration of these mechanical systems, and that using variational autoencoders can be the key to obtaining a smooth representation that lends itself to serve as the base for learning the dynamics in the latent space. Future work will focus on enabling dexterous manipulation of deformable objects by combining these learned models with model-based control techniques [48].

References

[1] V. E. Arriola-Rios, P. Guler, F. Ficuciello, D. Kragic, B. Siciliano, and J. L. Wyatt, "Modeling of deformable objects for robotic manipulation: A tutorial and review," *Front. Robotics AI*, vol. 7, 2020.

[2] H. Yin, A. Varava, and D. Kragic, "Modeling, learning, perception, and control methods for deformable object manipulation," *Sci. Robotics*, vol. 6, no. 54, p. eabd8803, 2021.

[3] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li, *et al.*, "Challenges and outlook in robotic manipulation of deformable objects," *IEEE Robotics & Autom. Mag.*, vol. 29, no. 3, pp. 67–77, 2022.

[4] F. Coltraro, J. Amorós, M. Alberich-Carramiñana, and C. Torras, "An inextensible model for the robotic manipulation of textiles," *Appl. Math. Model.*, vol. 101, pp. 832–858, 2022.

[5] A. Longhini, M. Moletta, A. Reichlin, M. C. Welle, A. Kravberg, Y. Wang, D. Held, Z. Erickson, and D. Kragic, "Elastic context: Encoding elasticity for data-driven models of textiles," 2022.

[6] R. Laezza and Y. Karayiannidis, "Learning shape control of elastoplastic deformable linear objects," in *2021 IEEE Int. Conf. on Robotics Autom. (ICRA)*, pp. 4438–4444, 2021.

[7] M. Gazzola, L. Dudte, A. McCormick, and L. Mahadevan, "Forward and inverse problems in the mechanics of soft filaments," *Royal Soc. open science*, vol. 5, no. 6, p. 171628, 2018.

[8] M. Bessa, K. Elkhodary, W. Liu, T. Belytschko, and B. Moran, *Nonlinear Finite Elements for Continua and Structures, Second Edition. Solution Manual.* 01 2013.

[9] A. Petit, V. Lippiello, G. A. Fontanelli, and B. Siciliano, "Tracking elastic deformable objects with an rgb-d sensor for a pizza chef robot," *Robotics Auton. Syst.*, vol. 88, pp. 187–201, 2017.

[10] F. Ficuciello, A. Migliozzi, E. Coevoet, A. Petit, and C. Duriez, "Fem-based deformation control for dexterous manipulation of 3d soft objects," in *2018 IEEE/RSJ Int. Conf. on Intell. Robots Syst. (IROS)*, pp. 4007–4013, IEEE, 2018.

[11] P. Schegg and C. Duriez, "Review on generic methods for mechanical modeling, simulation and control of soft robots," *Plos one*, vol. 17, no. 1, p. e0251059, 2022.

[12] L. Besselaar and C. Della Santina, "One-shot learning closed-loop manipulation of soft slender objects based on a planar polynomial curvature model," in *2022 IEEE 5th Int. Conf. on Soft Robotics (RoboSoft)*, pp. 518–524, IEEE, 2022.

[13] G. P. Liu, *Nonlinear identification and control: a neural network approach*. Springer Science & Business Media, 2012.

[14] O. Nelles and O. Nelles, *Nonlinear dynamic system identification*. Springer, 2020.

[15] K. Yamada, I. Maruta, and K. Fujimoto, "Subspace state-space identification of nonlinear dynamical system using deep neural network with a bottleneck," *IFAC-PapersOnLine*, vol. 56, no. 1, pp. 102–107, 2023.

[16] J. Weigand, M. Deflorian, and M. Ruskowski, "Input-to-state stability for system identification with continuous-time runge–kutta neural networks," *Int. J. Control*, vol. 96, no. 1, pp. 24–40, 2023.

[17] J. Liu, P. Borja, and C. Della Santina, "Physics-informed neural networks to model and control robots: a theoretical and experimental investigation," *Adv. Intell. Syst. press*, 2024.

[18] M. Q. Mohammed, K. L. Chung, and C. S. Chyi, "Review of deep reinforcement learning-based object grasping: Techniques, open challenges, and recommendations," *IEEE Access*, vol. 8, pp. 178450–178481, 2020.

[19] A. Cherubini, V. Ortenzi, A. Cosgun, R. Lee, and P. Corke, "Model-free vision-based shaping of deformable plastic materials," *The Int. J. Robotics Research*, vol. 39, no. 14, pp. 1739–1759, 2020.

[20] P. Zhou, J. Zhu, S. Huo, and D. Navarro-Alarcon, "Lasesom: A latent and semantic representation framework for soft object manipulation," *IEEE Robotics Autom. Lett.*, vol. 6, no. 3, pp. 5381–5388, 2021.

[21] X. Ma, D. Hsu, and W. S. Lee, "Learning latent graph dynamics for visual manipulation of deformable objects," 2021.

[22] R. Lee, D. Ward, V. Dasagi, A. Cosgun, J. Leitner, and P. Corke, "Learning arbitrary-goal fabric folding with one hour of real robot experience," in *Conf. on Robot Learn.*, pp. 2317–2327, PMLR, 2021.

[23] X. Lin, Y. Wang, Z. Huang, and D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *Proc. 5th Conf. on Robot Learn.* (A. Faust, D. Hsu, and G. Neumann, eds.), vol. 164 of *Proceedings of Machine Learning Research*, pp. 256–266, PMLR, 08–11 Nov 2022.

[24] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto, "Learning predictive representations for deformable objects using contrastive estimation," in *Proc. 2020 Conf. on Robot Learn.* (J. Kober, F. Ramos, and C. Tomlin, eds.), vol. 155 of *Proceedings of Machine Learning Research*, pp. 564–574, PMLR, 16–18 Nov 2021.

[25] R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, and K. Goldberg, "VisuoSpatial foresight for physical sequential fabric manipulation," , vol. 46, no. 1, pp. 175–199.

[26] Y. Qiu, J. Zhu, C. Della Santina, M. Gienger, and J. Kober, "Robotic fabric flattening with wrinkle direction detection," *arXiv preprint arXiv:2303.04909*, 2023.

[27] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations," 2017.

[28] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of coordinates and governing equations," *Proc. National Acad. Sci.*, vol. 116, pp. 22445–22451, oct 2019.

[29] P. Sharma, W. T. Chung, B. Akoush, and M. Ihme, "A review of physics-informed machine learning in fluid mechanics," *Energies*, vol. 16, no. 5, 2023.

[30] K. Kashinath, M. Mustafa, A. Albert, J.-L. Wu, C. Jiang, S. Esmaeilzadeh, K. Azizzadenesheli, R. Wang, A. Chattopadhyay, A. Singh, A. Manepalli, D. Chirila, R. Yu, R. Walters, B. White, H. Xiao, H. A. Tchelepi, P. Marcus, A. Anandkumar, P. Hassanzadeh, and Prabhat, "Physics-informed machine learning: case studies for weather and climate modelling," , vol. 379, no. 2194, p. 20200093.

[31] S. L. Brunton, J. Nathan Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. W. Blom-Schieber, T. Hogan, and D. McDonald, "Data-driven aerospace engineering: Reframing the industry with machine learning," *AIAA J.*, vol. 59, no. 8, pp. 2820–2847, 2021.

[32] M. Raissi, A. Yazdani, and G. E. Karniadakis, "Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations," *Science*, vol. 367, no. 6481, pp. 1026–1030, 2020.

[33] E. Tretschk, A. Tewari, M. Zollhöfer, V. Golyanik, and C. Theobalt, "DEMEA: Deep mesh autoencoders for non-rigidly deforming objects," in *Comput. Vis. – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), Lecture Notes in Computer Science, pp. 601–617, Springer International Publishing.

[34] N. Sharp, C. Romero, A. Jacobson, E. Vouga, P. Kry, D. I. Levin, and J. Solomon, "Data-free learning of reduced-order kinematics," in *ACM SIGGRAPH 2023 Conf. Proc.*, pp. 1–9, 2023.

[35] H. Bertiche, M. Madadi, and S. Escalera, "Neural cloth simulation," *ACM Trans. on Graph. (TOG)*, vol. 41, no. 6, pp. 1–14, 2022.

[36] N. Kairanda, M. Habermann, C. Theobalt, and V. Golyanik, "Neuralclothsim: Neural deformation fields meet the kirchhoff-love thin shell theory," *arXiv preprint arXiv:2308.12970*, 2023.

[37] L. Fulton, V. Modi, D. Duvenaud, D. I. W. Levin, and A. Jacobson, "Latent-space dynamics for reduced deformable simulation," *Comput. Graph. Forum*, 2019.

[38] S. Shen, Y. Yin, T. Shao, H. Wang, C. Jiang, L. Lan, and K. Zhou, "High-order differentiable autoencoder for nonlinear model reduction," *arXiv preprint arXiv:2102.11026*, 2021.

[39] C. Romero, D. Casas, M. M. Chiaramonte, and M. A. Otaduy, "Contact-centric deformation learning," *ACM Trans. on Graph. (TOG)*, vol. 41, no. 4, pp. 1–11, 2022.

[40] M. Lepri, D. Bacciu, and C. Della Santina, "Neural autoencoder-based structure-preserving model order reduction and control design for high-dimensional physical systems," *IEEE Control Syst. Lett.*, 2024.

[41] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013. Number: arXiv:1312.6114.

[42] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[43] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Trans. on Robotics*, 2023.

[44] C. Della Santina, C. Duriez, and D. Rus, "Model-based control of soft robots: A survey of the state of the art and open challenges," *IEEE Control Syst. Mag.*, vol. 43, no. 3, pp. 30–65, 2023.

[45] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *Int. Conf. on Learn. Represent.*, 2017.

[46] C. D. Santina, "The soft inverted pendulum with affine curvature," in *2020 59th IEEE Conf. on Decision Control (CDC)*, pp. 4135–4142, 2020.

[47] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, “Dream to control: Learning behaviors by latent imagination,” *arXiv preprint arXiv:1912.01603*, 2019.

[48] J. Liu, P. Borja, and C. Della Santina, “Physics-informed neural networks to model and control robots: a theoretical and experimental investigation,” *arXiv preprint arXiv:2305.05375*, 2023.