SIAM J. IMAGING SCIENCES
Vol. 13, No. 1, pp. 141–175

© 2020 Society for Industrial and Applied Mathematics

Dictionary Learning for Two-Dimensional Kendall Shapes*

Anna Song†, Virginie Uhlmann‡, Julien Fageot§, and Michael Unser¶

Abstract. We propose a novel sparse dictionary learning method for planar shapes in the sense of Kendall, namely configurations of landmarks in the plane considered up to similitudes. Our shape dictionary method provides a good trade-off between algorithmic simplicity and faithfulness with respect to the nonlinear geometric structure of Kendall's shape space. Remarkably, it boils down to a classical dictionary learning formulation modified using complex weights. Existing dictionary learning methods extended to nonlinear spaces map the manifold either to a reproducing kernel Hilbert space or to a tangent space. The first approach is unnecessarily heavy in the case of Kendall's shape space and causes the geometrical understanding of shapes to be lost, while the second one induces distortions and theoretical complexity. Our approach does not suffer from these drawbacks. Instead of embedding the shape space into a linear space, we rely on the hyperplane of centered configurations, including preshapes from which shapes are defined as rotation orbits. In this linear space, the dictionary atoms are scaled and rotated using complex weights before summation. Furthermore, our formulation is more general than Kendall's original one: it applies to discretely defined configurations of landmarks as well as continuously defined interpolating curves. We implemented our algorithm by adapting the method of optimal directions combined to a Cholesky-optimized order recursive matching pursuit. An interesting feature of our shape dictionary is that it produces visually realistic atoms, while guaranteeing reconstruction accuracy. Its efficiency can mostly be attributed to a clear formulation of the framework with complex numbers. We illustrate the strong potential of our approach for the characterization of datasets of shapes up to similitudes and the analysis of patterns in deforming two-dimensional shapes.

Key words. Kendall's shape space, sparse dictionary learning, 2D shape analysis, interpolating curves, splines

AMS subject classifications. 94A12, 65J22, 51M99, 65D18, 65D05

DOI. 10.1137/19M126044X

1. Introduction.

Shape analysis is highly relevant to biomedical imaging and computer vision. Among many other applications, it may be deployed to retrieve the main features from a collection of shapes, compare shapes as in morphometrics [3, 12], classify or recognize objects

*Received by the editors May 7, 2019; accepted for publication (in revised form) November 22, 2019; published electronically February 4, 2020.

https://doi.org/10.1137/19M126044X

Funding: The work of the first author was supported by École Normale Supérieure and École Polytechnique Fédérale de Lausanne. The second, third, and fourth authors were supported by Swiss National Science Foundation grant 200020_162343/1. The work of the second author was also partially supported by EMBL core fundings. The work of the third author was also partially supported by Swiss National Science Foundation grant P2ELP2_181759.

†Department of Mathematics and Applications, École Normale Supérieure (ENS), 75005, Paris, France (anna.song.maths@gmail.com).

‡European Bioinformatics Institute (EMBL-EBI), CB10 1SD, Cambridge, UK (uhlmann@ebi.ac.uk).

§Signals, Information, and Networks Group (SING), Harvard University, Cambridge, MA 02138 (julienfageot@fas.harvard.edu).

¶Biomedical Imaging Group (BIG), École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, CH-1015, VD, Switzerland (michael.unser@epfl.ch).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

[10, 41], describe the dynamics of a moving object or organism [50, 6], or study the distribution of data in a shape space [13, 46] relative to a mean shape [49]. Depending on the application, the concept of shape has different meanings [10, 60, 35]. Traditionally, shapes are handled as vectors $\mathbf{x}_1, \dots, \mathbf{x}_K \in \mathbb{R}^d$ and, for instance, correspond to silhouettes of objects outlined with landmarks. Standard signal analysis tools such as principal component analysis (PCA) can then be used to find the main modes of variation in the dataset of shapes [4, 13, 49, 50].

Alternatively, one can improve results obtained with PCA by using more refined tools, such as sparse dictionary learning [14, 31, 38, 37]. Given a *dictionary* of representative elements referred to as *atoms*, a signal is reconstructed from a *sparse* linear combination of them that minimizes the approximation error. This *sparse coding* is motivated by the assumption that natural signals are sparse [36, 40, 11, 54, 18]. The notion of sparsity has already proven its importance in an extensive range of problems, from image denoising and signal recovery to recognition and classification [36, 14, 7, 56]. Moreover, when the dictionary is learned from the data, significant improvements can be made on the signal reconstruction [40, 15, 38], leading to the so-called sparse dictionary learning approach, classically formulated as

$$\inf_{\mathbf{D},\mathbf{A}} \sum_{k=1}^{K} \|\mathbf{x}_k - \mathbf{D}\boldsymbol{\alpha}_k\|^2 + \lambda \mathrm{Sp}(\boldsymbol{\alpha}_k), \tag{1.1}$$

where $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_J) \in \mathbb{R}^{d\times J}$ is the dictionary and $\mathbf{A} = (\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_K) \in \mathbb{R}^{J\times K}$ contains the *weights* used to reconstruct $\mathbf{x}_k$. A bounding constraint on the atoms, $\|\mathbf{d}_j\| \le 1$, is added to ensure the well-posedness of the minimization problem. Sparsity is promoted by the term $\lambda \mathrm{Sp}(\boldsymbol{\alpha}_k)$, which penalizes nonzero coefficients in the weights, with $\lambda > 0$ a parameter. Often, one relies on $\mathrm{Sp}(\boldsymbol{\alpha}_k) = |\boldsymbol{\alpha}_k|_0$ or $\mathrm{Sp}(\boldsymbol{\alpha}_k) = |\boldsymbol{\alpha}_k|_1$, which denote the ℓ_0 constraint and the ℓ_1 norm, respectively. When $\hat{\mathbf{D}}$ and $\hat{\mathbf{A}}$ approximately optimize (1.1), $\hat{\mathbf{D}}\hat{\boldsymbol{\alpha}}_k$ is the *reconstruction* of the original data $\mathbf{x}_k$.

In some applications, the data are preprocessed in order to discard the influence of uninformative features such as the specific position, size, and orientation of the silhouettes. As in Procrustes analysis [13, 49], original silhouettes are scaled, translated, and rotated so as to optimally match a reference silhouette, typically the mean of the dataset. However, for data with high variability, this approach sometimes fails to produce visually *interpretable* representative elements (modes or atoms) because, after alignment, silhouettes are still not position-, scale-, and orientation-independent. To address this, we should handle data as *shapes* in the sense of Kendall [29, 30]. By definition, two geometric objects have the same shape if they are equivalent up to (direct) *similitudes* (i.e., up to translation, scaling, and rotation). Shapes are hence considered as nonlinear objects. Our work is motivated by the lack of a simple and efficient sparse dictionary learning method dedicated to Kendall's shape space. Such an analysis would be truly *invariant to similitudes* and benefit from both the efficiency of sparse dictionary learning and a valuable shape analysis framework [29, 30, 13, 47, 49, 28, 2].

An additional interesting feature of our approach is that it is suitable for the analysis of shapes defined from discretely defined configurations of landmarks as well as continuously defined curves. In Kendall's original formulation, the shape space is built upon *configurations of landmarks* in $(\mathbb{R}^d)^N$ considered up to similitudes [30, 13, 47]. Here, we extend Kendall's shape space to *interpolating curves* linearly generated by finitely many basis functions. This

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

extended framework applies in particular to *spline curves* generated by piecewise polynomials. They constitute a common representation for parametric curves [53, 5, 21, 52, 51, 44]. Our work is hence related to [44], which relies on the isometry between spline curves and configurations of landmarks to apply dictionary learning after prealignment.

Related works. Standard dictionary learning methods cannot be straightforwardly extended to Kendall's shape space because the latter is not a vector space but a Riemannian manifold. The difficulty resides in the *nonlinear* geometric structure of this space, in which a meaning must be given to linear combinations of atoms. Until recently, most dictionary learning approaches were devoted to data lying in linear spaces. In recent years, however, a few works have focused on the extension of sparse dictionary learning to nonlinear spaces such as the Grassmann manifold, the manifold of symmetric positive-definite matrices, Kendall's shape space (as in the two works [2, 28] that are closely related to ours), and more general Riemannian manifolds. This can be achieved in two ways: either by mapping the manifold to a Hilbert space, typically a reproducing kernel Hilbert space (RKHS) [25, 8, 34, 28, 24, 23], or by projecting the data onto a tangent space, once at a reference point, or iteratively at multiple points [17, 59, 22, 57, 55, 1, 27, 2]. Both approaches flatten the nonlinear space by mapping it to a linear one, so as to make weighted sums of atoms possible.

On the one hand, the mapping of Kendall's space to an RKHS through the kernel trick, as done in [28], follows a classical procedure and enables, after mapping, the use of several established techniques. However, it loses the simple structure of the underlying shape space because the mapping is not explicit. On the other hand, projecting the dataset on one tangent space unfaithfully represents the original distances outside of a neighborhood around the pole [8, 25]. In [2], an analysis with multiple projections instead is proposed by the authors. They adapt to Kendall's three-dimensional (3D) shape space a sparse dictionary learning method first proposed in [57] for general Riemannian manifolds, with the analysis of trajectories of shapes for 3D action recognition as targeted application.

Although the approach of [57, 2] has proven its success for action recognition tasks, some of its characteristics are less relevant here. It requires one to fold and unfold the manifold several times on tangent spaces using $\log_{\mathbf{x}_k}$ and $\exp_{\mathbf{x}_k}$ mappings. Most importantly, an affine constraint is added in order to ensure a nontrivial solution, but this requirement modifies the original problem, as pointed out by [26]. In addition, the original shapes $\mathbf{x}_k$ themselves are still needed in their reconstruction, which is undesirable here since the goal is to reconstruct a dataset from the dictionary and weights only. As we do not intend to tackle the same applications, our method should be considered as an alternative for learning shapes.

Overview of our method. In contrast to all previous approaches, the sparse dictionary learning method that we propose is mathematically simple while remaining faithful to the nonlinear structure of the shape space. We avoid kernel methods because they provide no additional geometrical understanding to the problem and are too sophisticated in regard of the simple structure of the shape space. We do not rely on tangent projections either, thereby avoiding both distortions and theoretical complexity.

Instead, the key idea of our approach can be summarized as follows. Kendall's shape space is the quotient of a *preshape sphere* $\mathcal{S}$ by the group of rotations. The manifold $\mathcal{S}$ is the sphere inside the linear space composed of all centered configurations. We compute all linear combinations in this vector space. Our atoms are *preshapes* in $\mathcal{S}$ (i.e., centered and normalized

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

configurations). We use complex numbers in the linear combination so as to scale and rotate them. We then add them together to obtain a configuration whose shape is close to that of the original one. This proximity is measured through a particular metric in Kendall's shape space. For the classical full Procrustes distance, our method simply leads to the optimization problem

$$\inf_{\mathbf{D},\mathbf{A}:\substack{\mathbf{d}_j\in\mathcal{S}\\ |\boldsymbol{\alpha}_k|_0\le N_0}} \sum_{k=1}^{K} |\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}^2 , \tag{1.2}$$

where $|\cdot|_{\boldsymbol{\Phi}}$ is an ℓ^2 norm specific to the representation (landmarks or interpolating curves), $\boldsymbol{\alpha}_k$ are complex weights, and the data $\mathbf{z}_k$ and atoms $\mathbf{d}_j$ are (complex) preshapes. Our work therefore proposes a natural extension of the standard dictionary learning method with a strong theoretical justification.

Contributions.

1. We extend Kendall's original framework to continuously defined interpolating curves. For this purpose, we put forward the concept of *configuration*, which indifferently represents a discrete object (configuration of landmarks) or a continuous one (interpolating curve). We thus extend and embed the work from [44] inside the more general (and appropriate) framework of Kendall's shape analysis.
2. Our entire approach of the 2D shape space is formulated in terms of complex numbers and Hermitian inner products. The geometrical interpretation of complex numbers provides a clear understanding of the framework.
3. Our main contribution consists in a simple and efficient sparse dictionary learning method, which we call the *2D Kendall shape dictionary* (2DKSD), dedicated to the analysis of 2D shapes in the sense of Kendall. Our approach provides faithful reconstructions with respect to the nonlinear geometric structure of Kendall's shape space, while remaining mathematically light. Using the full Procrustes distance [13] to compare the original and reconstructed shapes, 2DKSD boils down to *a simple and nearly classical dictionary learning formulation* (1.2) that relies on complex weights instead of real ones. It allows one to scale and rotate the atoms inside the weighted sum individually for each data point, instead of aligning the dataset to a reference shape as a preprocessing step.
4. Our implementation of this simple formulation is an adaptation of the algorithm used in the `SPAMS` software [36] to the Hermitian framework and is freely available online.[1] More precisely, we combine the method of optimal directions (MOD) [16] to an order recursive matching pursuit (ORMP) [9] with a Cholesky-based optimization.
5. Thanks to the complex setting, our method provides better reconstruction accuracy than approaches relying on the real setting. The atoms of 2DKSD are also visually more realistic and similar to shapes present in the original dataset.

Outline of the article. In section 2, we introduce the notion of planar configurations $\mathbf{z}$ and describe the action of similitudes over them. Section 3 is devoted to Kendall's space of 2D shapes, reformulated for general configurations. We briefly recall the structure of the shape

[1] https://github.com/ansonang3/2DKSD.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

space and its three classical metrics. Then, in section 4, we expose our main contribution: a sparse dictionary learning method dedicated to the analysis of 2D Kendall shapes, which for a well-chosen shape metric leads to a nearly standard formulation with complex weights (1.2). We expose the implementation of 2DKSD in section 5. Finally, we validate our approach in section 6 by experimenting on shapes extracted from real image datasets, before concluding in section 7.

2. Configurations in the plane and similitudes.

2.1. A fundamental example: Landmarks (discrete). Traditionally, the silhouette of a 2D object is sampled by an *ordered* set of points, called *landmarks*, that often mark salient features of the boundary. The object itself is then represented by this collection of N points, which we call a *configuration of N landmarks*, and is an element of $\mathbb{C}^N$ denoted by the bold letter $\mathbf{z}$. We are now interested in how (direct) similitudes transform configurations. In what follows, the word *similitudes* refers to direct similitudes.

It is elementary but important to observe that the action of a similitude in the plane, denoted by $(a, b) \in \mathbb{C}^* \times \mathbb{C}$ (where $\mathbb{C}^* := \mathbb{C} \setminus \{0\}$), has a nice expression using complex numbers,

$$\forall z \in \mathbb{C}, \qquad (a, b) \odot z = az + b, \tag{2.1}$$

where the multiplication by a applies to z a scaling of modulus $|a|$ and a rotation of angle $\arg a$, while the addition with b encodes the effect of a translation. We recall that arg is the complex argument: for z in $\mathbb{C}$, $z = |z|e^{\mathrm{i}\arg(z)}$, with $\arg(z) \in [0, 2\pi)$. Similitudes form a group $(\mathcal{G}, \circ)$ whose composition law is given by $(a', b') \circ (a, b) = (a'a, a'b + b')$.

Similarly, the action of similitudes on configurations of landmarks also has a natural expression,

$$\forall \mathbf{z} \in \mathbb{C}^N, \qquad (a, b) \odot \mathbf{z} = a\mathbf{z} + b\mathbf{u},$$

where $\mathbf{u} = (1, \ldots, 1)$ is the configuration collapsing to 1. In Figure 1 (left), the two configurations $\mathbf{z}_1$ and $\mathbf{z}_2$ are obtained from each other by a similitude. We say that they are *equivalent up to similitudes*.

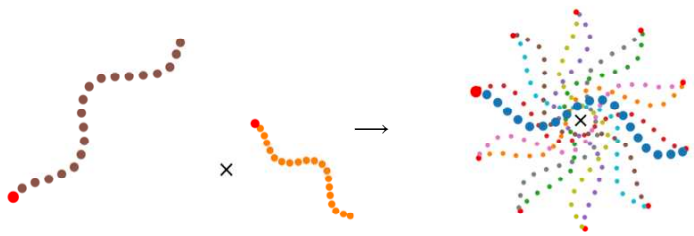


Figure 1. *The notion of shape in the sense of Kendall: Nondegenerate configurations up to similitudes. $\mathbf{z}_1 \sim \mathbf{z}_2$ (left) are nondegenerate configurations, equivalent up to direct similitudes. The cross indicates the origin. Hence, they share the same shape $[\mathbf{z}]$ (right), which can also be identified to a corresponding preshape (blue) up to rotations.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

2.2. Interpolating curves (continuous). We now move to a more involved concept of configuration: interpolating curves, which are used as an alternative to landmarks for representing silhouettes. In the plane, *interpolating curves* with N degrees of freedom are built by interpolating N complex coefficients $\mathbf{z}[0], \dots, \mathbf{z}[N-1]$ using linearly independent *basis functions* $\phi_n \in \mathbb{L}^2([0,1], \mathbb{R})$, $n = 0, \dots, (N-1)$. They have the general expression

$$\forall t \in [0,1], \quad r(t) = \sum_{n=0}^{N-1} \mathbf{z}[n]\phi_n(t). \tag{2.2}$$

If $r(0) = r(1)$, then r is a *closed interpolating curve*; otherwise it is an *open interpolating curve*. The vector $\mathbf{z} \in \mathbb{C}^N$ is called "the" *control vector* of r (as we show below, $\mathbf{z}$ is uniquely defined).

The ϕ_n are often taken to be continuous, as in the case of *interpolating spline curves*, or simply *spline curves*, for which the ϕ_n are piecewise polynomial. Interpolating curves are considered as continuously defined objects, in opposition to discretely defined landmarks. Yet, they are in fact intermediate objects between the continuous and discrete settings. As illustration, we describe the construction of interpolating curves relying on cubic B-spline interpolation in Appendix A.4.

Isometry. Interpolating curves are elements of the space of square-integrable curves $\mathcal{H} := \mathbb{L}^2([0,1], \mathbb{C})$. We endow $\mathcal{H}$ with the standard Hermitian inner product

$$(r \mid s)_{\mathcal{H}} := \int_0^1 \bar{r}\, s = \int_0^1 (r_1 s_1 + r_2 s_2) \; + \mathrm{i} \int_0^1 (r_1 s_2 - r_2 s_1), \qquad |r|_H = \sqrt{(r \mid r)_{\mathcal{H}}}, \tag{2.3}$$

where $r = r_1 + \mathrm{i} r_2 \in \mathcal{H}$, and similarly for s. Let Γ denote the linear map

$$\Gamma : \begin{array}{ccc} \mathbb{C}^N & \to & \mathcal{H} \\ \mathbf{z} & \mapsto & \sum_{n=0}^{N-1} \mathbf{z}[n]\phi_n \end{array} \tag{2.4}$$

and $\mathbf{\Phi} \in \mathbb{R}^{N \times N}$ the Gram matrix of the basis functions ϕ_n, defined by

$$\mathbf{\Phi}[n,m] := \langle \phi_n, \phi_m \rangle_{\mathbb{L}^2([0,1],\mathbb{R})} = \int_0^1 \phi_n(t)\phi_m(t)\, \mathrm{d}t. \tag{2.5}$$

The matrix $\mathbf{\Phi}$ is real-valued, symmetric, positive-definite. Let us endow $(\mathbb{C}^N, \mathbf{\Phi})$ with the Hermitian inner product (or briefly, *Hermitian product*) associated to $\mathbf{\Phi}$: the product of $\mathbf{z}, \mathbf{w} \in \mathbb{C}^N$ is $(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} = \mathbf{z}^* \mathbf{\Phi} \mathbf{w}$, where $\mathbf{z}^* := \bar{\mathbf{z}}^T$ refers to the conjugate transpose of $\mathbf{z}$. The corresponding norm is then $|\mathbf{z}|_{\mathbf{\Phi}} := \sqrt{\mathbf{z}^* \mathbf{\Phi} \mathbf{z}}$. In Appendix A.1, we detail how the Hermitian product is related to the real scalar product. As a recall of the conjugate symmetry, please note that $\mathbf{z}^* \mathbf{\Phi} \mathbf{w} = \overline{\mathbf{w}^* \mathbf{\Phi} \mathbf{z}}$.

Proposition 2.1 (isometry). *The linear map Γ is an isometry from $(\mathbb{C}^N, \mathbf{\Phi})$ to $(\mathcal{H}, (\ \mid\ )_{\mathcal{H}})$, in the sense that*

$$\forall\ \mathbf{z}, \mathbf{w} \in \mathbb{C}^N, \quad (\Gamma(\mathbf{z}) \mid \Gamma(\mathbf{w}))_{\mathcal{H}} = (\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}. \tag{2.6}$$

Therefore, interpolating curves form an N-dimensional subspace $\Gamma(\mathbb{C}^N) \subset \mathcal{H}$ isometric to $(\mathbb{C}^N, \mathbf{\Phi})$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Proof. We develop the product $(\sum_{n=0}^{N-1} \mathbf{z}[n]\phi_n \mid \sum_{m=0}^{N-1} \mathbf{w}[m]\phi_m)_{\mathcal{H}}$ to obtain

$$\sum_{n,m} \overline{\mathbf{z}[n]}\mathbf{w}[m]\langle \phi_n, \phi_m \rangle_{\mathbb{L}^2([0,1],\mathbb{R})},$$

which is exactly $\mathbf{z}^*\mathbf{\Phi}\mathbf{w}$. ∎

As a consequence, an interpolating curve $r \in \Gamma(\mathbb{C}^N)$ is completely identified to its control vector $\mathbf{z} = \Gamma^{-1}(r) \in \mathbb{C}^N$. This result is important for the extension of Kendall's theory to interpolating curves[2] and ensures that they can be identified to configurations.

Action of similitudes over interpolating curves. The action of any similitude $(a, b) \in \mathbb{C}^* \times \mathbb{C}$ on curves in $\mathcal{H}$ has once again a nice formulation,

$$\forall r \in \mathcal{H}, \qquad (a, b) \odot r = ar + b \ , \tag{2.7}$$

where $\mathbb{1}$ refers to the constant function $\forall t \in [0, 1],\ t \mapsto 1$ in $\mathcal{H}$. Suppose that $\mathbb{1} \in \Gamma(\mathbb{C}^N)$, namely, that constant curves are generated by the same basis functions ϕ_n. Under this assumption, it is legitimate to define $\mathbf{u} := \Gamma^{-1}(\mathbb{1})$. The action of similitudes is induced on $\mathbb{C}^N$ according to

$$\forall \mathbf{z} \in \mathbb{C}^N, \qquad (a, b) \odot \mathbf{z} = \Gamma^{-1}((a, b) \odot \Gamma(\mathbf{z})) = \Gamma^{-1}(ar + b\mathbb{1}) = a\mathbf{z} + b\mathbf{u}. \tag{2.8}$$

This is well defined, because $ar + b\mathbb{1}$ remains in $\Gamma(\mathbb{C}^N)$ for any interpolating curve r.

2.3. A more general concept: Configurations. The two previous frameworks—landmarks configurations and interpolating curves—can be encompassed inside one common and more general framework.

Definition 2.2. *A* configuration *of dimension $N \in \mathbb{N}^*$ in the plane is a complex vector $\mathbf{z} \in \mathbb{C}^N$.*

The notion of configuration plays a major role throughout this article and allows us to handle planar objects, such as curves or landmarks configurations, parameterized by $\{1, \dots, N\}$ through the configurations $\mathbf{z} \in \mathbb{C}^N$.

Set an integer $N \geq 1$. Let us consider a Hermitian matrix $\mathbf{\Phi} = \mathbf{\Phi}^* \in \mathbb{C}^{N\times N}$ that is positive-definite: $\forall \mathbf{z} \in \mathbb{C}^N \setminus \{0\}$, $\mathbf{z}^*\mathbf{\Phi}\mathbf{z} > 0$. Note that $\mathbf{\Phi}$ is not necessarily real-valued here. We endow $\mathbb{C}^N$ with the Hermitian inner product $(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} := \mathbf{z}^*\mathbf{\Phi}\mathbf{w}$. Working with landmarks in the usual way, $\mathbf{\Phi}$ refers to the identity matrix $\mathrm{Id} \in \mathbb{C}^{N\times N}$. In the case of interpolating curves, $\mathbf{\Phi}$ is the Gram matrix of the basis functions ϕ_n, assumed to be linearly independent, and to linearly generate $\mathbb{1}$.

Now let $\mathbf{u}$ denote a nonzero configuration, which we call the *shift configuration*, and consider the group action $\odot : (\mathcal{G}, \circ) \times \mathbb{C}^N \to \mathbb{C}^N$ expressed as $(a, b) \odot \mathbf{z} \mapsto a\mathbf{z} + b\mathbf{u}$, where $(\mathcal{G}, \circ)$ is the group of similitudes. The shift configuration conveys an important geometrical interpretation and is the vector determining the action of translations on $\mathbb{C}^N$, hence the

[2]The question of defining a good notion of curve shape and, in particular, a suitable representation for the curve (such as the square-root velocity representation [48]), is outside the scope of this article. We refer interested readers to [47].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

term "shift." Together with $\mathbf{\Phi}$, they will define the shape space built in the next section. In our fundamental examples, we take $\mathbf{u} = (1, \dots, 1)$ for landmarks and $\mathbf{u} = \Gamma^{-1}(\mathbb{1})$ for interpolating curves. In both cases, $\mathbf{u}$ specifies the coefficients of the unit vector or function in the corresponding basis of the representation.

In what follows, $\mathbb{C}\mathbf{u}$ denotes the complex vector line generated by $\mathbf{u}$, and $(\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}$ denotes the complex hyperplane of configurations orthogonal to $\mathbf{u}$.

Definition 2.3. *If* $\mathbf{z} \in \mathbb{C}\mathbf{u}$*, then* $\mathbf{z}$ *is said to be degenerate, and it is nondegenerate otherwise.*

In our examples of landmarks or interpolating curves, degenerate configurations represent objects that collapse to a single point in the plane and are not interesting in practice.

Definition 2.4. *A configuration* $\mathbf{z}$ *is said to be* centered *if* $\mathbf{u}^*\mathbf{\Phi}\mathbf{z} = 0$ *or, equivalently,* $\mathbf{z} \in (\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}$.

Centering $\mathbf{z}$ *consists in orthogonally projecting* $\mathbf{z}$ *onto* $(\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}$.

For any configuration $\mathbf{z}$, there exists a unique complex number $b \in \mathbb{C}$, called the *center* of $\mathbf{z}$, so that $\mathbf{z} - b\mathbf{u}$ is orthogonal to $\mathbf{u}$. It is given by $b = \frac{\mathbf{u}^*\mathbf{\Phi}\mathbf{z}}{|\mathbf{u}|^2_{\mathbf{\Phi}}}$, and $\mathbf{z} - b\mathbf{u}$ is then the centered version of $\mathbf{z}$. In the case of landmarks configurations, the center is the usual arithmetic mean. The operation $\mathbf{z} \mapsto (\mathbf{z} - \frac{\sum \mathbf{z}[i]}{n}\mathbf{u})$ centers $\mathbf{z}$. Note that $|\mathbf{u}|_{\mathbf{\Phi}} = \sqrt{n}$. Regarding interpolating curves, the center of $\mathbf{z}$ coincides with the temporal mean of r, given by $\bar{r} := \int_0^1 r(t)\,\mathrm{d}t \in \mathbb{C} = (\mathbb{1} \mid r)_{\mathcal{H}} = \mathbf{u}^*\mathbf{\Phi}\mathbf{z} = \frac{\mathbf{u}^*\mathbf{\Phi}\mathbf{z}}{|\mathbf{u}|^2_{\mathbf{\Phi}}}$. Note that, contrarily to landmarks, $|\mathbf{u}|_{\mathbf{\Phi}} = |\mathbb{1}|_{\mathbb{L}^2([0,1],\mathbb{C})} = 1$.

Definition 2.5. *A configuration* $\mathbf{z}$ *is called a* preshape *if it is* centered and normalized*,*

$$\mathbf{u}^*\mathbf{\Phi}\mathbf{z} = 0 \quad \textit{and} \quad |\mathbf{z}|_{\mathbf{\Phi}} = 1.$$

To visualize the various operations, we encourage the reader to refer to Figure 2. We also summarize some notation in Table 1.

3. Kendall's space of planar shapes.

We now describe Kendall's shape space in the planar case [13, 47]. When developing this framework, we make a systematic use of the shift configuration $\mathbf{u}$ and Hermitian inner product $\mathbf{\Phi}$ introduced in the previous section. In particular, interpolating curves can be referred to as configurations. We recall that we consider the action of similitudes written as $(a, b) \odot \mathbf{z} = a\mathbf{z} + b\mathbf{u}$.

3.1. Preshape sphere and shape space.

A *shape*, in the sense of Kendall, is an equivalence class of nondegenerate configurations considered up to similitudes (see Figure 1). It is what remains after discarding redundant geometric information given by the position (center), scaling, and orientation. The shape space is then the quotient of the nondegenerate subspace by the action of similitudes. We first get rid of translation and scaling, thus obtaining centered and normalized configurations which form the *preshape sphere* (see Figure 2). Then, we quotient by the group of rotations $U(1) := \{z \in \mathbb{C} \mid |z| = 1\}$ to obtain Kendall's shape space.

Definition 3.1. *The* preshape sphere $\mathcal{S} = \{\mathbf{z} \in \mathbb{C}^N \mid \mathbf{u}^*\mathbf{\Phi}\mathbf{z} = 0, \mathbf{z}^*\mathbf{\Phi}\mathbf{z} = 1\}$ *is the set of centered and normalized configurations* $\mathbf{z} \in \mathbb{C}^N$*. It is the quotient of* $\mathbb{C}^N$ *by the action of translations* $\mathbf{z} \mapsto \mathbf{z} + b\mathbf{u}$*,* $b \in \mathbb{C}$*, and of scalings* $\mathbf{z} \mapsto \lambda\,\mathbf{z}$*,* $\lambda \in \mathbb{R}^*$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

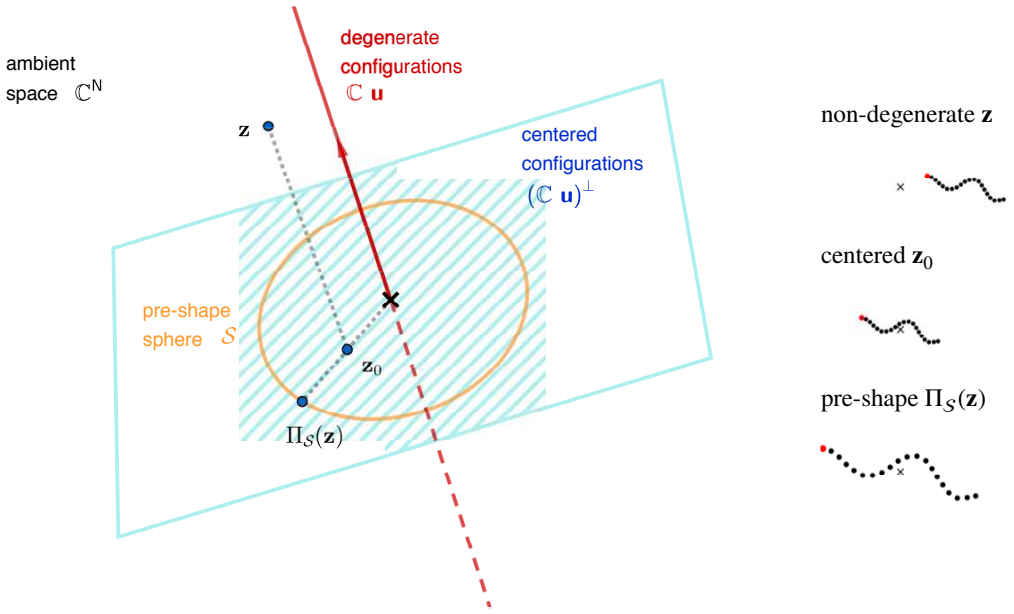


Figure 2. *Illustration of the spaces and of the "preshaping" operations. Atoms are preshapes that are linearly combined inside the (complex) hyperplane of centered configurations. For the purpose of illustration, elements are drawn in analogy to a real setting.*

Table 1
Spaces and elements.

Element	Notation	Space	Real dim
general configuration	$\mathbf{z}$	$\mathbb{C}^N$	$2N$
shift configuration	$\mathbf{u}$	$\mathbb{C}^N$	$2N$
centered configuration	$\mathbf{z}_0 = \mathbf{z} - \frac{\mathbf{u}^*\mathbf{\Phi}\mathbf{z}}{\lvert\mathbf{u}\rvert_{\mathbf{\Phi}}^2}\mathbf{u}$	$(\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}} \simeq \mathbb{C}^{N-1}$	$2N-2$
preshape (centered, normalized)	$\Pi_{\mathcal{S}}(\mathbf{z}) = \frac{\mathbf{z}_0}{\lvert\mathbf{z}_0\rvert_{\mathbf{\Phi}}}$	$\mathcal{S} = \{\lvert\mathbf{z}\rvert_{\mathbf{\Phi}} = 1\} \cap (\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}$	$2N-3$
shape (preshape up to rotations)	$[\mathbf{z}]$	$\Sigma = \mathcal{S}/U(1)$	$2N-4$

Since it consists of the unit-norm elements of the complex hyperplane $(\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}$ orthogonal to $\mathbf{u}$, $\mathcal{S}$ is a smooth compact real $(2N-3)$-manifold, as can be seen in

$$\mathcal{S} \;=\; \{|\mathbf{z}|_{\mathbf{\Phi}} = 1\} \;\cap\; (\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}}. \tag{3.1}$$

Any nondegenerate configuration is uniquely associated to a *preshape* by centering and then normalizing it (but the converse is not true).

Definition 3.2. *The preshape uniquely associated to a nondegenerate configuration $\mathbf{z} \in \mathbb{C}^N$ is the projection of $\mathbf{z}$ onto $\mathcal{S}$, denoted by*

$$\Pi_{\mathcal{S}}(\mathbf{z}) = \frac{\mathbf{z}_0}{|\mathbf{z}_0|_{\mathbf{\Phi}}}, \qquad \mathbf{z}_0 = \mathbf{z} - \frac{\mathbf{u}^*\mathbf{\Phi}\mathbf{z}}{|\mathbf{u}|_{\mathbf{\Phi}}^2}\mathbf{u}, \tag{3.2}$$

where $\mathbf{z}_0$ is the centered version of $\mathbf{z}$ (see Figure 2 and Table 1).

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Definition 3.3. *The* shape space

$$\Sigma := \mathcal{S}/U(1) \tag{3.3}$$

is the preshape sphere quotiented by the action of the group of rotations $\mathbf{z} \mapsto \mathrm{e}^{\mathrm{i}\theta}\ \mathbf{z}$, $\theta \in [0, 2\pi)$. *The orbit of a preshape* $\mathbf{z} \in \mathcal{S}$ *under the action of* $U(1)$ *is denoted by* $[\mathbf{z}] \in \Sigma$ *and is called the* shape *of* $\mathbf{z}$. *More generally, given a nondegenerate configuration* $\mathbf{z}$, *we define its shape* $[\Pi_{\mathcal{S}}(\mathbf{z})]$ *to be that of the unique preshape associated to* $\mathbf{z}$.

Kendall's shape space Σ is a compact smooth real $(2N-4)$-manifold, identified to the complex projective space [29, 20, 13, 47]

$$\mathbb{CP}^{N-2} = (\mathbb{C}^{N-1} \setminus \{0\})/\mathbb{C}^* \simeq \mathbb{S}^{2N-3}/U(1). \tag{3.4}$$

In this expression, $\mathbb{C}^{N-1} \setminus \{0\}$ refers to the set of centered nondegenerate configurations of landmarks. In the literature, a typical approach for quotienting out translations is to discard the last coordinate in $\mathbf{z} \in \mathbb{C}^N$. For centered configurations of landmarks, one indeed has that $\mathbf{z}[N-1] = -\sum_{n=0}^{N-2} \mathbf{z}[n]$. In the general case, $\mathbb{C}^{N-1} \setminus \{0\}$ can be replaced by $(\mathbb{C}\mathbf{u})^{\perp_{\mathbf{\Phi}}} \setminus \{0\}$ and $\mathbb{S}^{2N-3}$ by $\mathcal{S}$, conserving the identification of Σ to $\mathbb{CP}^{N-2}$.

3.2. Distances in the shape space. Three classical distances are usually defined on the shape space: the full (Procrustes), partial (Procrustes), and geodesic distances, denoted as d_F, d_P, and ρ, respectively [13, 47]. Their evaluations $\mathrm{dist}([\mathbf{z}], [\mathbf{w}])$ enjoy simple and closed-form expressions that involve only the preshapes $\mathbf{z}$ and $\mathbf{w}$. Readers interested in the practical use of these metrics and not in their geometrical definition may directly skip to (3.10), (3.11), and (3.12). For a quick geometrical intuition without mathematical details, we refer readers to Figures 3 and 4.

Distances and optimal transformations. In this paragraph, $\mathbf{z}$ and $\mathbf{w}$ denote preshapes by default.

Definition 3.4 (partial distance and optimal rotation). *Let* $\mathbf{z}, \mathbf{w} \in \mathcal{S}$. *We define*

$$d_P([\mathbf{z}], [\mathbf{w}]) = \min_{\theta \in [0, 2\pi)} |\mathrm{e}^{\mathrm{i}\theta}\ \mathbf{z} - \mathbf{w}|_{\mathbf{\Phi}}. \tag{3.5}$$

We say that we optimally rotate $\mathbf{z}$ *along* $\mathbf{w}$ *when, if unique, the* optimal rotation angle $\theta(\mathbf{z}, \mathbf{w})$ *is applied to* $\mathbf{z}$ *(see Figure 3).*

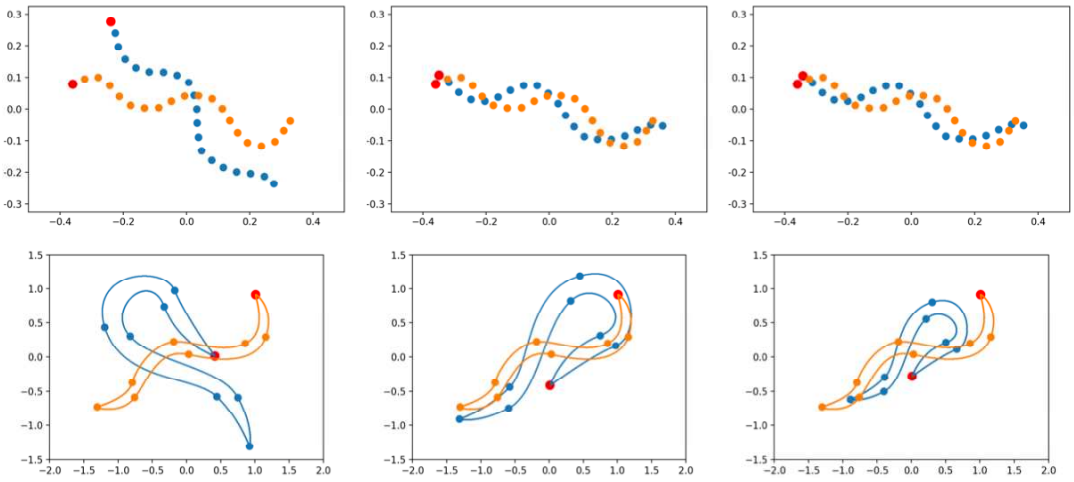
Definition 3.5 (full distance and optimal alignment). *Let* $\mathbf{z}, \mathbf{w} \in \mathcal{S}$. *We define*

$$d_F([\mathbf{z}], [\mathbf{w}]) = \min_{a \in \mathbb{C}} |a\ \mathbf{z} - \mathbf{w}|_{\mathbf{\Phi}} \tag{3.6}$$

$$= |\mathrm{P}_{\mathbb{C}\mathbf{z}}\mathbf{w} - \mathbf{w}|_{\mathbf{\Phi}}, \tag{3.7}$$

where $\mathrm{P}_{\mathbb{C}\mathbf{z}}$ *is the orthogonal projection onto the complex vector line generated by* $\mathbf{z}$ *(with respect to* $\mathbf{\Phi}$*). In* (3.6), *the* optimal alignment factor *is given by* $a(\mathbf{z}, \mathbf{w}) = \mathbf{z}^*\mathbf{\Phi}\mathbf{w}$, *and its modulus* $\lambda(\mathbf{z}, \mathbf{w})$ *is called the* optimal scaling factor *(see Figure 3). We say that we* optimally align $\mathbf{z}$ along $\mathbf{w}$ *when* $a(\mathbf{z}, \mathbf{w})$ *is applied to* $\mathbf{z}$ *to obtain* $\mathrm{P}_{\mathbb{C}\mathbf{z}}\mathbf{w} = a(\mathbf{z}, \mathbf{w})\ \mathbf{z} = \mathbf{z}^*\mathbf{\Phi}\mathbf{w}\ \mathbf{z}$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.



Initial preshapes
$\mathbf{z}$ (blue) and $\mathbf{w}$ (orange).

Optimally rotated
$\tilde{\mathbf{z}} = \mathrm{e}^{\mathrm{i}\arg(\mathbf{z}^*\Phi\mathbf{w})}\mathbf{z}$ along $\mathbf{w}$
$d_P([\mathbf{z}],[\mathbf{w}]) = |\tilde{\mathbf{z}} - \mathbf{w}|_\Phi$

Optimally aligned
$\tilde{\mathbf{z}} = (\mathbf{z}^*\Phi\mathbf{w})\,\mathbf{z}$ along $\mathbf{w}$
$d_F([\mathbf{z}],[\mathbf{w}]) = |\tilde{\mathbf{z}} - \mathbf{w}|_\Phi$

Figure 3. *Examples of optimal rotation and alignment. Upper row: landmarks with $N = 20$. Lower row: closed Hermite spline curves with $N = 16$ [52]. Left column: original preshapes (centered and normalized) $\mathbf{z}$ in blue and $\mathbf{w}$ in orange. Middle column: optimal rotation of $\mathbf{z}$ along $\mathbf{w}$. Right column: optimal alignment of $\mathbf{z}$ along $\mathbf{w}$. The angles for both optimal rotation and alignment are the same and are equal to $\theta = \arg(\mathbf{z}^*\Phi\mathbf{w})$. The optimal scaling in the alignment is the modulus $\lambda(\mathbf{z},\mathbf{w}) = |\mathbf{z}^*\Phi\mathbf{w}|$; when $\mathbf{z}$ and $\mathbf{w}$ are highly correlated, $\lambda(\mathbf{z},\mathbf{w})$ is close to 1 and the optimally aligned image of $\mathbf{z}$ is similar to the optimally rotated one. For landmarks (upper row), $|\mathbf{z}-\mathbf{w}|_\Phi = \sum_n |\mathbf{z}[n]-\mathbf{w}[n]|^2$ is the usual distance, here with $\mathbf{z},\mathbf{w}$ being highly correlated. For Hermite-spline curves (lower row), the distance $|\mathbf{z}-\mathbf{w}|_\Phi$ is equal to the usual curve distance $|r-s|_{\mathbb{L}^2([0,1],\mathbb{C})} = \sqrt{\int_0^1 |r(t)-s(t)|^2\,\mathrm{d}t}$ between their images r, s through the isometry Γ (section 2.2).*

$$\rho([\mathbf{z}],[\mathbf{w}]) = \arccos|\mathbf{z}^*\Phi\mathbf{w}|$$

Figure 4. *Geodesic path on Σ. Preshapes corresponding to shapes regularly met along the geodesic path joining $[\mathbf{z}]$ to $[\mathbf{w}]$. The two endpoints are the preshapes $\tilde{\mathbf{z}} = \mathrm{e}^{\mathrm{i}(\arg \mathbf{z}^*\Phi\mathbf{w})}\mathbf{z}$ and $\mathbf{w}$ optimally rotated along each other. The preshapes themselves describe the shortest geodesic path that joins $\tilde{\mathbf{z}}$ to $\mathbf{w}$ on the preshape sphere $\mathcal{S}$. The distance $\rho([\mathbf{z}],[\mathbf{w}])$ then corresponds to the angle measured from the center of the preshape sphere along this geodesic arc. Here, we illustrate this concept with open Hermite-spline curves [52, 45].*

Proposition 3.6. *Let $\mathbf{z},\mathbf{w} \in \mathcal{S}$ and suppose that $\mathbf{z}^*\Phi\mathbf{w} \neq 0$. Then, the optimal rotation angle is given by*

$$\theta(\mathbf{z},\mathbf{w}) = \arg(\mathbf{z}^*\Phi\mathbf{w}). \tag{3.8}$$

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

As a consequence, the angle involved in the optimal alignment factor is the same as the optimal angle itself: $\arg(a(\mathbf{z},\mathbf{w})) = \theta(\mathbf{z},\mathbf{w})$. *Otherwise, when* $\mathbf{z}^*\mathbf{\Phi}\mathbf{w} = 0$, *the distances are maximal and equal to* $d_F([\mathbf{z}],[\mathbf{w}]) = 1$ *and* $d_P([\mathbf{z}],[\mathbf{w}]) = \sqrt{2}$.

Proof. If $\mathbf{z}^*\mathbf{\Phi}\mathbf{w} = 0$, then we use the orthogonality of $\mathbf{z}$ and $\mathbf{w}$ to conclude. Otherwise, since

$$\arg\min_{\theta\in[0,2\pi)} |\mathrm{e}^{\mathrm{i}\theta}\cdot\mathbf{z}-\mathbf{w}|^2_{\mathbf{\Phi}} = \arg\max_{\theta\in[0,2\pi)} \underbrace{\Re(\mathrm{e}^{\mathrm{i}\theta}\cdot\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}}_{\leq|(\mathrm{e}^{\mathrm{i}\theta}\cdot\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}|}, \tag{3.9}$$

θ is optimal when $(\mathrm{e}^{\mathrm{i}\theta}\cdot\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} = e^{-i\theta}(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} \in \mathbb{R}^+$, in other words, when $\theta = \arg(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}$. ∎

Note that the orthogonality condition $\mathbf{z}^*\mathbf{\Phi}\mathbf{w} = 0$ means that $\mathbf{z}$ and $\mathbf{w}$ are decorrelated: whatever rotation we apply on $\mathbf{z}$, the distance $|\mathrm{e}^{\mathrm{i}\theta}\cdot\mathbf{z}-\mathbf{w}|_{\mathbf{\Phi}}$ remains maximal. In the real setting, this corresponds to the case where $\mathbf{z}$ is orthogonal to all rotations of $\mathbf{w}$ with respect to the real inner product (see Appendix A.1).

Corollary 3.7. *Let* $\mathbf{z},\mathbf{w}\in\mathcal{S}$. *The distances have closed-form expressions given by*

$$d_P([\mathbf{z}],[\mathbf{w}]) = \left|\frac{\mathbf{z}^*\mathbf{\Phi}\mathbf{w}}{|\mathbf{z}^*\mathbf{\Phi}\mathbf{w}|}\mathbf{z}-\mathbf{w}\right|_{\mathbf{\Phi}} = \sqrt{2-2|\mathbf{z}^*\mathbf{\Phi}\mathbf{w}|} \in [0,\sqrt{2}], \tag{3.10}$$

$$d_F([\mathbf{z}],[\mathbf{w}]) = |(\mathbf{z}^*\mathbf{\Phi}\mathbf{w})\ \mathbf{z}-\mathbf{w}|_{\mathbf{\Phi}} = \sqrt{1-|\mathbf{z}^*\mathbf{\Phi}\mathbf{w}|^2} \in [0,1]. \tag{3.11}$$

In Figure 3, we illustrate the fact that the distances d_P and d_F measure the shortest reachable norm $|\tilde{\mathbf{z}}-\mathbf{w}|_{\mathbf{\Phi}}$ if $\tilde{\mathbf{z}}$ is the image of $\mathbf{z}$ after any rotation, or scaling and rotation.

Proposition 3.8 (Riemannian distance). *Let* $\mathbf{z},\mathbf{w}\in\mathcal{S}$. *The Riemannian distance in the shape space (see Figure 4), also called geodesic distance and denoted by* $\rho([\mathbf{z}],[\mathbf{w}])$, *is equal to*

$$\rho([\mathbf{z}],[\mathbf{w}]) = \arccos|\mathbf{z}^*\mathbf{\Phi}\mathbf{w}| = \arccos\lambda(\mathbf{z},\mathbf{w}) \quad\in\quad [0,\pi/2]. \tag{3.12}$$

Proposition 3.9 (relationships between d_F, d_P, and ρ). *The three distances* d_F, d_P, *and* ρ *are related to each other as*

$$d_P^2 = 2-2\sqrt{1-d_F^2}, \tag{3.13}$$

$$d_F = \sin\rho, \tag{3.14}$$

$$d_P = 2\sin(\rho/2). \tag{3.15}$$

We found it worth presenting the Riemannian distance ρ alongside d_F and d_P, although our 2DKSD method does not require the use of the Riemannian structure. In fact, the geometric structure of Σ is entirely inherited from that of $\mathcal{S}$: this justifies using the preshape sphere, whose geometry is well understood, to think about the shape space itself. In particular, the Riemannian structure of Σ is a quotient structure of $\mathcal{S}$, as discussed in Appendix A.5. As a consequence, tangent spaces, geodesics, and exponentials benefit from explicit expressions on Σ. For readers interested in the Riemannian structure of the shape space (not restricted to the planar case), we refer to [13, 47], where the usual definition relying on landmarks is used. For those seeking good references about Riemannian geometry, in particular that of complex projective spaces, we recommend [20, 19].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

4. Shape dictionary. The simplest dictionary learning approach on a dataset of configurations consists in considering them as elements of a point cloud in $(\mathbb{C}^N)^K \simeq (\mathbb{R}^{2N})^K$, and then applying an existing dictionary learning method. However, as discussed in the introduction, a preprocessing of the data can be necessary in order to attenuate differences of position, scaling, or orientation across the dataset. These geometric dissimilarities indeed lead to undesirable artifacts after learning, such as smoothed and unfaithful reconstructions or distorted and redundant atoms (in the sense that they share roughly the same shape but are associated to, e.g., different orientations). An example of preprocessing is proposed in [44], where the data are first optimally aligned along a mean configuration $\mathbf{z}_{\text{mean}}$, with $[\Pi_{\mathcal{S}}(\mathbf{z}_{\text{mean}})] \in \Sigma$ being the Fréchet mean of $[\Pi_{\mathcal{S}}(\mathbf{z}_1)], \dots, [\Pi_{\mathcal{S}}(\mathbf{z}_K)] \in \Sigma^K$ with respect to the distance d_F (see Appendix A.3).

A limitation of this kind of "align-first" approach is that the alignment step depends on the whole dataset and needs to be started anew for each new data input. In section 6, we compare the results obtained by aligning first to those produced by our algorithm. Instead, we prefer to directly work in the shape space with suitable shape metrics, without having to preprocess the data.

Our 2DKSD method takes advantage of *complex weights* to rotate and scale the atoms before summing them to reconstruct each original shape. Hence, instead of prealigning the data, we rotate and scale *inside the weighted sums* $\mathbf{D}\boldsymbol{\alpha}_k$, individually for each data shape. Also, we show that, for a good choice of the error metric that serves to compare the original and reconstructed shapes, *the 2DKSD boils down to a nearly classical and very simple dictionary learning formulation.* Weights are, however, complex vectors, and the dataset and dictionary atoms must satisfy some (light) constraints imposed by Kendall's framework. This a priori unexpected result, shown in Proposition 4.2, is interesting both mathematically and numerically.

Statement of the shape dictionary. We suppose that elements of the dataset are preshapes $\mathbf{z}_1, \dots, \mathbf{z}_K \in \mathcal{S}$ (centered and normalized). Let dist denote any distance on Σ, such as one of the three distances d_F, d_P, ρ.

Definition 4.1. *The* 2DKSD general formulation *of the problem to be solved is written*

$$\inf_{\substack{\mathbf{d}_j \in \mathcal{S}, \\ \mathbf{D},\mathbf{A}:\ \mathbf{D}\boldsymbol{\alpha}_k \in \mathcal{S} \\ |\boldsymbol{\alpha}_k|_0 \le N_0}} \quad \sum_{k=1}^{K} \operatorname{dist}\left([\mathbf{z}_k], [\mathbf{D}\boldsymbol{\alpha}_k]\right)^2. \tag{4.1}$$

The dictionary corresponds to $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_J) \in \mathbb{C}^{N \times J}$, whose atoms $\mathbf{d}_j \in \mathcal{S}$ are preshapes. Importantly, the weights $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_K \in \mathbb{C}^J$ are *complex* numbers which apply scalings and rotations to the atoms $\mathbf{d}_j$ before summing them to $\mathbf{D}\boldsymbol{\alpha}_k$. This linear combination is sparse, since the (hard) sparsity constraint $|\boldsymbol{\alpha}_k|_0 \le N_0$ enforces that at most N_0 coefficients are nonzero. We also impose $\mathbf{D}\boldsymbol{\alpha}_k$ and $\mathbf{z}_k$ to be preshapes before comparing their shapes. If $\hat{\mathbf{D}}$ and $\hat{\mathbf{A}}$ are (approximately) optimizers of (4.1), the original shape $[\mathbf{z}_k]$ is reconstructed[3] as $[\hat{\mathbf{D}}\hat{\boldsymbol{\alpha}}_k]$.

[3]When discussing the sparse coding step, we shall show that, for dist $= d_F$, one can compute $\hat{\boldsymbol{\alpha}}_k$ so that the original preshape itself is reconstructed as $\hat{\mathbf{D}}\hat{\boldsymbol{\alpha}}_k$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

The unit-norm constraints imposed on $\mathbf{d}_j$ and $\mathbf{D}\boldsymbol{\alpha}_k$ are in fact nonessential. Up to a rescaling of the weights $\boldsymbol{\alpha}_k$, we can more generally consider $\mathbf{d}_j$ as a centered configuration of norm smaller than 1, thus allowing possibly collapsing atoms $\mathbf{d}_j = \mathbf{0}$. This does not change the problem since this corresponds to a case where $\mathbf{d}_j$ would be ignored in the linear combinations of the original optimization problem (4.1). Hence, we could replace $\mathbf{d}_j \in \mathcal{S}$ by $\mathbf{d}_j \perp_{\boldsymbol{\Phi}} \mathbf{u}; |\mathbf{d}_j|_{\boldsymbol{\Phi}} \leq 1$. We have decided instead to keep atoms as preshapes to have a uniform scaling across the dictionary.

Also, $\mathbf{D}\boldsymbol{\alpha}_k$ only needs to be a nondegenerate configuration in $\mathbb{C}^N \setminus \mathbb{C}\mathbf{u}$ for the shape $[\Pi_{\mathcal{S}}(\mathbf{D}\boldsymbol{\alpha}_k)]$ to be defined. Since each $\mathbf{d}_j$ is already centered, $\mathbf{D}\boldsymbol{\alpha}_k$ is necessarily centered, and so we only need to suppose that $\mathbf{D}\boldsymbol{\alpha}_k \neq \mathbf{0}$ does not collapse. We also note that $\Pi_{\mathcal{S}}(\mathbf{D}\boldsymbol{\alpha}_k) = \mathbf{D}\frac{\boldsymbol{\alpha}_k}{|\mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}}$ is then simply the normalized version of $\mathbf{D}\boldsymbol{\alpha}_k$. Hence, the infimum in (4.1) is equal to

$$\inf_{\substack{\mathbf{d}_j \in \mathcal{S} \\ \mathbf{D},\mathbf{A}:\ \mathbf{D}\boldsymbol{\alpha}_k \in \mathcal{S} \\ |\boldsymbol{\alpha}_k|_0 \leq N_0}} \sum_{k=1}^{K} \operatorname{dist}\left([\mathbf{z}_k], [\mathbf{D}\boldsymbol{\alpha}_k]\right)^2 = \inf_{\substack{\mathbf{d}_j \in \mathcal{S} \\ \mathbf{D},\mathbf{A}:\ \mathbf{D}\boldsymbol{\alpha}_k \neq \mathbf{0} \\ |\boldsymbol{\alpha}_k|_0 \leq N_0}} \sum_{k=1}^{K} \operatorname{dist}\left([\mathbf{z}_k], \left[\mathbf{D}\frac{\boldsymbol{\alpha}_k}{|\mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}}\right]\right)^2. \tag{4.2}$$

Proposition 4.2. *Using the full Procrustes distance* dist $= d_F$ *as minimization criterion,* (4.2) *leads to the* 2DKSD simple formulation

$$\inf_{\mathbf{D}:\mathbf{d}_j \in \mathcal{S}} \sum_{k=1}^{K} \min_{\boldsymbol{\alpha}_k : |\boldsymbol{\alpha}_k|_0 \leq N_0} |\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}^2, \tag{4.3}$$

where we recall that $\mathbf{z}_k$ *are preshapes. This formulation is equivalent to the one in* (1.2).

Proof. We express the minimization problem (4.2) as

$$\inf_{\mathbf{D}:\left\{\substack{\mathbf{d}_j \perp_{\boldsymbol{\Phi}} \mathbf{u} \\ |\mathbf{d}_j|_{\boldsymbol{\Phi}} \leq 1}\right.} \sum_{k=1}^{K} \inf_{\boldsymbol{\alpha}_k:\left\{\substack{\mathbf{D}\boldsymbol{\alpha}_k \neq \mathbf{0} \\ |\boldsymbol{\alpha}_k|_0 \leq N_0}\right.} d_F\left([\mathbf{z}_k], \left[\mathbf{D}\frac{\boldsymbol{\alpha}_k}{|\mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}}\right]\right)^2 \tag{4.4}$$

since, after fixing $\mathbf{D}$, the minimization breaks into K independent elementary terms that correspond to a sparse coding of the shapes $[\mathbf{z}_k]$. Each term has the form

$$\inf_{\boldsymbol{\alpha}:\left\{\substack{\mathbf{D}\boldsymbol{\alpha} \neq \mathbf{0} \\ |\boldsymbol{\alpha}|_0 \leq N_0}\right.} d_F\left([\mathbf{z}], \left[\mathbf{D}\frac{\boldsymbol{\alpha}}{|\mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}}\right]\right)^2 = \inf_{\boldsymbol{\alpha}:\left\{\substack{\mathbf{D}\boldsymbol{\alpha} \neq \mathbf{0} \\ |\boldsymbol{\alpha}|_0 \leq N_0}\right.} |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\boldsymbol{\Phi}}^2 = \inf_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\boldsymbol{\Phi}}^2, \tag{4.5}$$

where we successively applied (3.7), used that the vector lines generated by $\mathbf{D}\frac{\boldsymbol{\alpha}}{|\mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}}$ and $\mathbf{D}\boldsymbol{\alpha}$ is the same, and dropped[4] the inequality $\mathbf{D}\boldsymbol{\alpha} \neq \mathbf{0}$. Then, we apply the results of Lemma A.2 and find that

$$\min_{\boldsymbol{\alpha}:\left\{\substack{\mathbf{D}\boldsymbol{\alpha} \neq \mathbf{0} \\ |\boldsymbol{\alpha}|_0 \leq N_0}\right.} d_F\left([\mathbf{z}], \left[\mathbf{D}\frac{\boldsymbol{\alpha}}{|\mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}}\right]\right)^2 = \min_{\boldsymbol{\alpha}:\left\{\substack{\mathbf{D}\boldsymbol{\alpha} \neq \mathbf{0} \\ |\boldsymbol{\alpha}|_0 \leq N_0}\right.} |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\boldsymbol{\Phi}}^2 = \min_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\boldsymbol{\Phi}}^2 \tag{4.6}$$

$$= \min_{|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}^2, \tag{4.7}$$

[4]The case $\mathbf{D}\boldsymbol{\alpha} = \mathbf{0}$ does not induce a lower infimum value because $|\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\boldsymbol{\Phi}}^2 = |\mathbf{z}|_{\boldsymbol{\Phi}}^2$ is then maximal.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

where the three first terms share a common minimizer. Therefore, a new formulation of the original problem is given by (4.3) and, equivalently, by (1.2). ∎

5. Dictionary update and sparse coding. We numerically solved the 2DKSD problem (4.3) involving the full distance d_F. Our code is fully available online.

As summarized in Algorithms 5.1 and 5.2, the implementation is directly adapted from the algorithm used in the SPAMS software [37, 36] to a complex Hermitian framework. Recall that $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_J)$ is the dictionary and $\mathbf{A} = (\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_K)$ contains the weighting complex vectors. To minimize the corresponding loss functional

$$E(\mathbf{D}, \mathbf{A}) = \sum_{k=1}^{K} |\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\boldsymbol{\Phi}}^2, \tag{5.1}$$

we use a classical procedure. After initializing the dictionary with random elements of the dataset, the algorithm alternates between a sparse coding step in which $\mathbf{A}$ minimizes (5.1), and where $\mathbf{D}$ is fixed [33, 37, 36], and a dictionary update step where $\mathbf{A}$ is fixed.

5.1. Dictionary update with MOD. To update the dictionary, it is natural to think about performing gradient descent over $\mathbf{D}$ for updating the dictionary. Let us denote by $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_K) \in \mathbb{C}^{N \times K}$ the matrix whose columns contain the dataset. The gradient can be conveniently made explicit as $\boldsymbol{\nabla}_{\mathbf{D}} E(\mathbf{D}, \mathbf{A}) = 2\boldsymbol{\Phi}\left(\mathbf{D}\mathbf{A}\mathbf{A}^* - \mathbf{Z}\mathbf{A}^*\right)$. Here, however, we prefer to rely on the MOD introduced by [16], as done in the SPAMS software [37, 36]. Given a fixed value of $\mathbf{A}$, the dictionary is updated so that it solves $\min_{\mathbf{D}} E(\mathbf{D}, \mathbf{A})$, and its columns are then projected back onto $\mathcal{S}$. It is known that, in the landmarks case $\boldsymbol{\Phi} = \mathrm{Id}$, a particular solution to the least-squares problem $\min_{\mathbf{D} \in \mathbb{C}^{N \times J}} E(\mathbf{D}, \mathbf{A})$ is

$$\hat{\mathbf{D}} = \mathbf{Z}\mathbf{A}^+ = \mathbf{Z}\mathbf{V}\Sigma^+\mathbf{U}^*, \tag{5.2}$$

where $\mathbf{A}^+ = \mathbf{V}\Sigma^+\mathbf{U}^*$ is the pseudoinverse of $\mathbf{A}$, and the SVD decomposition of $\mathbf{A}$ is $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^*$. The pseudoinverse Σ^+ corresponds to the matrix Σ where nonzero diagonal elements

Algorithm 5.1. 2D Kendall shape dictionary with method of optimal directions.

procedure 2DKSD WITH MOD
Dataset: preshapes $\mathbf{z}_1, \dots, \mathbf{z}_K$
Parameters: number of iterations T, sparsity N_0, number of atoms J
Initialization: initial dictionary $\mathbf{D}_0$ of preshapes
Output: dictionary $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_J)$ of preshapes

$\mathbf{D} \leftarrow \mathbf{D}_0$ ▷ Dictionary initialization
for $t = 1, \dots, T$ **do**
  for $k = 1, \dots, K$ **do**
    $\boldsymbol{\alpha}_k \simeq \arg\min_{|\boldsymbol{\alpha}|_0 \le N_0} |\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}^2$ ▷ Sparse coding with COMPLEX ORMP (Algorithm 5.2)
  $\mathbf{D} \leftarrow \Pi_{\mathcal{S}}(\mathbf{Z}\mathbf{A}^+)$, where $\mathbf{A} = (\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_K)$. ▷ Dictionary update with MOD (equation (5.2))

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Algorithm 5.2. Complex order recursive matching pursuit.

procedure COMPLEX ORMP
Data: one preshape $\mathbf{z}$
Parameters: sparsity N_0
Input dictionary: $\mathbf{D} = (\mathbf{d}_1, \dots, \mathbf{d}_J)$
Output: weight $\boldsymbol{\alpha} \in \mathbb{C}^J$

$l = 1, \quad l_{\max} = N_0, \quad I[0] = \emptyset$
$\mathbf{r}^{(0)} = \mathbf{z}, \quad \boldsymbol{\delta}_j^{(0)} = \mathbf{d}_j$ for $j = 1, \dots, J$ ▷ Initialization
while $l \leq N_0$ **do**
  $j[l] = \left\{ \arg\max \frac{|\boldsymbol{\delta}_j^{(l-1)*} \boldsymbol{\Phi} \mathbf{r}^{(l-1)}|}{|\boldsymbol{\delta}_j^{(l-1)}|_{\boldsymbol{\Phi}}} \mid j \notin I[l-1], |\boldsymbol{\delta}_j^{(l-1)}|_{\boldsymbol{\Phi}} \neq 0 \right\}$
  $I[l] = I[l-1] \cup \{j[l]\}$
  if this maximum is zero **then**
    $l_{\max} = l - 1$
    break
  $\mathbf{q}^{(l)} = \frac{\boldsymbol{\delta}_{j[l]}^{(l-1)}}{|\boldsymbol{\delta}_{j[l]}^{(l-1)}|_{\boldsymbol{\Phi}}}$ ▷ New orthonormal basis element
  $\boldsymbol{\delta}_j^{(l)} = \boldsymbol{\delta}_j^{(l-1)} - (\mathbf{q}^{(l)*} \, \boldsymbol{\Phi} \, \boldsymbol{\delta}_j^{(l-1)}) \, \mathbf{q}^{(l)}$ for $j = 1, \dots, J$ ▷ Project on the orthogonal subspace
  $\mathbf{r}^{(l)} = \mathbf{r}^{(l-1)} - (\mathbf{q}^{(l)*} \, \boldsymbol{\Phi} \, \mathbf{r}^{(l-1)}) \, \mathbf{q}^{(l)}$ for $j = 1, \dots, J$ ▷ Update the remainder
  $l \leftarrow l + 1$
Use (5.5) to build $\boldsymbol{\alpha}$ ▷ Solve $\mathrm{P}_{\mathbb{C}\{\mathbf{d}_j\}_{j \in I[l_{\max}]}}(\mathbf{z}) = \mathbf{D}_{I[l_{\max}]} \boldsymbol{\alpha}_{I[l_{\max}]}$

are replaced by their multiplicative inverse. We assert in Lemma A.3 of Appendix A.2 that, interestingly, when $\boldsymbol{\Phi} \neq \mathrm{Id}$, the update is the same and does not depend on $\boldsymbol{\Phi}$.

Let us remark that the solution (5.2) cancels the gradient (one can see it by checking that $\mathbf{A}^+\mathbf{A}\mathbf{A}^* = \mathbf{A}^*$), and also that the update does not involve any previous value assigned to $\mathbf{D}$. When $(\mathbf{A}\mathbf{A}^*)$ is invertible,[5] we have $\mathbf{A}^+ = \mathbf{A}^*(\mathbf{A}\mathbf{A}^*)^{-1}$ and the solution is then unique and equal to $\hat{\mathbf{D}} = \mathbf{Z}\mathbf{A}^*(\mathbf{A}\mathbf{A}^*)^{-1}$.

After this first operation, the nonzero columns of the dictionary are projected back onto the preshape sphere by applying the normalization

$$\forall j = 1, \dots, J, \quad \mathbf{d}_j \leftarrow \mathbf{d}_j / |\mathbf{d}_j|_{\boldsymbol{\Phi}}.$$

This is sufficient as (5.2) already centers the atoms, since the data themselves are centered. We replace columns which have become zero, as well as underutilized ones, by preshapes randomly picked in the original dataset.

[5] This happens whenever the rank of $\mathbf{A}$ is equal to its number of lines or, equivalently, when the columns of $\mathbf{A}^*$ are independent.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

5.2. Sparse coding with ORMP. The sparse coding problem for a data point $\mathbf{z}$ is formulated as

$$\min_{\substack{\boldsymbol{\alpha}\in\mathbb{C}^J \\ |\boldsymbol{\alpha}|_0\leq N_0}} |\mathbf{z}-\mathbf{D}\boldsymbol{\alpha}|_{\boldsymbol{\Phi}}^2, \tag{5.3}$$

where $\mathbf{D}$ is known. Equivalently, the nonzero coefficients of the solution $\hat{\boldsymbol{\alpha}}$ correspond to a set of indices $\hat{I}$ that solves

$$\min_{\substack{I\subset\{1,\dots,J\} \\ |I|\leq N_0}} |\mathbf{z}-\mathrm{P}_{\mathbb{C}\{\mathbf{d}_j\}_{j\in I}}(\mathbf{z})|^2, \tag{5.4}$$

where $\mathbb{C}\{\mathbf{d}_j\}_{j\in I}$ denotes the complex vector space generated by the columns indexed by I.

Let us suppose that a solution $\hat{\boldsymbol{\alpha}}$ has been found. Then, the shape $[\mathbf{z}]$ is approximately reconstructed as $[\mathbf{D}\frac{\hat{\boldsymbol{\alpha}}}{|\mathbf{D}\hat{\boldsymbol{\alpha}}|_{\boldsymbol{\Phi}}}]$, whenever $\mathbf{D}\hat{\boldsymbol{\alpha}}\neq\mathbf{0}$. Note that, in fact, the original preshape $\mathbf{z}$ itself (with preserved orientation) can also be approximated as $\mathbf{D}\frac{\hat{\boldsymbol{\alpha}}}{|\mathbf{D}\hat{\boldsymbol{\alpha}}|_{\boldsymbol{\Phi}}}$: there is no need to rotate $\mathbf{D}\hat{\boldsymbol{\alpha}}$ in order to bring it close to the original preshape. As stated in the proof of Lemma A.2, this is because the solution $\hat{\boldsymbol{\alpha}}$ satisfies $\mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z}=\mathbf{D}\hat{\boldsymbol{\alpha}}$, implying that $\mathbf{D}\hat{\boldsymbol{\alpha}}$ corresponds to the optimal alignment of the preshape $\mathbf{D}\frac{\hat{\boldsymbol{\alpha}}}{|\mathbf{D}\hat{\boldsymbol{\alpha}}|_{\boldsymbol{\Phi}}}$ along $\mathbf{z}$ (see Definition 3.5). Also, $\mathbf{D}\frac{\hat{\boldsymbol{\alpha}}}{|\mathbf{D}\hat{\boldsymbol{\alpha}}|_{\boldsymbol{\Phi}}}$ is optimally rotated along $\mathbf{z}$ (see Proposition 3.6).

Order recursive matching pursuit. A possible way to approximate the solution of (5.3) is to rely on a complex version of orthogonal matching pursuit (OMP) [42, 43] adapted to the Hermitian framework $(\mathbb{C}^n,\boldsymbol{\Phi})$. Instead, we prefer to use ORMP, with a Cholesky-based optimization [9, 36], as it was empirically found to be more efficient than an adapted complex OMP. The ORMP algorithm is explained in [9] and a (non-optimized) implementation is presented in Algorithm 5.2. ORMP starts with an empty set of indices $I[0]=\emptyset$, a remainder $\mathbf{r}^{(0)}=\mathbf{z}$, and a family of vectors $\{\boldsymbol{\delta}_1^{(0)},\dots,\boldsymbol{\delta}_J^{(0)}\}=\{\mathbf{d}_1,\dots,\mathbf{d}_J\}$. At each step, the algorithm increases $I[l-1]$ by a new element $j[l]$ maximizing $\frac{|\boldsymbol{\delta}_j^{(l-1)*}\boldsymbol{\Phi}\mathbf{r}^{(l-1)}|}{|\boldsymbol{\delta}_j^{(l-1)}|_{\boldsymbol{\Phi}}}$ (wherever defined), in such a manner that $I[l]=I[l-1]\cup\{j[l]\}$ minimizes (5.4) under the nesting constraint $I[l-1]\subsetneq I$ and $|I[l]|=l$. Then, it updates the vectors $\{\boldsymbol{\delta}_1^{(l)},\dots,\boldsymbol{\delta}_J^{(l)}\}$ in a way that is similar to the Gram–Schmidt process.[6] It stops either when N_0 indices have been found or when $\mathbf{z}$ is spanned by $l_{\max}$ independent atoms and a new index $j[l_{\max}+1]$ does not contribute in minimizing (5.4) further. Finally, the output weight $\boldsymbol{\alpha}$ is a solution to $\mathrm{P}_{\mathbb{C}\{\mathbf{d}_j\}_{j\in I[l_{\max}]}}(\mathbf{z})=\mathbf{D}_{I[l_{\max}]}\boldsymbol{\alpha}_{I[l_{\max}]}$. Its nonzero coefficients are given by

$$\boldsymbol{\alpha}_{I[l_{\max}]}=(\mathbf{D}_{I[l_{\max}]}^*\boldsymbol{\Phi}\mathbf{D}_{I[l_{\max}]})^{-1}\mathbf{D}_{I[l_{\max}]}^*\boldsymbol{\Phi}\ \mathbf{z}. \tag{5.5}$$

Optimizing ORMP. Thanks to the so-called Cholesky optimization of ORMP [9, 37, 36], it is in fact possible to speed up computations considerably. The Gram matrix $\mathbf{D}^*\boldsymbol{\Phi}\mathbf{D}$ is

[6] The difference being that the order is imposed by $j[1],\dots,j[N_0]$, so that the final orthonormal family is $\{\boldsymbol{\delta}_{j[1]}^{(0)},\dots,\boldsymbol{\delta}_{j[N_0]}^{(N_0-1)}\}$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

precomputed before coding the data $\mathbf{z}$ (and reused for other data). At each step $1 \leq l \leq N_0$, only the elements

$$\frac{\boldsymbol{\delta}_{j[l]}^{(l-1)*}\boldsymbol{\Phi}\mathbf{d}_j}{|\boldsymbol{\delta}_{j[l]}^{(l-1)}|_{\boldsymbol{\Phi}}} = \frac{\boldsymbol{\delta}_{j[l]}^{(l-1)*}\boldsymbol{\Phi}\boldsymbol{\delta}_j^{(l-1)}}{|\boldsymbol{\delta}_{j[l]}^{(l-1)}|_{\boldsymbol{\Phi}}}, \qquad |\boldsymbol{\delta}_j^{(l)}|_{\boldsymbol{\Phi}}^2, \qquad \mathbf{d}_j^*\boldsymbol{\Phi}\mathbf{r}^{(l-1)}, \quad \text{and} \quad \frac{\boldsymbol{\delta}_j^{(l-1)*}\boldsymbol{\Phi}\mathbf{r}^{(l-1)}}{|\boldsymbol{\delta}_j^{(l-1)}|_{\boldsymbol{\Phi}}} = \frac{\mathbf{d}_j^*\boldsymbol{\Phi}\mathbf{r}^{(l-1)}}{|\boldsymbol{\delta}_j^{(l-1)}|_{\boldsymbol{\Phi}}}$$

are updated, for $j = 1, \ldots, J$. The coefficients of the decomposition of $\mathbf{q}^{(l)} = \frac{\boldsymbol{\delta}_{j[l]}^{(l-1)}}{|\boldsymbol{\delta}_{j[l]}^{(l-1)}|_{\boldsymbol{\Phi}}}$ in the linearly independent family $\{\mathbf{d}_{j[1]}, \ldots, \mathbf{d}_{j[l]}\}$ are also iteratively computed. If these coefficients are stacked into a column vector $\mathbf{c}^{(l)} \in \mathbb{C}^{N_0}$ where the last $N_0 - l$ coefficients are zero, the matrix $\mathbf{U} = (\mathbf{c}^{(1)}, \ldots, \mathbf{c}^{(N_0)}) \in \mathbb{C}^{N_0 \times N_0}$ is upper triangular with strictly positive real scalars on the diagonal. Furthermore, it is the inverse conjugate of the Cholesky factor $\mathbf{L}$ involved in the factorization $\mathbf{D}_{I[N_0]}^*\boldsymbol{\Phi}\mathbf{D}_{I[N_0]} = (\mathbf{U}^*)^{-1}\mathbf{U}^{-1} = \mathbf{L}\mathbf{L}^*$. The final weight $\boldsymbol{\alpha}$ is then given by

$$\boldsymbol{\alpha}_{I[N_0]} = \mathbf{U}\left[\frac{\boldsymbol{\delta}_{j[1]}^{(0)*}\boldsymbol{\Phi}\mathbf{r}^{(0)}}{|\boldsymbol{\delta}_{j[1]}^{(0)}|_{\boldsymbol{\Phi}}}, \ldots, \frac{\boldsymbol{\delta}_{j[N_0]}^{(N_0-1)*}\boldsymbol{\Phi}\mathbf{r}^{(N_0-1)}}{|\boldsymbol{\delta}_{j[N_0]}^{(N_0-1)}|_{\boldsymbol{\Phi}}}\right]^T.$$

All computations can be inferred from the updates in Algorithm 5.2 and are not detailed here. They can be transparently investigated in the available source code.

6. Computational results. We ran our algorithm on 5 datasets that vary in size, type, and shape.[7]

- Dataset 1[8] consists of $K > 5500$ skeletons of *Caenorhabditis elegans* nematodes delineated as discretely defined configurations of $N = 20$ landmarks. The dataset results from the concatenation of videos featuring freely crawling nematodes aged between 2 and 18 days and presenting various locomotion behaviors. The dataset was reconstructed using $N_0 = 5$ atoms out of the $J = 10$ atoms learned by 2DKSD.
- Dataset 2[9] contains $K > 6300$ skeletons of four *Caenorhabditis elegans* nematodes delineated as continuously defined open Hermite-spline curves [52, 45] with $N = 12$ degrees of freedom. The animals laid in a shared container and were constrained by the lack of space, leading to looping or wavy shapes. The dataset was reconstructed using $N_0 = 5$ atoms out of the $J = 10$ atoms learned by 2DKSD.
- Dataset 3[10] features $K = 40$ hands of four different people, as outlined by $N = 56$ landmarks. The dataset was reconstructed using $N_0 = 3$ atoms out of the $J = 10$ atoms learned by 2DKSD.
- Datasets 4 and 4b[11] are constituted of $K > 1500$ leaves outlines and come in the form of configurations of $N = 200$ landmarks and cubic B-splines [53, 5] with $N = 40$

[7] Experimental results and datasets from this section can be found at https://github.com/ansonang3/2DKSD.

[8] Images were extracted from videos available from the *C. elegans* behavioral database [58].

[9] Data courtesy of [39].

[10] Data courtesy of [49].

[11] Data extracted from the binary images of the Kaggle leaf dataset, https://www.kaggle.com/c/leaf-classification.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

degrees of freedom, respectively. For most shapes, the first landmark marks either the stem or the leaf tip. Both datasets were (independently) reconstructed using $N_0 = 5$ atoms out of the $J = 20$ atoms learned by 2DKSD.

- Dataset 5 is a synthetic dataset generated by deforming original silhouettes extracted from the binary images of the MPEG-7 database[12]. For each original preshaped configuration $\mathbf{z}_{\text{or}} = (\mathbf{z}^1_{\text{or}}, \dots, \mathbf{z}^N_{\text{or}}) \in \mathbb{C}^N$, N random Gaussian planar vectors were generated and smoothed with the kernel $K = (|\mathbf{z}^i_{\text{or}} - \mathbf{z}^j_{\text{or}}|^2)_{i,j}$ of the squared distances between landmarks. These deformations were then added to $\mathbf{z}_{\text{or}}$. The process was repeated 10 times, resulting in a dataset of $K = 14000$ shapes featuring 70 classes of objects, each containing 200 configurations of $N = 200$ landmarks. The dataset was reconstructed using $N_0 = 30$ atoms out of the $J = 230$ atoms learned by 2DKSD.

Comparison to the "align-first" method. The problem, which we call "align-first," relating most to the one we are solving, is stated as

$$\inf_{\substack{\mathbf{D}', \mathbf{A}': \\ |\boldsymbol{\alpha}'_k|_0 \le N_0}} \sum_{k=1}^{K} \left|\widetilde{\mathbf{z}}_k - \mathbf{D}'\boldsymbol{\alpha}'_k\right|^2_{\boldsymbol{\Phi}}, \qquad \text{where } \boldsymbol{\alpha}'_k \in \mathbb{R}^J, \tag{6.1}$$

where $\mathbf{A}' = (\boldsymbol{\alpha}'_1, \dots, \boldsymbol{\alpha}'_K)$, and $\widetilde{\mathbf{z}}_1, \dots, \widetilde{\mathbf{z}}_K \in \mathcal{S}$ are preshapes optimally rotated along a reference preshape. Typically, we chose to rotate them along a mean preshape $\mathbf{z}_{\text{mean}}$ (see Appendix A.3). This standard problem is essentially the same as in (4.3), except that the original dataset is preprocessed and, more notably, the weights $\boldsymbol{\alpha}'_k$ are real vectors.

In practice, we took advantage of the MOD combined with the real version of the Cholesky-optimized ORMP provided by the `SPAMS` toolbox [37]. This is an especially appropriate comparison, since our algorithm is a direct adaptation of the latter to the Hermitian framework. Thanks to an isometric transformation through $\sqrt{\boldsymbol{\Phi}}$, (6.1) could be reformulated as a standard ℓ^2 problem.

Comparison to complex PCA. As a valuable and complementary comparison, we also considered the complex version of PCA. It also relies on complex weights, but atoms are chosen in a specific way, which we recall here in our setting. Suppose that the average configuration has been discarded in each element of the dataset, and let us still use $\mathbf{z}_1, \dots, \mathbf{z}_K$ to denote this new dataset (the average configuration coordinates are simply the coefficient-by-coefficient arithmetic means). We find the first complex PCA mode by solving

$$\arg\min_{|\mathbf{w}_1|_{\boldsymbol{\Phi}} \le 1} \sum_{k=1}^{K} |\mathbf{z}_k - (\mathbf{w}_1^* \boldsymbol{\Phi} \mathbf{z}_k)\mathbf{w}_1|^2_{\boldsymbol{\Phi}} = \arg\min_{|\mathbf{w}_1|_{\boldsymbol{\Phi}} \le 1} \sum_{k=1}^{K} |\mathbf{z}_k|^2_{\boldsymbol{\Phi}} - |\mathbf{w}_1^* \boldsymbol{\Phi} \mathbf{z}_k|^2 = \arg\max_{|\mathbf{w}_1|_{\boldsymbol{\Phi}} \le 1} \sum_{k=1}^{K} |\mathbf{w}_1^* \boldsymbol{\Phi} \mathbf{z}_k|^2. \tag{6.2}$$

Let us set $\mathbf{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_K)$ as before. The matrix $\mathbf{Z}\mathbf{Z}^*\boldsymbol{\Phi}$ is self-adjoint with respect to the Hermitian product $\boldsymbol{\Phi}$, because $\forall \mathbf{x}, \mathbf{y} \in \mathbb{C}^N, (\mathbf{Z}\mathbf{Z}^*\boldsymbol{\Phi}\ \mathbf{x} \mid \mathbf{y})_{\boldsymbol{\Phi}} = (\mathbf{x} \mid \mathbf{Z}\mathbf{Z}^*\boldsymbol{\Phi}\ \mathbf{y})_{\boldsymbol{\Phi}}$. By the spectral theorem, this implies that there exists a $\boldsymbol{\Phi}$-orthonormal basis $\mathbf{V} \in \mathbb{C}^{N \times N}$ with $\mathbf{V}^*\boldsymbol{\Phi}\mathbf{V} = \mathrm{Id}$ in which $\mathbf{Z}\mathbf{Z}^*\boldsymbol{\Phi}$ is expressed as a diagonal matrix $\mathbf{D}$:

$$\mathbf{Z}\mathbf{Z}^*\boldsymbol{\Phi} = \mathbf{V}\mathbf{D}\mathbf{V}^*\boldsymbol{\Phi} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}.$$

[12] Data courtesy of [32]. See http://www.dabi.temple.edu/~shape/MPEG7/dataset.html.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Table 2
RMSE and parameters used to generate the results of Figures 7 to 11.

Dataset	N	K	N_0	J	Indicative run-time	RMSE, **ours**	RMSE, **align-first**
1	20	> 5500	5	10	21 s *	2.36%	4.19%
2	12	> 6300	5	10	24 s *	4.04%	7.95%
3	56	40	3	10	< 0.4 s	2.48%	2.73%
4	200	> 1500	5	20	18 s	6.45%	7.12%
4b	40	> 1500	5	20	12 s	6.41%	7.01%
5	200	14000	30	230	441 s **	1.57%	2.71%

Since $\mathbf{Z}\mathbf{Z}^*\mathbf{\Phi}$ is positive semidefinite with respect to $\mathbf{\Phi}$, we know that $\mathbf{D} = \operatorname{diag}(\lambda_1, \dots, \lambda_N)$ has (real) nonnegative values that can be ordered as $\lambda_1 \geq \cdots \geq \lambda_N \geq 0$. A solution of (6.2) then corresponds to a unit-norm eigenvector of $\mathbf{Z}\mathbf{Z}^*\mathbf{\Phi}$ corresponding to λ_1. The second complex PCA mode $\mathbf{w}_2$ is the solution to a problem analogous to (6.2), except that we add the constraint $\mathbf{w}_2 \perp_{\mathbf{\Phi}} \mathbf{w}_1$. It is given by a unit-norm eigenvector corresponding to λ_2. Further modes are iteratively found with the constraint to be $\mathbf{\Phi}$-orthogonal to previous modes.

The dictionary is then the collection of the modes $(\mathbf{w}_1, \dots, \mathbf{w}_J)$ and the best N_0-term approximation of $\mathbf{z}_k$ is $\sum_{j=1}^{N_0} (\mathbf{w}_j^* \mathbf{\Phi} \mathbf{z}_k) \mathbf{w}_j$. The corresponding loss is then equal to $\sum_{k=1}^{K} |\mathbf{z}_k|_{\mathbf{\Phi}}^2 - \lambda_1 - \cdots - \lambda_{N_0}$. As in the previous paragraph, we employ $\sqrt{\mathbf{\Phi}}$ to transform the problem into a standard form suitable for computations.

6.1. Results. In Figures 7, 8, 9, 10, 11, and 12, we show the dictionary learned by 2DKSD and the align-first method with specific parameters for each dataset (Table 2), alongside four examples of reconstructed data superimposed over the original one. In each case, the reconstruction errors

$$|\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\mathbf{\Phi}} = \frac{|\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\mathbf{\Phi}}}{|\mathbf{z}_k|_{\mathbf{\Phi}}}$$

are expressed in percentages and can be understood as an absolute error multiplied by 100, or as a proportion of $|\mathbf{z}_k| = 1$ as well. Note that, as mentioned in section 5.2, the 2DKSD ensures that these errors are equal to $d_F([\mathbf{z}_k], [\mathbf{D}\boldsymbol{\alpha}_k])$, which is not the case for the align-first method. We also indicate the root mean square reconstruction errors (RMSE), defined as

$$\sqrt{\frac{E}{K}} = \sqrt{\frac{1}{K}\sum_{k=1}^{K} |\mathbf{z}_k - \mathbf{D}\boldsymbol{\alpha}_k|_{\mathbf{\Phi}}^2} \quad \text{and} \quad \sqrt{\frac{E'}{K}} = \sqrt{\frac{1}{K}\sum_{k=1}^{K} |\widetilde{\mathbf{z}}_k - \mathbf{D}'\boldsymbol{\alpha}'_k|_{\mathbf{\Phi}}^2},$$

respectively, and also expressed in percentages. We summarize them in Table 2, where we specify the parameters used in the examples, as well as a typical run-time[13] for $T = 30$ iterations in Algorithm 5.1, which is sufficient for the loss to converge in all datasets, and including the duration of the final sparse coding of the whole dataset. To reduce ORMP run-time for larger datasets (such as Datasets 1 and 2, marked with a star * in the table), we applied Algorithm 5.2 in parallel and independently on each data point $\mathbf{z}_k$. To learn overcomplete dictionaries on very big datasets (such as Dataset 5, marked with two stars ** in the table),

[13] All experiments were run on a standard laptop with an Intel Core i5-7200U CPU running 4 cores at 2.50GHz, with 7,7 Gb of RAM.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

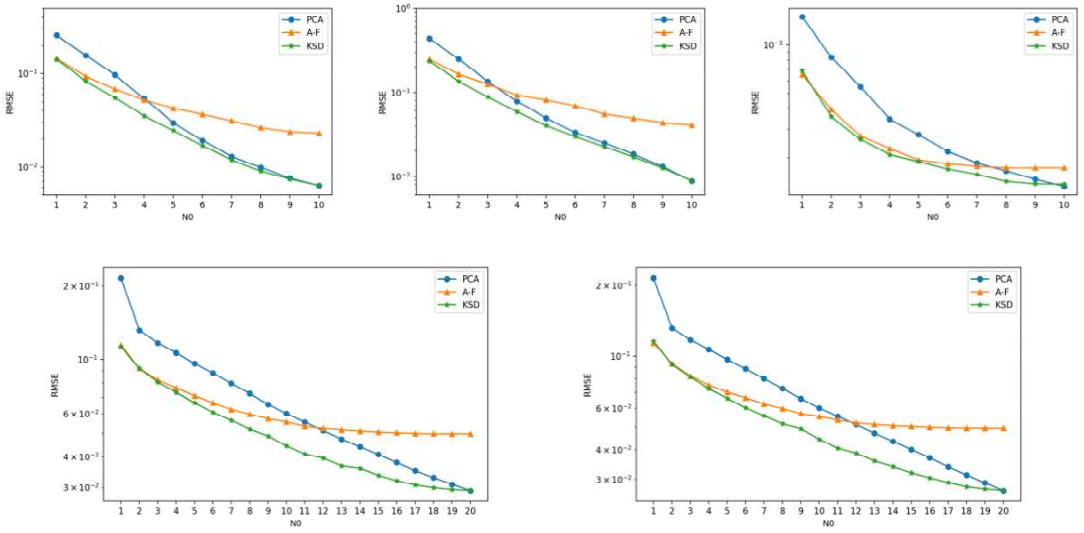


Figure 5. *RMSE of complex PCA, (real) align-first, and* 2*DKSD, in logarithmic scale. From left to right, top to bottom: Datasets* 1 *to* 4*b. In each graph, we plot the RMSE as a function of the number* N_0 *of atoms used in the reconstructions. For complex PCA, we used the first* N_0 *modes. For align-first and* 2*DKSD, we learned a dictionary parameterized by* (N_0, J) *for each value of* N_0*, with* J *set to* 10 *(top row) or* 20 *(bottom row).*

we chose to update the weights $\boldsymbol{\alpha}_k$ stochastically and in parallel by batches of 4096, after initializing them to zero. Obviously, additional efforts could be made to optimize the implementation and reduce run-time on big datasets, but this falls out of the scope of the present work and is kept for the future.

In Figure 5, we illustrate in logarithmic scale the RMSE obtained with complex PCA v.s. (real) align-first v.s. 2DKSD, after learning atoms from Datasets 1 to 4b. For varying values of N_0, we compare the RMSE of a reconstruction relying on the first N_0 PCA modes, and the RMSEs of an align-first or 2DKSD dictionary learned with the parameters (N_0, J). $J \geq N_0$ is fixed to the same values used in the examples shown in the figures and is such that $J < N$. This comparison is relevant for two reasons. First, the dictionary of PCA atoms is a natural candidate when learning an undercomplete dictionary containing fewer atoms than the dimension of the configuration space. Second, while complex PCA relies on complex weights but determines the atoms in a specific way, the align-first method is less constrained in the choice of atoms but relies on real weights. Our 2DKSD method is therefore a good compromise between them. We have also explored the $J > N$ case, which leads to an overcomplete dictionary (Dataset 5, Figure 12).

Comparison with align-first and complex PCA. We argue that the align-first method provides less satisfying results than our shape dictionary, both mathematically and numerically. Mathematically, the optimized loss functional (6.1), denoted by E', is necessarily larger than our loss E (5.1) since weights are constrained to lie on the real line. Intuitively, they can

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

scale but not rotate the atoms before summing them. In contrast, the complex weights in our method scale and rotate the atoms, resulting in a smaller loss E. Numerically, this leads to a visible difference in the RMSEs, as seen in Figure 5. In particular, the loss obtained with our algorithm E can be a third or a fourth of the one resulting from align-first E', as in Figures 7 and 8. Besides, the dictionary computed by align-first often contains distorted, irregular or unrealistic atoms, which reflect the effort required to fit the data, in spite of the prealignment. When analyzing datasets of shapes, it is sometimes preferable to obtain realistic atoms and visually accurate reconstructions instead of perfect reconstructions but unrealistic atoms. Visually realistic atoms indeed offer a way to hypothesize on the nature of the variability of the dataset.

Figure 5 demonstrates the benefits of using complex numbers in the linear combinations. For increasing values of N_0, the align-first RMSE indeed becomes significantly larger even when compared to that of complex PCA. As another illustration of their relevance, Figure 6 features two different weighted sums of the same three atoms that allow reconstructing very dissimilar shapes. It is not surprising for the first hand to be correctly reconstructed as, at first order, its shape is similar to that of the first atom, which is weighted by the coefficient of largest magnitude. The remainder is compensated by the other two atoms. For the second hand, however, it is more surprising to see that hands with open fingers, scaled, rotated, and then summed together, manage to produce a hand with closed fingers.

Last, Figure 5 shows that 2DKSD is more appropriate than complex PCA for $N_0 < J$. When learning a KSD dictionary that uses all the $N_0 = J < N$ atoms to reconstruct data, the RMSE is close to that of complex PCA. In fact, the minimal RMSEs should mathematically be equal whenever the average configuration of the dataset corresponds to the zero configuration, or if it is not discarded from the dataset (see the details of complex PCA discussed above).

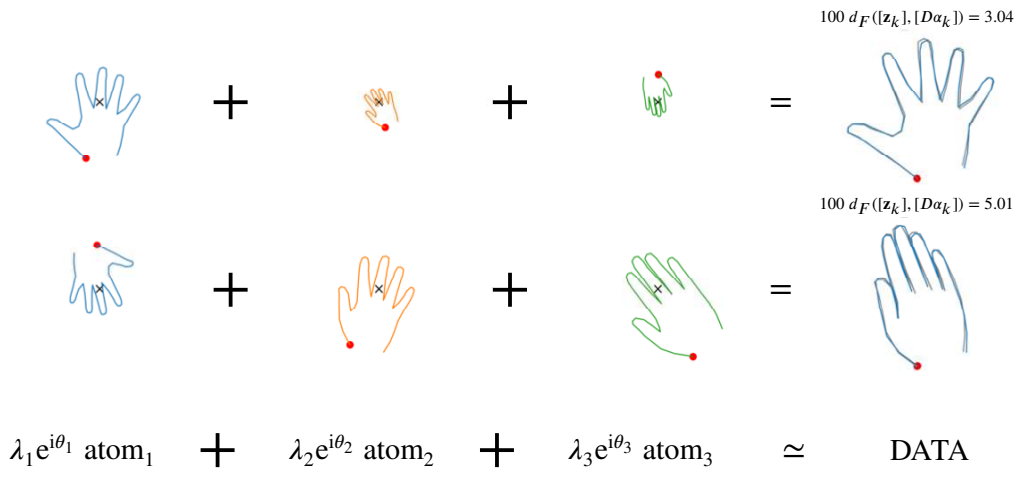


Figure 6. *Complex linear combinations offer more freedom in the reconstruction. Left: three atoms weighted with different complex coefficients. Right: their sum (blue) superimposed over reconstructed data (gray). Black crosses indicate the origin.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

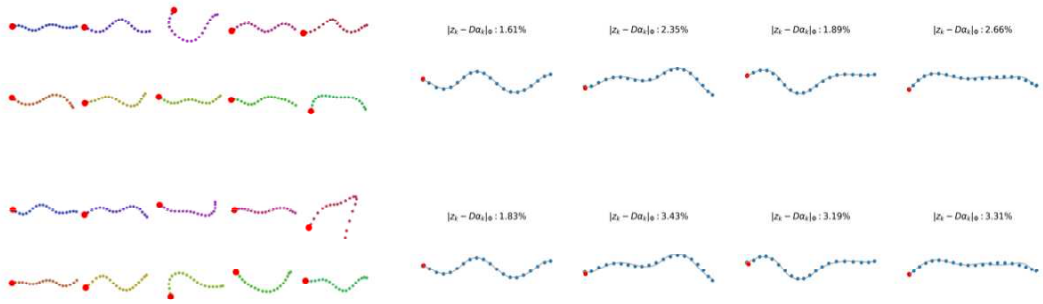


Figure 7. *Dataset* 1. *Top: our method. Bottom: align-first method. Left: shape dictionary of* $J = 10$ *atoms, taking* $N_0 = 5$ *out of them to reconstruct the* $K > 5500$ *shapes represented as configurations of dimension* $N = 20$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 2.36% *and* 4.19%, *respectively.*

Indeed, both solutions find the J-dimensional subspace that is the closest to the data points, i.e., the subspace spanned by the J PCA modes.

Comments on the examples comparing 2DKSD and align-first. As is seen in Figures 7 and 8, nematode shapes seem to be efficiently reconstructed using not more than five atoms. For both Datasets 1 and 2, the reconstructions are visually satisfying and numerically accurate. The results for Dataset 2 are remarkable in that, although the original shapes do not have the characteristic smoothness of freely moving worms, the algorithm has less difficulty in the

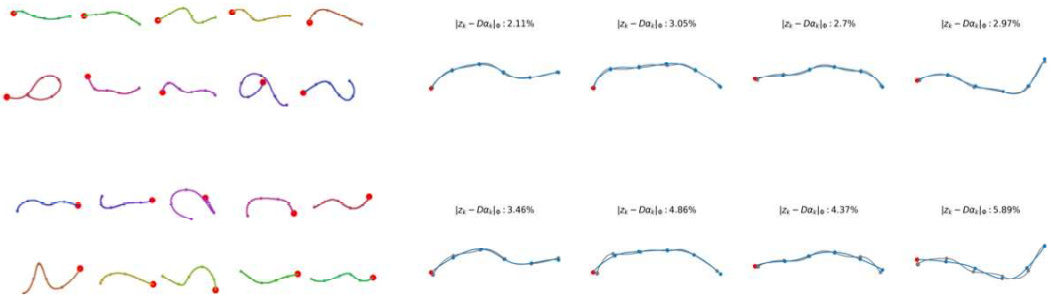


Figure 8. *Dataset* 2. *Top: our method. Bottom: align-first method. Left: shape dictionary of* $J = 10$ *atoms, taking* $N_0 = 5$ *out of them to reconstruct the* $K > 6300$ *shapes represented as configurations of dimension* $N = 12$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 4.04% *and* 7.95%, *respectively. We display the curves and the control points.*

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

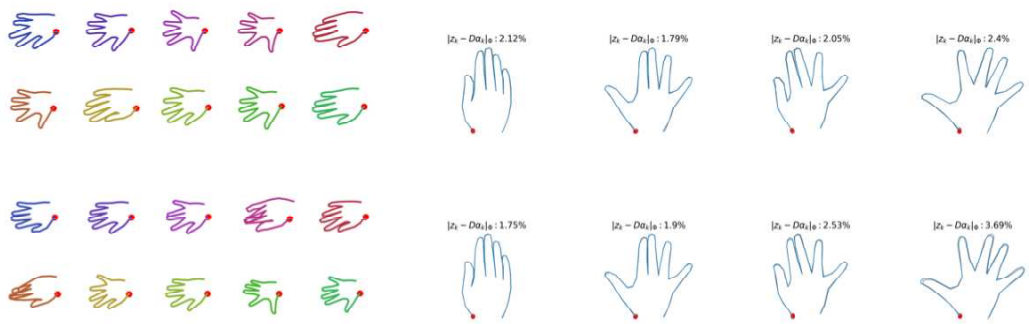


Figure 9. *Dataset* 3. *Top: our method. Bottom: align-first method. Left: shape dictionary of* $J = 10$ *atoms, taking* $N_0 = 3$ *out of them to reconstruct the* $K = 40$ *shapes represented as configurations of dimension* $N = 56$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 2.48% *and* 2.73%, *respectively. We display configurations of landmarks as continuous curves.*

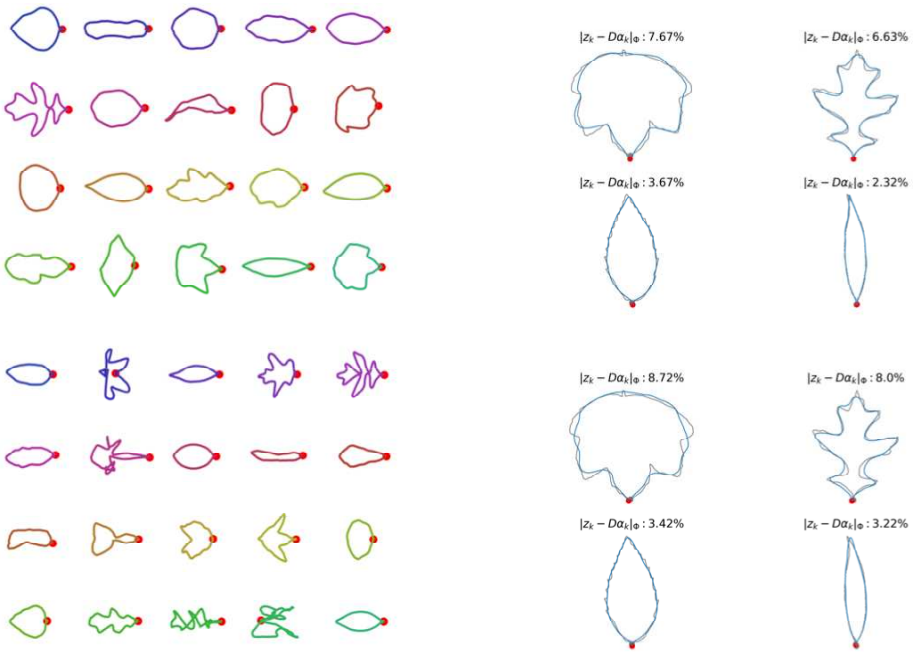


Figure 10. *Dataset* 4. *Top: our method. Bottom: align-first method. Left: shape dictionary of* $J = 20$ *atoms, taking* $N_0 = 5$ *out of them to reconstruct the* $K > 1500$ *shapes represented as configurations of dimension* $N = 200$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 6.45% *and* 7.12%, *respectively. We display the configurations of landmarks as continuous curves.*

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

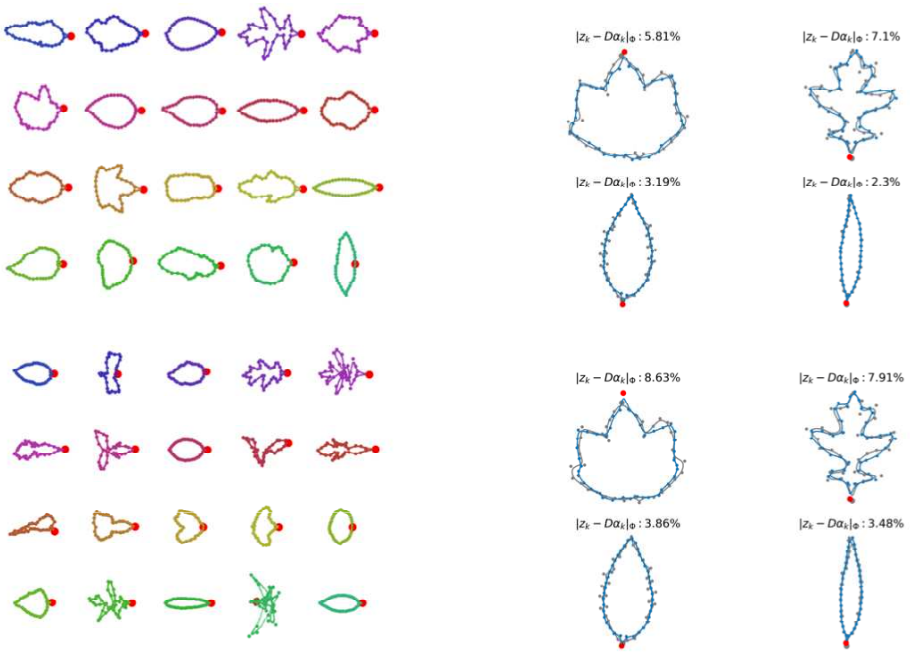


Figure 11. *Dataset 4b. Top: our method. Bottom: align-first method. Left: shape dictionary of* $J = 20$ *atoms, taking* $N_0 = 5$ *out of them to reconstruct the* $K > 1500$ *shapes represented as configurations of dimension* $N = 40$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 6.41% *and* 7.01%, *respectively. We display the curves and also the control points, which explains the nonsmooth effects.*

reconstruction than align-first. Moreover, it produces realistic atoms, whose curve parameterization is well-balanced, in contrast to those of align-first, where consecutive control points are not regularly distributed along the curve.

In Figure 9 reporting the results for Dataset 3, both methods result in very similar losses, but the atoms are less realistic with align-first. They exhibit self-intersecting and stretched fingers. In this case, it is an informative feature as it indicates to which extent fingers "tend" to be closed. Our algorithm is then an alternative to the align-first method.

The results in Figures 10 and 11 on the leaf datasets 4 and 4b are similar regardless of the representation (landmarks or B-splines) used. Our loss E is less than 84% of the comparison loss E', and we obtain a dictionary with realistic leaves, while align-first contains twisted shapes.

Finally, we observe that the overcomplete dictionaries learned by both methods on Dataset 5, using 20 of their atoms shown in Figure 12, look fairly similar. Thus, 2DKSD offers no particular advantage over align-first regarding the visual realisticness of the atoms in that case but leads to a significantly smaller RMSE. It also reproduces better high-frequency features of the silhouettes, as in the cow and lizard shapes.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

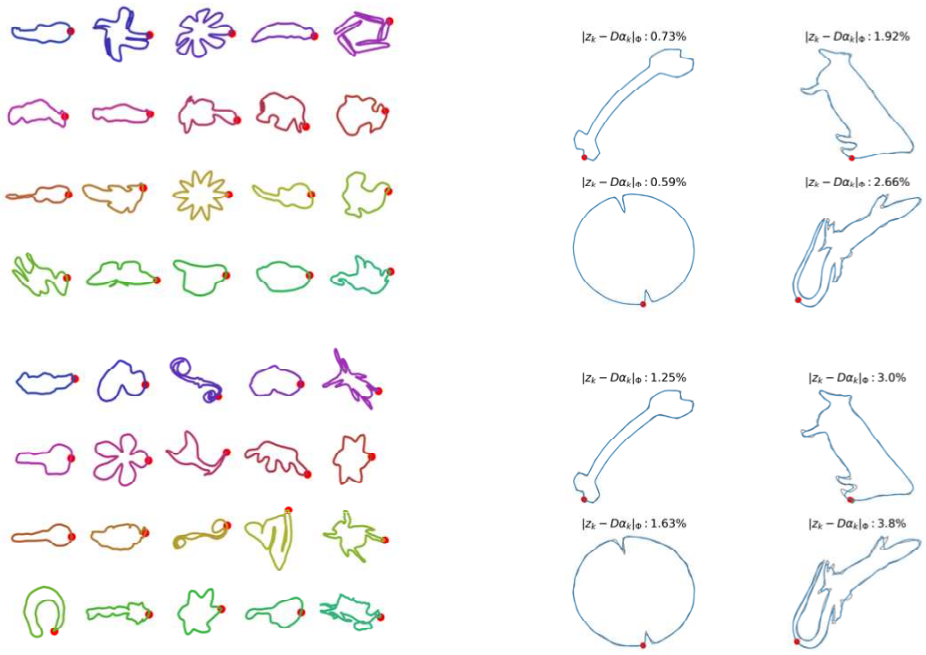


Figure 12. *Dataset* 5. *Top: our method. Bottom: align-first method. Left: first* 20 *atoms of the shape dictionary of* $J = 230$ *atoms, taking* $N_0 = 30$ *out of them to reconstruct the* $K = 14000$ *shapes represented as configurations of dimension* $N = 200 < J$. *Right: four examples of reconstruction (blue) over the original data (gray). The RMSEs are* 1.57% *and* 2.71%, *respectively.*

7. Conclusion.

Our 2D Kendall shape dictionary approach is a natural adaptation of usual learning techniques to Kendall's nonlinear manifold and does not require sophisticated operations. For the classical full Procrustes metric, it simplifies to a nearly standard formulation in which complex weights are used when combining the atoms of the dictionary. As our main contribution, this formulation has the double property of enjoying *both a strong theoretical justification and a simple algorithmic framework*. We have demonstrated the positive impact of using a complex setting: datasets do not need to be aligned along a reference mean because atoms are freely scaled and rotated before being summed to reconstruct shapes independently. This flexibility increases reconstruction accuracy and allows dictionary atoms to remain visually realistic. We have also extended Kendall's space of planar shapes, initially defined for discrete configurations of landmarks, to continuously defined interpolating curves, by introducing the general notion of configuration. Hopefully, our method is a promising tool for characterizing complex phenotypes from biological images.

Appendix A.

A.1. Hermitian products hold twice as much information than scalar products.

In what follows, we make the identification $\mathbb{C}^N \simeq \mathbb{R}^{2N}$ through the mapping $\mathbf{z} \mapsto z = (\Re(\mathbf{z}), \Im(\mathbf{z}))^T$,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\Re(\mathbf{z}) = (\Re(\mathbf{z}[0]), \dots, \Re(\mathbf{z}[N-1]))$ and likewise for the imaginary part. We use bold and italics to distinguish the complex and real counterparts of the vectors. If $(\mathbb{C}^N, \mathbf{\Phi})$ is endowed with a Hermitian inner product $(\mathbf{z}, \mathbf{w}) \mapsto (\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} := \mathbf{z}^* \mathbf{\Phi} \mathbf{w}$ associated to the matrix $\mathbf{\Phi} \in \mathbb{C}^{N \times N}$, then one can define on $\mathbb{R}^{2N}$ the scalar product $\langle \cdot, \cdot \rangle_{\mathbf{\Phi}}$ canonically associated to it as

$$\langle z, w \rangle_{\mathbf{\Phi}} := \Re\left((\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} \right), \tag{A.1}$$

defining the same norm $\langle z, z \rangle_{\mathbf{\Phi}} = (\mathbf{z} \mid \mathbf{z})_{\mathbf{\Phi}}$. Due to the sesquilinearity of the Hermitian product, we have that $(\mathrm{i}\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} = -\mathrm{i}(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}$, so that $\Re\left((\mathrm{i}\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}\right) = \Im\left((\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}}\right)$, and then

$$(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} = \langle z, w \rangle_{\mathbf{\Phi}} + \mathrm{i} \langle R_{\pi/2} \odot z, w \rangle_{\mathbf{\Phi}}, \tag{A.2}$$

where by $R_\theta \odot z = \cos\theta \ (\Re(\mathbf{z}), \Im(\mathbf{z}))^T + \sin\theta \ (-\Im(\mathbf{z}), \Re(\mathbf{z}))^T$ we denote the image (in the real setting) of $\mathbf{z}$ by a rotation by θ, namely, the real counterpart of $e^{i\theta}\mathbf{z}$. Here, $R_{\pi/2} \odot z = (-\Im(\mathbf{z}), \Re(\mathbf{z}))^T$.

As a consequence, $(\mathbf{z} \mid \mathbf{w})_{\mathbf{\Phi}} = 0$ if and only if $\langle z, w \rangle_{\mathbf{\Phi}} = 0$ and $\langle R_{\pi/2} \odot z, w \rangle_{\mathbf{\Phi}} = 0$. In fact, a stronger property can be deduced: $\forall \theta \in [0, 2\pi)$, $\langle R_\theta \odot z, w \rangle_{\mathbf{\Phi}} = \langle z, R_\theta \odot w \rangle_{\mathbf{\Phi}} = 0$. This is because $R_\theta \odot z = \cos\theta \ z + \sin\theta \ R_{\pi/2} \odot z$, and we conclude by bilinearity of the scalar product, and then by exchanging the roles of z and w. In other words, *Hermitian orthogonality can be understood as a real orthogonality between any rotated image of* $\mathbf{z}$ *and* $\mathbf{w}$ *(and conversely).*

Remark A.1. Suppose that $\mathbf{\Phi} = \mathbf{A} + \mathrm{i}\mathbf{B}$ with $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N \times N}$. The scalar product is computed as $\langle z, w \rangle_{\mathbf{\Phi}} = z^T \mathbf{\Psi} w$, with $\mathbf{\Psi} = \left(\begin{smallmatrix} \mathbf{A} & -\mathbf{B} \\ \mathbf{B} & \mathbf{A} \end{smallmatrix}\right)$.

A.2. Two lemmas. In this section, we establish two lemmas. The first one is used in the proof of Proposition 4.2, and the second one states that the dictionary update (5.2) does not involve the matrix $\mathbf{\Phi}$.

Lemma A.2. *Let* $\hat{\boldsymbol{\alpha}} \in \mathbb{C}^J$ *be a minimizer of* $\min_{|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathbf{D}\boldsymbol{\alpha}|_{\mathbf{\Phi}}$. *If* $\mathbf{D}\hat{\boldsymbol{\alpha}} \neq \mathbf{0}$, *then any term in* (4.5) *is also minimized by* $\hat{\boldsymbol{\alpha}}$. *Otherwise,* $\mathbf{z}$ *is orthogonal to the columns of* $\mathbf{D}$ *and the distances in* (4.5) *are all maximized to* 1. *In that case, any* $\widetilde{\boldsymbol{\alpha}}$ *satisfying* $\mathbf{D}\widetilde{\boldsymbol{\alpha}} \neq \mathbf{0}$ *is a minimizer of any term in* (4.5).

Proof. The quantity $|\mathbf{z} - \mathbf{D}\boldsymbol{\alpha}|_{\mathbf{\Phi}}$ reaches a minimum in $\{\boldsymbol{\alpha} \mid |\boldsymbol{\alpha}|_0 \leq N_0\}$. It corresponds to the minimal distance to $\mathbf{z}$ of a subspace generated by at most N_0 atoms $\mathbf{d}_j$ (see previous discussions about sparse coding). Let $\hat{\boldsymbol{\alpha}}$ denote the corresponding minimizer. It satisfies $\min_{|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathbf{D}\boldsymbol{\alpha}|_{\mathbf{\Phi}} = |\mathbf{z} - \mathbf{D}\hat{\boldsymbol{\alpha}}|_{\mathbf{\Phi}}$. By definition of a projection, $|\mathbf{z} - \mathbf{D}\hat{\boldsymbol{\alpha}}| \geq |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z}|$. Moreover, if $\lambda \in \mathbb{C}$ is such that $\mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z} = \lambda \mathbf{D}\hat{\boldsymbol{\alpha}}$, then $|\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z}| = |\mathbf{z} - \mathbf{D}(\lambda\hat{\boldsymbol{\alpha}})|$, where $|\lambda\hat{\boldsymbol{\alpha}}|_0 \leq N_0$. The last term is then greater than or equal to $\min_{|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathbf{D}\boldsymbol{\alpha}|_{\mathbf{\Phi}}$, hence it comes that

$$|\mathbf{z} - \mathbf{D}\hat{\boldsymbol{\alpha}}|_{\mathbf{\Phi}} = |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z}|_{\mathbf{\Phi}}. \tag{A.3}$$

Using the same arguments, one easily shows that this also corresponds to $\min_{|\boldsymbol{\alpha}|_0 \leq N_0} |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{D}\boldsymbol{\alpha}}\mathbf{z}|_{\mathbf{\Phi}}$. As a useful remark, notice that the equality (A.3) implies that $\mathrm{P}_{\mathbb{C}\mathbf{D}\hat{\boldsymbol{\alpha}}}\mathbf{z} = \mathbf{D}\hat{\boldsymbol{\alpha}}$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Coming back to the original equalities (4.5), if $\mathbf{D}\hat{\boldsymbol{\alpha}} \neq \mathbf{0}$, then all terms are minimized by $\hat{\boldsymbol{\alpha}}$. Otherwise, if $\mathbf{D}\hat{\boldsymbol{\alpha}} = \mathbf{0}$, then $\mathbf{z}$ is orthogonal to the vectors $\mathbf{d}_j$, and $\mathbf{z}$ is not correlated to any shape generated by the $\mathbf{d}_j$, inducing a maximal distance d_F. In particular, any $\widetilde{\boldsymbol{\alpha}}$ such that $|\widetilde{\boldsymbol{\alpha}}|_0 \leq N_0$ and $\mathbf{D}\widetilde{\boldsymbol{\alpha}} \neq \mathbf{0}$ can be taken as a minimizer of any term in (4.5). ∎

Lemma A.3. *E being defined in (5.1), a solution to $\min_{\mathbf{D}\in\mathbb{C}^{n\times J}} E(\mathbf{D},\mathbf{A})$, for a fixed value of $\mathbf{A}$, is given by (5.2), and its expression is independent from the matrix $\boldsymbol{\Phi}$.*

Proof. We rewrite the problem into a standard ℓ^2 form:

$$\begin{aligned}\min_{\mathbf{D}} E(\mathbf{D},\mathbf{A}) = \min_{\mathbf{D}} \sum_{k=1}^{K} |\mathbf{D}\alpha_k - \mathbf{z}_k|_{\boldsymbol{\Phi}}^2 &= \min_{\mathbf{D}} \sum_{k} |\sqrt{\boldsymbol{\Phi}}\mathbf{D}\alpha_k - \sqrt{\boldsymbol{\Phi}}\mathbf{z}_k|^2 \\ &= \min_{\mathbf{D}} \|\sqrt{\boldsymbol{\Phi}}(\mathbf{D}\mathbf{A} - \mathbf{Z})\|_{\mathrm{Fro}}^2 \\ &= \min_{\mathbf{D}} \|(\mathbf{A}^*\mathbf{D}^* - \mathbf{Z}^*)\sqrt{\boldsymbol{\Phi}}\|_{\mathrm{Fro}}^2 \\ &= \min_{\mathbf{H}} \sum_{i=1}^{n} |\mathbf{A}^*\mathbf{h}_i - \mathbf{g}_i|^2,\end{aligned}$$

$\mathbf{h}_i$ and $\mathbf{g}_i$ denoting the columns of $\mathbf{H} = \mathbf{D}^*\sqrt{\boldsymbol{\Phi}}$ and $\mathbf{G} = \mathbf{Z}^*\sqrt{\boldsymbol{\Phi}}$. It is known that the elementary problems are solved by $\hat{\mathbf{h}}_i = (\mathbf{A}^*)^+\mathbf{g}_i$. This gives $\hat{\mathbf{H}}^* = \mathbf{G}^*\mathbf{A}^+$ and finally $\hat{\mathbf{D}} = \mathbf{Z}\mathbf{A}^+$, after inverting with $(\sqrt{\boldsymbol{\Phi}})^{-1}$. ∎

A.3. Mean shape. The mean shape of a dataset can be defined as the Fréchet mean of the points $[\mathbf{z}_k]$ scattered on Kendall's manifold [13], with respect to one of the three distances d_F, d_P, or ρ. It is the unique global minimizer $\mathbf{z}_{\mathrm{mean}}$ of

$$\sum_{k=1}^{K} \mathrm{dist}([\mathbf{z}_{\mathrm{mean}}],[\mathbf{z}_k])^2,$$

when it exists. The Fréchet mean with respect to the full distance $\mathrm{dist} = d_F$ (see Figure 13) can be found as the shape of the eigenvector associated to the greatest eigenvalue of the operator [13, 47]

$$\sum_{k} \mathrm{P}_{\mathbb{C}\mathbf{z}_k},$$

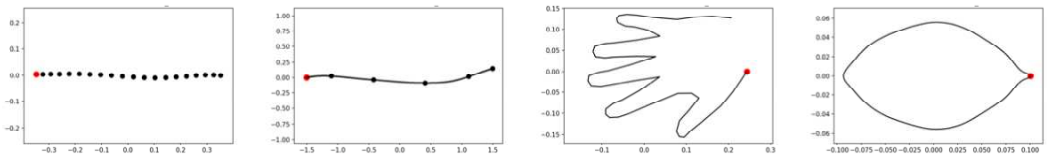


Figure 13. *Fréchet mean shape of Datasets 1, 2, 3, and 4 with respect to $\mathrm{dist} = d_F$. We display one representative preshape, up to rotations.*

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

where $\mathbf{z}_k \in \mathcal{S}$ are preshapes and where $\mathrm{P}_{\mathbb{C}\mathbf{z}_k} = \mathbf{z}_k\mathbf{z}_k^*\boldsymbol{\Phi}$ is the orthogonal projector onto the complex vector line generated by $\mathbf{z}_k$ relatively to the Hermitian product $\boldsymbol{\Phi}$. To prove it, let us consider the problem in the preshape sphere. We obtain

$$\arg\min_{\mathbf{z}\in\mathcal{S}} \sum |\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{z}_k}\mathbf{z}|^2_{\boldsymbol{\Phi}} = \arg\max_{\mathbf{z}\in\mathcal{S}} \sum |\mathrm{P}_{\mathbb{C}\mathbf{z}_k}\mathbf{z}|^2_{\boldsymbol{\Phi}} = \arg\max_{\mathbf{z}\in\mathcal{S}} \mathbf{z}^*\boldsymbol{\Phi}\sum_k \mathrm{P}_{\mathbb{C}\mathbf{z}_k}\mathbf{z}, \tag{A.4}$$

where we used that $|\mathbf{z} - \mathrm{P}_{\mathbb{C}\mathbf{z}_k}\mathbf{z}|^2_{\boldsymbol{\Phi}} = 1 - |\mathrm{P}_{\mathbb{C}\mathbf{z}_k}\mathbf{z}|^2_{\boldsymbol{\Phi}}$. A similar proof for landmarks ($\boldsymbol{\Phi} = \mathrm{Id}$) can be found in [13, p. 178].

Link with the mean-shape curve of [44]. In this article related to ours, a *mean-shape curve* of a family of curves $\{r_1, \dots, r_K\}$ in $H := \mathbb{L}^2([0,1], \mathbb{R}^2)$ endowed with the usual norm is defined as an optimal curve

$$r_{\mathrm{mean}} \in \arg\min_{\substack{|r|_H=1 \\ \bar{r}=0}} \sum_{k=1}^{K} |r - \mathrm{P}_k r|^2_H = \arg\max_{\substack{|r|_H=1 \\ \bar{r}=0}} \sum_{k=1}^{K} |\mathrm{P}_k r|^2_H, \tag{A.5}$$

where P_k is the *similarity projector*, i.e., the orthogonal projection onto the subspace S_{r_k} of dimension 4 associated to r_k containing all the images up to similitude transforms of r_k. It is written

$$S_r = \left\{ \lambda \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} r + \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \mid \lambda \in \mathbb{R}, \theta \in [0, 2\pi), \alpha, \beta \in \mathbb{R} \right\} \tag{A.6}$$

$$= \mathbb{R}\left\{ \begin{pmatrix} r^x \\ r^y \end{pmatrix}, \begin{pmatrix} -r^y \\ r^x \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}. \tag{A.7}$$

It can be shown that r_{mean} belongs to the eigenspace associated to the *second* greatest eigenvalue of $\sum_k \mathrm{P}_k$ as

$$\sum_{k=1}^{K} \mathrm{P}_k r_{\mathrm{mean}} = \lambda_2 r_{\mathrm{mean}}, \qquad |r_{\mathrm{mean}}| = 1, \tag{A.8}$$

and, as a consequence, the mean-shape curve of interpolating curves is also an interpolating curve (as soon as constants are themselves interpolating curves), since $r_{\mathrm{mean}} \in \mathrm{Im}(\sum \mathrm{P}_k)$. Aligning the dataset to the mean-shape curve then consists in taking

$$\widetilde{r_k} := \mathrm{P}_k r_{\mathrm{mean}}.$$

In fact, in the complex setting, if $r_1, \dots, r_K \in \mathbb{L}^2([0,1], \mathbb{C})$ are centered and normalized, the orbit of the mean-shape interpolating curve $[r_{\mathrm{mean}}]$ up to rotations *is then exactly the Fréchet mean* of the orbits $[r_1], \dots, [r_K]$ in the curve counterpart of the shape space with respect to the full distance d_F. In other words, if $\mathbf{z}_1, \dots, \mathbf{z}_K$ are the control vectors of $r_1, \dots, r_K$, then the control vector $\mathbf{z}_{\mathrm{mean}} \in \mathcal{S}$ of r_{mean} is one representative, up to rotations, of the Fréchet mean of $[\mathbf{z}_1], \dots, [\mathbf{z}_K]$. Also, the notion of alignment proposed in [44] coincides with that involved in the full distance (see (3.7)), if working with preshapes.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

A.4. B-spline curves. We find it is worthwhile to present our setting for B-splines, which are popular tools to construct interpolating curves [53, 5]. We provide explicit expressions for the basis functions ϕ_n. Spline curves (interpolating curves, more generally) motivate the introduction of a Hermitian product $\mathbf{\Phi}$ and a notion of shift configuration $\mathbf{u}$, for which projecting on the subspace orthogonal to $\mathbf{u}$ is akin to centering the spline curve. Readers can rely on this example to adapt our framework to other representations, such as Hermite-spline curves.

B-spline curves have parameters called *control points* that are regularly spaced along a continuous parameter t. A B-spline with M control points holds $N = M$ degrees of freedom, as in

$$\forall t \in [0,1], \quad r(t) = \sum_{n=0}^{M-1} \mathbf{z}[n]\phi_n(t), \tag{A.9}$$

where the basis functions ϕ_n can be obtained, for instance, from the cubic B-spline generator function

$$\beta^3(t) = \begin{cases} |t|^3/2 - t^2 + 2/3 & \text{for } 0 \le |t| \le 1, \\ (2-|t|)^3/6 & \text{for } 1 \le |t| \le 2, \\ 0 & \text{otherwise.} \end{cases} \tag{A.10}$$

Closed cubic B-splines curves are then obtained with

$$\text{for } n = 0, \forall t \in [0,1], \qquad \phi_0(t) = \beta^3(Mt) + \beta^3(Mt - M), \tag{A.11}$$

$$\text{for } n = 1, \forall t \in [0,1], \qquad \phi_1(t) = \beta^3(Mt - 1) + \beta^3(Mt - 1 - M), \tag{A.12}$$

$$\forall n = 2, \dots, M-2, \forall t \in [0,1], \qquad \phi_n(t) = \beta^3(Mt - n), \tag{A.13}$$

$$\text{for } n = M-1, \forall t \in [0,1], \qquad \phi_{M-1}(t) = \beta^3(Mt + 1) + \beta^3(Mt + 1 - M). \tag{A.14}$$

One can check that $\mathbb{1}$ is a cubic B-spline and that it corresponds to the shift configuration $\mathbf{u} = (1, \dots, 1)$. We know that the temporal mean of the spline curve, defined after Definition 2.4, coincides with the product $\frac{\mathbf{u}^*\mathbf{\Phi}\mathbf{z}}{|\mathbf{u}|^2_{\mathbf{\Phi}}} = \mathbf{u}^*\mathbf{\Phi}\mathbf{z} = \overline{\mathbf{z}^*\mathbf{\Phi}\mathbf{u}}$. In the case of closed B-splines, it is simply the usual arithmetic mean of the control points

$$\bar{r} = \frac{1}{M} \sum_{m=0}^{M-1} \mathbf{z}[m], \tag{A.15}$$

which is subtracted to $\mathbf{z}$ when centering the configuration. Please note that we have more sophisticated expressions for closed and open Hermite-spline curves, which we detail in a future work.

A.5. Riemannian structure of the shape space. We provide a concise description of the Riemannian structure of the shape space Σ (our method does not use it in a crucial way). Using the same conventions as in Appendix A.1, we use bold and italics to denote the complex and real version of a vector by $\mathbf{z} \in \mathbb{C}^N$ and $z \in \mathbb{R}^{2N}$, respectively. The real inner product $(z, w) \mapsto \Re(\mathbf{z}^*\mathbf{\Phi}\mathbf{w})$ canonically associated to the Hermitian inner product $\mathbf{\Phi}$ is

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

denoted as $\langle \cdot, \cdot \rangle_{\mathbf{\Phi}}$. The shape space based on $(\mathbb{R}^{2N}, \langle \cdot, \cdot \rangle_{\mathbf{\Phi}})$ has a Riemannian structure that is similar to the usual complex projective space $\mathbb{CP}^{N-2}$. The latter is the shape space based on $(\mathbb{R}^{2N}, \langle \cdot, \cdot \rangle_{\text{can}})$, defined for landmarks and the standard inner product [29, 13, 47, 20, 19].

The Riemannian structure of Σ is inherited from that of $\mathcal{S}$, a sphere in finite dimension whose structure is well understood. Being a smooth submanifold of $(\mathbb{R}^{2N}, \langle \cdot, \cdot \rangle_{\mathbf{\Phi}})$, $\mathcal{S}$ is endowed with the induced Riemannian metric. The tangent space to $\mathcal{S}$ at z is

$$T_z(\mathcal{S}) = \{v \in \mathbb{R}^{2N} \mid \langle z, v \rangle_{\mathbf{\Phi}} = 0\}, \tag{A.16}$$

which is equipped with the Riemannian inner product $\forall v_1, v_2 \in T_z\mathcal{S}, \langle v_1, v_2 \rangle_z := \langle v_1, v_2 \rangle_{\mathbf{\Phi}}$.

The space space Σ is the quotient of $\mathcal{S}$ by the group of planar rotations $U(1)$. The action of $U(1)$ on the Riemannian manifold $\mathcal{S}$ is smooth, free, and proper on $\mathcal{S}$, with $U(1)$ being a Lie group acting by isometries, meaning that

$$\forall \theta \in U(1), \langle v_1, v_2 \rangle_z = \langle \mathrm{R}_\theta \odot v_1, \mathrm{R}_\theta \odot v_2 \rangle_{\mathrm{R}_\theta \odot z}. \tag{A.17}$$

The resulting quotient Σ consequently inherits a (unique) Riemannian structure on $\Sigma = \mathcal{S}/U(1)$ such that $\mathbf{\Pi} : \begin{pmatrix} \mathcal{S} \to \Sigma \\ z \mapsto [\mathbf{z}] \end{pmatrix}$ is a Riemannian submersion [20, 19]. This structure can be described as follows [20, 19, 47]. At each point z of $\mathcal{S}$, the tangent space $T_z(\mathcal{S})$ can be split into two subspaces orthogonal to each other with respect to $\langle \cdot, \cdot \rangle_{\mathbf{\Phi}}$ as

$$T_z(\mathcal{S}) = T_z([z]) \overset{\perp}{\oplus} H_z(\mathcal{S}), \tag{A.18}$$

where the *vertical subspace* $T_z([z]) = \mathrm{Ker}(d\pi_z) = \mathbb{R}(R_{\pi/2} \odot z)$ is the space tangent at z to the orbit $[z]$, while the orthogonal complement is the *horizontal subspace*

$$H_z(\mathcal{S}) = \{v \in \mathbb{R}^{2N} \mid \langle z, v \rangle_{\mathbf{\Phi}} = 0 \text{ and } \langle z, R_{\pi/2} \odot v \rangle_{\mathbf{\Phi}} = 0\} \simeq \{\mathbf{v} \in \mathbb{C}^N \mid \mathbf{z}^* \mathbf{\Phi} \mathbf{v} = 0\}. \tag{A.19}$$

The last identification uses the properties of the Hermitian inner product seen in Appendix A.1. By definition, the tangent space $T_{[z]}(\Sigma)$ is then identified to the horizontal subspace $H_z(\mathcal{S})$ through the isometry $d\pi_z : v \in H_z(\mathcal{S}) \mapsto [v] \in T_{[z]}(\Sigma)$. The Riemannian inner product on $T_{[z]}(\Sigma)$ is then obtained as

$$\langle\langle [v_1], [v_2] \rangle\rangle_{[z]} := \langle v_1, v_2 \rangle_z, \tag{A.20}$$

which does not depend on the particular choice of the representative preshape z.

Geodesics and Riemannian distance ρ (Definition 3.8) can then be defined on Σ. The Riemannian distance $\rho_{\mathcal{S}}$ on $\mathcal{S}$ is equal to the length of the shortest path joining z to w,

$$\rho_{\mathcal{S}}(z, w) = \arccos \langle z, w \rangle_{\mathbf{\Phi}}. \tag{A.21}$$

The corresponding shortest geodesic arc $\{\boldsymbol{\alpha}(t)\}$ that joins z to w is

$$\forall t \in [0, 1], \quad \boldsymbol{\alpha}(t) = \frac{1}{\sin r}(\sin((1-t)r)z + \sin(t\ r)w), \quad \text{where } r = \rho_{\mathcal{S}}(z, w). \tag{A.22}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

The shortest geodesic path between the shapes $[\mathbf{z}]$ and $[\mathbf{w}]$ is the quotient path $\{[\tilde{\boldsymbol{\alpha}}(t)]\}$, where $\tilde{\boldsymbol{\alpha}}$ is the geodesic in $\mathcal{S}$ that joins $\tilde{z} \in [\mathbf{z}]$ to w, where $\tilde{z}$ is chosen so that the corresponding (complex) preshape $\tilde{\mathbf{z}}$ is optimally rotated along $\mathbf{w}$. The Riemannian distance in the shape space is then the length of $\tilde{\boldsymbol{\alpha}}(t)$, computed as $\arccos\langle \tilde{z}, w\rangle_{\boldsymbol{\Phi}} = \arccos(\Re(\tilde{\mathbf{z}}^*\boldsymbol{\Phi}\mathbf{w}))$. It also corresponds to the geodesic distance between $\tilde{z}$ and the set $[\mathbf{w}]$. From an argument in the proof of Proposition 3.6, we know that, since $\tilde{\mathbf{z}}$ is optimally rotated, it is also equal to $\arccos|\tilde{\mathbf{z}}^*\boldsymbol{\Phi}\mathbf{w}| = \arccos|\mathbf{z}^*\boldsymbol{\Phi}\mathbf{w}|$, thus resulting in the equality of Definition 3.8.

Acknowledgments. A.S. is especially thankful to Jean Feydy for drawing her attention to reference [36]. She is grateful for the support from École Normale Supérieure and École Polytechnique Fédérale de Lausanne.

REFERENCES

[1] R. Anirudh, P. Turaga, J. Su, and A. Srivastava, *Elastic functional coding of human actions: From vector-fields to latent variables*, in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3147–3155, https://doi.org/10.1109/CVPR.2015.7298934.

[2] A. Ben Tanfous, H. Drira, and B. Ben Amor, *Coding Kendall's shape trajectories for 3D action recognition*, in Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, https://hal.archives-ouvertes.fr/hal-01713295.

[3] C. V. Bennett and A. Goswami, *Morphometric analysis of cranial shape in fossil and recent euprimates*, Anatomy Res. Internat., 2012 (2012), https://doi.org/10.1155/2012/478903.

[4] F. Bookstein, *Morphometric Tools for Landmark Data: Geometry and Biology*, Geometry and Biology, Cambridge University Press, New York, 1997, https://books.google.fr/books?id=amwT1ddIDwAC.

[5] P. Brigger, J. Hoeg, and M. Unser, *B-spline snakes: A flexible tool for parametric contour detection*, IEEE Trans. Image Process., 9 (2000), pp. 1484–1496, https://doi.org/10.1109/83.862624.

[6] S. D. Buckingham and D. B. Sattelle, *Strategies for automated analysis of C. Elegans locomotion*, Invertebrate Neurosci., 8 (2008), 121, https://doi.org/10.1007/s10158-008-0077-3.

[7] E. J. Candés, J. K. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), pp. 489–509, http://dblp.uni-trier.de/db/journals/tit/tit52.html#CandesRT06.

[8] R. Caseiro, J. F. Henriques, P. Martins, and J. Batista, *Semi-intrinsic mean shift on Riemannian manifolds*, in Computer Vision—ECCV 2012, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, eds., Springer, Berlin, 2012, pp. 342–355.

[9] S. F. Cotter, R. Adler, R. D. Rao, and K. Kreutz-Delgado, *Forward sequential algorithms for best basis selection*, IEEE Trans. Vision Image Signal Process., 146 (1999), pp. 235–244, https://doi.org/10.1049/ip-vis:19990445.

[10] L. da Fontoura Costa and R. Cesar, *Shape Analysis and Classification: Theory and Practice*, Image Processing Series, CRC Press, Boca Raton, FL, 2010, https://books.google.fr/books?id=x_wiWedtc0cC.

[11] D. L. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306, http://dblp.uni-trier.de/db/journals/tit/tit52.html#Donoho06.

[12] A. G. Drake, M. Coquerelle, and G. Colombeau, *3D morphometric analysis of fossil canid skulls contradicts the suggested domestication of dogs during the late paleolithic*, Scientific Reports, 5 (2015), https://doi.org/10.1038/srep08299.

[13] I. Dryden and K. Mardia, *Statistical Shape Analysis: With Applications in R*, Wiley Ser. Probab. Statist., Wiley, New York, 2016, https://books.google.fr/books?id=CUimDAAAQBAJ.

[14] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, New York, 2010.

[15] M. Elad and M. Aharon, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. Image Process., 15 (2006), pp. 3736–3745, http://dblp.uni-trier.de/db/journals/tip/tip15.html#EladA06.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

[16] K. Engan, S. O. Aase, and J. H. Husøy, *Method of optimal directions for frame design*, in Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP99, 1999, pp. 2443–2446.

[17] H. E. Çetingül and R. Vidal, *Intrinsic mean shift for clustering on Stiefel and Grassmann manifolds*, in Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1896–1902.

[18] J. Fageot, E. Bostan, and M. Unser, *Wavelet statistics of sparse and self-similar images*, SIAM J. Imaging Sci., 8 (2015), pp. 25, 2951–2975, https://doi.org/10.1137/151003015.

[19] J. Gallier and J. Quaintance, *Notes on Differential Geometry and Lie Groups*, 2011, https://www.seas.upenn.edu/~jean/diffgeom-spr-I.pdf.

[20] S. Gallot, D. Hulin, and J. Lafontaine, *Riemannian Geometry*, Universitext, Springer, Berlin, 2004, https://books.google.fr/books?id=6F4Umpws_gUC.

[21] R. Gonzalo, *Segmentation and Tracking in High-Throughput Bioimaging*, EPFL thesis 5657, Swiss Federal Institute of Technology Lausanne, 2013.

[22] K. Guo, P. Ishwar, and J. Konrad, *Action recognition using sparse representation on covariance manifolds of optical flow*, in Proceedings of the 7th IEEE International Conference on Advanced Video and Signal Based Surveillance, 2010, pp. 188–195.

[23] M. T. Harandi, R. I. Hartley, B. C. Lovell, and C. Sanderson, *Sparse coding on symmetric positive definite manifolds using Bregman divergences*, IEEE Trans. Neural Networks Learning Systems, 27 (2016), pp. 1294–1306.

[24] M. T. Harandi, R. I. Hartley, C. Shen, B. C. Lovell, and C. Sanderson, *Extrinsic methods for coding and dictionary learning on Grassmann manifolds*, Int. J. Comput. Vis., 114 (2015), pp. 113–136.

[25] M. T. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell, *Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach*, in Computer Vision—ECCV 2012, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, eds., Springer, Berlin, pp. 216–229.

[26] M. T. Harandi, C. Sanderson, C. Shen, and B. C. Lovell, *Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution*, in Proceedings of the 2013 IEEE International Conference on Computer Vision, 2013, pp. 3120–3127.

[27] Z. Huang, C. Wan, T. Probst, and L. V. Gool, *Deep learning on Lie groups for skeleton-based action recognition*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1243–1252.

[28] S. Jayasumana, M. Salzmann, H. Li, and M. Harandi, *A framework for shape analysis via Hilbert space embedding*, in Proceedings of the 2013 IEEE International Conference on Computer Vision, 2013, pp. 1249–1256.

[29] D. G. Kendall, *The diffusion of shape*, Adv. Appl. Probab., 9 (1977), pp. 428–430, http://www.jstor.org/stable/1426091.

[30] D. G. Kendall, *Shape manifolds, Procrustean metrics, and complex projective spaces*, Bull. London Math. Soc., 16 (1984), pp. 81–121, https://doi.org/10.1112/blms/16.2.81.

[31] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski, *Dictionary learning algorithms for sparse representation*, Neural Comput., 15 (2003), pp. 349–396, https://doi.org/10.1162/089976603762552951.

[32] L. J. Latecki, R. Lakämper, and U. Eckhardt, *Shape descriptors for non-rigid shapes with a single closed contour*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2000, pp. 424–429.

[33] H. Lee, A. Battle, R. Raina, and A. Y. Ng, *Efficient sparse coding algorithms*, in Proceedings of the 19th International Conference on Neural Information Processing Systems, 2006, pp. 801–808, http://dl.acm.org/citation.cfm?id=2976456.2976557.

[34] P. Li, Q. Wang, W. Zuo, and L. Zhang, *Log-Euclidean kernels for sparse representation and dictionary learning*, in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1601–1608.

[35] S. Loncaric, *A survey of shape analysis techniques*, Pattern Recognition, 31 (1998), pp. 983–1001, https://doi.org/10.1016/S0031-2023(97)00122-2.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[36] J. Mairal, F. Bach, and J. Ponce, *Sparse modeling for image and vision processing*, Found. Trends Comput. Graphics Vision., 8 (2014), pp. 85–283, https://doi.org/10.1561/0600000058.
[37] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, *Online dictionary learning for sparse coding*, in Proceedings of the 26th Annual International Conference on Machine Learning, ACM, New York, 2009, pp. 689–696, https://doi.org/10.1145/1553374.1553463.
[38] J. Mairal, F. R. Bach, and J. Ponce, *Task-driven dictionary learning*, IEEE Trans. Pattern Anal. Machine Intelligence, 34 (2012), pp. 791–804, http://dblp.uni-trier.de/db/journals/pami/pami34.html#MairalBP12.
[39] D. Migliozzi, M. Cornaglia, L. Mouchiroud, V. Uhlmann, M. A. Unser, J. Auwerx, and M. A. M. Gijs, *Multimodal imaging and high-throughput image-processing for drug screening on living organisms on-chip*, J. Biomed. Optics, 24 (2018), pp. 1–9, https://doi.org/10.1117/1.JBO.24.2.021205.
[40] B. Olshausen and D. Field, *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*, Nature, 381 (1996), pp. 607–609.
[41] M. Ovsjanikov, A. M. Bronstein, M. M. Bronstein, and L. J. Guibas, *Shape Google: A computer vision approach to isometry invariant shape retrieval*, in Proceedings of the IEEE 12th International Conference on Computer Vision Workshops, 2009, pp. 320–327.
[42] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, *Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition*, in Proceedings of the 27th Annual Asilomar Conference on Signals, Systems, and Computers, 1993, pp. 40–44.
[43] G. Peyré, *The numerical tours of signal processing*, Comput. Sci. Eng., 13 (2011), pp. 94–97, https://doi.org/10.1109/MCSE.2011.71.
[44] D. Schmitter and M. Unser, *Landmark-based shape encoding and sparse-dictionary learning in the continuous domain*, IEEE Trans. Image Process., 27 (2018), pp. 365–378, https://doi.org/10.1109/TIP.2017.2762582.
[45] I. Schoenberg, *Cardinal interpolation and spline functions. V. The B-splines for cardinal Hermite interpolation*, Linear Algebra Appl., 7 (1973), pp. 1–42, https://doi.org/10.1016/0024-3795(73)90034-7.
[46] C. Small, *The Statistical Theory of Shape*, Springer Ser. Statist., Springer, New York, 2012, https://books.google.fr/books?id=C43bBwAAQBAJ.
[47] A. Srivastava and E. Klassen, *Functional and Shape Data Analysis*, Springer Ser. Statist., Springer, New York, 2016.
[48] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn, *Shape analysis of elastic curves in Euclidean spaces*, IEEE Trans. Pattern Anal. Machine Intelligence, 33 (2011), pp. 1415–1428, https://doi.org/10.1109/TPAMI.2010.184.
[49] M. B. Stegmann and D. D. Gomez, *A Brief Introduction to Statistical Shape Analysis*, Technical University of Denmark, 2002, http://www2.imm.dtu.dk/pubdb/p.php?403.
[50] G. J. Stephens, B. Johnson-Kerner, W. Bialek, and W. S. Ryu, *Dimensionality and dynamics in the behavior of C. elegans*, PLOS Comput. Biol., 4 (2008), pp. 1–10, https://doi.org/10.1371/journal.pcbi.1000028.
[51] V. Uhlmann, *Landmark Active Contours for Bioimage Analysis: A Tale of Points and Curves*, EPFL thesis 7951, Swiss Federal Institute of Technology Lausanne, 2017.
[52] V. Uhlmann, J. Fageot, and M. Unser, *Hermite snakes with control of tangents*, IEEE Trans. Image Process., 25 (2016), pp. 2803–2816, https://doi.org/10.1109/TIP.2016.2551363.
[53] M. Unser, A. Aldroubi, and M. Eden, *B-spline signal processing. I. Theory*, IEEE Trans. Signal Process., 41 (1993), pp. 821–833, https://doi.org/10.1109/78.193220.
[54] M. Unser and P. Tafti, *An Introduction to Sparse Stochastic Processes*, Cambridge University Press, New York, 2014, https://books.google.fr/books?id=DKYZBAAAQBAJ.
[55] R. Vemulapalli, F. Arrate, and R. Chellappa, *Human action recognition by representing 3D skeletons as points in a Lie group*, in Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 588–595.
[56] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, *Robust face recognition via sparse representation*, IEEE Trans. Pattern Anal. Machine Intelligence, 31 (2009), pp. 210–227, http://dblp.uni-trier.de/db/journals/pami/pami31.html#WrightYGSM09.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[57] Y. Xie, J. Ho, and B. Vemuri, *On a nonlinear generalization of sparse coding and dictionary learning*, in Proceedings of the International Conference on Machine Learning, 2013, pp. 1480–1488.
[58] E. Yemini, T. Jucikas, L. J. Grundy, A. E. X. Brown, and W. R. Schafer, *A database of C. elegans behavioral phenotypes*, Nature Methods, 10 (2013), pp. 877–879.
[59] C. Yuan, W. Hu, X. Li, S. Maybank, and G. Luo, *Human action recognition under log-Euclidean Riemannian metric*, in Computer Vision—ACCV 2009, H. Zha, R.-i. Taniguchi, and S. Maybank, eds., Springer, Berlin, 2010, pp. 343–353.
[60] D. Zhang and G. Lu, *Review of shape representation and description techniques*, Pattern Recognition, 37 (2004), pp. 1–19, https://doi.org/10.1016/j.patcog.2003.07.008.

Downloaded 05/05/20 to 128.179.255.51. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.