Sparse Summarization of Robotic Grasping Data

Martin Hjelm Carl Henrik Ek Renaud Detry Hedvig Kjellström Danica Kragic

***Abstract*— We propose a new approach for learning a summarized representation of high dimensional continuous data. Our technique consists of a Bayesian non-parametric model capable of encoding high-dimensional data from complex distributions using a sparse summarization. Specifically, the method marries techniques from probabilistic dimensionality reduction and clustering. We apply the model to learn efficient representations of grasping data for two robotic scenarios.**

I. Introduction

Many problems in robotics deal with high dimensional data and one of the important questions is how to represent it in as simple and efficient form as possible.

Even with solid domain knowledge and intuition, understanding the data can be hampered and we can assume low-dimensionality is desirable for understanding and finding the right form of representation. To that end it is common to reduce dimensionality as a pre-processing step using one of the many dimensionality reduction techniques available [1]–[4]. Another important class of methods is those that prefer or require additional dimensionality reduction in terms of clustered or discrete data. As an example, most features in computer vision such as HOGs, shapes and SIFTs [5]–[7] rely on an efficient discretization and representation of a very high-dimensional feature space. As another example, in machine learning, the structure of a graphical model has a profound effect on its descriptive power. Little general progress has been made for learning structure from continuous data. For discrete data, however, a range of methods exists (for a review see [8]). As a tool for visualizing and summarizing data, clustering, and discretization can be very effective. In [9] clustering was used to extract a prototypical grasp, which allowed generalization of grasps to novel objects. However, discretization is a "hard-choice" where once a data-point is associated to a state its relationship to the original feature representation is lost.

We present a method that performs dimensionality reduction and clustering simultaneously. We learn a latent space via the augmentation of the observed data together with a sparse, generative mapping. The generative mapping is coupled with the latent space representation of the augmented data in such a way as to enforce the latent representation of the observed data to be explainable through the augmented data, which is encouraged to be uncorrelated.

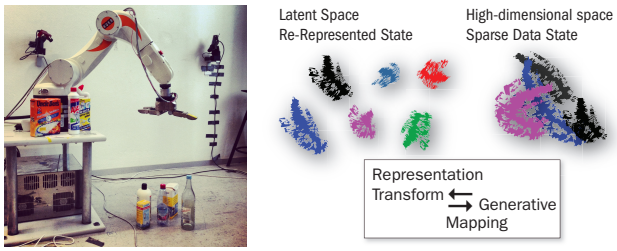


Fig. 1: Simplifying inference by forcing a re-representation: relating objects, actions and tasks using a generative approach.

II. Related Work

Dimensionality reduction methods are central to many application domains. Formally such methods make the assumption that the observed representation of the data has been generated from an underlying representation through a generative mapping. Such methods are divided in (i) spectral methods that find a mapping from the observed data to the new parameterization and (ii) generative models, that find a representation that can generate the observed data. Spectral methods aim to model the inverse of the generative mapping and are therefore more restricted, considering only the set of solutions where the generative mapping takes the form of a bijection [10]. Generative methods are much more flexible but additional information must be provided to limit the space of solutions. In this paper we will exploit the flexibility of the generative approach.

A generative method that has seen wide success is the Gaussian Process Latent Variable Model (GP-LVM) [1]. In the GP-LVM framework the generative mapping is modeled using a flexible Gaussian Process ($\mathcal{GP}$) prior [11] and the resulting representation is referred to as the latent representation. One of the main benefits of the model is that it is straightforward to incorporate priors on the latent representation. In the original presentation of the model an uninformative prior was used to regularize the solution space [1]. However, many different priors have been suggested, encoding different preferences on the representations. [12] presents a model that enforces the latent locations to respect the local distance in the observed space. In [13] the authors propose a prior based on class information, learning a representation that reflects the class division. Similarly [14] constrains the latent space to respect a certain topology. Wang et al. [15] uses a prior that encourages the latent space to reflect the dynamics of the data by using an auto-regressive prior that is suitable for dynamic modeling. The representations learned using such priors have had a big

M. Hjelm, C. H. Ek, R. Detry, H. Kjellström and D. Kragic are with the Centre for Autonomous Systems, Computer Vision and Active Perception Lab, CSC, KTH Royal Institute of Technology, Stockholm, Sweden. Email: {martinhjelm,chek,detryr,hedvig,danik}@csc.kth.se

This work was supported by the Swedish Foundation for Strategic Research, the Belgian National Fund for Scientific Research (FNRS) and the EU project TOMSY (IST-FP7-270436). We would also like to thank Andreas Damianou for constructive discussions and insightful comments.

impact on modeling of high-dimensional dynamic data such as human pose [16]. In robotics this has been further developed with the proposal of additional regularization which encourage temporally regular shaped latent spaces for data over multiple repetitions such that each matches a simple template sequence [17].

The wide variety of priors that have been discussed above, all share the same goal: *finding a representation that will suit a specific task or model*. We present a prior that generates latent representations aimed at summarizing high dimensional data using a sparse clustered representation. Our approach is an extension of the work presented in [18], [19] but provides a much more principled formulation that significantly increases the strength and applicability of the model.

III. Model

Given a set of observed data $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]$ represented in a space or parameterization Y we wish to find a new parameterization X that summarizes and represents the observed data. Formally there is a mapping f that relates elements $\mathbf{x}_i$ in the latent representation X to its corresponding observed parameterization $\mathbf{y}_i \in Y$, $f : X \rightarrow Y$. For a set of observed data $\mathbf{Y}$ there exists an infinite number of possible representations that respect the above relationship. Here, we adopt a latent variable approach where the mapping f will take functional form. We are interested in finding a well separated and grouped representation such that it can easily be summarized in terms of a clustering. For the latent representation we adopt a GP-LVM model, assuming the observed data to be generated from the latent representation through a functional mapping with additive Gaussian noise. This leads to the likelihood of the model $p(\mathbf{Y}|\mathbf{f})$, where $\mathbf{f}$ is the instantiation of the function. The GP-LVM proceeds by assuming that each dimension of the observed data is independent, given the latent locations, and by placing a $\mathcal{GP}$ prior over the mapping.

A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution [11]. A $\mathcal{GP}$ is uniquely determined by its mean and covariance function, where the covariance function relates the influence of the other random variables in the collection that is: $f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$, where μ and $k(\cdot, \cdot)$ are the mean and the covariance function respectively.

The benefit of the GP-LVM compared to other latent variable models is that the mapping f can analytically be integrated out from the likelihood. This leads to a marginal likelihood which averages over every possible mapping f,

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{i=1}^{D} \int p(\mathbf{Y}^i|\mathbf{F}) p(\mathbf{F}|\mathbf{X}) dF = \prod_{i=1}^{D} \mathcal{N}(\mathbf{Y}^i|0, K). \quad (1)$$

Here $\mathbf{Y}^i$ corresponds to the i:th dimension of the observed data, K the specific covariance.

The $\mathcal{GP}$ prior in the model is extremely flexible and can, with an appropriately chosen kernel function, be made to have a non-zero probability for a large range of functions allowing for a very representative model.

Learning implies seeking the latent location $\mathbf{X}$ and the hyper-parameters θ and can be found by maximizing the posterior of the model

$$\arg\max_{\mathbf{X},\theta} p(\mathbf{X}, \theta|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X}, \theta)\, p(\mathbf{X}) \quad (2)$$

In the original model presented in [1] an uninformative prior was used for the latent space to find a solution that was as unrestricted as possible. However, when additional information is available, such as in [13], or when a specific structure of the space is desired such can easily be accommodated within the framework by formulation of a more specific prior $p(\mathbf{X})$ [17]. We will now proceed to explain how a prior that facilitates a clustered representation can be formulated.

A. Latent space priors

It is illustrative to think about the problem from another direction, imagining already clustered, well-represented lower dimensional input that are related to some higher dimensional target values via some function. This is essentially a regression problem and to solve it we want to use $\mathcal{GP}$ regression. However, the regression must meet the condition that some points in the data - the cluster centers - represent the same information as the data in the clusters e.g. if we were to remove some data points the solution should still be the same. Therefore we augment our dataset with additional input and target value pairs $(\mathbf{U}, \mathbf{f}_u)$ that have this explanatory capability. We refer to the pairs as *inducing inputs* and *inducing points*. The complete probability is now written as

$$p(\mathbf{Y}, \mathbf{f}, \mathbf{f}_u|\mathbf{X}, \mathbf{U}) = p(\mathbf{Y}|\mathbf{f}, \mathbf{f}_u)\, p(\mathbf{f}|\mathbf{f}_u)\, p(\mathbf{f}_u) \quad (3)$$

where $\mathbf{f}_u$ are the inducing points in the observed space and $\mathbf{U}$ the corresponding latent inputs. This is the same formulation as in sparse $\mathcal{GP}$ regression [20]. However there the augmentation data – the inducing points and inputs – are considered as additional variables for approximating the true posterior but here they are cluster centers with the power to represent the data points that belong to the clusters.

If we go back to our GP-LVM and use the augmentation idea to model our representation of the data to a latent clustering; we realize that if the cluster centers should be responsible for explaining the points belonging to the cluster then the inducing points in the observed space should be as uncorrelated as possible. By reducing the correlation between the inducing points we are forcing them to choose which points in the dataset to explain. The GP-LVM will thus be forced to find a balance between a good latent representation and one that clusters the latent variables. A good latent representation will mean a mapping to the observed data that is as probable as possible. The decorrelation property and the explanatory capacity of the augmentation data will mean a latent representation that is as separated as possible and where the cluster centers explain the cluster points as good as possible.

In practice such behavior will manifest itself when the covariance function evaluated on the inducing inputs $\mathbf{U}$ is diagonal. For example in a grasp representation this means

recovering a set of independent postures that are capable of *inducing* the full range of possible postures in the data. To that end, we are motivated by the inducing prior that was defined in [18], [19] which penalizes the $\mathcal{L}_1$ norm of the off-diagonal elements of a kernel matrix evaluated on the inducing variables,

$$p(\mathbf{U}|\theta,\beta) \sim \mathcal{N}(\sqrt{D(\mathbf{U},\theta)}|0,\beta_U^{-1}), \tag{4}$$
$$D(\mathbf{U},\theta_u) = \sum_{ij}^{M} \lambda_{ij}\; k(\mathbf{u}_i,\mathbf{u}_j,\theta_u), \quad \lambda_{ij} = \begin{cases} 0 & i=j \\ 1 & i\neq j \end{cases},$$

where $k(u_i,u_j,\theta_u)$ is the covariance function of the $\mathcal{GP}$ prior on the inducing points $\mathbf{f}_u$, θ the support and β_U^{-1} the precision parameter or what we will later refer to as the constraint parameter. This will force the covariance matrix of the inducing points, in the $\mathcal{GP}$ prior, into a more diagonal form since any non-zero off-diagonal values will be penalized.

Introducing the augmented clustering data into the GP-LVM we can marginalize out f_u in the same manner as f was marginalized out, leading to the following posterior,

$$p(\mathbf{X},\mathbf{U},\theta|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{X},\mathbf{U},\theta)\, p(\mathbf{U}|\theta)\, p(\mathbf{X}), \tag{5}$$

where the optimization now is also over the cluster centers.

In the original model proposed in [18], a discretization of the latent space i.e. the inducing inputs of the posterior above were sought. However, the prior over them, $P(\mathbf{U})$ and the generative mapping was decoupled meaning that it did not fully exploit and model the relation between U and f_u. In the optimization process they were separate from each other in the sense that the kernel support parameter was not shared. This means that in the optimization process the parameter on $\mathbf{f}_u$ focused on explaining the observed data $\mathbf{Y}$, and was not affected by the prior on $\mathbf{U}$ to the same extent; if a certain form of the latent representation part would be highly likely.

By coupling the support parameter in the cluster prior and the inducing $\mathcal{GP}$ prior we enforce consistency. This happens since the support parameter for the inducing $\mathcal{GP}$ prior is now less likely to move in a direction that would increase the representative probability at the expense of the cluster prior probability. Hence the explanatory capacity of the inducing inputs for the latent variables in terms of clusters becomes much more effective. In the experimental section we will show results comparing the original uncoupled approach with the new approach of shared kernel function parameters. Since our formulation of the augmentation of the data is mathematically equivalent to sparse $\mathcal{GP}$ regression we can utilize the same methods. We therefore use the sparse, variational approximation of [21] for the $\mathcal{GP}$ prior.

The maximization of the posterior, i.e., the learning, is done via conjugate gradient descent. The latent variables are assigned to the cluster centers / inducing inputs with which they have the largest covariance. In this case this is equivalent to choosing the minimum Euclidian distance since we are using a spherical kernel. This is an attractive aspect of our method since it opens up for other kernels or other distance measures for assigning points to clusters. It also means that our assignments are not hard assignments unless we specify so, since the covariance gives a degree of association. In our case this is useful since it allows us to be either coarse or fine-tuned when selecting grasp generalizations for example. In a more general setting it allows for agglomerative clustering depending on the cut-off value of the covariance one set to rule the association.

IV. Experiments

We use three datasets for evaluation: a synthetic dataset allowing us to have full control over the characteristics of the data, and two recently presented grasping datasets.

A. Initialization, Parameters, Number of clusters

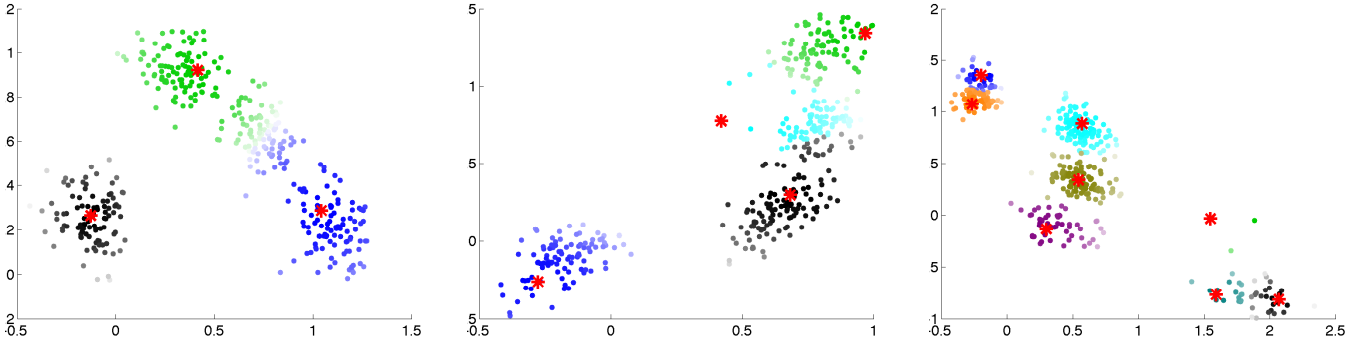
The initialization process can be thought of as a pre-step to help the clustering. We can think of the log posterior we are trying to optimize as an energy function being the sum of two parts $E = E_{\text{Representation}} + E_{\text{Clustering}}$, where $E_{\text{Representation}}$ is our GP-LVM posterior, the representative part and $E_{\text{Clustering}}$ is the clustering part, our prior on the latent variables and cluster centers. This energy function has several local minima due to the large representation space, the latent variables and inducing inputs configuration in the latent space. Initializing using standard probabilistic PCA (PPCA) [3] as is the default case for the GP-LVM might not be the most beneficial, since it will create strong local minima for the representative part while not taking into account the clustering part. The conjugate gradient descent is highly likely to get stuck in local minima close to the initialization. To give the optimization process carte blanche if the PPCA is too biased, we utilize a random initialization for the latent variables while the inducing variables are chosen using K-means to ensure an even distribution of the cluster centers.

The precision parameter of the inducing prior can be thought of as a parameter to tighten or relax our constraints on the correlation between the inducing points. It affects the separation of the clusters where a tight (large) constraint will force the latent points closer to the cluster centers and a more relaxed (small) constraint will have a lesser clustering effect.

That the number of cluster centers affects the solution is a natural and desirable property of the model. But the effect of the ratio of data points to clusters on the solution is more subtle. This can be understood by thinking about the assumption of the inducing inputs and points as explanatory for the observed data. If the ratio is too big and the data too noisy, the explanatory capacity and the power of the cluster center prior will be reduced since the covariance will have a natural diagonal and therefore the representative part will take over in the optimization.

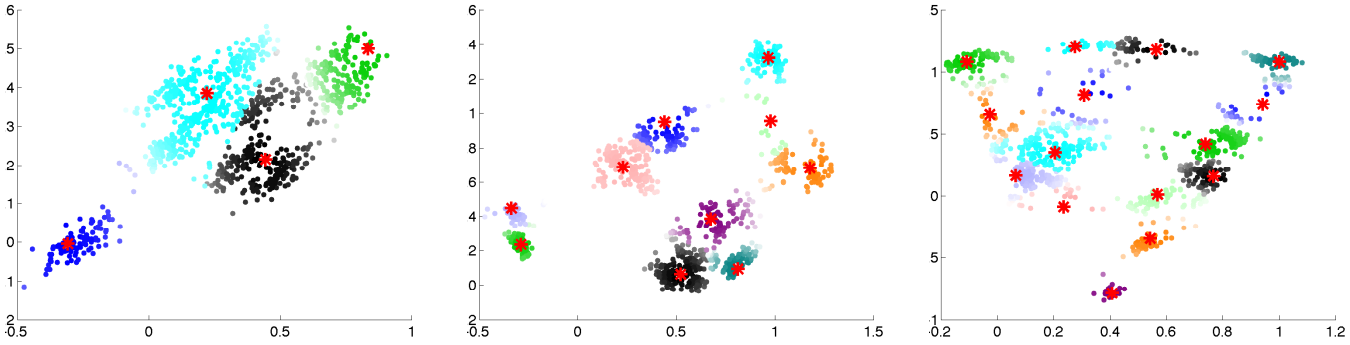
B. Synthetic dataset

As a good litmus test for our approach we generated two test datasets consisting of 400 points sampled from four (Fig.2) and ten (Fig.3) different two-dimensional Gaussians with random means and identical covariances on the unit square. The points were then linearly projected into ten dimensions where Gaussian noise was added. During generation we also saw to it that the sampled points would overlap,



(a) 3 clusters (b) 4 clusters (c) 8 clusters

Fig. 2: Synthetic dataset generated by sampling from 4 Gaussians and projected into to 10 dimensions with added noise. The red stars specifies the cluster centers and the color gradient specifies degree of covariance. In (a) we use 3 cluster points, which results in a pulling a part of the data as two points split the explaining of the 4 underlying clusters. (b) Using 4 cluster centers we find 4 Gaussian looking clusters. The blue center is forced left due to the constraint parameter. In (c) we choose 8 clusters this divides the data such that some bigger clusters of latent variables are explained by two cluster centers.



(a) 4 clusters (b) 10 clusters (c) 15 clusters

Fig. 3: Synthetic dataset generated in the same way as in figure 2 but with 10 Gaussian clusters. The behavior is consistent with the analysis in figure 2.

to not make it too simple for a PPCA-K-means solution. We choose the following scenarios to test our model:

1) True amount of clusters: The true amount of clusters is, of course, an ambiguous statement dependent on the definition criteria for the clusters, as well as how hard the lines are drawn between categories. If one tries to group apples and pears the line is quite clear, but if one chooses damaged fruit there is suddenly a sliding scale. The notion of the true amount remains in the questions we ask and the way we structure the data. Thus, without encoding a preference on the latent variables via the prior, the representation and clustering is going to be the one most likely to have generated the observed data but not necessarily the one we deem to be the true. Of course, some representations are going to be more likely than others. With this in mind, Figure 2b shows that if we ask for four clusters corresponding to the clusters underlying the observed data, our method delivers four Gaussian-looking clusters.

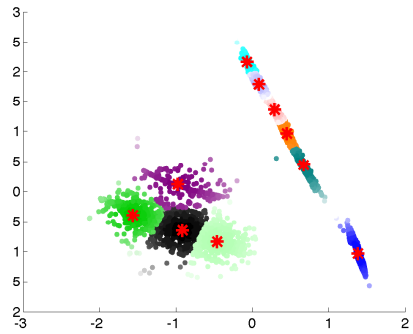
2) Too few clusters: Using too few cluster centers has the effect that the model tries to push together the data even when there is an inherent and more probable separation. Basically the inducing inputs are forced to represent more of the data, stretching and compressing the latent variables, as seen in Figures 2a and 3a.

3) Too many clusters: Enforcing a less likely clustering onto the data means less separation and divisions of natural clusters between two or more centers. Again, Figures 2c and 3c exemplify the trade-off between the representation and clustering, with the representation being some larger chunks of latent points crammed together while the cluster prior tries to divide the data.

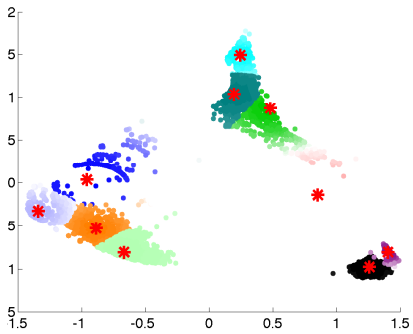
C. Grasping data

We continue by applying the proposed method to the same dataset that was previously used by the original method using the de-coupled prior [18], [19]. The dataset consists of a twenty-dimensional representation of a hand configuration performing different grasps. Different grasps are related to objects depending on how the objects are used. By discretizing the data according to the learned clustered representation, the authors learn an efficient factorization of the data. The following experiments clarify the benefits of the coupled prior by comparing the results of our method with the original one.

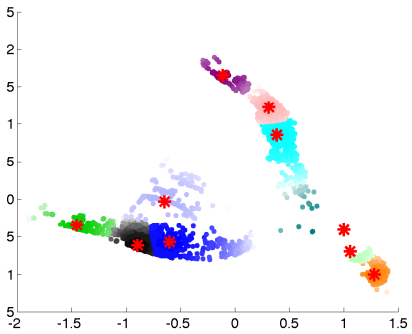
We first run the GPLVM algorithm without a prior using ten inducing points on the data. Comparing the results, 4b, with the initialization, 4a, we can see that the difference is mostly in spreading the data. Introducing the un-coupled prior, 4c-d, the data points get even more drawn out and clusters starts to form. The uncoupled prior is still too weak



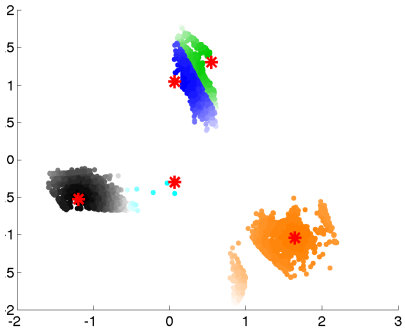
(a) PPCA-Kmeans init with 10 clusters.



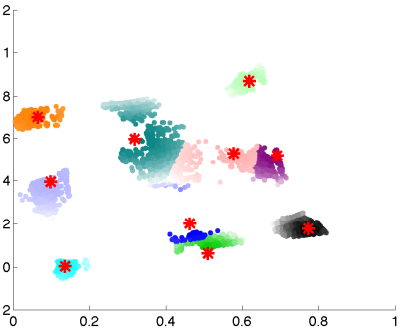
(b) 10 clusters, no prior on the inducing inputs.



(c) 10 clusters using the uncoupled prior.



(d) 5 clusters using the coupled prior.



(e) 10 clusters using the coupled prior.

Fig. 4: Clustering of the Fcon Armar dataset [19]. We first do a simple PPCA-K-means initialization in (a) and then a GPLVM solution from that initialization without the cluster point prior in (b). The difference between the initialization and GPLVM is not big as can be expected. Adding the uncoupled prior to the GPLVM results in a slightly more clustered and spread out latent space. However comparing it with the results seen in (d) and (e) where the coupling is in place it is easy to see how much more powerful that solution is. In (e) the data is almost pulled apart despite the strong representation part.

compared to the representation part to start separating the data from the initialization, resulting in the solution being close to the no-prior. The large amount of data compared to the number of centers makes the solution sensitive to the constraint parameter, the random initialization, and the initial placement of the cluster centers. For strong results we must rely on a PPCA-K-means initialization, implying that the uncoupled prior in this case is not strong enough to move past uninteresting minima. When we introduce the coupling something interesting happens: the data starts getting more pulled apart, as seen in 4d and 4e. In 4d one cluster center is responsible only for a few data points. This can be explained by some observed data having high variance. Increasing the number of cluster centers, 4e, gives a more even distribution of the data points but some clusters still shares two cluster centers. This suggests that there can be a number of clusters that is more probable with respect to the representative part or that the intra-cluster variance makes the centers need more inducing inputs than one to be explained.

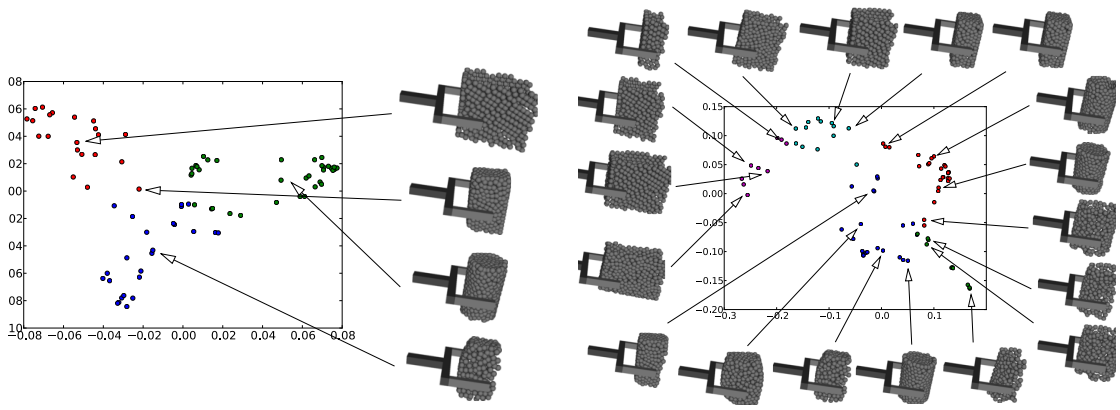
D. Grasp Shape Experience

We now present an instance of the sparse summarization problem in the context of robot grasp learning. To efficiently grasp new objects, it has been argued that robots should learn from experience and transfer acquired grasping skills to new objects as they come [22], [23]. We have recently argued that it can be done by learning the shape of parts by which objects are often grasped [24], [25]. The core of our approach is to compare the shapes of surface segments extracted around a robot's hand while it is trained to grasp different objects. Our rationale is that shapes that are observed across multiple grasps should be helpful for grasping new objects. We suggested a two-step solution to this problem: (1) measure the similarity in shape between all pairs of grasps in the dataset, and (2) cluster the grasps in the space induced by the similarity measure. As it is costly to teach grasps to a robot, the dataset is sparse, which makes clustering challenging.

We now present our work on the data and similarity measure presented in [26]. The data was collected by teleoperating a robot to grasp eight objects in several different ways[1]. Then, surface segments of various extents were segmented out of the objects around the grasping points, and a shape similarity measure was applied to all pairs of such segments. The similarity measures were then ordered into a matrix.

We start out by applying PCA to the similarity matrix and project the data down into twenty dimensions. Our prior belief is that the data will generalize well by the division into three categories. Hence, we ran our method using three cluster centers, the results of which may be seen in Fig. 5a. The clustering turns out to be similar to the PPCA-K-means initialization and consistent in the clustering. This can be understood by realizing that if the natural division is three, then the data can be expected to have a high between-cluster variance implying strong minima for the representation part, which in practice would be close to the initialization. However, when we move on to using five clusters (Fig.5) we see a big change from the three cluster grouping. With more cluster

[1] `http://renaud-detry.net/research-gt.php`



(a) Random initialization with 3 cluster centers

(b) Random initialization 5 clusters using prior

Fig. 5: GSE data clustering solution using 3 and 5 clusters. Using 3 cluster centers groups the data similar to a PPCA-K-means solution meaning that the representation part has a possible strong minima. The 5 cluster solution in (b) infuses a finer granularity. The representation part now has to take much more of the constraints of the clustering part into consideration and thereby forces a more grouped and different latent configuration.

centers, the prior now splits the data into a finer granularity, forcing the representation to move away from the original expression and form more refined clusters. We also make the observation that the clustering is consistent in that the same points roughly gets assigned to the same clusters when using a random initialization of the latent space in repeated trials. This means that there is a clear underlying natural grouping, which we find. Furthermore, since the clustering is confident in the optimal solution, it implies that the posterior has few local minima.

V. Conclusion

We have presented an approach to learn a sparse, low-dimensional representation of high-dimensional robotic grasping data. The model is general and provides a summarization of any continuous data. We are currently evaluating the possibilities of integrating the coupled inducing point prior model with the full Bayesian variational approach to the GP-LVM [27]. Further, we are interested in integrating the proposed prior with a dynamic model which should facilitate learning of key-frames for dynamic data.

References

[1] N. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *The Journal of Machine Learning Research*, 6:1783–1816, 2005.

[2] M. Cox and T. Cox. Multidimensional Scaling. In *Handbook of data visualization*, pages 315–347. Springer, 2008.

[3] Michael E Tipping and C.M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B*, 61(3):611–622, 1999.

[4] N. Lawrence. A Unifying Probabilistic Perspective for Spectral Dimensionality Reduction: Insights and New Models. *The Journal of Machine Learning Research*, pages 1609–1638, 2012.

[5] N Dalal and B Triggs. Histograms of Oriented Gradients for Human Detection. In *IEEE CVPR*, pages 886–893, 2005.

[6] G Mori, S Belongie, and J Malik. Efficient shape matching using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1832–1837, 2005.

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[8] P. Leray and O. Francois. BNT structure learning package: Documentation and experiments. Technical report, Laboratoire PSI - INSA, 2004.

[9] R. Detry, C.H. Ek, M. Pronobis, J. Piater, and D. Kragic. Generalizing Grasps Across Partly Similar Objects. In *IEEE ICRA*, 2012.

[10] C. H. Ek. Shared Gaussian Process Latent Variable Models. *PhD Thesis*, 2009.

[11] C.E. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[12] N. Lawrence and J. Quiñonero-Candela. Local distance preservation in the GP-LVM through back constraints. In *Proceedings of the 23rd international conference on Machine learning*, pages 513–520. ACM, 2006.

[13] R. Urtasun and T. Darrell. Discriminative Gaussian Process Latent Variable Model for Classification. In *ICML*, pages 927–934, 2007.

[14] R. Urtasun, D. Fleet, A. Geiger, J. Popovic, T. Darrell, and N. Lawrence. Topologically-Constrained Latent Variable Models. *ICML*, 2008.

[15] J M Wang, David J Fleet, and A Hertzmann. Gaussian Process Dynamical Models for Human Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008.

[16] R. Urtasun, D. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamical models. *IEEE CVPR*, 1:238–245, 2006.

[17] S Bitzer and S. Vijayakumar. Latent Spaces for Dynamic Movement Primitives. *International Conference on Humanoid Robots*, 2009.

[18] C.H. Ek, D. Song, and D. Kragic. Learning Conditional Structures in Graphical Models from a Large Set of Observation Streams through efficient Discretisation. In *IEEE ICRA, Workshop on Manipulation under Uncertainty*. Royal Institute of Technology, 2011.

[19] D. Song, C.H. Ek, K. Huebner, and D. Kragic. Multivariate discretization for bayesian network structure learning in robot grasping. In *IEEE ICRA*, pages 1944–1950, 2011.

[20] J. Quiñonero-Candela and C.E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[21] M. Titsias. Variational Learning of Inducing Variables in Sparse Gaussian Processes. *Journal of Machine Learning Research - Proceedings Track*, 5:567–574, 2009.

[22] J. Coelho, J. Piater, and R. Grupen. Developing haptic and visual perceptual categories for reaching and grasping with a humanoid robot. In *Robotics and Autonomous Systems*, volume 37, 2000.

[23] A. Morales, E. Chinellato, A. H. Fagg, and A. P. del Pobil. Using experience for assessing grasp reliability. *International Journal of Humanoid Robotics*, 1(4):671–691, 2004.

[24] D. Renaud, C.H. Ek, M. Madry, J. Piater, and D Kragic. Generalizing grasps across partly similar objects. In *IEEE ICRA*, 2012.

[25] O. Kroemer, E. Ugur, E. Oztop, and J. Peters. A kernel-based approach to direct action perception. In *IEEE ICRA*, 2012.

[26] R. Detry, C.H. Ek, M. Madry, and K. Danica. Learning a dictionary of prototypical grasp-predicting parts from grasping experience. In *IEEE ICRA*, 2013. to appear.

[27] M. Titsias and N. Lawrence. Bayesian Gaussian Process Latent Variable Model. In *International Conference on Airtificial Inteligence and Statistical Learning*, 2010.