
Gaussian Process Latent Variable Alignment Learning

Ieva Kazlauskaitė

University of Bath, UK
Electronic Arts

i.kazlauskaitė@bath.ac.uk

Carl Henrik Ek

University of Bristol, UK

Neill D. F. Campbell

University of Bath, UK
Royal Society

n.campbell@bath.ac.uk

Abstract

We present a model that can automatically learn alignments between high-dimensional data in an unsupervised manner. Our proposed method casts alignment learning in a framework where both alignment and data are modelled simultaneously. Further, we automatically infer groupings of different types of sequences within the same dataset. We derive a probabilistic model built on non-parametric priors that allows for flexible warps while at the same time providing means to specify interpretable constraints. We demonstrate the efficacy of our approach with superior quantitative performance to the state-of-the-art approaches and provide examples to illustrate the versatility of our model in automatic inference of sequence groupings, absent from previous approaches, as well as easy specification of high level priors for different modalities of data.

1 Introduction

Learning from sequential data is challenging as data might be sampled at different and uneven rates, sequences might be collected out of phase, *etc.* Consider the following scenarios: humans performing a task may take more or less time to complete parts of it, climate patterns are often cyclic though particular events take place at slightly different times in the year, the mental ability of children varies depending on their age, neuronal spike waveforms contain temporal jitter, replicated scientific experiments often vary in timing. However, most sample statistics, *e.g.* mean and variance, are designed to capture variation in amplitude rather than phase/timing. This leads to increased sample variance, blurred fundamental data structures and an inflated number of principal components needed to describe the data. There-

fore, the data needs to be aligned in order for dependencies such as these to be recovered. This is a non-trivial task that is often performed as a pre-processing stage to modelling.

Traditionally, the notion of sequence similarity comes from a measure of pairwise similarity integrated across the sequences. This local measure often leads to highly non-convex optimisations problems making alignments challenging to learn. In this paper we take a different approach where we encapsulate alignment and modelling within a single framework. By simultaneously modelling the sequences and the alignment we can capture global structure thereby circumventing the difficulties associated with an objective function based on pairwise similarity.

Further difficulties arise when the the dataset contains observations from several distinct functions. Consider, for example, a set of motion capture experiments that include tasks such as running, jumping and sitting down. Data for each of these three types of sequences can be aligned to themselves but a global alignment between them may not exist. In traditional approaches, the observed data must be grouped into the distinct sequence types before alignment. To overcome this limitation, our approach also produces a generative model over the sequences themselves. This means we can simultaneously infer both alignments and their grouping.

Methods for learning alignments can broadly be classified into two categories. The first learns a function to warp the input dimension while the second directly learns the transformed sequences. There are several benefits to learning a warping function as it allows us to resample the data and, by constraining the class of functions, we can also incorporate global constraints on the alignment. However, specifying a parametric function is challenging and often results in difficult optimisation tasks. Directly learning transformed sequences avoids having to specify a parametrisation. However, this comes at the cost of removing all but the most rudimentary global constraints on the warping function since the optimal alignment is completely specified by the pairwise similarity. In contrast, we propose a novel approach that learns the warping function using a probabilistic model. Underpinning our methodology is the use of Gaussian process priors that allows us to approach this learning in

a Bayesian framework achieving principled regularisation without reducing the solution space.

Our proposed model overcomes a number of problems with the existing literature and confers three main contributions:

1. We model the observed data directly with a generative process, rather than interpolating between observations, that allows us to reject noise in a principled manner.
2. The generative model of the aligned data allows a fully unsupervised approach that performs simultaneous clustering and alignment.
3. We use continuous, non-parametric processes to model explicitly the warping functions throughout; this allows the specification of sensible priors rather than unintuitive or heuristic choices of parametrisations.

2 Background

Pairwise similarity There has been a significant amount of work in learning alignments from data. Most approaches are based on the assumption of the existence of a pairwise similarity measure between the instances of each sequence. The classical approach to minimise the distance between two sequences is called Dynamic Time Warping (DTW), and is based on computing an affinity matrix of the two sequences to be aligned [5]. The solution corresponds to the path through this matrix that leads to the minimal combined pairwise cost. The optimal solution is found by backtracking through the affinity matrix and can be estimated using Dynamic Programming [26]. DTW finds the optimal alignment based on a pairwise distance between each element in two sequences. Such formulation imposes a number of limitations. DTW returns an alignment but not a parametrised warping, and it is not trivial to encode a preference towards different warps as this would be a global characteristic while DTW is a local algorithm.

Multiple sequences In its original form DTW aligns two sequences only but several extensions allow it to process multiple sequences at once, most notably Procrustes dynamic time warping (PDTW), Procrustes derivative dynamic time warping (PDDTW), and Iterative Motion Warping (IMW) [16, 10, 14]. All of these methods are applied directly in the observation space which is a limitation when the data contains a significant amount of noise. The main algorithms that address this limitation are Canonical Time Warping (CTW) and Generalized Time Warping (GTW) [41, 39]. Both of these approaches perform feature extraction and find a subspace that maximises the linear correlation of data samples. Similarly to our approach, GTW is parametrised using monotonic warping functions. However, in all these methods the spatial alignment and time warping are coupled. Another extension, called Generalized Canonical Time Warping (GCTW) combines CCA with DTW to simultaneously align multiple sequences of multi-modal data [40]. GCTW relies on additional heuristic energy terms

and on coarse-to-fine optimisation to get the energy method to converge to a good local minimum.

Feature extraction More recently, deep neural networks were employed to perform temporal alignments [33], [34]. The proposed method, called Deep Canonical Time Warping (DCTW), performs non-linear feature extraction and it performs competitively on larger audio-visual datasets. A different method proposed by [24] uses continuous hidden Markov models, where the latent trace is an underlying representation of the set of observable sequences. [15] introduced hyperalignment that finds isometric transformations of trajectories in voxel space that result in an accurate match of the time-series data. An extension to this model was proposed by [25] who address the issues of scalability and feature extension through the use of the kernel trick. The authors note that classification accuracy relies on intelligent feature selection.

Manifold alignment Similar to our approach, [7] propose an unsupervised manifold alignment method. It is based on finding alignment by enforcing several constraints such as geometry structure and feature matching, geometry preservation and integer constraints. The approach shows promising results but is very computationally expensive. Another non-linear feature extraction method [37] named Manifold Time Warping relies on constructing a k-nearest neighbour graph and then performing DTW to align a pair of sequences.

Implicit transformation Another approach to alignment is to use an implicit transformation of the sequences. In [9, 8] the authors propose a kernel function that is capable of mapping sequences of different length to an implicit feature space. Another similar approach is [3] which describes a range of different kernels on sequences, this method is flexible and allows for learning implicit feature space mappings for sequences of not only different lengths but also different dimensionality. These methods work well experimentally but as the alignment is implicit we cannot re-align sequences or construct novel ones.

Shape analysis A different line of work, often referred to as elastic registration or shape analysis is considered in the functional data analysis literature. In [12] the authors propose an extension to DTW by replacing the Euclidean distance with a Mahalanobis distance. By having a parametrisable distance function the authors are able to learn the metric function from a set of paired observations. [19] study the group-theoretic approach to warps by using the group of warping functions to describe the equivalence relation between signals. In particular, the authors use the Fisher-Rao Riemannian metric and the resulting geodesic distance to align signals with random warps, scalings and translations. Square root velocity function (SRVF) facilitates the use of Fisher-Rao distance between functions by estimating the \mathbb{L}^2 norm between their SRVFs [30, 18]. [35] proposed a generative model that combines elastic shape analysis of curves and functional principal component analysis (fPCA).

Another recent extension called Elastic functional coding relies on trajectory embeddings on Riemannian manifolds and results in manifold functional variant of PCA [2].

Gaussian processes Our model makes use of Gaussian processes as priors over warpings, sequences and their groupings. A Gaussian process (GP) [27] is a random process specified by a mean $m(x)$ and a covariance function $k_\theta(x, x')$. The covariance function is parametrised by a set of hyper-parameters θ while the mean is often considered as constant zero. The index set of the two functions is infinite which allows GPs to be interpreted as non-parametric priors over the space of functions. Even though the process is infinite, an instantiation of the process is finite and reduces to a Gaussian distribution. In a regression setting we observe a set of noisy samples $\mathcal{D} = \{x_n, y_n\}_{n=1}^N$ of a latent function $f(\cdot)$ such that $y_n = f(x_n) + \varepsilon_n$. By placing a GP prior over the latent function $f \sim \mathcal{GP}(m(x), k(x, x'))$ the instantiations of the function at the training data $\{f_n = f(x_n)\}_{n=1}^N$ are Gaussian as $F \sim \mathcal{N}(m(X), k(X, X))$ where F and X are concatenations of the function instantiations and the observed input locations respectively. By choosing a Gaussian noise model the functions can be marginalised out in closed form due to the self-conjugate property of the Gaussian distribution. The Gaussian process latent variable model (GP-LVM) [21] is a model that uses GP priors to learn latent variables. The model assumes that each dimension of the observed data Y have been generated from a latent variable X through some latent function f . By placing a GP prior over f and marginalising out this mapping, the latent representation of the data can be recovered. The model is very flexible and has been implemented across a wide range of different applications, for example [6, 13, 36].

Warped GPs In [29] and [23] the authors construct a GP with a warped input space to account for differences in observations (e.g. inputs may vary over many orders of magnitude), and show that a warped GP finds the standard preprocessing transforms, such as the logarithm, automatically. In comparison, our approach leads to a warped output space of the GP-LVM, and uses the additional knowledge of possible misalignments in the high-dimensional space to regularise the problem of building a low-dimensional latent space. Concurrent with our work, [11] use GPs for modelling sequences of neural population spike-trains and the corresponding temporal warps. The proposed approach is an extension to GP factor analysis [38] and uses a linear combination of shared and private latent processes to encourage alignment of sequences for different trails. Unlike our work, [11] do not recover a clustering of the sequences and thus require the groups of sequences for alignment to be known a-priori.

3 Methodology

Alignment learning is the task of recovering a set of monotonic warping functions that have been used to create

samples of a latent sequence. Fig. 1 provides an overview of our approach. We are provided with a number of noisy time warped observations of a set of unobserved latent sequences and our task is to infer both this set of sequences and the time warps that give rise to the observations.

Let us assume that we have J noisy sequence observations $\{Y_j\}$ (Fig. 1a) where each observed sequence comprises N time samples, $\mathbf{Y} = (y_{(j,n)}) \in \mathbb{R}^{J \times N}$. We consider each sequence to be generated as a sample from a latent function $f_j(x)$ (Fig. 1b) under a monotonic warping $g_j(x)$ as $y_{(j,n)} = f_j(g_j(x_n)) + \varepsilon_{jn}$ where the samples have been corrupted by additive Gaussian noise $\varepsilon_{jn} \sim \mathcal{N}(0, \beta_j^{-1})$. Due to the close association of sequences and temporal data, we use the word time to refer to the input domain of the sequence, however our method is general and applicable to any ordered index set.

The aligned sequences, which are unobserved, are given by the corresponding functions without the time warp $s_{(j,n)} = f_j(x_n) + \varepsilon_{jn}$, $\mathbf{S} = (s_{(j,n)}) \in \mathbb{R}^{J \times N}$ (as illustrated in Fig. 1d and Fig. 1c). This means that we can encode our warping function as the transformation from a *known* sampling of an *unknown* aligned sequence to the *unknown* sampling of the *known* observations for each sequence. However, as described in the introduction and illustrated in Fig. 1d, we wish to design a model that is not restricted to the case where all the observations arise from a *single* latent function (for example, in Fig. 1d there are two unknown true sequences).

To account for the possible existence of multiple latent functions, we consider a generative model for the aligned sequences themselves. By specifying that the generative process be as simple as possible, we encourage the clustering of these sequences, which allows to automatically find the smallest number of latent functions explaining the data. We encode this as the aligned sequences being generated via a smooth mapping $h(\cdot)$ from a low dimensional space $Z \in \mathbb{R}^Q$ as

$$S_j = h(Z_j) + \hat{\varepsilon} \quad \text{s.t.} \quad S_j = f_j(x) + \varepsilon_j \quad \forall j \quad (1)$$

where $\hat{\varepsilon} \sim \mathcal{N}(0, \gamma^{-1}I)$ and $\varepsilon_j \sim \mathcal{N}(0, \beta_j^{-1}I)$. This low dimensional manifold is visualised, for our toy example, in Fig. 1f where Z is a 2D space and the locations of the aligned sequences S_j are shown as coloured points matching the corresponding aligned sequences in Fig. 1e. We see that the two different sequences are clustered appropriately by their location in the manifold and the sequences are correctly aligned. We use a probabilistic model of the aligned sequences, which allows us to quantify uncertainty of the low-dimensional manifold representations (the heatmap in Fig. 1f).

3.1 Probabilistic Model

In this section we specify the two components of our model that correspond to the constraint introduced in Eq. (1). The first part corresponds to fitting the data that explains the observed sequences and specifies the latent functions, while the second part enforces a simple, low-dimensional

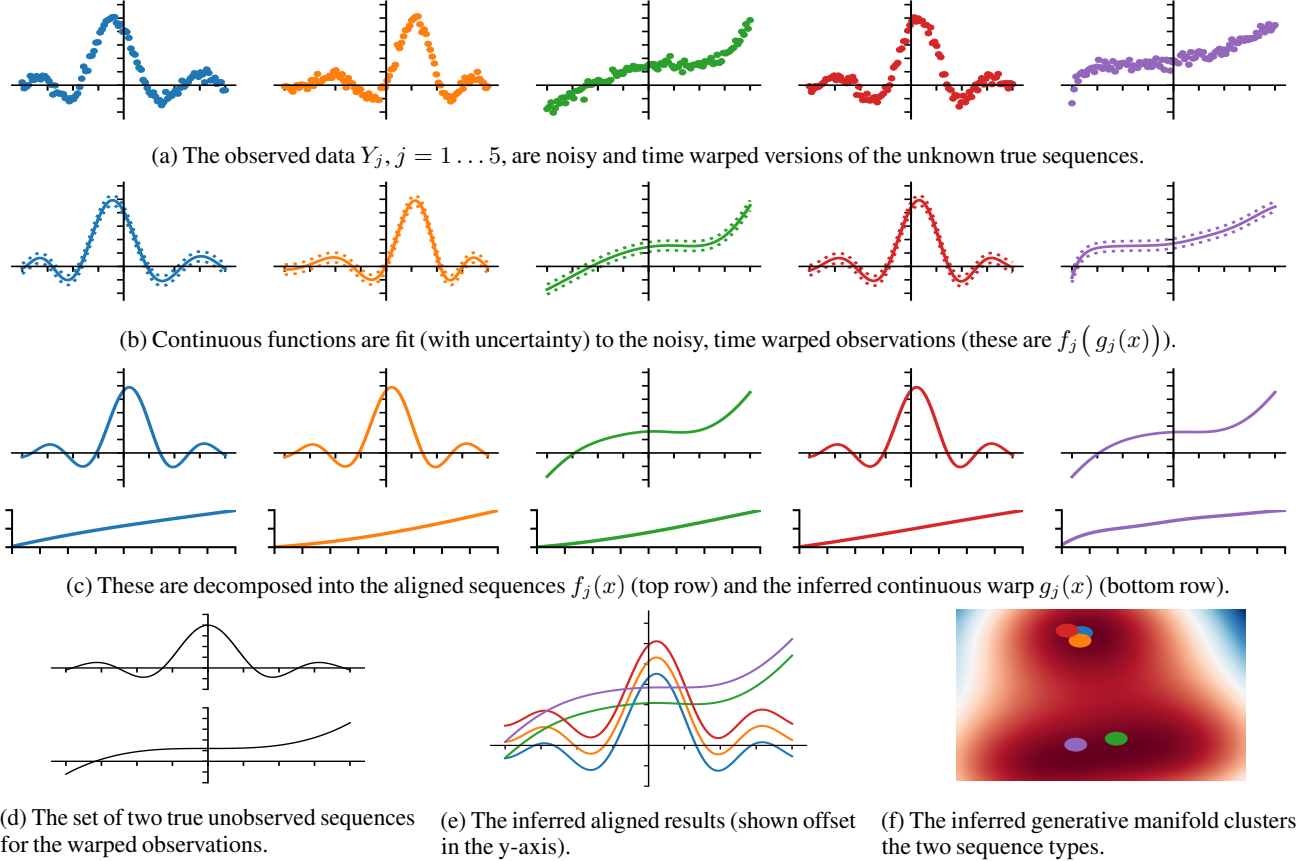


Figure 1: Overview of our model on a toy example. We are presented with a set noisy observations (a) that we assume to be time warped versions from a set of true sequences (d). We fit continuous functions (b) to the observations and then decompose and cluster them into aligned versions with continuous time warps (c). This results in a generative model where the aligned sequences (e) are produced from a manifold (f) that reveals the clustering of the observations into the two fundamental sequences of (d).

structure of the aligned sequences. Given noisy observed data, we do not impose this constraint exactly, but rather define both model components as probabilistic models and interpret this constraint as one of the aligned sequences having high likelihood under both model components simultaneously. If the aligned sequences \mathbf{S} were known, this interpretation would correspond exactly to fitting a model to the observed data by maximising the data likelihood. Since \mathbf{S} are unobserved, we refer to them as *pseudo-observations*; similarly to [32], we augment the probability space with a set of pseudo-observations which are constrained by the two components of our model. We then fit the model by maximising the joint marginal likelihood of observations \mathbf{Y} and pseudo-observations \mathbf{S} , while optimising not only w.r.t. the model parameters, but also w.r.t. the pseudo-observations \mathbf{S} .

Model over time We have $Y_j \in \mathbb{R}^N$ as the observed sequences, and let $X \in \mathbb{R}^N$ denote an observed uniform sampling of time. We introduce a random variable $G_j \in \mathbb{R}^N$ to encode the time warp function sampled at X such that $G_j \sim g_j(X)$. The random variables for the functions $f_j(\cdot)$ are more involved since the functions are evaluated at different locations. Let $F_j^G \sim f_j(G_j)$ denote the output

of the function sampled at the time warped locations G_j and let $F_j^X \sim f_j(X)$ denote the function evaluated at the uniform sample locations X . The observations Y_j are the noise-corrupted versions of F_j^G , and similarly, we call the noise-corrupted version of F_j^X pseudo-observations, since they are not observed and should be inferred.

We now define the priors over the generating and warping functions $f_j(\cdot)$ and $g_j(\cdot)$. Specifying a parametric mapping is challenging and it severely limits the possible functions we can recover. In this paper, we make use of flexible non-parametric Gaussian Process (GP) priors which allows us to provide significant structure to the learning problem without reducing the possible solution space. The two random variables connected with $f_j(\cdot)$ may then be jointly specified under a GP prior where the covariance, with hyperparameters θ , is evaluated at G_j and X for F_j^G and F_j^X respectively as

$$p\left(\begin{bmatrix} F_j^X \\ F_j^G \end{bmatrix} \middle| G_j, X_j, \theta_j\right) \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} k_{\theta_j}(X, X) & k_{\theta_j}(X, G_j) \\ k_{\theta_j}(G_j, X) & k_{\theta_j}(G_j, G_j) \end{bmatrix}\right). \quad (2)$$

Warping functions We encode our preference for smooth warping functions $g_j(\cdot)$ by making $p(G_j | X)$ a GP prior

with a smooth kernel function. We can ensure monotonicity by an appropriate parametrisation of the G_j using an auxiliary input. Without loss of generality, these are constrained to be monotonic in the range $[-1, 1]$ using a set of auxiliary variables $U_j \in \mathbb{R}^N$ such that

$$[G_j]_n := 2 \sum_{k=1}^n [\text{softmax}(U_j)]_k - 1. \quad (3)$$

Importantly, all warping functions are continuous and generative which means we are able to resample the data. Therefore, we write that $p(G_j | X, \omega) \sim \mathcal{N}(0, k_{\omega_j}(X, X))$ with hyperparameters ω_j . An alternative to our parameterisation is GPs with monotonicity information [28], however, this approach does not guarantee that the posterior predictive is monotonic.

Model over sequences We would like a constraint that aligns similar sequences to each other while keeping dissimilar sequences apart without us specifying which sequences belong together. We consider using dimensionality reduction as a means of preserving similarities in the prediction space as well as imposing the preference for dissimilar data points to be placed far apart in the latent space. In particular, we propose to use a GP-LVM that places independent GPs over the data features and optimises the locations of the low-dimensional latent points that correspond to each sequence.

To this end, we let the random variable $Z_j \in \mathbb{R}^Q$ be the embedded manifold location of the sequence. The random variable H_j denotes the output of the mapping function evaluated at Z_j such that $H_j \sim h(Z_j)$. To ease notation, we use bold symbols to denote the concatenation across J such that, for example, $\mathbf{Z} = [Z_1, \dots, Z_J]$. We encode the preference for a smooth mapping by placing a GP prior over the mapping $h(\cdot)$ so that we have $p(\mathbf{H} | \mathbf{Z}, \psi) \sim \mathcal{N}(0, k_{\psi}(\mathbf{Z}, \mathbf{Z}))$ where ψ are the hyperparameters of the covariance kernel. The pseudo-observations are modelled by the GPLVM by adding independent Gaussian noise to \mathbf{H} . Next we consider the joint distribution of the model to derive an objective which simultaneously ensures that (i) the observed data \mathbf{Y} is fitted well by the corresponding GPs f at the warped locations, (ii) the pseudo observations are fitted well by the corresponding GPs f at the fixed sampling locations, (iii) the pseudo observations are such that they exhibit a simple structure that is captured by the latent variable model.

Joint distribution The joint distribution (ignoring the hyperparameters and noise terms for clarity) decomposes as

$$p(\mathbf{S}, \mathbf{Y}, \mathbf{F}^X, \mathbf{F}^G, \mathbf{G}, \mathbf{H}, \mathbf{Z} | \mathbf{X}) = p(\mathbf{Y} | \mathbf{F}^G) p(\mathbf{S} | \mathbf{H}, \mathbf{F}^X) p(\mathbf{H} | \mathbf{Z}) p(\mathbf{F}^X, \mathbf{F}^G | \mathbf{G}, \mathbf{X}) p(\mathbf{G} | \mathbf{X}) p(\mathbf{Z}). \quad (4)$$

The terms $p(\mathbf{H} | \mathbf{Z})$, $p(\mathbf{F}^X, \mathbf{F}^G | \mathbf{G}, \mathbf{X})$ and $p(\mathbf{G} | \mathbf{X})$ are the GP priors defined previously and $p(\mathbf{Z}) \sim \mathcal{N}(\mathbf{0}, I)$ is the latent prior. We note that $p(\mathbf{F}^X, \mathbf{F}^G | \mathbf{G}, \mathbf{X})$ and $p(\mathbf{G} | \mathbf{X})$ factorise fully over J .

Likelihood terms The likelihood of the observations under i.i.d. Gaussian noise with precision β_j is $p(\mathbf{Y} | \mathbf{F}^G) = \prod_j p(Y_j | F_j^G) = \prod_j \mathcal{N}(Y_j | F_j^G, \beta_j^{-1} I)$. The likelihood of the pseudo-observations is more involved since it encodes the relationship of Eq. (1). We define the likelihood $p(\mathbf{S} | \mathbf{H}, \mathbf{F}^X)$ as an equal mixture:

$$p(\mathbf{S} | \mathbf{H}, \mathbf{F}^X) = \frac{1}{2} \left(\prod_n \mathcal{N}(S_n | \mathbf{H}_n, \gamma^{-1} I_J) + \prod_j \mathcal{N}(S_j | \mathbf{F}_j^X, \beta_j^{-1} I_N) \right) \quad (5)$$

where S_j refers to the rows and S_n refers to the columns of \mathbf{S} .

This explicitly encourages the two components (the one over h and the one over f) to coincide so that the pseudo-observations are explained by both components of the model simultaneously. In order to find the maximum likelihood solution, we use the fact that $\log(\frac{1}{2} a + \frac{1}{2} b) \geq \frac{1}{2} \log(a) + \frac{1}{2} \log(b)$, and we maximise the lower bound on the log-likelihood to find the parameters of the two models, the latent variables, and the pseudo-observations.

Approximations The integrals over \mathbf{H} and \mathbf{F}^X are regular GP marginalisations, which can be computed in closed-form. The integral corresponding to \mathbf{F}^G includes a composition of GPs, $(f \circ g)$, which does not have a closed-form solution. Following [22], we approximate this integral using a point estimate, and since \mathbf{G} is directly optimised, it allows us to use the monotonic parametrisation of Eq. (3) without the need to integrate over the corresponding parameters.

Learning We place priors over the hyperparameters $\{\gamma, \psi, \theta_j, \beta_j\}$ as log-Normal distributions with zero mean and unit variance. We also place an additional prior on the raw sample points U_j to encourage smooth warps, and improve training as,

$$\log p(\{U_j\}) = \sum_{j=1}^J \log \mathcal{N}(U_j | \mathbf{0}, I_N). \quad (6)$$

We optimise the following marginal log-likelihood (excluding the terms corresponding to the priors on the hyperparameters): $\log p(\mathbf{S}, \mathbf{Y}, \mathbf{Z} | X) = \log p(\mathbf{S}, \mathbf{Y} | X) + \log p(\mathbf{S} | \mathbf{Z}) + \log p(\mathbf{Z})$ w.r.t. the pseudo observations \mathbf{S} , the latent variables \mathbf{Z} and the hyperparameters of the model to obtain the MAP estimates.

Implementation We implement our model using the TensorFlow [1] framework and minimise the negative marginal log-likelihood objective using the Adam optimizer [17]. By default, we used standard squared exponential covariance functions for all the Gaussian process priors. In some of the experiments, different covariance functions were used when the data or warping functions were less smooth (e.g. the Matérn covariance). The complexity of our method is limited by the inversion of the covariance matrices and therefore scales with $\mathcal{O}(JN^3 + J^3)$. However, there are standard

sparse approaches available to scale to longer sequences. We also implemented the sparse variational method of Titsias [32] which reduces the complexity to $\mathcal{O}(JNM^2 + J^3)$, where M is a specified number of inducing points for the sparse approximation. This method performed well for M an order of magnitude smaller than the full N . We note that the use of a sparse approximation fits naturally with the rest of our model as it increases the smoothness of the observations, which may simplify the alignment task.

3.2 Comparison of Variants of our Model

Our proposed model is fully non-parametric and models both the warping and the generating functions at the same time. Methods that rely on the standard \mathbb{L}^2 metric in the input space are ill-posed and thus require a regularisation term. This leads to an optimisation problem that suffers from poor local minima and relies on the use of a coarse-to-fine approach. In order to highlight the limitations of using the standard \mathbb{L}^2 metric in the input space, we describe a model that performs a parametric re-sampling of the data which corresponds to removing our model of the warpings but retaining a model of the data. In effect we take a traditional pairwise minimisation approach but include a probabilistic model of the data which has the effect of regularising the optimisation problem.

Parametric warps We use a parametric re-sampling function $\tilde{g}^{(j)}(\cdot)$ similar to [40] consisting of K monotonically increasing basis functions. For each input sequence Y_j , we learn a set of weights $\mathbf{w}^{(j)} \in \mathbb{P}^K$. By enforcing that the weights lie on the surface of the k^{th} order probability simplex \mathbb{P} the resulting function is guaranteed to be monotonic. The task is now to find the set of weights $\{\mathbf{w}_k^j\}_{k=1}^K$ such that re-sampling the data according to the warping functions results in the aligned sequences. As we do not have access to \mathbf{S} , we use the same latent variable model as previously and refer to this model as *GP-LVM+basis*. The model can be learned using gradient descent. The parametric model described above, as well as some previous approaches, rely on hand-picked basis functions to define the warps. This results in poor accuracy when the set of basis functions is small and in high computational complexity when the set is large.

Energy alignment We demonstrate the efficacy of using the alignment GP-LVM to perform simultaneous clustering and alignment by replacing it with an energy minimisation objective that is similar to the previous literature, *e.g.* [19]. The latent variable model part of the objective is replaced with an energy minimisation term between each of the S_j and the mean of all the sequences $\{S_j\}_{j=1}^J$. In § 4 we show the results of this method with the GP warping functions (*energy+GP*) and with the basis function warpings as described above (*energy+basis*).

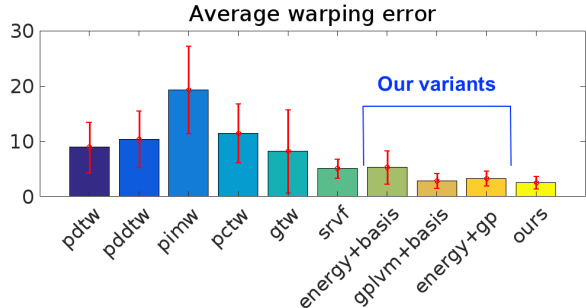


Figure 2: Comparison to state-of-the-art: average error on 25 datasets proposed by Zhou *et al.* [39].

4 Experiments

We now discuss the experimental evaluation of our proposed model. We use standard squared exponential covariance functions for all the GP priors, unless stated otherwise. We show comparisons to current state-of-the-art approaches from data mining and functional data analysis communities using publicly available reference implementations¹. The accuracy is primarily measured in terms of the warping error, *i.e.* the mean squared error (MSE) between the known true warps and the estimated warps. The alignment error, the MSE between the pairs of aligned sequences, is easily misinterpreted since it is a local measurement. In particular, it does not capture the degenerate cases where the local maxima and minima in the input sequences are shifted to non-corresponding extrema; this is particularly true in datasets with periodic components. Other examples of degenerate behaviour are multiple dimensions collapsing to a single point and warps that rely on translating and rescaling every input in each dimension that leads to over-fitting (an example of this is IMW alignment [39]). All of these result in high alignment accuracy but produce poor quality results.

Datasets with quantifiable comparisons For this experiment, we use the dataset proposed by Zhou and De la Torre [39]. It consists of sequences that are generated by temporally transforming latent 2D shapes under known warping transformations that allow quantitative evaluation of the estimated warps. To better assess the quality of the alignments, we run 25 tests with randomly selected size of the dataset, dimensionality and temporal transformations. Our approach outperforms other methods on these datasets, see Fig. 2 and the supplementary material, and produces accurate alignments irrespective of the size of the dataset, dimensionality and structure of the sequences. The variant of our method that uses parametric warps (*gplvm+basis*) performs competitively on these datasets, motivating the use of a Gaussian process objective for alignment. Furthermore, we see that our non-parametric approach to modelling the time warps improves the flexibility of the model; out of the two models

¹See [42] for the implementation of DTW, DDTW, IMW, CTW, GTW, and [31] for the implementation of SRVF.

MSE (SD)	SRVF	GP-LVM+BASIS	OURS
ALIGNMENT	6.4 (± 1.7)	8.4 (± 2.7)	5.9 (± 1.1)
WARPING	30.0 (± 10.4)	9.7 (± 4.9)	9.7 (± 5.7)

Table 1: Quantitative comparison of alignments and warps for the best competing method on dataset with multiple true sequences (alignment and grouping task).

that rely on energy minimisation as the alignment objective, *energy+basis* and *energy+gplvm*, the latter one demonstrates lower warping error and significantly lower standard deviation on this dataset. This result supports the premise that even though the non-parametric representation allows for any smooth monotonic warp, the probabilistic framework places sufficient structure to make the problem well posed and avoid over-fitting. An example of warps and alignments for this experiment are available in the supplementary material.

Dataset for clustering In our second experiment, we consider a dataset that contains multiple clusters of sequences. This task requires the sequences to be aligned within each cluster. None of PDTW, PCTW, GTW nor the energy minimisation methods are able to perform this task as they have no knowledge of the underlying structure of the dataset. The SRVF algorithm performs clustering by first aligning the data in terms of amplitude and phase, then performing fPCA based on the estimated summary statistics, and finally modelling the original data using joint Gaussian or non-parametric models on the fPCA representations. We compare the performance of the SRVF algorithm with our approach as well as the variant of our approach with fixed basis functions.

We consider a dataset that contains three distinct groups of functions that were generated by temporally transforming three random 2D curves as described previously. All three approaches rely on the structure of the data alone to recognise the existence of the clusters and Fig. 3 shows that all three methods are able to align the data within clusters. The performance of the methods is contrasted by calculating the MSE among all pairs of sequences within each group (alignment error) and the MSE between the true warping functions and the warping functions calculated using each of the methods (warping error). For this comparison we repeat the test 25 times with randomly selected initial curves, number of dimensions and number of sequences per group. The quantitative comparison in Table 1 shows that our method consistently achieves the lowest alignment errors (i.e. with lowest standard deviation (SD) on the set of datasets).

Our method, as well as the parametric variant of it, also achieves low warping errors in comparison to SRVF which implies that they are able to reconstruct the original temporal transformations more accurately than SRVF. This behaviour is apparent in Fig. 3 where the warping functions produced by our method, and the parametric version of it, resemble the true warps while SRVF estimates noticeably different warping functions; this results in unpredictable distortions

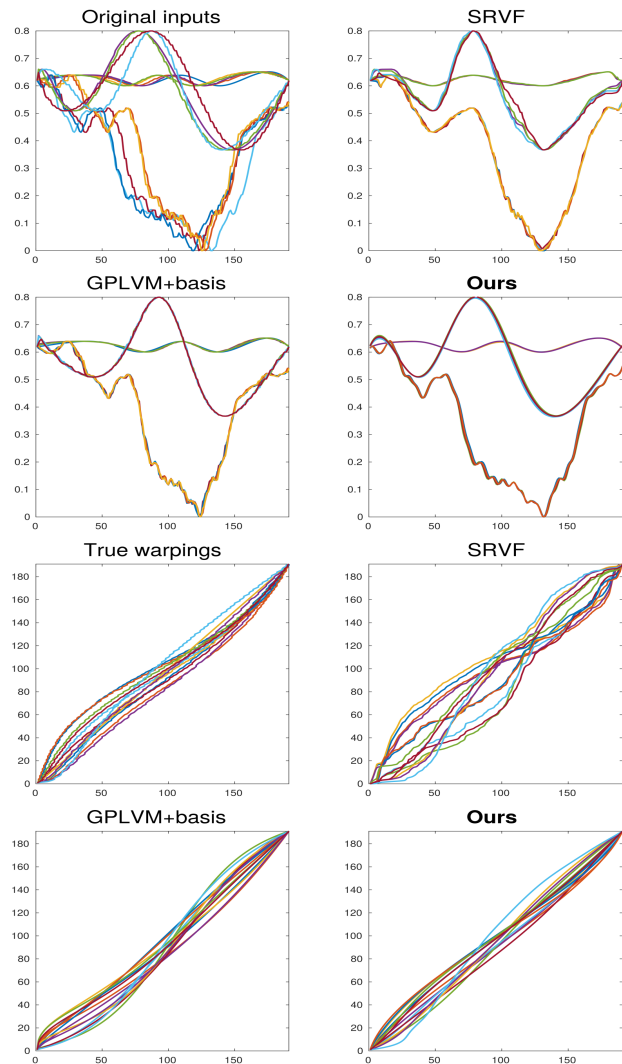


Figure 3: Alignment of 15 sequences that belong to 3 different clusters (top 4 graphs) and the corresponding warping functions.

in the aligned dataset. These results reflect the differences between the SRVF method and our approach; while SRVF is cast as an optimisation problem over a constrained domain, the domain of our probabilistic formulation is much larger but, importantly, structured from the assumptions encoded in the prior. This provides a better regularisation ultimately leading to the improvement in the recovered warpings.

Motion capture data We evaluate the performance of our model on a set of motion capture data from the CMU database [20], where each input sequence corresponds to a short clip of motion and the data is represented as quaternion locations of the joints of the subject performing the motion. We use the motion of subject no 64 from the CMU dataset that correspond to golf related motions such as a swing, a putt, and placing and picking up of a ball. We consider five instances of three different motions that need to be temporally aligned within the three groups. Fig. 4 illustrates how our model favours the simplified, i.e. aligned, inputs. The corresponding

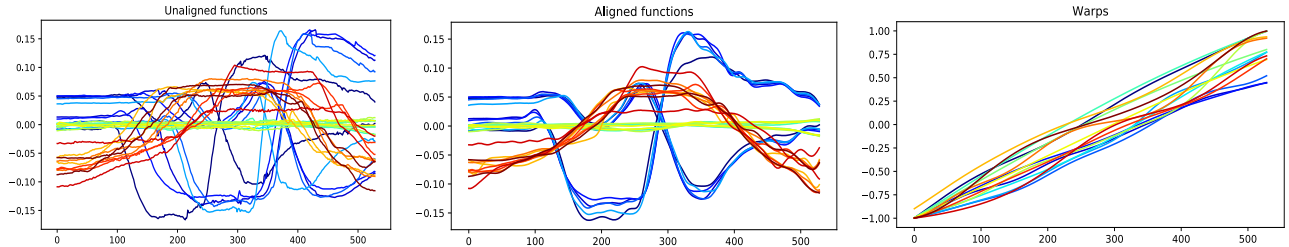


Figure 4: GP-LVM alignment demonstrates the preference for a simplified explanation when the model is given the ability to align the data.

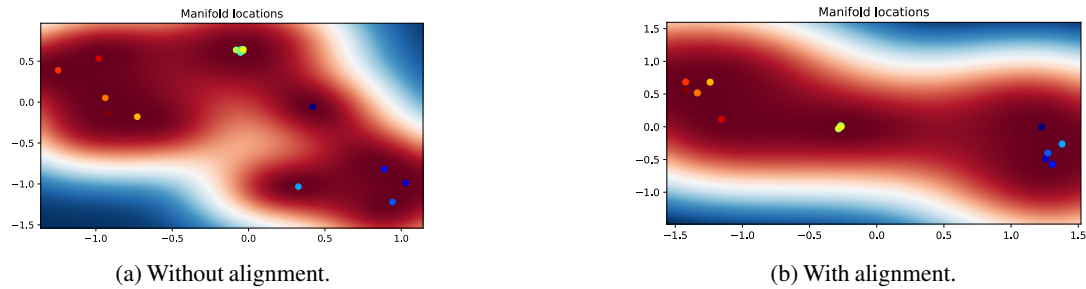
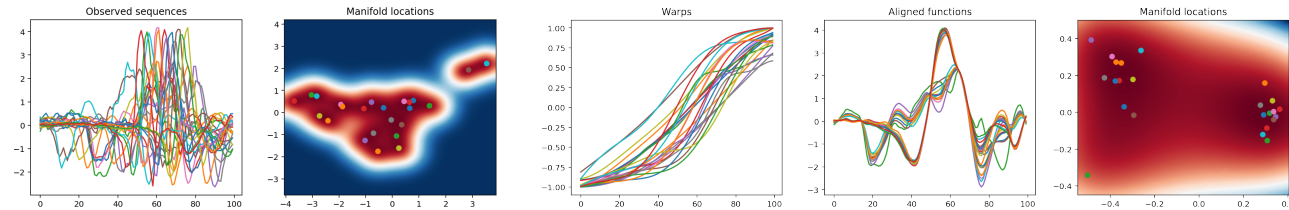


Figure 5: 2D manifolds produced without and with alignment in the GP-LVM. Using the alignments emphasizes the existence of multiple clusters of data and aligns data points within each cluster.



(a) The clustering of the unaligned observed sequences does not reveal the two types of heartbeats. (b) Accounting for the alignment of sequences allows us to discover automatically the two different types of heartbeats.

Figure 6: Alignment of heartbeats data [4].

manifolds produced using a traditional GP-LVM (*i.e.* without alignment) and a manifold produced using our approach are shown in Fig. 5. Our model produces a fine alignment of the input sequences within each of the groups, and consequently the resulting two-dimensional manifold offers a good separation of the three groups. We note that the manifold produced using GP-LVM without alignment contains more isolated areas, which means the model is less capable of generalising between the warps. Therefore, our implicitly aligned model is able to generate smoother transitions in the manifold, producing high quality predicted outputs of novel alignments.

Heartbeats data This dataset contains heartbeat sounds, and it is known that a normal heart sound has a clear "lub dub, lub dub" pattern which varies temporally depending on the age, health, and state of the subject [4]. Our approach automatically aligns and clusters the heart sounds recorded by a digital stethoscope. Instead of using a pre-processing step with a low-pass filter to account for the noise in the high frequencies, we use a Matérn 3/2 kernel that takes into account the rapid variations in the recordings while also limiting the effect of the uninformative high frequency noise. Fig. 6

illustrates how simultaneous fitting and alignment allows us to correctly discover and cluster the two types of heartbeats.

5 Conclusion and Future Work

We have presented a probabilistic model that is able to implicitly align inputs that contain temporal variations. Our approach models the observed data directly producing a generative model of the functions rather than interpolating between observations. In addition, using a GP-LVM for alignment builds an unsupervised generative model that has the benefit of simultaneous clustering and aligning the input sequences. Furthermore, we proposed a continuous, non-parametric explicit model of the time warping functions that removes issues such as quantisation artefacts and the need for ad-hoc pre-processing. We demonstrated that the proposed approaches perform competitively on alignment tasks, and outperform the existing methods on the task of simultaneous alignment and clustering. In the future we will consider the use of Bayesian GP-LVM for automatic model selection and will test the framework on additional datasets, including multi-modal data.

Acknowledgments

This work has been supported by EPSRC CDE (EP/L016540/1) and CAMERA (EP/M023281/1) grants as well as the Royal Society. IK would like to thank the Frostbite Physics team at EA.

References

- [1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. Software available from tensorflow.org.
- [2] R. Anirudh, P. Turaga, J. Su, and A. Srivastava. Elastic Functional Coding of Human Actions: From Vector-Fields to Latent Variables. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [3] A. Baisero, F. T. Pokorný, and C. H. Ek. On a Family of Decomposable Kernels on Sequences. *arXiv.org*, abs/1501.06284, 2015.
- [4] P. Bentley, G. Nordehn, M. Coimbra, and S. Mannor. The PASCAL Classifying Heart Sounds Challenge 2011 (CHSC2011) Results. <http://www.peterjbentley.com/heartchallenge>.
- [5] D. J. Berndt and J. Clifford. Using Dynamic Time Warping to Find Patterns in Time Series. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, 1994.
- [6] N. D. F. Campbell and J. Kautz. Learning a Manifold of Fonts. *ACM Transactions on Graphics*, 33(4), 2014.
- [7] Z. Cui, H. Chang, S. Shan, and X. Chen. Generalized Unsupervised Manifold Alignment. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.
- [8] M. Cuturi. Fast Global Alignment Kernels. In *International Conference on Machine Learning (ICML)*, 2011.
- [9] M. Cuturi, J. P. Vert, O. Birkenes, and T. Matsui. A Kernel for Time Series Based on Global Alignments. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007.
- [10] I. L. Dryden and K. V. Mardia. *Statistical Shape Analysis, with Applications in R. Second Edition*. 2016.
- [11] Lea Duncker and Maneesh Sahani. Temporal alignment and latent gaussian process factor inference in population spike trains. *bioRxiv*, 2018.
- [12] D. Garreau, R. Lajugie, S. Arlot, and F. Bach. Metric Learning for Temporal Sequence Alignment. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.
- [13] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović. Style-based Inverse Kinematics. In *ACM SIGGRAPH*, 2004.
- [14] Eugene Hsu, Kari Pulli, and Jovan Popović. Style translation for human motion. *ACM Trans. Graph.*, 24(3):1082–1089, 2005.
- [15] V. H. James, J. S. Guntupalli, A. C. Connolly, Y. O. Halchenko, B. R. Conroy, M. I. Gobbini, M. Hanke, and P. J. Ramadge. A Common, High-Dimensional Model of the Representational Space in Human Ventral Temporal Cortex. *Neuron*, 72(2):404–416, 2011.
- [16] E. J. Keogh and M. J. Pazzani. Derivative Dynamic Time Warping. In *SIAM International Conference on Data Mining*, 2001.
- [17] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2014.
- [18] S. Kurtek, A. Srivastava, E. Klassen, and Z. Ding. Statistical Modeling of Curves using Shapes and Related Features. *Journal of the American Statistical Association*, 107(499):1152–1165, 2012.
- [19] S. Kurtek, A. Srivastava, and W. Wu. Signal Estimation Under Random Time-warpings and Nonlinear Signal Alignment. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.
- [20] Carnegie Mellon Graphics Lab. Motion Capture Database . ["http://mocap.cs.cmu.edu/info.php"](http://mocap.cs.cmu.edu/info.php), 2016.
- [21] N. D. Lawrence. Probabilistic Non-Linear Principal Component Analysis with Gaussian Process Latent Variable Models. *Journal of Machine Learning Research (JMLR)*, 6:1783–1816, 2005.
- [22] N D Lawrence and A J Moore. Hierarchical gaussian process latent variable models. In *Proceedings of the 24th international conference on Machine learning*, pages 481–488. ACM, 2007.
- [23] M. Lázaro-Gredilla. Bayesian Warped Gaussian Processes. In *Advances in Neural Information Processing Systems (NIPS)*. 2012.

- [24] J. Listgarten, R. M. Neal, S. T. Roweis, and A. Emili. Multiple Alignment of Continuous Time Series. In *Advances in Neural Information Processing Systems (NIPS)*. 2005.
- [25] A. Lorbert and P. J. Ramadge. Kernel Hyperalignment. In *Advances in Neural Information Processing Systems (NIPS)*. 2012.
- [26] M. Müller. *Information Retrieval for Music and Motion*. 2007.
- [27] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. 2005.
- [28] Jaakko Riihimäki and Aki Vehtari. Gaussian processes with monotonicity information. In Yee Whye Teh and Mike Titterton, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 645–652, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.
- [29] E. Snelson, Z. Ghahramani, and C. E. Rasmussen. Warped Gaussian Processes. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.
- [30] A. Srivastava, W. Wu, S. Kurtek, E. Klassen, and J. S. Marron. Registration of Functional Data Using Fisher-Rao Metric. *ArXiv*, abs/1103.3817, 2011.
- [31] A. Srivastava, W. Wu, S. Kurtek, E. Klassen, and J. S. Marron. Elastic Functional Data Analysis. <http://ssamg.stat.fsu.edu/software>, 2018.
- [32] Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR.
- [33] G. Trigeorgis, M. A. Nicolaou, S. Zafeiriou, and B. W. Schuller. Deep Canonical Time Warping. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [34] G. Trigeorgis, M. A. Nicolaou, S. Zafeiriou, and B. W. Schuller. Deep Canonical Time Warping for Simultaneous Alignment and Representation Learning of Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2017.
- [35] J. D. Tucker, W. Wu, and A. Srivastava. Generative Models for Functional Data using Phase and Amplitude Separation. *Computational Statistics and Data Analysis*, 61(Supplement C):50–66, 2013.
- [36] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua. Priors for People Tracking from Small Training Sets. In *International Conference on Computer Vision (ICCV)*, 2005.
- [37] H. T. Vu, C. J. Carey, and S. Mahadevan. Manifold Warping: Manifold Alignment over Time. 2012.
- [38] Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1881–1888. Curran Associates, Inc., 2009.
- [39] F. Zhou. Generalized Time Warping for Multi-modal Alignment of Human Motion. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [40] F. Zhou and F. de la Torre. Generalized Canonical Time Warping. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 38(2), 2016.
- [41] F. Zhou and F. de la Torre. Canonical Time Warping for Alignment of Human Behavior. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.
- [42] F. Zhou and F. de la Torre. Software for Canonical Time Warping. http://www.f-zhou.com/ta_code.html, 2018.