Geodesic Regression on the Grassmannian

Yi Hong[1], Roland Kwitt[2], Nikhil Singh[1], Brad Davis[3],
Nuno Vasconcelos[4] and Marc Niethammer[1,5]

[1] Department of Computer Science, UNC Chapel Hill, NC, United States
[2] Department of Computer Science, Univ. of Salzburg, Austria
[3] Kitware Inc., Carrboro, NC, United States
[4] Statistical and Visual Computing Lab, UCSD, CA, United States
[5] Biomedical Research Imaging Center, UNC Chapel Hill, NC, United States

Abstract. This paper considers the problem of regressing data points on the Grassmann manifold over a scalar-valued variable. The Grassmannian has recently gained considerable attention in the vision community with applications in domain adaptation, face recognition, shape analysis, or the classification of linear dynamical systems. Motivated by the success of these approaches, we introduce a principled formulation for regression tasks on that manifold. We propose an *intrinsic* geodesic regression model generalizing classical linear least-squares regression. Since geodesics are parametrized by a starting point and a velocity vector, the model enables the synthesis of new observations on the manifold. To exemplify our approach, we demonstrate its applicability on three vision problems where data objects can be represented as points on the Grassmannian: the prediction of traffic speed and crowd counts from dynamical system models of surveillance videos and the modeling of aging trends in human brain structures using an affine-invariant shape representation.

Keywords: Geodesic regression; Grassmann manifold; Traffic speed prediction; Crowd counting; Shape regression.

1 Introduction

Data objects in many computer vision problems admit a subspace representation. Examples include feature sets obtained after dimensionality reduction via PCA, or observability matrix representations of linear dynamical systems. Assuming equal dimensionality, such subspace representations allow to interpret the data as points on the Grassmann manifold $\mathcal{G}(p,n)$, *i.e.*, the manifold of p-dimensional linear subspaces of $\mathbb{R}^n$. The seminal work of [10] and the introduction of efficient processing algorithms to manipulate points on the Grassmannian [12] has led to a variety of principled approaches to solve different vision and learning problems. These include domain adaptation [13,29], gesture recognition [19], face recognition under illumination changes [20], or the classification of visual dynamic processes [27]. Other works have explored subspace estimation via conjugate gradient decent [21], mean shift clustering [6], and the definition

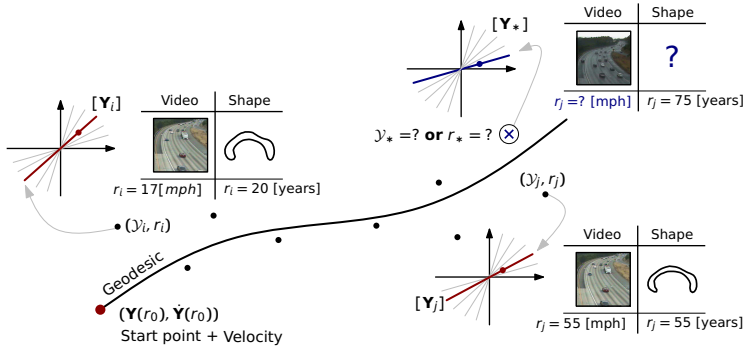


Fig. 1. Illustration of Grassmannian geodesic regression and inference. At the point marked $\otimes$, the inference objective for (i) traffic videos is to predict the independent variable r_* (here: speed), whereas for (ii) corpus callosum shapes we seek the manifold-valued $\mathcal{Y}_*$ for a value of the independent variable (here: age). For illustration, elements on the Grassmannian are visualized as lines through the origin, *i.e.*, $\mathcal{Y}_i \in \mathcal{G}(1,2)$.

of suitable kernel functions [14,18] that can be used with a variety of machine learning techniques.

While many vision applications primarily focus on performing classification or recognition tasks on the Grassmannian, the problem of regression has gained little attention (see §2). Yet, this statistical methodology has the potential to address many problems in a principled way. For instance, it enables predictions of an associated scalar-valued variable while, at the same time, respecting the geometry of the underlying space. Further, in scenarios such as shape regression, we are specifically interested in summarizing continuous trajectories that capture variations in the manifold-valued variable as a function of the scalar-valued independent variable. Fig. 1 illustrates these two inference objectives. While predictions about the scalar-valued variable could, in principle, be formulated within existing frameworks such as Gaussian process regression, *e.g.*, by using Grassmann kernels [14,18], it is not clear how to or if it is possible to address the second inference objective in such a formulation.

Contribution. We propose a formulation that directly fits a geodesic to a collection of data points. This is beneficial for several reasons. First, it is a *simple* and natural extension of linear regression to the Grassmannian; second, it provides a compact representation of the complete geodesic path; third, since the geodesic is parametrized by a starting point and a velocity, we can freely move along it and synthesize additional observations; fourth, it opens up the possibility of statistical analysis on Grassmannian geodesics; finally, this concept easily extends to more complex models, such as piecewise regression. The approach is extremely versatile which we demonstrate on three vision problems where data objects admit a representation on the Grassmannian. First, we show that the geodesic regression model can predict traffic speed and crowd counts from dynamical system representations of surveillance video clips *without* any pre-

processing. Second, we show that this model allows us to capture aging trends of human brain structures under an affine-invariant representation of shape [3]. These three different vision problems are solved in a common framework with minor parameter adjustments. While the applications presented in this paper are limited, our method should, in principle, be widely applicable to other problems on the Grassmann manifold, previously proposed in the vision literature.

The paper is structured as follows: §2 reviews closely related work; §3 introduces our formulation of *Grassmannian geodesic regression (GGR)* and presents two numerical solution strategies. §4 shows experimental results and §5 concludes the paper with a discussion of the main results, limitations and future work.

2 Related Work

While differential geometric concepts, such as geodesics and intrinsic higher-order curves, have been well studied [23,5], their use for regression has only recently gained interest. A variety of methods extending concepts of regression in Euclidean spaces to *nonflat* manifolds have been proposed. Rentmeesters [24], Fletcher [11] and Hinkle *et al.* [15] address the problem of geodesic fitting on Riemannian manifolds, mostly focusing on symmetric spaces. Niethammer *et al.* [22] generalized linear regression to the manifold of diffeomorphisms to model image time-series data, followed by works extending this concept [16,25,26].

In principle, we can distinguish between two groups of approaches: first, *geodesic shooting* based strategies which address the problem using adjoint methods from an optimal-control point of view [22,16,25,26]; the second group comprises strategies which are based on optimization techniques that leverage *Jacobi fields* to compute the required gradients [11,24]. Unlike Jacobi field approaches, solutions using adjoint methods do not require computation of the curvature explicitly and easily extend to higher-order models, *e.g.*, polynomials [15], splines [26], or piecewise regression models. Our approach is a representative of the first category which ensures extensibility to more advanced models.

In the context of computer-vision problems, Lui [19] recently adapted the known Euclidean least-squares solution to the Grassmann manifold. While this strategy works remarkably well for the presented gesture recognition tasks, the formulation does not guarantee to minimize the sum-of-squared geodesic distances within the manifold. Since, in the regression literature, this is the natural extension of least-squares to Riemannian manifolds, the geometric and variational interpretation of [19] remains unclear. In contrast, we address the problem from an energy-minimization point of view which allows us to guarantee, by design, consistency with the geometry of the manifold.

To the best of our knowledge, the closest works to ours are [2] and [24]. Batzies *et al.* [2] discusses only a theoretical characterization of the geodesic fitting problem on the Grassmannian, but does not provide a numerical strategy for estimation. In contrast, we derive alternative optimality conditions using principles from optimal-control. These optimality conditions not only form the basis for our shooting approach, but also naturally lead to a convenient iterative

algorithm. By construction, the obtained solution is guaranteed to be a geodesic. As discussed above, Rentmeesters [24] follows the Jacobi field approach. While both optimization methods have the same computational complexity for the gradient, *i.e.*, $O(np^2)$ on the Grassmannian $\mathcal{G}(p,n)$, it is non-trivial to generalize [24] to higher-order or piecewise models. Our approach, on the other hand, offers an alternative, simple solution that is (i) extensible and (ii) easy to implement.

3 Grassmannian Geodesic Regression (GGR)

To develop the framework for GGR, we first briefly review the Riemannian structure of the Grassmannian. For a more detailed treatment of this topic we refer the reader to [10,4,1]. We then discuss exact geodesic matching for two points and inexact geodesic matching for multiple points in §3.1 and present two strategies to solve these problems in §3.2 and §3.3.

Riemannian structure of the Grassmann manifold. The *Grassmann* manifold $\mathcal{G}(p,n)$ is defined as the set of p-dimensional linear subspaces of $\mathbb{R}^n$, typically represented by an orthonormal matrix $\mathbf{Y} \in \mathbb{R}^{n\times p}$, such that the column vectors span $\mathcal{Y}$, *i.e.*, $\mathcal{Y} = \text{span}(\mathbf{Y})$. The Grassmannian can equivalently be defined as a quotient space within the special orthogonal group $SO(n)$ as $\mathcal{G}(p,n) := \mathcal{SO}(n)/(\mathcal{SO}(n-p)\times\mathcal{SO}(p))$. The *canonical metric* $g_{\mathcal{Y}} : \mathcal{T}_{\mathcal{Y}}\mathcal{G}(p,n)\times\mathcal{T}_{\mathcal{Y}}\mathcal{G}(p,n) \to \mathbb{R}$ on $\mathcal{G}(p,n)$ is given by

$$g_{\mathcal{Y}}(\boldsymbol{\Delta}_{\mathcal{Y}}, \boldsymbol{\Delta}_{\mathcal{Y}}) = \text{tr } \boldsymbol{\Delta}_{\mathcal{Y}}^{\top}\boldsymbol{\Delta}_{\mathcal{Y}} = \text{tr } \mathbf{C}^{\top}(\mathbf{I}_n - \mathbf{Y}\mathbf{Y}^T)\mathbf{C} \ , \tag{1}$$

where $\mathbf{I}_n$ denotes the $n\times n$ identity matrix, $\mathcal{T}_{\mathcal{Y}}\mathcal{G}(p,n)$ is the tangent space at $\mathcal{Y}$, $\mathbf{C} \in \mathbb{R}^{n\times p}$ arbitrary and $\mathbf{Y}$ is a *representer* for $\mathcal{Y}$. Under this choice of metric, the arc-length of the geodesic connecting two subspaces $\mathcal{Y}, \mathcal{Z} \in \mathcal{G}(p,n)$ is related to the *canonical angles* $\phi_1, \dots \phi_p \in [0, \pi/2]$ between $\mathcal{Y}$ and $\mathcal{Z}$ as $d^2(\mathcal{Y},\mathcal{Z}) = ||\boldsymbol{\phi}||_2^2$. In what follows, we slightly change notation and use $d^2(\mathbf{Y},\mathbf{Z})$, with $\mathcal{Y} = \text{span}(\mathbf{Y})$ and $\mathcal{Z} = \text{span}(\mathbf{Z})$. In fact, the (squared) geodesic distance can be computed from the SVD decomposition $\mathbf{U}(\cos\boldsymbol{\Sigma})\mathbf{V}^{\top} = \mathbf{Y}^{\top}\mathbf{Z}$ as $d^2(\mathbf{Y},\mathbf{Z}) = ||\cos^{-1}(\text{diag}\,\boldsymbol{\Sigma})||^2$ (cf. [12]), where $\boldsymbol{\Sigma}$ is a diagonal matrix with principal angles ϕ_i.

Finally, consider a curve $\gamma : [0,1] \to \mathcal{G}(p,n), r \mapsto \gamma(r)$ with $\gamma(0) = \mathcal{Y}_0$ and $\gamma(1) = \mathcal{Y}_1$, where $\mathcal{Y}_0$ represented by $\mathbf{Y}_0$ and $\mathcal{Y}_1$ represented by $\mathbf{Y}_1$. The *geodesic equation* for such a curve on $\mathcal{G}(p,n)$ is given (in terms of representers) by

$$\ddot{\mathbf{Y}}(r) + \mathbf{Y}(r)[\dot{\mathbf{Y}}(r)^{\top}\dot{\mathbf{Y}}(r)] = \mathbf{0}, \text{ with } \dot{\mathbf{Y}}(r) \doteq \frac{d}{dr}\mathbf{Y}(r) \ . \tag{2}$$

Eq. (2) also defines the Riemannian exponential map on the Grassmannian as an ODE for convenient numerical computations. Integrating the geodesic equation, starting with initial conditions, "shoots" the geodesic forward in time.

3.1 Exact/Inexact geodesic matching

Exact matching between two points. To generalize linear regression in Euclidean space to geodesic regression on the Grassmannian, we replace the *line*

equation by the geodesic equation (2), *i.e.*, the Euler-Lagrange equation of

$$E(\mathbf{Y}(r)) = \int_{r_0}^{r_1} \text{tr } \dot{\mathbf{Y}}(r)^\top \dot{\mathbf{Y}}(r)\, dr, \text{ such that } \mathbf{Y}(r_0) = \mathbf{Y}_0,\ \mathbf{Y}(r_1) = \mathbf{Y}_1 \quad (3)$$

and $\dot{\mathbf{Y}}(r) = (\mathbf{I}_n - \mathbf{Y}(r)\mathbf{Y}(r)^\top)\mathbf{C}$. To generalize residuals, we need the derivative of the squared geodesic distance of points to the regression geodesic with respect to its base point, *i.e.*, $\nabla_{\mathbf{Y}_0} d^2(\mathbf{Y}_0, \mathbf{Y}_1)$. Since the squared distance can be formulated as $d^2(\mathbf{Y}_0, \mathbf{Y}_1) = \min_{\mathbf{Y}(r)} E(\mathbf{Y}(r))$ for $r_0 = 0$ and $r_1 = 1$, we can derive $\nabla_{\mathbf{Y}_0} d^2(\mathbf{Y}_0, \mathbf{Y}_1)$, at optimality, as $\nabla_{\mathbf{Y}_0} d^2(\mathbf{Y}_0, \mathbf{Y}_1) = -2\dot{\mathbf{Y}}(0)$ (see supplementary material for details). The geodesic connecting the subspaces spanned by $\mathbf{Y}_0, \mathbf{Y}_1$, and its initial condition $\dot{\mathbf{Y}}(0)$ can be efficiently computed following [12], resulting in an efficient computation of $\nabla_{\mathbf{Y}_0} d^2(\mathbf{Y}_0, \mathbf{Y}_1)$ which will be used to solve the regression problem with multiple points. Since the geodesic can connect two points *exactly*, we refer to the case of two points as the *exact* matching problem.

Inexact matching for multiple points. In order to fit a geodesic, given by an initial point $\mathbf{Y}(r_0)$ and an initial velocity $\dot{\mathbf{Y}}(r_0)$, to a collection of points $\{\mathbf{Y}_i\}_{i=0}^{N-1}$ at N measurement instances $\{r_i\}_{i=0}^{N-1}$, exact matching is relaxed to *inexact* matching through the minimization of the energy

$$E(\mathbf{Y}(r_0), \dot{\mathbf{Y}}(r_0)) = \alpha \int_{r_0}^{r_{N-1}} \text{tr } \dot{\mathbf{Y}}(r)^\top \dot{\mathbf{Y}}(r)\, dr + \frac{1}{\sigma^2} \sum_{i=0}^{N-1} d^2(\mathbf{Y}(r_i), \mathbf{Y}_i), \quad (4)$$

fulfilling the constraints for initial conditions $\mathbf{Y}(r_0)^\top \mathbf{Y}(r_0) = \mathbf{I}_p$, $\mathbf{Y}(r_0)^\top \dot{\mathbf{Y}}(r_0) = \mathbf{0}$, and the geodesic equation of (2); $\alpha \geq 0$ and $\sigma > 0$. The search for the curve $\mathbf{Y}(r)$ that minimizes this energy is denoted as *inexact* matching. As in the Euclidean case, $\mathbf{Y}(r_0)$ and $\dot{\mathbf{Y}}(r_0)$ can be interpreted as the initial *intercept* and *slope* that parametrize the geodesic. The first term in (4) is a norm-penalty on the slope of the geodesic, whereas α and σ are balancing constants. In practice, α is typically set to 0, unless we have specific prior knowledge about the slope, similar to a slope-regularized least-squares fit.

3.2 Approximate solution by pairwise searching

One possibility to finding a geodesic that best approximates all data points $\{\mathbf{Y}_i\}$ is to adopt an extension of the well-known random sample consensus (RANSAC) procedure. This consists of picking pairs of points $\{\mathbf{Y}_a, \mathbf{Y}_b\}$; assuming $r_a < r_b$, we can compute the corresponding initial velocity $\dot{\mathbf{Y}}(r_a)$ (using the procedures of [12]) and then integrate the geodesic equation (2) forward and backward to span the full measurement interval of all data points $\{\mathbf{Y}_i\}$. As for a geodesic, $\dot{\mathbf{Y}}(r)^\top \dot{\mathbf{Y}}(r) = const.$, we can measure the regression energy in (4), given the geodesic specified by $\{\mathbf{Y}_a, \mathbf{Y}_b\}$, to evaluate model fit. By either randomly sampling a sufficient number of pairs of data points, or (for small datasets) exhaustively sampling all possible pairs, we obtain the approximate solution as the geodesic of the data point pair with the smallest energy. This solution, denoted

as *GGR (pairwise searching)*, can be used directly, or to initialize the iterative numerical solution described in §3.3. Note that by dividing points into inliers and outliers, given distance thresholds, this defines a RANSAC-like estimation methodology on the Grassmannian.

3.3 Optimal solution by geodesic shooting

To solve the energy minimization problem in (4), we discuss the shooting solution for the special case $N = 2$ first; the general solution then follows accordingly. Specializing (4) to $N = 2$ and $\mathbf{Y}(r_0) = \mathbf{Y}_0$, the geodesic determined by two representers, $\mathbf{Y}_0$ and $\mathbf{Y}_1$, can be obtained by minimizing the shooting energy

$$E(\mathbf{Y}(r_0), \dot{\mathbf{Y}}(r_0)) = \alpha \text{ tr } \dot{\mathbf{Y}}(r_0)^\top \dot{\mathbf{Y}}(r_0) + \frac{1}{\sigma^2} d^2(\mathbf{Y}(r_1), \mathbf{Y}_1) \tag{5}$$

subject to constraints for initial conditions and the geodesic equation. To simplify computations, we replace the second order geodesic constraint by a system of first order. That is, we introduce auxiliary variables $\mathbf{X}_1(r) = \mathbf{Y}(r)$ and $\mathbf{X}_2(r) = \dot{\mathbf{Y}}(r)$ to rewrite the shooting energy of (5) and its constraints. By adding the constraints through Lagrangian multipliers, computing the associated variation, collecting terms and integration by parts, we obtain the optimality conditions with boundary conditions and constraints as shown in the forward and backward steps of Algorithm 1. Since the geodesic is determined by the unknown initial conditions, we need the gradients with respect to the sought-for initial conditions $\nabla_{\mathbf{X}_1(r_0)} E$ and $\nabla_{\mathbf{X}_2(r_0)} E$, which are also given in Algorithm 1[6].

The extension to the full GGR formulation is conceptionally straightforward. The goal is now to fit a best-approximating geodesic, *cf.* (4), to N data points $\{\mathbf{Y}_i\}_{i=0}^{N-1}$. Unlike the case for $N = 2$, instead of a fixed initial condition and one inexact final matching condition, we have (i) both initial $\mathbf{Y}(r_0)$ and $\dot{\mathbf{Y}}(r_0)$ free and (ii) multiple inexact matching terms. This creates jump conditions for the Lagrangian multiplier $\lambda_1(r)$ at each measurement instant when integrating backward in time. Algorithm 1 performs this computation.

4 Experiments

In the experiments, we demonstrate the versatility of our approach on three vision problems with data objects represented on the Grassmannian. First, on traffic speed prediction and crowd counting based on linear dynamical system models of surveillance video clips and second, on modeling the aging trend that is visible in the 2D shape of the human *corpus callosum*.

Dynamical systems as points on the Grassmannian. We demonstrate GGR in the context of modeling video clips by linear dynamical systems (LDS), commonly referred to as *dynamic texture* models [9] in the computer vision literature. For videos, represented by a collection of vectorized frames $\mathbf{y}_1, \ldots, \mathbf{y}_\tau$ with

[6] More details about the derivation are included in the supplementary material.

Algorithm 1: Grassmannian geodesic regression (GGR)

Data: $\{(r_i, \mathbf{Y}_i)\}_{i=0}^{N-1}$, $\alpha \geq 0$ and $\sigma > 0$
Result: $\mathbf{Y}(r_0)$, $\dot{\mathbf{Y}}(r_0)$
Set initial $\mathbf{Y}(r_0)$ and $\dot{\mathbf{Y}}(r_0)$, *e.g.*, using pairwise searching of §3.2.
while *not converged* **do**

Solve $\begin{cases} \dot{\mathbf{X}}_1 = \mathbf{X}_2, \ \mathbf{X}_1(r_0) = \mathbf{Y}(r_0), \\ \dot{\mathbf{X}}_2 = -\mathbf{X}_1(\mathbf{X}_2^\top \mathbf{X}_2), \ \mathbf{X}_2(r_0) = \dot{\mathbf{Y}}(r_0) \end{cases}$ forward for $r \in [r_0, r_{N-1}]$.

Solve $\begin{cases} \dot{\lambda}_1 = \lambda_2 \mathbf{X}_2^\top \mathbf{X}_2, \ \lambda_1(r_{N-1}+) = \mathbf{0}, \\ \dot{\lambda}_2 = -\lambda_1 + \mathbf{X}_2(\lambda_2^\top \mathbf{X}_1 + \mathbf{X}_1^\top \lambda_2), \ \lambda_2(r_{N-1}) = \mathbf{0} \end{cases}$ backward with jump conditions

$$\lambda_1(r_i-) = \lambda_1(r_i+) - \frac{1}{\sigma^2} \nabla_{\mathbf{X}_1(r_i)} d^2(\mathbf{X}_1(r_i), \mathbf{Y}_i)$$

and $\nabla_{\mathbf{X}_1(r_i)} d^2(\mathbf{X}_1(r_i), \mathbf{Y}_i)$ computed as in §3.1. For multiple measurements at a given r_i, the jump conditions for each measurement are added up.

Compute gradient with respect to initial conditions:

$$\nabla_{\mathbf{X}_1(r_0)} E = -(\mathbf{I}_n - \mathbf{X}_1(r_0)\mathbf{X}_1(r_0)^\top)\lambda_1(r_0-) + \mathbf{X}_2(r_0)\lambda_2(r_0)^\top \mathbf{X}_1(r_0),$$
$$\nabla_{\mathbf{X}_2(r_0)} E = 2\alpha \mathbf{X}_2(r_0) - (\mathbf{I}_n - \mathbf{X}_1(r_0)\mathbf{X}_1(r_0)^\top)\lambda_2(r_0).$$

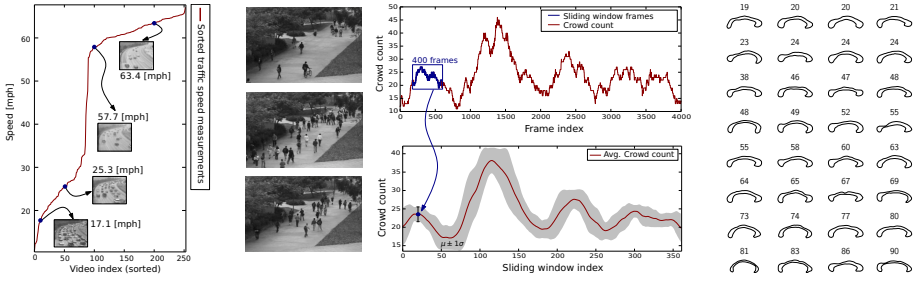
Use a line search with these gradients to update $\mathbf{Y}(r_0)$ and $\dot{\mathbf{Y}}(r_0)$ as described in Algorithm 2 in Appendix A.

end

$\mathbf{y}_i \in \mathbb{R}^n$, the standard dynamic texture model has the form: $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{w}_k$, $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{W})$; $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k$, $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$, with $\mathbf{x}_k \in \mathbb{R}^p$, $\mathbf{A} \in \mathbb{R}^{p \times p}$ and $\mathbf{C} \in \mathbb{R}^{n \times p}$. When relying on the prevalent (approximate) estimation approach of [9], the matrix $\mathbf{C}$ is, by design, of (full) rank p (*i.e.*, the number of states) and by construction we obtain an *observable* system, where the *observability* matrix $\mathbf{O} = [\mathbf{C}\ (\mathbf{CA})\ (\mathbf{CA}^2)\ \cdots\ (\mathbf{CA}^{p-1})]^\top \in \mathbb{R}^{np \times p}$ also has full rank. System identification is not unique in the sense that systems $(\mathbf{A}, \mathbf{C})$ and $(\mathbf{TAT}^{-1}, \mathbf{CT}^{-1})$ with $\mathbf{T} \in \mathcal{GL}(p)$[7] have the same transfer function. Hence, the realization subspace spanned by $\mathbf{O}$ is a point on the Grassmannian $\mathcal{G}(p, n)$ and the observability matrix is a representer of this subspace. In our experiments, we identify an LDS model for a video clip by its $np \times p$ orthonormalized observability matrix.

Shapes as points on the Grassmannian. We also apply GGR in the context of landmark-based shape analysis. A *shape matrix* is constructed based on its m landmarks, $\mathbf{L} = \{(x_1, y_1, ...); (x_2, y_2, ...); \ldots; (x_m, y_m, ...)\}$. Using SVD on the shape matrix, *i.e.*, $\mathbf{L} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, we obtain an affine-invariant shape representation from the left-singular vectors $\mathbf{U}$ [3]. This establishes a mapping from the shape matrix to a point on the Grassmannian (with $\mathbf{U}$ as the representative).

[7] $\mathcal{GL}(p)$ is the general linear group of $p \times p$ invertible matrices.



(a) Traffic surveillance (b) Crowd surveillance (c) Shapes

Fig. 2. Illustration of the datasets: (a) surveillance videos of highway traffic [7] for speed regression; (b) surveillance videos of a sidewalk [8] for regressing *average* crowd count and (c) corpus callosum shapes [11] for shape regression.

4.1 Datasets

Synthetic sine/cosine signals. To first demonstrate GGR on a toy-example, we embed 25 synthetic 2D sine/cosine signals, sampled at 630 points in $[0, 10\pi]$, in $\mathbb{R}^{24}$; the signal frequencies are uniformly sampled in $(0,\ 10)$. The 2D signals $\mathbf{s} \in \mathbb{R}^{2\times 630}$ are then linearly projected via $\overline{\mathbf{s}} = \mathbf{U}\mathbf{s}$, where $\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{24})$ and $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$. Finally, white Gaussian noise with $\sigma = 0.1$ is added to $\overline{\mathbf{s}}$. Given a collection of training signals, our objective is to predict the signal frequency based on the LDS models of the 24-dimensional data.

UCSD traffic dataset [7]. This dataset was introduced in the context of clustering traffic flow patterns with LDS models. It contains a collection of short traffic video clips, acquired by a surveillance system monitoring highway traffic. There are 253 videos in total and each video is roughly matched to the speed measurements from a highway-mounted speed sensor. We use the pre-processed video clips introduced in [7] which were converted to grayscale and spatially normalized to 48×48 pixels with zero mean and unit variance. Our rationale for using an LDS representation for speed prediction is the fact that clustering and categorization experiments in [7] showed compelling evidence that dynamics are indicative of the traffic class. We argue that the notion of speed of an object (*e.g.*, a car) could be considered a property that humans infer from its visual dynamics.

UCSD pedestrian dataset [8]. We use the `Peds1` subset of the UCSD pedestrian dataset which contains 4000 frames with a ground-truth people count (both directions and total) associated with each frame. Similar to [8] we ask the question whether we can infer the number of people in a scene (or clip) without actually detecting the people. While this has been done by resorting to crowd/motion segmentation and Gaussian process regression on low-level features extracted from these segments, we go one step further and try to avoid any preprocessing at all. In fact, our objective is to infer an *average* people count from an LDS

representation of short video segments (*i.e.*, within a temporal sliding window). This is plausible because the visual dynamics of a scene change as people appear in it. Further, an LDS does not only model the dynamics, but also the appearance of videos; both aspects are represented in the observability matrix of the system. We remark, though, that such a strategy does not allow for fine-grain frame-by-frame predictions as in [8]. Yet, it has the advantages of not requiring any pre-selection of features or possibly unstable preprocessing steps such as the aforementioned crowd segmentation.

In our setup, we split the 4000 frames into 37 video clips of 400 frames each, using a sliding window with steps of 100 frames, and associate an average people count with each clip, see Fig. 2(b). The video clips are spatially down-sampled to a resolution of 60×40 pixel (original: 238×158) to keep the observability matrices at a reasonable size. Since the overlap between the clips potentially biases the experiments, we introduce a weighted variant of system identification (see Appendix B) with weights based on a Gaussian function centered at the middle of the sliding window and a standard deviation of 100. While this ensures stable system identification, by still using 400 frames, it reduces the impact of the overlapping frames on the parameter estimates. With this strategy, the average crowd count is localized to a smaller region.

Corpus callosum shapes [11]. To demonstrate GGR for modeling shape changes, we use a collection of 32 *corpus callosum* shapes with ages varying from 19 to 90 years, shown in Fig. 2(c). Each shape is represented by 64 2D boundary landmarks, and is projected to a point on the Grassmannian using the left-singular vectors obtained from the SVD decomposition of the 64×2 shape matrix.

4.2 Results

We compare the performance of (i) *GGR (pairwise searching)* (*i.e.*, the approximate solution), (ii) *Full GGR*, and (iii) *Full piecewise GGR*. For (iii), the regression space is subdivided into regression intervals and a full regression solution is computed for each interval independently. Given a (test) measurement, a regressor is estimated for all intervals. We search over each interval and find the closest point on the geodesic with the smallest distance. The value of the regressor at this optimal point is then regarded as the predicted value for the measurement. For full GGR, we set $\alpha = 0$ because no prior information is known about the measurements, and $\sigma^2 = 1$. Two segments were used in *Full piecewise GGR* and the breakpoint (separating the regression intervals) varied with the dataset, but was roughly chosen to separate the data into two equal-sized groups or two classes. While this is certainly an ad–hoc choice and could be fully data-driven, our choice of two segments is only to demonstrate the easy extensibility of our method to a piecewise regression formulation. To compare the three GGR variants, we report the *mean absolute error (MAE)*, computed over all folds in a cross validation (CV) setup with a dataset-dependent number of folds.

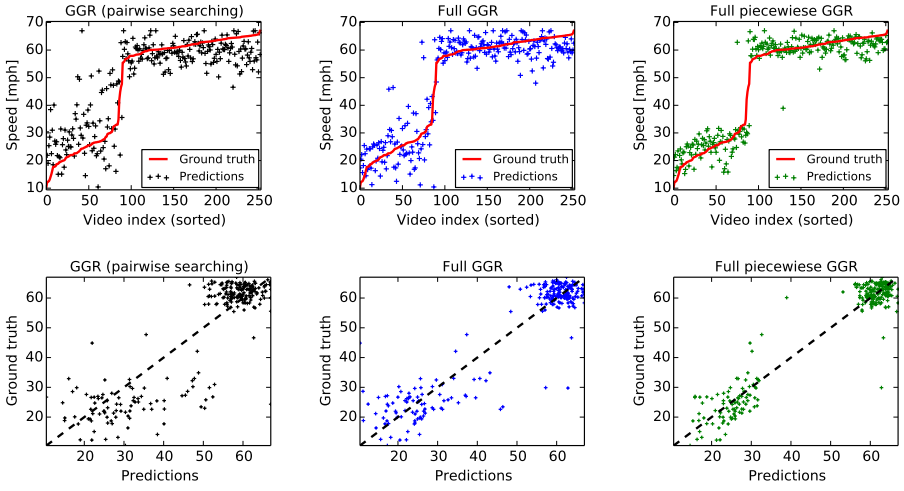


Fig. 3. Visualization of traffic speed predictions via 5-fold cross validation. The top row shows the predictions vs. the videos sorted by *speed*; the bottom row shows the correlation with the ground-truth.

Signal frequency prediction (toy data). For this experiment, the number of LDS states is set to $p = 2$ which is, in theory, sufficient to capture sine/cosine signals. We divide the 25 signals into 5 groups for 5-fold CV. For *Full piecewise GGR*, we regress on the signals with frequencies in the two intervals $(0, 5)$ and $[5, 10)$. The testing MAE ranges from 0.49e-15±0.32e-15 for both *GGR (pairwise searching)* and *Full GGR* to 0.58e-15±0.28e-15 for *Full piecewise GGR*, *cf.* Table 1. On this toy data, this shows that all our regression formulation(s) can essentially capture the data perfectly.

Traffic speed prediction. For each video clip, we estimate LDS models with $p = 10$ states. The breakpoint of *Full piecewise GGR* is set at 50 [mph], which roughly divides the videos into two categories, *i.e.*, fast and slow. Results are reported for 5-fold CV. A visualization of the predictions is shown in Fig. 3 with the predictions versus the sorted speed measurements, as well as the correlation with the ground-truth. As we can see from the MAEs in Table 1, the results gradually improve as we switch from *GGR (pairwise searching)* to *Full GGR* and *Full piecewise GGR*, with a top MAE of 3.35 ± 0.38 [mph] for testing.

Crowd counting. For each of the 37 video clips we extract from the `Peds1` dataset, we estimate LDS models with $p = 10$ states using weighted system identification as described in Appendix B. For *Full piecewise GGR*, the breakpoint is set to a count of 23 people; this choice separates the 37 videos into two groups of roughly equal size. Results are reported for 4-fold CV. From the results shown in Fig. 4, we see that both *Full GGR* and *Full piecewise GGR* provide visually close predictions to the ground-truth. From Table 1, we further see that

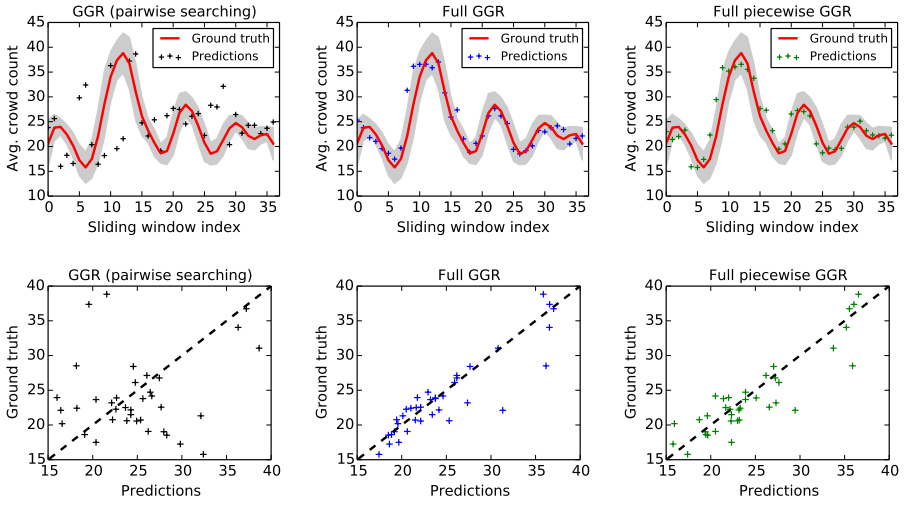


Fig. 4. Visualization of crowd counting results via 4-fold cross validation. The top row shows the crowd count predictions as a function of the sliding window index, overlaid on the ground-truth counts; the bottom row shows the predictions versus the ground-truth. The gray bands indicate the weighted standard deviation ($\pm 1\sigma$) of the number of people in the sliding window.

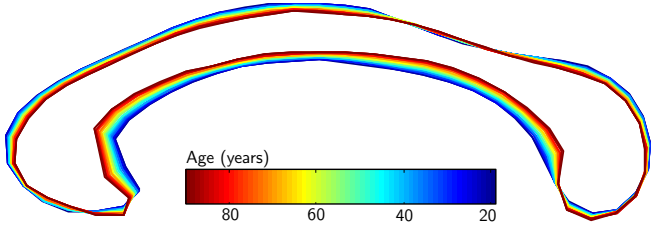


Fig. 5. Corpus callosum shapes along *Full GGR* geodesic; colored by age in years.

these two GGR variants have significantly better performance than the pairwise searching strategy. In fact, *Full GGR* achieves the top prediction by improving from 5.14 ± 0.64 to 1.65 ± 0.79. Although, *Full piecewise GGR* has lowest training error among the three variants, its testing error is higher than for *Full GGR*, indicating an overfit to the data.

Corpus callosum aging. We generate corpus callosum shapes along the geodesic fit by *Full GGR*, as shown in Fig. 5. The shapes are recovered from the points along the geodesic on the Grassmann manifold through scaling by the mean singular values of the SVD results. As we can see, the shape shrinks from blue to red, corresponding to 19 and 90 years of age; this demonstrates the thinning trend of the corpus callosum with age and is consistent with [11].

	GGR (pairwise searching)		Full GGR		Full piecewise GGR	
	Train	Test	Train	Test	Train	Test
Signal freq. ($\times 10^{-15}$)	**0.50 ± 0.09**	**0.49 ± 0.32**	**0.50 ± 0.09**	**0.49 ± 0.32**	0.52 ± 0.06	0.58 ± 0.28
Traffic speed	6.46 ± 0.55	6.20 ± 0.77	2.98 ± 0.20	4.59 ± 0.43	**1.65 ± 0.13**	**3.35 ± 0.38**
Crowd counting	4.27 ± 0.33	5.14 ± 0.64	0.81 ± 0.21	**1.65 ± 0.79**	**0.63 ± 0.08**	2.05 ± 0.88

Table 1. Mean absolute errors (MAE, computed via cross validation) ±1 standard deviation on both training and testing data. Either *Full GGR* or *Full piecewise GGR* give the best results. *Full piecewise GGR* leads to overfitting for the crowd counting case, hence *Full GGR* is preferable in this case.

It is critical to note that since the Grassmann manifold has non-negative sectional curvature, conjugate points do exist. This implies that there can be multiple geodesics that connect any two points, resulting in a potentially non-unique solution for the regression problem. However, Wong [28] proves that geodesics are unique as long as subspace angles ϕ_i are less than $\pi/2$. We evaluated all subspace angles in our experiments against this criteria and found *no* violation which ensures that all estimated geodesics were unique. While the issue of conjugate points exists with any manifold of non-negative curvature, this criteria can certainly serve as a sanity check for any solution to the regression problem.

5 Discussion

In this paper, we developed a general theory for Grassmannian geodesic regression. This allowed us to compute regression geodesics that explain the variation in the data on the Grassmannian. We demonstrated the utility of our method for modeling a dependent Grassmannian-valued variable in the form of observability matrices from LDS and affine-invariant shape data, with respect to a scalar-valued independent variable. We also showed that our formulation naturally extends to piecewise regression models.

The experimental results on the traffic speed data show that the dynamics captured by the LDS models correlate with traffic speed, leading to predictions with an MAE error of 3.35 ± 0.38 [mph]. This is an encouraging result, especially since the dataset has an unbalanced design and requires no higher-level preprocessing (*e.g.*, tracking). For crowd counting, an MAE of 1.65 ± 0.79 does not beat the frame-by-frame counting results in [8] (1.31 for frame counting on `Peds1`, and 0.59 for our measure of *average* counting). However, in our case, information is captured by the LDS model directly from the raw image data, whereas frame-by-frame counting typically requires a collection of suitable features and thus involves more preprocessing. Additionally, our approach is not

directly comparable to [8], since regressing an average people count is influenced by the variation of the counts within the LDS estimation window.

In our shape regression experiment, we show that the resulting estimated geodesic effectively summarizes the trajectory of changes in the corpus callosum for a population. In fact, the corpus callosum exhibits a clear thinning with progressing age. Since the estimated geodesic summarizes the complete nonlinear variability of aging related biological changes, and is compactly represented by its initial conditions, this modeling opens the possibility of nonlinear statistics on changes in (anatomical) shapes.

Some open questions need to be addressed in future work. For example, piecewise GGR has the advantage of greater flexibility but inherently depends upon the *optimal* number of segments. While the breakpoints could, in principle, be chosen in a data-driven way, the increased flexibility makes the model susceptible to overfitting issues (especially with unbalanced data). Furthermore, since we fit the segments independently, this results in discontinuous piecewise geodesic curves. Thanks to the adjoint method it is, however, possible to derive a *continuous-piecewise* GGR variant by constraining the geodesics to match at the segment boundaries (see supplementary material for details).

Another interesting avenue to pursue in future work would be to leverage the concept of *time-warping* in which the time-axis is bent according to some parametric function. This increases flexibility and could be beneficial in vision applications where we have specific prior knowledge about the data, *e.g.*, traffic speed measurements exhibiting saturation in the upper and lower ranges. The general strategy to incorporate time-warping into the regression formulation is developed in [17] and exemplified on the Grassmannian, using the numerical machinery developed in this work.

A Line search on the Grassmannian

Performing a line search is not as straightforward as in Euclidean space since we need to assure that the constraints for $\mathbf{Y}(r_0)$ and $\dot{\mathbf{Y}}(r_0)$ are fulfilled for any given step. In particular, changing $\mathbf{Y}(r_0)$ will change the associated tangent vector $\dot{\mathbf{Y}}(r_0)$. Once, we have updated $\mathbf{Y}(r_0)$ to $\mathbf{Y}^u(r_0)$ by moving along the geodesic defined by $\mathbf{Y}(r_0)$ and the gradient of the energy with respect to this initial point, *i.e.*, $\nabla_{\mathbf{X}_1(r_0)}E$, we can transport the tangent $\dot{\mathbf{Y}}(r_0)$ to $\mathbf{Y}^u(r_0)$ using the closed form solution for *parallel transport* of [10]. In particular,

$$\dot{\mathbf{Y}}^u(r_0) = [\mathbf{Y}(r_0)\mathbf{V}\ \mathbf{U}] \begin{pmatrix} -\sin t\boldsymbol{\Sigma} \\ \cos t\boldsymbol{\Sigma} \end{pmatrix} \mathbf{U}^\top + (\mathbf{I}_n - \mathbf{U}\mathbf{U}^\top)\dot{\mathbf{Y}}(r_0) \tag{6}$$

where $\mathbf{H} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ is the compact SVD of the tangent vector at $\mathbf{Y}(r_0)$ along the geodesic connecting $\mathbf{Y}(r_0)$ and $\mathbf{Y}^u(r_0)$. Algorithm 2 lists the line search procedure in full technical detail.

Algorithm 2: Grassmannian equivalent of $x^{k+1} = x^k - \Delta t g$, where Δt is the timestep and g is the gradient.

Data: $\mathbf{Y}(r_0)$, $\dot{\mathbf{Y}}(r_0)$, $\nabla_{\mathbf{Y}(r_0)} E$, $\nabla_{\dot{\mathbf{Y}}(r_0)} E$, Δt
Result: Updated $\mathbf{Y}^u(r_0)$ and $\dot{\mathbf{Y}}^u(r_0)$
Compute $\dot{\overline{\mathbf{Y}}}^u(r_0) = \dot{\mathbf{Y}}(r_0) - \Delta t \nabla_{\mathbf{X}_2(r_0)} E$
Compute $\mathbf{Y}^u(r_0)$ by flowing for Δt along geodesic with initial condition $(\mathbf{Y}(r_0), -\nabla_{\mathbf{X}_1(r_0)} E)$ (using algorithm in [10])
Transport $\dot{\overline{\mathbf{Y}}}^u(r_0)$ along the geodesic connecting $\mathbf{Y}(r_0)$ to $\mathbf{Y}^u(r_0)$, using (6), resulting in $\dot{\overline{\mathbf{Y}}}^u_T(r_0)$
Project updated initial velocity onto the tangent space (for consistency): $\dot{\mathbf{Y}}^u(r_0) \leftarrow (\mathbf{I}_n - \mathbf{Y}^u(r_0)\mathbf{Y}^u(r_0)^\top)\dot{\overline{\mathbf{Y}}}^u_T(r_0)$.

B Temporally localized system identification

To support a non-uniform weighting of samples during system identification, we propose a *temporally* localized variant of [9]. This is beneficial in situations where we need a considerable number of frames for stable system identification, yet not all samples should contribute equally to the LDS parameter estimates. Specifically, given the measurement matrix $\mathbf{M} = [\mathbf{y}_1, \cdots, \mathbf{y}_\tau]$ and a set of weights $\mathbf{w} = [w_1, \cdots, w_\tau]$, such that $\sum_i w_i = \tau$, we perform a weighted SVD of $\mathbf{M}$, *i.e.*,

$$\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{M}\mathrm{diag}(\sqrt{\mathbf{w}}) \ . \tag{7}$$

Then, as in [9], $\mathbf{C} = \mathbf{U}$ and $\mathbf{X} = \boldsymbol{\Sigma}\mathbf{V}^\top$. Once the state matrix $\mathbf{X}$ has been determined, $\mathbf{A}$ can be computed as $\mathbf{A} = \mathbf{X}_2^\tau \mathbf{W}^{\frac{1}{2}} (\mathbf{X}_1^{\tau-1}\mathbf{W}^{\frac{1}{2}})^\dagger$, where † denotes the pseudoinverse, $\mathbf{X}_2^\tau = [\mathbf{x}_2, \cdots, \mathbf{x}_\tau]$, $\mathbf{X}_1^{\tau-1} = [\mathbf{x}_1, \cdots, \mathbf{x}_{\tau-1}]$ and $\mathbf{W}^{\frac{1}{2}}$ is a diagonal matrix with $W_{ii}^{\frac{1}{2}} = [\frac{1}{2}(w_i + w_{i+1})]^{1/2}$.

Acknowledgements This work was supported by NSF grants EECS-1148870, EECS-0925875 and IIS-1208522.

References

1. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press (2008)
2. Batzies, E., Machado, L., Silva Leite, F.: The geometric mean and the geodesic fitting problem on the Grassmann manifold, unpublished manuscript (available at http://www.mat.uc.pt/preprints/ps/p1322.pdf)
3. Begelfor, E., Werman, W.: Affine invariance revisited. In: CVPR (2006)
4. Boothby, W.: An Introduction to Differentiable Manifolds and Riemannian Geometry. Academic Press (1986)
5. Camarinha, M., Leite, F.S., Crouch, P.: Splines of class C^k on non-Euclidean spaces. IMA J. Math. Control Info. 12(4), 399–410 (1995)

6. Çetingül, H., Vidal, R.: Intrinsic mean shift for clustering on Stiefel and Grassmann manifolds. In: CVPR (2009)
7. Chan, A., Vasconcelos, N.: Classification and retrieval of traffic video using auto-regressive stochastic processes. In: Intelligent Vehicles (2005)
8. Chan, A., Vasconcelos, N.: Counting people with low-level features and Bayesian regression. Trans. Image Process. 12(4), 2160–2177 (2012)
9. Doretto, G., Chiuso, A., Wu, Y., Soatto, S.: Dynamic textures. Int. J. Comput. Vision 51(2), 91–109 (2003)
10. Edelman, A., Arias, T., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. 20(2), 303–353 (1998)
11. Fletcher, T.P.: Geodesic regression and the theory of least squares on Riemannian manifolds. Int. J. Comput. Vision 105(2), 171–185 (2012)
12. Gallivan, K., Srivastava, A., Xiuwen, L., Dooren, P.V.: Efficient algorithms for inferences on Grassmann manifolds. In: Statistical Signal Processing Workshop. pp. 315–318 (2003)
13. Gopalan, R., Li, R., Chellappa, R.: Domain adaption for object recognition: An unsupervised approach. In: ICCV (2011)
14. Hamm, J., Lee, D.: Grassmann discriminant analysis: A unifying view on subspace learning. In: ICML (2008)
15. Hinkle, J., Fletcher, P.T., Joshi, S.: Intrinsic polynomials for regression on Riemannian manifolds. J. Math. Imaging Vis. pp. 1–21 (2014)
16. Hong, Y., Joshi, S., Sanchez, M., Styner, M., , Niethammer, M.: Metamorphic geodesic regression. In: MICCAI (2012)
17. Hong, Y., Singh, N., Kwitt, R., Niethammer, M.: Time-warped geodesic regression. In: MICCAI (2014)
18. Jayasumana, S., Hartley, R., Salzmann, M., Li, H., Harandi, M.: Optimizing over radial kernels on compact manifolds. In: CVPR (2014)
19. Lui, Y.: Human gesture recognition on product manifolds. JMLR 13, 3297–3321 (2012)
20. Lui, Y., Beveridge, J., Kirby, M.: Canonical Stiefel quotient and its application to generic face recognition in illumination spaces. In: BTAS (2009)
21. Mittal, S., Meer, P.: Conjugate gradient descent on Grassmann manifolds for robust subspace estimation. Image Vision Comput. 30, 417–427 (2012)
22. Niethammer, M., Huang, Y., Vialard, F.X.: Geodesic regression for image time-series. In: MICCAI (2011)
23. Noakes, L., Heinzinger, G., Paden, B.: Cubic splines on curved spaces. IMA J. Math. Control Info. 6(4), 465–473 (1989)
24. Rentmeesters, Q.: A gradient method for geodesic data fitting on some symmetric Riemannian manifolds. In: CDC-ECC (2011)
25. Singh, N., Hinkle, J., Joshi, S., Fletcher, P.: A vector momenta formulation of diffeomorphisms for improved geodesic regression and atlas construction. In: ISBI (2013)
26. Singh, N., Niethammer, M.: Splines for diffeomorphic image regression. In: MICCAI (2014)
27. Turuga, P., Veeraraghavan, A., Srivastrava, A., Chellappa, R.: Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. IEEE Trans. Pattern Anal. Mach. Intell. 33(11), 2273–2285 (2011)
28. Wong, Y.C.: Differential geometry of Grassmann manifolds. Proc. Natl. Acad. Sci. USA 57(3), 589–594 (1967)
29. Zheng, J., Liu, M.Y., Chellappa, R., Phillips, P.: A Grassmann manifold-based domain adaption approach. In: ICML (2012)