EFFICIENT ALGORITHMS FOR INFERENCES ON GRASSMANN MANIFOLDS

Kyle A. Gallivan, Anuj Srivastava†, Xiuwen Liu†*

Florida State University
Tallahassee, FL, USA 32306

*Paul Van Dooren**

Université catholique de Louvain
Louvain-la-Neuve, Belgium

ABSTRACT

Linear representations and linear dimension reduction techniques are very common in signal and image processing. Many such applications reduce to solving problems of stochastic optimizations or statistical inferences on the set of all subspaces, i.e. a Grassmann manifold. Central to solving them is the computation of an "exponential" map (for constructing geodesics) and its inverse on a Grassmannian. Here we suggest efficient techniques for these two steps and illustrate two applications: (i) For image-based object recognition, we define and seek an optimal linear representation using a Metropolis-Hastings type, stochastic search algorithm on a Grassmann manifold. (ii) For statistical inferences, we illustrate computation of sample statistics, such as mean and variances, on a Grassmann manifold.

1. INTRODUCTION

Studies of linear systems is very common in all branches of science and engineering. Linear systems are both easier to design and analyze, and hence, linear approximations of more general systems are quite popular. High dimensional systems are commonly studied after undergoing linear dimension reduction. Examples include image component analysis where images are projected onto low-dimensional (linear) subspaces, such as principal subspaces or independent component subspaces, before statistical algorithms are applied. In signal processing, the problems of transmitter detection and tracking using sensor array data are intimately related to estimation/tracking of principal subspaces of the observed data. Such problems, and many others, are now being viewed as those of optimization or inferences on Grassmann manifolds, the sets of linear subspaces of a vector space.

Consider the Grassmann manifold of all k-dimensional subspaces of $\mathbb{R}^n$, denoted by $\mathcal{G}_{n,k}$. Several textbooks describe the structure of $\mathcal{G}_{n,k}$ with a focus on its geometry and calculus. Edelman et al. [1] use the differential geometry of Grassman and other orthogonally constrained manifolds in order to provide gradient solutions to optimization problems. Srivastava et al. derived the geodesics and analyzed the associated structure via Lie group theory [2, 3] for addressing the problem of subspace tracking as that of nonlinear filtering on $\mathcal{G}_{n,k}$. Liu et al. [4] have described a stochastic gradient technique for solving an optimization problem on $\mathcal{G}_{n,k}$ relating linear representations of images.

In this paper we focus on deriving efficient algorithms for use in above-mentioned applications. Towards that goal, a convenient approach is to view $\mathcal{G}_{n,k}$ as the quotient space $SO(n)/(SO(k) \times SO(n-k))$ where $SO(n)$ is the Lie group of $n \times n$ real-valued rotation matrices. A Lie group is a differentiable manifold with a group structure. $SO(n)$ forms a group with matrix multiplication as the group operation. If subspaces by represented by their orthonormal bases in $\mathbb{R}^{n\times k}$, then the equivalence with respect to the subgroup $SO(n-k)$ is already accounted for and only the subgroup $SO(k)$ needs to be removed. In other words, for an orthonormal basis $S \in \mathbb{R}^{n\times k}$, all the bases contained in the set $\{SU : U \in SO(k)\}$, called the *orbit* of S, span the same subspace and should be treated as equivalent. Figure 1 pictorially illustrates this idea where each subspace, corresponding to an equivalence class of bases, is denoted by a vertical line.

An advantage of this approach is to utilize well-known results from Lie group theory in deriving algorithms on $SO(n)$. It is well known that geodesic paths on $SO(n)$ are given by one-parameter exponential flows, i.e. $t \mapsto \exp(tB)$, where $B \in \mathbb{R}^{n\times n}$ is a skew-symmetric matrix. Viewing $\mathcal{G}_{n,k}$ as a quotient space of $SO(n)$ one can specify geodesics on $\mathcal{G}_{n,k}$ as well. Geodesics in $SO(n)$ are also geodesics in $\mathcal{G}_{n,k}$ as long as they are perpendicular to the orbits generated by the subgroup $SO(k) \times SO(n-k)$. This implies that geodesics in $\mathcal{G}_{n,k}$ are given by one-parameter exponential flows $t \mapsto \exp(tB)$ where skew-symmetric B is further restricted to be of the form

$$B = \begin{pmatrix} 0 & A^T \\ -A & 0 \end{pmatrix}, \ A \in \mathbb{R}^{(n-k)\times k}. \qquad (1)$$

Please refer to [3] for details. Superscript T denotes the matrix transpose. The sub-matrix A specifies the direction and the speed of geodesic flow. In Figure 1, the flows should be horizontal, or perpendicular to the vertical orbits, to be geodesics in $\mathcal{G}_{n,k}$.

Geodesics are central to solving several problems on $\mathcal{G}_{n,k}$. For instance, the solution of an optimization problem can be achieved using a piecewise-geodesic flow driven by a gradient vector field [1]. $\mathcal{G}_{n,k}$ becomes a metric space using the geodesic lengths as a metric, or one can define means and covariances of probability distributions on $\mathcal{G}_{n,k}$ using geodesic paths. There are two key computations that are needed in evaluating geodesics on $\mathcal{G}_{n,k}$. Let $\mathcal{S}_0$, $\mathcal{S}_1$ be two k-dimensional subspaces of $\mathbb{R}^n$, represented by the bases S_0 and S_1, respectively, and let $A \in \mathbb{R}^{(n-k)\times k}$ be any matrix. The process generated by the one-parameter flow $\Psi(t) = Q\exp(tB)J$, where $Q \in SO(n)$ such that $Q^T S_0 = J$ and $J = \begin{bmatrix} I_k \\ 0_{n-k,k} \end{bmatrix}$, is a geodesic flow in $\mathcal{G}_{n,k}$ that starts from $\mathcal{S}_0$. Here, B is the skew-symmetric, block-diagonal matrix given in Eqn. 1.

We outline three specific tasks for which we provide efficient algorithms. These tasks are required in any problem of optimization or statistical inferences on $\mathcal{G}_{n,k}$.

1. **Task 1**: Given the skew-symmetric and block-diagonal structure of B (Eqn. 1), we are interested in a technique for effi-

*Supported by NSF Grant CCR-9912415.

†Supported by NSF Grants NMA-201012010 and DMS-0101429.

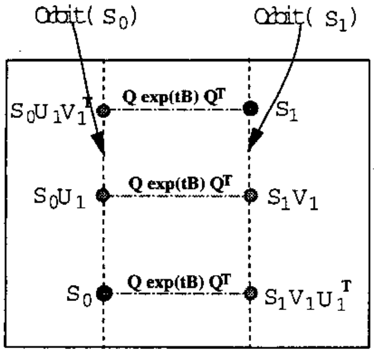

Fig. 1. A pictorial illustration of Grassmann manifold as a quotient space. S_0 and S_1 are bases of two different k-dimensional subspaces of $\mathbb{R}^n$. Geodesics in $\mathcal{G}_{n,k}$ flow perpendicular to the orbits.

cient computation of $\Psi(t)$, for several values of t, without resorting to the full $O(n^3)$ exponentiation of B. In Figure 1, this task amounts to computing the horizontal (broken) line starting from S_0.

2. **Task 2**: Given S_0 and S_1, one is often interested in finding an appropriate direction matrix A such that geodesic along that direction, and starting at S_0, reaches S_1 in unit time . In Figure 1 the goal is to find the direction B (and hence A) of geodesic flow from one orbit to another.

3. **Task 3**: Given S_0 and S_1, find the geodesic path Ψ that starts from S_0 and passes through the orbit of S_1 in unit time. This can be accomplished using the first two tasks but in cases where we do not need to make explicit the direction A of the geodesic flow, it can be done more efficiently.

The first computation is for exponentiation while the second one is for its inverse or "logarithm" on $\mathcal{G}_{n,k}$. In this paper, we utilize the geometry of $\mathcal{G}_{n,k}$ and some past results from linear algebra, the CS decomposition in particular, to derive efficient algorithms for these two computations. Then, we demonstrate these ideas in the context of two applications, one in image component analysis and image-based object recognition and other in computing statistics from sample points on $\mathcal{G}_{n,k}$.

This paper is organized as follows: Section 2 analyzes geodesics on $\mathcal{G}_{n,k}$ and uses standard results from linear algebra to address the three tasks outlined earlier. Section 4 presents two applications of these ideas in image analysis and sample statistics.

2. ALGORITHMS FOR EFFICIENT COMPUTATIONS

Let S_0 and S_1 be two matrices in $\mathbb{R}^{n\times k}$ whose columns are orthogonal bases for the k-dimensional spaces $\mathcal{S}_0$ and $\mathcal{S}_1$ and $Q = (S_0 \;\; C_0)$ be an $n \times n$ orthogonal completion of S_0. The computation of Q, given S_0, is discussed later in Section 2.1. Let $U_1\Gamma V_1^T$ be a singular value decomposition (SVD) of the $k \times k$ matrix $S_0^T S_1$. This decomposition is important for several reasons. First, it helps in finding the nearest elements on the orbit of S_1 given any element on the orbit of S_0. For instance, the element nearest to S_0 is $S_1V_1U_1^T$ while the element nearest to $S_0U_1V_1^T$ is S_1 itself as shown in Figure 1. Secondly, elements of Γ relate to the angles of rotation from $\mathcal{S}_0$ to $\mathcal{S}_1$.

As Figure 1 suggests, the geodesic connecting $\mathcal{S}_0$ and $\mathcal{S}_1$ can be stated in several similar ways depending upon the starting point. A convenient way is to connect the bases $\bar{S}_0 = S_0U_1$ and $\bar{S}_1 = S_1V_1$, the so-called *canonical bases.* The geodesic $\Psi(t) = Q\exp(tB)J$ can be re-written in terms of the canonical bases by multiplying on right by U_1:

$$\begin{aligned} \bar{\Psi}(t) &= Q\exp(tB)Q^T\bar{S}_0 \\ &= QUR(t)U^TQ^T\bar{S}_0 \end{aligned} \quad (2)$$

where $\exp(tB) = UR(t)U^T$. The matrix $U \in SO(n)$ is block diagonal $U = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix}$, where U_1 is as defined earlier and $U_2 \in SO(n-k)$. The matrix $R(t) \in \mathbb{R}^{n\times n}$ takes the form:

$$R(t) = \begin{pmatrix} \tilde{R}(t) & 0 \\ 0 & I_{n-2k} \end{pmatrix}, \text{ where } \tilde{R}(t) = \begin{pmatrix} \Gamma(t) & \Sigma(t) \\ -\Sigma(t) & \Gamma(t) \end{pmatrix}.$$

The matrices $\Gamma(t)$, $\Sigma(t) \in \mathbb{R}^{k\times k}$ are diagonal and nonnegative with elements $\gamma_i = \cos(t\theta_i)$ and $\sigma_i = \sin(t\theta_i)$ for $0 \le \theta_1 \le \cdots \le \theta_k \le \pi/2$ respectively. These θ_is from the angles of rotation from $\mathcal{S}_0$ to $\mathcal{S}_1$. A similar characterization of this geodesic flow can also be reached using the ideas presented in [5]. Substituting for $R(t)$ in Eqn. 2, we obtain:

$$\begin{aligned} \bar{\Psi}(t) &= QUR(t)U^TJU_1 = QUR(t)J \\ &= QU\begin{pmatrix} \Gamma(t) \\ \Sigma(t) \end{pmatrix} = Q\begin{pmatrix} U_1\Gamma(t) \\ -\tilde{U}_2\Sigma(t) \end{pmatrix}, \end{aligned} \quad (3)$$

where $\tilde{U}_2$ is an $(n-k)\times k$ matrix made up of the first k columns of U_2. In this notation, it can be shown that the sub-matrix $A \in \mathbb{R}^{(n-k)\times k}$ inside the matrix B (Eqn. 1) has the SVD $A = \tilde{U}_2\Theta U_1^T$, where Θ is a diagonal matrix with elements given by θ_is.

From a practical viewpoint, computation of geodesics in $\mathcal{G}_{n,k}$ must have complexity far below the $O(n^3)$ implied by the expression $\exp(tB)$. Rotating from one k-dimensional space or basis to another can involve at most $2k$ directions since, in the worst case, all k original directions must be replaced by k new ones. The form of $\tilde{R}(t)$ and the fact that B can have a rank of at most $2k$ (Eqn. 1) also support that idea. Therefore, we seek an algorithm that uses $O(nk^2)$ computations for computing geodesics and related terms. Furthermore, if it is necessary to evaluate the geodesic at many values of t, the cost per point must be kept to $O(nk)$. Edelman et al. [1] suggest a form of geodesic that satisfies these computational constraints when the initial basis $\Psi(0)$ is given along with a direction $\dot{\Psi}(0) \in \mathbb{R}^{n\times k}$. We seek computationally efficient algorithms for use in related situations.

Now we return to the three tasks laid out in the introduction. In all these cases we are given S_0 and need to determine a completion Q such that $Q^TS_0 = J$. This computation can be performed in $O(nk^2)$ computations as described later in Section 2.1.

Task 1: Here we are given: (i) a basis S_0 for the initial subspace $\mathcal{S}_0$ on $\mathcal{G}_{n,k}$ and (ii) a matrix $A \in \mathbb{R}^{(n-k)} \times k$ that determines direction of geodesic flow. The goal is to be sample the resulting geodesic at various values of t including $t = 1$.

Let $A = \tilde{U}_2\Theta U_1^T$ be the compact SVD of the direction matrix. From this decomposition, we can determine $\Gamma(t)$ and $\Sigma(t)$, and along with $\tilde{U}_2$, U_1 substitute them back in Eqn. 3 to evaluate $\Psi(t)$.

This idea is computationally feasible for evaluating only a small number of points on the geodesic due to the $O(nk^2)$ cost of applying Q. If the number of points to be evaluated is large, the following approach can be utilized. Since $\overline{\Psi}(t) = Q\exp(tB)JU_1$, we have

$$\dot{\overline{\Psi}}(0) = Q\begin{pmatrix} 0 \\ -A \end{pmatrix} U_1 = -C_0\tilde{U}_2\Theta = -D\Theta \qquad (4)$$

for $D \equiv C_0\tilde{U}_2$. Therefore,

$$\overline{\Psi}(t) = S_0U_1\Gamma(t) - (C_0\tilde{U}_2)\Sigma(t) = S_0U_1\Gamma(t) - D\Sigma(t). \quad (5)$$

To compute $\Psi(t)$, first compute D using Q, A, Θ, and U_1 (Eqn. 4), and then substitute them in Eqn. 5. An important advantage of using the geodesic between the canonical bases, as opposed to any other bases, is that the two matrices $\Gamma(t)$ and $\Sigma(t)$ are diagonal only for this representation. In the interest of numerical stability one can combine the two steps to obtain the second term in Eqn. 5 as $\dot{\overline{\Psi}}(0)(\Theta^{-1}\Sigma(t))$ more reliably.

The matrix D can be computed first in $O(nk^2)$ operations and then the cost of evaluating $\overline{\Psi}(t)$ at each value of t follows with $O(nk)$ operations.

Task 2: Here we are given two bases, S_0 and S_1, for the initial and final subspaces on the geodesic, and the goal is to find the direction matrix $A \in \mathbb{R}^{(n-k)\times k}$ of the geodesic connecting the two subspaces.

We first compute Q^TS_1 and then compute its thin CS decomposition, i.e.,

$$Q^TS_1 = \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} U_1 & 0 \\ 0 & U_2 \end{pmatrix}\begin{pmatrix} \Gamma(1) \\ -\Sigma(1) \\ 0 \end{pmatrix} V_1^T = \begin{pmatrix} U_1 & 0 \\ 0 & \tilde{U}_2 \end{pmatrix}\begin{pmatrix} \Gamma(1) \\ -\Sigma(1) \end{pmatrix} V_1^T$$

This decomposition costs $O(nk^2)$ and may also be viewed (and computed) as a generalized SVD [6]. Now A is easily recovered by determining Θ via the arcsin or arccos that is numerically reliable given the size of the angle, and evaluating $A = \tilde{U}_2\Theta U_1^T$. A can be also computed via a numerically more sensitive form $A = -YV_1\Sigma^{-1}\Theta U_1^T$. Note that if we have $\theta_i = 0$ (or close to it) then the i-th diagonal element of $\Sigma^{-1}\Theta$ is set to 1 in order to compute the correct values in A.

Task 3: Here we are given bases, S_0 and S_1, for the initial and final spaces on the geodesic, and the goal is to be sample $\overline{\Psi}(t)$ for several values of t without explicitly computing the direction matrix A.

From the SVD of $S_0^TS_1 = U_1\Gamma V_1^T$ and Eqn. 2, $\overline{\Psi}(t) = \overline{S}_0\Gamma(t) - D\Sigma(t)$ where D is as defined earlier. Clearly, we cannot afford to compute all of the large matrix D but as before we need the direction D in order to have a cost per t value of $O(nk)$. We do not have A so may not use the technique of Task 1. We do, however, have $\overline{S}_1 = S_1V_1$. Evaluating the flow at time $t = 1$, we have $\overline{S}_1 = \overline{S}_0\Gamma(1) - D\Sigma(1)$, and

$$\overline{D} \equiv -D\Sigma(1) = \overline{S}_1 - \overline{S}_0\Gamma(1)\,.$$

Now the geodesic flow can be written as: $\overline{\Psi}(t) = \overline{S}_0\Gamma(t) + \overline{D}\Omega(t)$, where $\Omega(t) \equiv \Sigma(1)^{-1}\Sigma(t)$. If θ_i is small we set $\omega_i(t) = \sin(t\theta_i)/\sin(\theta_i) \approx t$ in order to improve numerical reliability. The computation of $\overline{D}$ requires $O(nk)$ and the recurring cost is also $O(nk)$.

2.1. Key Computational Steps

The algorithms discussed above achieve the required complexity of $O(nk^2)$ preprocessing with $O(nk)$ cost per time point when sampling the geodesic curve. Algorithms for the SVD can be implemented reliably [6] and the computation of principal angles and vectors is addressed by Björck and Golub [7]. Stewart discusses a reliable algorithm to determine the CS decomposition in [8], and the work of Paige and Wei [9] provides useful generalizations.

The transformation Q can be computed via Householder reflectors with a complexity of $O(nk^2)$ [6]. Its form can be chosen so that its application to an $n \times k$ matrix also requires $O(nk^2)$. However, we can reduce the complexity of producing Q to $O(k^3)$ while insuring stability if we rotate the basis for S_0, i.e., $G^TS_0^T = \begin{pmatrix} L^T & G^TS_{02}^T \end{pmatrix}$, where $G \in SO(k)$ and chosen so that $L \in \mathbb{R}^{k\times k}$ is triangular with negative diagonal elements. This requires $O(k^3)$ computations and we have

$$Q^T = I_n - \begin{pmatrix} L - I_k \\ S_{02}G \end{pmatrix}(I - L)^{-1}\begin{pmatrix} L^T - I_k & G^TS_{02}^T \end{pmatrix},$$
$$S_1^TQ_0 = \begin{pmatrix} S_{11}^T & S_{12}^T \end{pmatrix}Q_0 = \begin{pmatrix} S_1^TS_0G^T & S_{12}^T + Z^TG^TS_{02}^T \end{pmatrix}$$

where $(L - I_k)Z = \begin{pmatrix} G^TS_0^TS_1 - S_{11} \end{pmatrix}$.

3. APPLICATIONS

We present two applications of the efficient algorithms described earlier. One relates to finding the best linear representation of images for application in image-based object recognition, while the second deals with computing means and covariances on Grassmann manifolds.

1. Optimal Component Analysis: High dimensionality of observed images implies that the task of recognizing objects (from images) will generally involve excessive memory storage and computation. It also prohibits effective use of statistical techniques in image analysis since statistical models on high-dimensional spaces are both difficult to derive and to analyze. This motivates a search for representations that can reduce image dimensions or induce representations that are relatively invariant to the unwanted perturbations. One idea is to project images linearly to some pre-defined low-dimensional subspace, and use the projected values for analyzing images. For instance, let S be an $n \times k$ orthogonal matrix denoting a basis of a k-dimensional subspace of $\mathbb{R}^n$ ($n >> k$), and let I be an image reshaped into an $n \times 1$ vector. Then, the vector $a(I) = U^TI \in \mathbb{R}^k$ becomes a k-dimensional representation of I. In this setup, several bases including principal component analysis (PCA) and Fisher discriminant analysis (FDA) have widely been used. Although they satisfy some optimality criteria, they may not necessarily be optimal for a specific application at hand.

We are interested in using linear representations of images in recognition of objects from their images, and define $F(S)$ to the recognition performance on a data set resulting from choosing S for projecting images into $\mathbb{R}^k$ (see [4] for details). We seek optimal subspace: $\hat{S} = \text{argmax}_{S\in\mathcal{G}_{n,k}} F(S)$, and utilize the following algorithm to solve for it.

Algorithm 1 Stochastic Gradient Search: *Let $X(0) \in \mathcal{G}_{n,k}$ be any initial condition. Set $t = 0$.*

1. *Calculate the gradient direction matrix $A(X_t)$ of F using numerical approximations as described in [4].*

2. *Generate $k(n-k)$ independent realizations, w_{ij}s, from standard normal density. Calculate a candidate value Y according to $\Psi(1)$ starting from X_t in the direction of $(A+\sqrt{D_t}W)$ (Task 1).*
3. *Compute $F(Y)$, $F(X_t)$, and set $dF = F(Y) - F(X_t)$.*
4. *Set $X_{t+1} = Y$ with probability $\min\{\exp(dF/D_t), 1\}$, else set $X_{t+1} = X_t$.*
5. *Decrease the temperature D_t to D_{t+1}, set $t = t+1$, and go to Step 1.*

Shown in Figure 2 are four examples of $F(X_t)$ plotted versus the time t, each starting from a different initial condition.

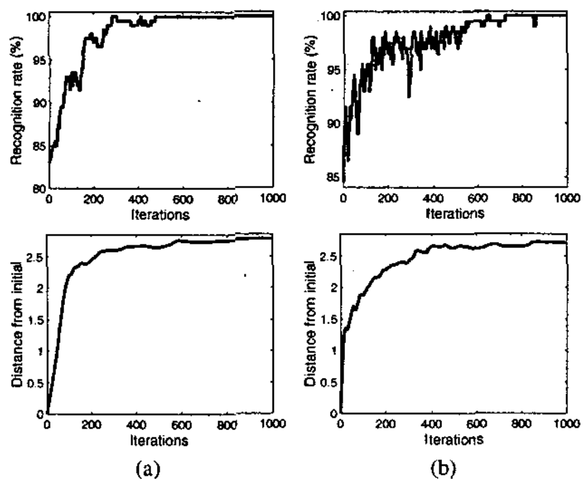

Fig. 2. Plots of $F(X_t)$ (top) and geodesic distance of X_t from X_0 (bottom) versus t for different initial conditions. (a) X_0 =PCA, (b)X_0 = ICA. For these curves, $n = 154$ and $k = 5$.

2. Sample Statistics of Subspaces: Any problem of statistical inference on $\mathcal{G}_{n,k}$ requires computation of sample statistics. In view of the nonlinearity of $\mathcal{G}_{n,k}$, it is not straightforward to define and compute even basic statistics such as means and covariances. There are two types of definitions popularly used : (i) Extrinsic statistics, where $\mathcal{G}_{n,k}$ is embedded in a larger Euclidean space, statistics are computed in this larger space and then projected back to $\mathcal{G}_{n,k}$ [10]. Non-uniqueness of embedding leads to non-uniqueness of statistics although the computations are relatively simple here. (ii) Intrinsic statistics, where the Riemannian structure of $\mathcal{G}_{n,k}$ is used to define uniquely statistics of interest [11]. The computation of intrinsic mean requires an iterative procedure with a need for both exponentiation and logarithm in each step. An algorithm for computing intrinsic mean and covariance of subspaces with orthonormal bases $S_i \in \mathbb{R}^{n\times k}$, $i = 1, \ldots, m$ on $\mathcal{G}_{n,k}$ is stated next.

Algorithm 2 *Set $j = 0$. Choose some time increment $\epsilon \leq \frac{1}{n}$. Choose a point $\mu_0 \in \mathcal{G}_{n,k}$ as an initial guess of the mean. (For example, one could just take $\mu_0 = S_1$.)*

1. *For each $i = 1, \ldots, m$ choose the tangent vector $B_i \in T_{\mu_j}(\mathcal{G}_{n,k})$ which is tangent to the shortest geodesic from μ_j to S_i, and whose norm is equal to the length of this shortest geodesic (Task 2). The vector $\bar{B} = \sum_{i=1}^{n} B_i$ form the direction matrix for updating μ_j.*
2. *Flow for time ϵ along the geodesic which starts at μ_j and has velocity vector $\bar{B}$ (Task 1). Call the point where you end up μ_{j+1}, i.e. $\mu_{j+1} = \Psi(\epsilon)$ starting at μ_j in the direction given by $\bar{B}$.*
3. *If converged, set $\mu = \mu_j$. Else set $j = j+1$, and go to Step 1.*
4. *Similar to Step 1, compute the directions B_is for geodesics from μ to S_is. Extract the sub-matrices A_is from B_is, and compute their sample covariance matrix after converting A_is into column vectors.*

4. CONCLUSION

We have presented efficient algorithms for two key tasks in solving problems on Grassmann manifolds: computation of exponential map for evaluating geodesics, and computation of its inverse for direction finding. Our current work focuses on numerical stability of these algorithms in the context of applications in image analysis.

5. REFERENCES

[1] A. Edelman, T. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM Journal of Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303–353, 1998.

[2] A. Srivastava, "A Bayesian approach to geometric subspace estimation," *IEEE Transactions on Signal Processing*, vol. 48, no. 5, pp. 1390–1400, 2000.

[3] A. Srivastava and E. Klassen, "Bayesian, geometric subspace tracking," *Journal for Advances in Applied Probability*, vol. 36, no. 1, pp. to appear, March 2004.

[4] X. Liu, A. Srivastava, and K. Gallivan, "Optimal linear representations of images for object recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. In review, 2002.

[5] C. Davis and W. M. Kahan, "The rotation of eigenvectors by a perturbation, III," *SIAM J. Numer. Anal.*, vol. 7, no. 1, pp. 1–46, 1970.

[6] Gene H. Golub and C. F. Vanloan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, 1989.

[7] Ake Björck and Gene H. Golub, "Numerical methods for computing angles between linear subspaces," *Mathematics of Computation*, vol. 27, pp. 579–594, 1973.

[8] G. W. Stewart, "Computing the CS decomposition of a partitioned orthonormal matrix," *Numerische Mathematik*, pp. 297–306, 1982.

[9] C. C. Paige and M. Wei, "History and generality of the CS decomposition," *Linear Algebra and Its Applications*, vol. 208/209, pp. 303–326, 1994.

[10] A. Srivastava and E. Klassen, "Monte carlo extrinsic estimators for manifold-valued parameters," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 299–308, February 2001.

[11] R. Bhattacharya and V. Patrangenaru, "Nonparametric estimation of location and dispersion on riemannian manifolds," *Journal for Statistical Planning and Inference*, vol. 108, pp. 23–36, 2002.