Disentangling by Subspace Diffusion

David Pfau, Irina Higgins, Aleksandar Botev, Sébastien Racanière
DeepMind
London, UK
{pfau, irinah, botev, sracaniere}@google.com

Abstract

We present a novel nonparametric algorithm for symmetry-based disentangling of data manifolds, the Geometric Manifold Component Estimator (GEOMANCER). GEOMANCER provides a partial answer to the question posed by Higgins et al. (2018): is it possible to learn how to factorize a Lie group solely from observations of the orbit of an object it acts on? We show that fully unsupervised factorization of a data manifold is possible *if* the true metric of the manifold is known and each factor manifold has nontrivial holonomy – for example, rotation in 3D. Our algorithm works by estimating the subspaces that are invariant under random walk diffusion, giving an approximation to the de Rham decomposition from differential geometry. We demonstrate the efficacy of GEOMANCER on several complex synthetic manifolds. Our work reduces the question of whether unsupervised disentangling is possible to the question of whether unsupervised metric learning is possible, providing a unifying insight into the geometric nature of representation learning.

1 Introduction

The ability to disentangle the different latent factors of variation in the world has been hypothesized as a critical ingredient in representation learning [1], and much recent research has sought a fully unsupervised way to learn disentangled representations from data [2–18]. Because the community has not settled on a definition of “disentangling", much of this work relies on heuristics and qualitative criteria to judge performance. For instance, datasets are often constructed by varying interpretable factors like object position, rotation and color and methods are judged by how well they recover these predefined factors [19, 20]. One commonly used definition is that a representation is disentangled if the data distribution can be modeled as a nonlinear transformation of a product of independent probability distributions [20]. This leads to a pessimistic result, that the different latent factors are not identifiable without side information or further assumptions.

To escape this pessimistic result, we can turn to a different, symmetry-based definition of disentangling [21], rooted in the Lie group model of visual perception [22–26]. Instead of a product of *distributions*, the symmetry-based approach to disentangling considers a representation disentangled if it matches the product of *groups* that define the symmetries in the world. If the actions that define the possible transformations in the world form a group $G = G_1 \times G_2 \times ... \times G_m$, then a representation is disentangled under this definition if it decomposes in such a way that the action of a single subgroup leaves all factors of the representation invariant except for one (See Supp. Mat., Sec. B for a formal definition).

The symmetry-based definition is appealing as it resolves an apparent contradiction in the parallelogram model of analogical reasoning [27]. For concepts represented as vectors $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ in a flat space, the analogy $\mathbf{a} : \mathbf{b} :: \mathbf{c} : \mathbf{d}$ can be completed by $\mathbf{d} = \mathbf{b} + \mathbf{c} - \mathbf{a}$, as $\mathbf{d}$ completes the fourth corner of a parallelogram. This model has worked well in practice when applied to embeddings learned by deep neural networks for words [28, 29] and natural images [30, 31], and often matches human judgments [32]. But for many natural transformations, including 3D rotation, it is not possible to form a representation in a flat vector space such that vector addition corresponds to composition of transformations. Instead, obser-

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

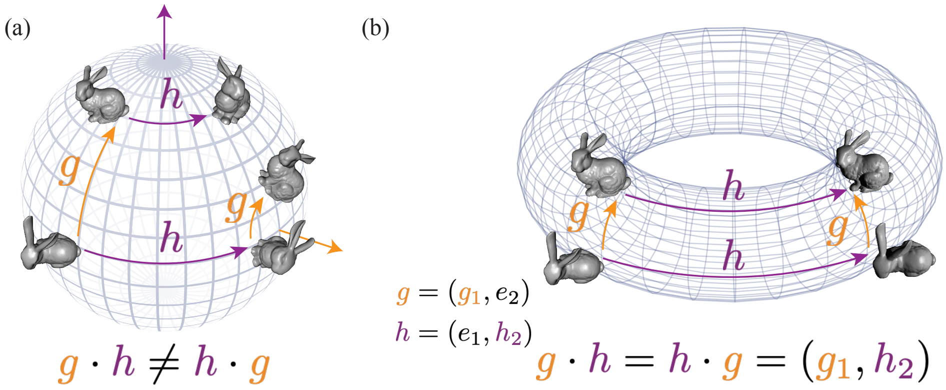

(a)
h
g
g
h
$g \cdot h \neq h \cdot g$
(b)
h
g
g
h
$g = (g_1, e_2)$
$h = (e_1, h_2)$
$g \cdot h = h \cdot g = (g_1, h_2)$

Figure 1: Two views of the orbit of an object under the action of a group, in this case images of the Stanford Bunny under changes in pose and illumination (a) Transformations from the same subgroup, in this case 3D rotations, do not in general commute, and the analogy is ambiguous. (b) Transformations from disentangled subgroups. The transformations commute and the analogy is unambiguous.

vations from such transformations can be naturally represented as coming from the orbit of a group [26]. In this setting, the parallelogram model breaks down – if a,b,c are representations, and g,$h \in G$ are the group elements such that $g \cdot a = b$ and $h \cdot a = c$, then the completion of the analogy $a:b::c:d$ is ambiguous, as $g \cdot h \cdot a \neq h \cdot g \cdot a$ for noncommutative operations (Figure 1). State-of-the-art generative models for single image classes typically elide this by restricting the dataset to a limited patch of the manifold, like forward-facing faces [33] or limited ranges of change in elevation [6, 34–38]. This ambiguity, however, is resolved if G is a product of subgroups. So long as g and h leave all factors invariant except for one, and each varies a different factor, then g and h do commute, and the analogy can be uniquely completed.

While this definition of disentangling is appealing, it does not provide an obvious recipe for how to learn the appropriate factorization of the action on the world state. Some group invariances and equivariances can be built into neural network architectures [39–46], and it has been shown how commutative group representations can be learned [47]. Methods have been proposed to learn symmetry-based disentangled representations when conditioned on interactions [16, 17], or to learn dictionaries of Lie group operators from neighboring pairs of data [48, 49], but a general algorithm to factorize noncommutative groups without any supervision remains elusive.

If we restrict our attention to Lie groups – groups that are also manifolds, like rotations – we could use the properties of infinitesimal transformations as a learning signal. Essentially, we would like to use failures of the parallelogram model *as a learning signal itself*. Those directions that lie on disentangled submanifolds will behave like vectors in a flat space when one is moved in the direction of the other, hence complying with the parallelogram model, while directions within each submanifold may be mixed together in arbitrary ways. Computing all of the directions that remain invariant provides a disentangled factorization of the manifold. These intuitions can be made precise by the de Rham decomposition theorem [50], a foundational theorem in holonomy theory, a branch of differential geometry.

Here we present an algorithm that turns these ideas into a practical method for disentangling, the Geometric Manifold Component Estimator (GEOMANCER). GEOMANCER differs from other disentangling algorithms in that it does not learn a nonlinear embedding of the data. Instead, it can either be applied directly to the data so long as the local metric information is known, or it can be applied as a post-processing step to learned embeddings. GEOMANCER is a nonparametric algorithm which learns a set of subspaces to assign to each point in the dataset, where each subspace is the tangent space of one disentangled submanifold. This means that GEOMANCER can be used to disentangle manifolds for which there may not be a global axis-aligned coordinate system. GEOMANCER is also able to discover the correct number of dimensions in each submanifold without prior knowledge. Our algorithm is particularly well suited for dealing with transformations which have nontrivial holonomy, such as 3D rotations. In contrast, most previous work [2–16, 18] has focused on transformations with trivial holonomy, such as translation in 2D.

GEOMANCER builds on classic work on nonparametric manifold learning, especially Laplacian Eigenmaps [51], Diffusion Maps [52] and extensions like Orientable and Vector Diffusion Maps [53–55], generalizing the idea of finding modes of a random walk diffusion operator on manifolds from scalars and vectors to matrices. These classic methods are primarily focused on learning a low-dimensional embedding of data. While GeoManCEr uses and extends many of these same mathematical techniques, its primary goal is to learn a factorization of the data manifold, which makes it an interesting new application of spectral graph theory.

GeoManCEr should not be confused with methods like Horizontal Diffusion Maps, which are for "synchronization" problems in manifold learning [56, 57]. These methods also use scalar and (co)vector diffusion, but not subspace diffusion. Much like other manifold learning problems, they are concerned with finding an embedding of points on a manifold, not a factorization. What distinguishes synchronization problems from the rest of manifold learning is that they are concerned with finding embeddings such that operations that relate pairs of data can be composed together in a cycle-consistent manner. GeoManCEr instead exploits the "cycle-inconsistency" of certain operations to distinguish entangled and disentangled directions around each point.

It is important to distinguish GeoManCEr from other manifold learning methods that go beyond learning a single undifferentiated manifold, especially Robust Multiple Manifold Structure Learning (RMMSL) [58]. RMMSL learns a *mixture* of manifolds which are all embedded in the same space. GeoManCEr by contrast, learns a single manifold which is itself a *product* of many submanifolds, where each submanifold exists in its own space. We next present a rapid overview of the relevant theory, followed by a detailed description of GeoManCEr, and finally show results on complex data manifolds.

2 Theory

We assume some basic familiarity with the fundamentals of Riemannian geometry and parallel transport, though we strongly encourage reading the review in Supp. Mat., Sec. A. For a more thorough treatment, we recommend the textbooks by Do Carmo [59] and Kobayashi and Nomizu [60]. Let x denote points on the k-dimensional Riemannian manifold $\mathcal{M}$ with metric $\langle\cdot,\cdot\rangle_{\mathcal{M}}$, possibly embedded in $\mathbb{R}^n$. We denote paths by $\gamma:\mathbb{R}\rightarrow\mathcal{M}$, tangent spaces of velocity vectors by $T_x\mathcal{M}$, cotangent spaces of gradient operators by $T_x^*\mathcal{M}$, and vectors in $T_x\mathcal{M}$ by $\mathbf{v}$, $\mathbf{w}$, etc. As $\mathcal{M}$ is a Riemannian manifold, it has a unique Levi-Civita connection, which allows us to define a covariant derivative and parallel transport, which gives a formal definition of two vectors in nearby tangent spaces being parallel.

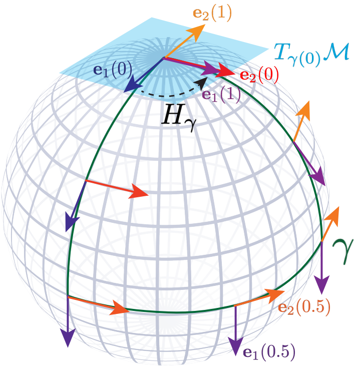


Figure 2: Holonomy on the sphere. The linear transform H_γ captures the amount vectors in $T_{\gamma(0)}\mathcal{M}$ are rotated by parallel transport around the path γ.

Holonomy and the de Rham decomposition The basic tools of differential geometry, especially parallel transport, can be used to infer the product structure of the manifold through the *holonomy group*. Consider a loop $\gamma:[0,1]\rightarrow\mathcal{M}$, $\gamma(0)=x$, $\gamma(1)=x$. Given an orthonormal basis $\{\mathbf{e}_i\}\subset T_x\mathcal{M}$ of the tangent space at x, we can consider the parallel transport of all basis vectors around the loop $\{\mathbf{e}_i(t)\}$, illustrated in Fig. 2. Then the vectors $\{\mathbf{e}_i(1)\}\subset T_x\mathcal{M}$ form the columns of a matrix H_γ that fully characterizes how any vector transforms when parallel transported around a loop. That is, for any parallel transport $\mathbf{v}(t)$ along γ, $\mathbf{v}(1)=H_\gamma\mathbf{v}(0)$. Note that if the affine connection preserves the metric , then $\{\mathbf{e}_i(t)\}$ will be an orthonormal basis for all t, so H_γ is in the orthonormal group $\mathrm{O}(k)$. Moreover if $\mathcal{M}$ is orientable, then the handedness of $\{\mathbf{e}_i(t)\}$ cannot change, so H_γ is in the special orthonormal group $\mathrm{SO}(k)$, that is, $\det(H_\gamma)=1$.

The linear transform H_γ is the *holonomy* of the loop γ. The space of all holonomies for all loops that start and end at x form the *holonomy group* $\mathrm{Hol}_x(\mathcal{M})$ at x. It can clearly be seen that this space is a group by considering different loops. The trivial loop $\gamma(t)=x$ has holonomy I, the identity. If γ has holonomy H_γ, then the loop $\gamma'(t)=\gamma(1-t)$ has holonomy H_γ^{-1}, so if H_γ is in the holonomy group, so is its inverse. And if γ_1 and γ_2 are loops, then the loop formed by first going around γ_1 followed by γ_2 has holonomy $H_{\gamma_2}H_{\gamma_1}$, so if two elements are in this group, then so is their product.

The structure of the holonomy group is extremely informative about the global structure of the manifold. If the manifold $\mathcal{M}$ is actually a product of submanifolds $\mathcal{M}_1\times\mathcal{M}_2\times\ldots\times\mathcal{M}_n$, with the corresponding product metric as its metric , then it is straightforward to show that the tangent space $T_x\mathcal{M}$ can be decomposed into orthogonal subspaces $T_x^{(1)}\mathcal{M},...,T_x^{(m)}\mathcal{M}$ such that the action of $\mathrm{Hol}_x(\mathcal{M})$ leaves each subspace invariant. That is, if $\mathbf{v}\in T_x^{(i)}\mathcal{M}$, then $H_\gamma\mathbf{v}\in T_x^{(i)}\mathcal{M}$ for all γ . These subspaces are each tangent to the respective submanifolds that make up $\mathcal{M}$. The more remarkable result is that the converse holds locally and, if the manifold is simply connected and geodesically complete, globally [50]. A manifold is simply connected if any closed loop can be continuously deformed into a single point, and it is geodesically complete if any geodesic can be followed indefinitely.

Theorem 1. de Rham Decomposition Theorem (de Rham, 1952), see also [60, Theorem 6.1]: Assume $\mathcal{M}$ is a simply connected and geodesically complete Riemannian manifold. If there exists a point $x\in\mathcal{M}$ and a proper subspace U that is invariant under the action of the holonomy group $\mathrm{Hol}_x(\mathcal{M})$, then $\mathcal{M}$ is a product Riemannian manifold $\mathcal{M}_1\times\mathcal{M}_2$ with $T_x\mathcal{M}_1=U$ and $T_x\mathcal{M}_2=U^\perp$. The tangent spaces to $\mathcal{M}_1$ and $\mathcal{M}_2$ at any other point y are obtained by parallel transporting U and $U^\perp$ respectively along any path from x to y.

The above theorem can be applied recursively, so that if the holonomy group leaves multiple pairwise orthogonal subspaces invariant, we can conclude that $\mathcal{M}$ is a product of multiple Riemannian manifolds. It seems quite remarkable that a property of the holonomy group in a single tangent space can tell us so much about the structure of the manifold. This is because the holonomy group itself integrates information over the entire manifold, so in a sense it is not really a local property at all.

This result is the main motivation for GEOMANCER – we aim to discover a decomposition of a data manifold by investigating its holonomy group. The holonomy group is a property of *all possible paths*, so it cannot be computed directly. Instead, we build a partial differential equation that inherits properties of the holonomy group and work with a numerically tractable approximation to this PDE.

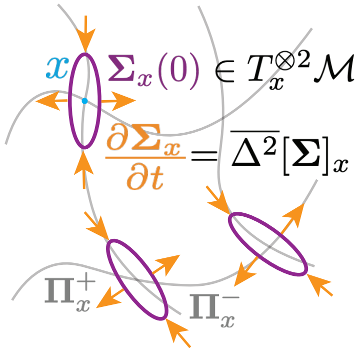

x
$\mathbf{\Sigma}_x(0)\in T_x^{\otimes 2}\mathcal{M}$
$\frac{\partial\mathbf{\Sigma}_x}{\partial t}=\overline{\Delta^2}[\mathbf{\Sigma}]_x$
$\mathbf{\Pi}_x^+$
$\mathbf{\Pi}_x^-$

Figure 3: Subspace diffusion. A field of symmetric semidefinite matrices $\mathbf{\Sigma}(t)$ evolves in time according to the differential equation $\dot{\mathbf{\Sigma}}=\overline{\Delta^2}[\mathbf{\Sigma}]$.

Subspace diffusion on manifolds While it is not feasible to compute properties of all loops on a manifold, the *average* properties of random walk diffusion on a manifold can be computed by studying the properties of the diffusion equation. Consider a particle undergoing a Brownian random walk on a manifold with diffusion rate τ. Then, given an initial probability density $p(x,0)$, the probability of finding the particle at x at time t evolves according to the diffusion equation:

$$\frac{\partial p(x,t)}{\partial t}=\tau\Delta^0[p](x,t) \qquad (1)$$

where Δ^0 is a linear operator called the *Laplace-Beltrami* operator [60, Note 14, Vol. 2], defined as the trace of the second covariant derivative $\Delta^0[f]=\mathrm{Tr}\nabla^2 f$. Even if the initial condition is a delta function, the change in probability is nonzero everywhere, so the Laplace-Beltrami operator encodes global information about the manifold, though it weights local information more heavily.

The Laplace-Beltrami operator, which acts on scalar functions, can be generalized to the *connection Laplacian* for rank-(p,q) tensor-valued functions. As the second covariant derivative of a rank-(p,q) tensor is a rank-$(p,q+2)$ tensor, we can take the trace over the last two dimensions to get the connection Laplacian. The connection Laplacian also has an intuitive interpretation in terms of random walks of vectors. Given a probability density $p(\mathbf{v},t)$ over $T\mathcal{M}$, the manifold of all tangent spaces on $\mathcal{M}$, the connection Laplacian on vectors Δ^1 gives the rate of change of the mean of the density at every point $\mu_x=\int_{T_x\mathcal{M}}p(\mathbf{v},t)\mathbf{v}d\mathbf{v}$, while the connection Laplacian on matrices Δ^2 gives the rate of change of the second moment $\mathbf{\Sigma}_x=\int_{T_x\mathcal{M}}p(\mathbf{v},t)\mathbf{v}\mathbf{v}^T d\mathbf{v}$ (Fig. 3).

Many of the properties of the holonomy group can be inferred from the second-order connection Laplacian Δ^2. In particular, for a product manifold, the eigenfunctions of Δ^2 contain information about the invariant subspaces of the holonomy group:

Theorem 2. Let $\mathcal{M}=\mathcal{M}_1\times...\times\mathcal{M}_m$ be a Riemannian product manifold, and let $T_x^{(1)}\mathcal{M},...,T_x^{(m)}\mathcal{M}$ denote orthogonal subspaces of $T_x\mathcal{M}$ that are tangent to each submanifold. Then the tensor fields

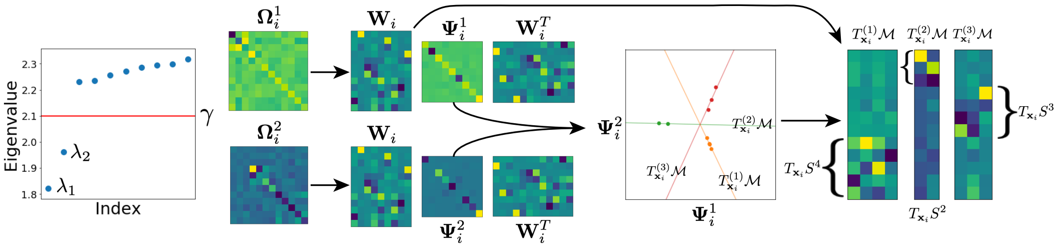


Figure 4: Illustration of the pipeline to go from eigenfunctions of $\overline{\Delta^2}$ to a set of bases for disentangled subspaces, shown on real data from the manifold $S^2 \times S^3 \times S^4$. Matrices $\mathbf{\Omega}_i^r$ up to the threshold in the spectrum γ are simultaneously diagonalized as $\mathbf{W}_i \mathbf{\Psi}_i^r \mathbf{W}_i^T$. Columns of $\mathbf{W}_i$ are clustered based on the cosine similarity of the diagonals of $\mathbf{\Psi}_i^r$ to form bases for the subspaces $T^{(1)}_{\mathbf{x}_i}\mathcal{M}$,...,$T^{(m)}_{\mathbf{x}_i}\mathcal{M}$. For clarity, the data is visualized in the embedding space rather than coordinates of the tangent space.

$\mathbf{\Pi}^{(i)} : \mathcal{M} \rightarrow T_x\mathcal{M} \otimes T_x^*\mathcal{M}$ for $i \in 1,\dots,m$, where $\mathbf{\Pi}^{(i)}_x$ is the linear projection operator from $T_x\mathcal{M} \rightarrow T_x^{(i)}\mathcal{M}$, go to 0 under the action of the connection Laplacian Δ^2.

We provide an informal argument and a more formal proof in Supp. Mat., Sec. B. There is an elegant parallel with the scalar Laplacian. The number of zero eigenvalues of the Laplacian is equal to the number of connected components of a graph or manifold, with each eigenvector being uniform on one component and zero everywhere else. For the second-order Laplacian, the zero eigenvalues correspond to factors of a product manifold, with the matrix-valued eigenfunction being the identity in the subspace tangent to one manifold and zero everywhere else. However, these are not in general the *only* eigenfunctions of Δ^2 with zero eigenvalue, and we will discuss in the next section how GEOMANCER avoids spurious eigenfunctions.

3 Method

Here we describe the actual Geometric Manifold Component Estimator algorithm (GEOMANCER). The main idea is to approximate the second-order connection Laplacian Δ^2 from finite samples of points on the manifold, and then find those eigenvectors with nearly zero eigenvalue that correspond to the disentangled submanifolds of the data. This allows us to define a set of *local* coordinates around every data point that are aligned with the disentangled manifolds.

Suppose we have a set of points $\mathbf{x}_1,\dots,\mathbf{x}_t \in \mathbb{R}^n$ sampled from some manifold embedded in $\mathbb{R}^n$. To construct Δ^2, we first build up approximations to many properties of the manifold. In our discussion here, we will assume the data is embedded in a space where the ℓ_2 metric in $\mathbb{R}^n$ matches the metric on the manifold. Such an embedding must exist for all manifolds [61], but how to learn it is an open question.

To start, we construct a symmetric nearest neighbors graph with edges $\mathcal{E} = \{e_{ij}\}$ between $\mathbf{x}_i$ and $\mathbf{x}_j$. This defines the set of possible steps that can be taken under a random walk. Next, we construct a set of tangent spaces, one per data point, by applying PCA to the difference between $\mathbf{x}_i$ and its neighbors $\mathbf{x}_j$ s.t. $e_{ij} \in \mathcal{E}$. The number of principal components k, equivalent to the dimensionality of the manifold, is a hyperparameter we are free to choose. This defines a set of local orthonormal coordinate systems $\mathbf{U}_i$ and local tangent vectors $\mathbf{v}_j$ s.t. $\mathbf{x}_j - \mathbf{x}_i \approx \mathbf{U}_i \mathbf{v}_j$ for neighboring points $\mathbf{x}_j$. This approach to constructing local tangent spaces is also used by many other manifold learning methods [53, 54, 62–64]. We will use these coordinates to construct the parallel transport from the point $\mathbf{x}_i$ to $\mathbf{x}_j$.

Graph Connection Laplacians To construct Δ^2, we need a generalization of graph Laplacians to higher order tensors. The graph Laplacian is a linear operator on scalar functions on a graph, defined as:

$$\Delta^0[f]_i = \sum_{j \text{s.t.} e_{ij} \in \mathcal{E}} f_i - f_j \tag{2}$$

Equivalently, if we represent functions over data points as a vector in $\mathbb{R}^t$ then the Laplacian can be given as a matrix Δ^0 in $\mathbb{R}^{t \times t}$ with $\Delta^0_{ij} = -1$ if $e_{ij} \in \mathcal{E}$ and $\Delta^0_{ii} = n_i$ where n_i is the number of neighbors of

```
Algorithm 1: Geometric Manifold Component Estimator (GEOMANCER)
Data: $\mathbf{x}_1,...,\mathbf{x}_t \in \mathbb{R}^n$ sampled from $\mathcal{M} = \mathcal{M}_1 \times ... \times \mathcal{M}_m$ with dimension $k$
1. Build the manifold:
    $e_{ij} \in \mathcal{E}$ if $\mathbf{x}_j \in \mathrm{knn}(\mathbf{x}_i)$ or $\mathbf{x}_i \in \mathrm{knn}(\mathbf{x}_j)$    ▷ Construct nearest neighbors graph
    $d\mathbf{X}_i = (\mathbf{x}_{j_1} - \mathbf{x}_i,...,\mathbf{x}_{j_{n_i}} - \mathbf{x}_i)$ for $j_1,...,j_{n_i}$ s.t. $e_{ij} \in \mathcal{E}$
    $\mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^T = \mathrm{SVD}(d\mathbf{X}_i)$, $T_{\mathbf{x}_i}\mathcal{M} \approx \mathrm{span}(\mathbf{U}_i)$    ▷ Estimate tangent spaces by local PCA
2. Build and diagonalize the connection Laplacian:
    $\mathbf{U}_{ij} \mathbf{\Sigma}_{ij} \mathbf{V}_{ij}^T = \mathrm{SVD}(\mathbf{U}_j^T \mathbf{U}_i)$
    $\mathbf{Q}_{ij} = \mathbf{U}_{ij} \mathbf{V}_{ij}^T$ for all $i,j$ s.t. $e_{ij} \in \mathcal{E}$    ▷ Construct connection
    $\Delta^2_{(ij)} = -\mathbf{Q}_{ij} \otimes \mathbf{Q}_{ij}$, $\Delta^2_{(ii)} = n_i \mathbf{I}$    ▷ Build blocks of 2nd-order graph connection Laplacian
    $\overline{\Delta^2_{(ij)}} = \mathbf{\Pi}_{\mathrm{tr}}^T \mathbf{\Pi}_{\mathrm{sym}}^T \Delta^2_{(ij)} \mathbf{\Pi}_{\mathrm{sym}} \mathbf{\Pi}_{\mathrm{tr}}$    ▷ Project blocks onto space of symmetric zero-trace matrices
    $\overline{\Delta^2} \boldsymbol{\phi}^r = \lambda_r \boldsymbol{\phi}^r$, $r = 1,...,R$    ▷ Compute bottom $R$ eigenfunctions/values of $\overline{\Delta^2}$
    $\mathrm{vec}(\mathbf{\Omega}_i^r) = \mathbf{\Pi}_{\mathrm{sym}} \mathbf{\Pi}_{\mathrm{tr}} \boldsymbol{\phi}_i^r$    ▷ Project eigenfunctions back to matrices
3. Align the results from different eigenvectors of the Laplacian:
    $\mathbf{W}_i \mathbf{\Psi}_i^r \mathbf{W}_i^T = \mathbf{\Omega}_i^r$ for all $r$ s.t. $\lambda_r < \gamma$    ▷ Simultaneously diagonalize matrices by FFDIAG [67]
    $\boldsymbol{\psi}_{ik} = (\Psi^1_{i,kk}...,\Psi^r_{i,kk},...,\Psi^{m-1}_{i,kk})$
    $\mathcal{C}^j = \{\boldsymbol{\psi}_{ik} | \boldsymbol{\psi}_{ik}^T \boldsymbol{\psi}_{\ ik'} / ||\boldsymbol{\psi}_{ik}|| ||\boldsymbol{\psi}_{ik'}|| > 0.5\}$    ▷ Cluster diagonals of $\mathbf{\Psi}_i$ by cosine similarity
    $T^{(j)}_{\mathbf{x}_i}\mathcal{M} = \mathrm{span}(\{\mathbf{w}_{ik} | \boldsymbol{\psi}_{ik} \in \mathcal{C}^j\})$    ▷ Columns of $\mathbf{W}_i$ in each cluster span the subspaces
Result: Orthogonal subspaces $T^{(1)}_{\mathbf{x}_i}\mathcal{M},...,T^{(m)}_{\mathbf{x}_i}\mathcal{M}$ at every point $\mathbf{x}_i$ tangent to $\mathcal{M}_1,...,\mathcal{M}_m$
```

$\mathbf{x}_i$. If the graph is approximating a Riemannian manifold, then in the limit of dense sampling the graph Laplacian becomes equivalent to the Laplace-Beltrami operator [65].

To generalize the graph Laplacian from scalars to vectors and tensors, we can replace the difference between neighboring scalars in Eqn. 2 with a difference between tensors. These must be tensors *in the same tangent space*, so any neighboring vectors must be parallel transported from $T_{\mathbf{x}_i}\mathcal{M}$ to $T_{\mathbf{x}_j}\mathcal{M}$, and similarly for higher-order tensors. On a graph, the parallel transport from $\mathbf{x}_i$ to $\mathbf{x}_j$ can be approximated by an orthonormal matrix $\mathbf{Q}_{ij}$ associated with e_{ij}, while the transport in the reverse direction is given by $\mathbf{Q}_{ji} = \mathbf{Q}_{ij}^T$. This leads to a natural definition for the first-order *graph connection Laplacian* [54]:

$$\Delta^1[\mathbf{v}]_i = \sum_{j \mathrm{s.t.} e_{ij} \in \mathcal{E}} \mathbf{v}_i - \mathbf{Q}_{ij}^T \mathbf{v}_j \tag{3}$$

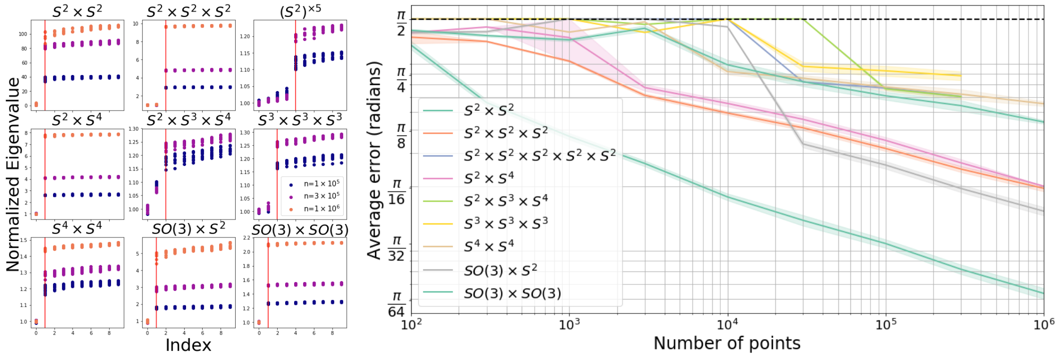
This is a linear operator on vector-valued functions. We can represent vector-valued functions as a single flattened vector in $\mathbb{R}^{tk}$, in which case the graph connection Laplacian is a block-sparse matrix in $\mathbb{R}^{tk \times tk}$. Generalizing to matrices yields the second-order graph connection Laplacian:

$$\Delta^2[\mathbf{\Sigma}]_i = \sum_{j \mathrm{s.t.} e_{ij} \in \mathcal{E}} \mathbf{\Sigma}_i - \mathbf{Q}_{ij}^T \mathbf{\Sigma}_j \mathbf{Q}_{ij} \tag{4}$$

which, again flattening matrix-valued functions to vectors in $\mathbb{R}^{tk^2}$, gives a block-sparse matrix in $\mathbb{R}^{tk^2 \times tk^2}$. The diagonal blocks $\Delta^2_{(ii)} = n_i \mathbf{I}$ while the block for edge e_{ij} is given by $\Delta^2_{(ij)} = -\mathbf{Q}_{ij} \otimes \mathbf{Q}_{ij}$.

While this gives a general definition of Δ^2 for graphs, we still need to define the connection matrices $\mathbf{Q}_{ij}$. When a manifold inherits its metric from the embedding space, the connection is given by the projection of the connection in the embedding space. As the connection in Euclidean space is trivial, the connection on the manifold is given by the orthonormal matrix that most closely approximates the projection of $T_{\mathbf{x}_j}\mathcal{M}$ onto $T_{\mathbf{x}_i}\mathcal{M}$. In the local coordinates defined by $\mathbf{U}_i$ and $\mathbf{U}_j$, the projection is given by $\mathbf{U}_j^T \mathbf{U}_i$. If $\mathbf{U}_{ij} \mathbf{\Sigma}_{ij} \mathbf{V}_{ij}^T$ is the SVD of $\mathbf{U}_j^T \mathbf{U}_i$, then $\mathbf{Q}_{ij} = \mathbf{U}_{ij} \mathbf{V}_{ij}^T$ gives the orthonormal matrix nearest to $\mathbf{U}_j^T \mathbf{U}_i$. This is closely related to the canonical (or principal) angles between subspaces [66], and this connection was also used by Singer and Wu [54] for the original graph connection Laplacian.

Eliminating Spurious Eigenfunctions We now have all the ingredients needed to construct Δ^2. However, a few modifications are necessary to separate out the eigenfunctions that are projections



(a) (b)

Figure 5: Results on synthetic manifolds. (a) The spectrum of $\overline{\Delta^2}$ for products of spheres and special orthogonal groups with different amounts of data. The spectrum is rescaled so the first eigenvalue equals 1. A clear gap emerges in the spectrum at the eigenvalue equal to the number of submanifolds (red line). (b) The average angle between the subspaces recovered by GEOMANCER and the true tangent spaces of the submanifolds. Past a critical threshold, the error declines with more training data.

onto submanifolds from those that are due to specific properties of the particular manifold. First, many manifolds have eigenfunctions of Δ^2 with zero eigenvalue that are skew-symmetric (Supp. Mat., Sec. C). Moreover, the action of Δ^2 on any function of the form $f_j\mathbf{I}$ will be the same as the action of Δ^0 on f_j, meaning eigenvalues of Δ^0 are present in the spectrum of Δ^2 as well. While these will not typically have eigenvalue zero, they may still be small enough to get mixed in with more meaningful results. To avoid both of these spurious eigenfunctions we project each block of Δ^2 onto the space of operators on symmetric zero-trace matrices, to yield a projected block $\overline{\Delta^2_{(ij)}}$ of the projected second-order graph connection Laplacian $\overline{\Delta^2}$. The eigenfunctions of interest can still be expressed as $\sum_j c^r_j \mathbf{\Pi}^{(j)}$ where $\mathbf{\Pi}^{(j)}_i$ is the orthogonal projection onto $T^{(j)}_{\mathbf{x}_i}\mathcal{M}$ and $\sum_j c^r_j \mathrm{dim}(T^{(j)}_{\mathbf{x}_i}\mathcal{M}) = 0$. We can then use standard sparse eigensolvers to find the lowest eigenvalues $\lambda_1,...,\lambda_R$ and eigenvectors $\boldsymbol{\phi}^1,...,\boldsymbol{\phi}^R$, which we split into individual vectors $\boldsymbol{\phi}^r_i$ for each point $\mathbf{x}_i$ and project back to full $k\times k$ matrices $\mathbf{\Omega}^r_i$. For details please refer to Sec. C.2.

Clustering Subspace Dimensions Once we have computed the smallest eigenvalues $\lambda_1,...,\lambda_R$ and matrices $\mathbf{\Omega}^1_1,...,\mathbf{\Omega}^R_t$ from $\overline{\Delta^2}$, we need to merge the results together into a set of orthogonal subspaces $T^{(1)}_{\mathbf{x}_i}\mathcal{M},...,T^{(m)}_{\mathbf{x}_i}\mathcal{M}$ at every point. The appropriate number of submanifolds m can be inferred by looking for a gap in the spectrum and stopping before $\lambda_m > \gamma$, similar to how PCA can identify the dimensionality of the best linear projection of a dataset. Due to the degeneracy of the eigenfunctions and the constraint that $\mathrm{tr}(\mathbf{\Omega}^r_i) = 0$, the results will be linear combinations of $\mathbf{\Pi}^{(j)}_i$ that we have to demix. As the projection matrices are orthogonal to one another, they can be expressed in the same orthonormal basis $\mathbf{W}_i$ as $\mathbf{\Pi}^{(j)}_i = \mathbf{W}_i\mathbf{D}^{(j)}_i\mathbf{W}^T_i$, where $\mathbf{D}^{(j)}_i$ are $0/1$-valued diagonal matrices such that $\sum_j \mathbf{D}^{(j)}_i = \mathbf{I}$. This is a simultaneous diagonalization problem, which generalizes eigendecomposition to multiple matrices that share the same eigenvectors. We solve this with the orthogonal FFDIAG algorithm [67], yielding a decomposition $\mathbf{\Omega}^r_i = \mathbf{W}_i\mathbf{\Psi}^r_i\mathbf{W}^T_i$ where $\mathbf{\Psi}^r_i \approx \sum_j c^r_j \mathbf{D}^{(j)}_i$.

The columns of $\mathbf{W}_i$ then need to be clustered, one cluster per disentangled subspace. Let $\boldsymbol{\psi}_{ik}$ be the vector made up of the k-th diagonal of $\mathbf{\Psi}^r_i$ for all $r = 1,...,m-1$. The simultaneous constraints on c^r_j and $\mathbf{D}^{(j)}_i$ push $m-1$-dimensional vectors $\boldsymbol{\psi}_{ik}$ to cluster together in the m corners of a simplex. Thus the vectors can simply be clustered by checking if the cosine similarity between two $\boldsymbol{\psi}_{ik}$ is greater than some threshold. Finally, a basis for every disentangled subspace can be constructed by taking all columns $\mathbf{w}_{ik}$ of $\mathbf{W}_i$ such that $\boldsymbol{\psi}_{ik}$ cluster together. An example is given in Fig. 4.

The complete algorithm for GEOMANCER is summarized in Alg. 1. The basic building blocks are just nearest neighbors, SVD and eigendecomposition. GEOMANCER requires very few hyperparameters – just the dimension k, the number of nearest neighbors, and the gap γ in the spectrum of $\overline{\Delta^2}$ at

Object	Latents				LEM	Pixels	β-VAE	Chance
	True	Rotated	Scaled	Linear	d=15		d=8	
Bunny	**0.024**	**0.024**	0.72±0.68	1.42±0.09	0.37	1.25	1.30±0.02	1.26±0.23
Dragon	**0.023**	**0.023**	0.96±0.70	1.26±0.37	0.32	1.26	1.15±0.09	1.26±0.23

Table 1: The average angle between the true disentangled subspaces and the subspaces recovered by GeoManCEr from different embeddings of Stanford 3D objects. The first 5 columns all use information from the true latents, including Laplacian Eigenmaps (LEM) with 15 embedding dimensions, while the next 2 columns only use pixels.

which to stop splitting tangent spaces, which can be chosen by simple heuristics. We demonstrate GeoManCEr's performance in the next section.

4 Experiments

To demonstrate the power of GeoManCEr, we investigate its performance on both synthetic manifolds and a dataset of rendered 3D objects. We avoid using existing performance metrics for disentangling, as most are based on the causal or probabilistic interpretation of disentangling [2, 4, 6, 20, 21, 68–70] rather than the symmetry-based one. Furthermore, many disentangling metrics assume that each disentangled factor is one-dimensional [2, 4, 6, 68, 69], which is not the case in our investigation here. Details of dataset generation, training, evaluation and additional results are given in Supp. Mat., Sec. D.

Synthetic Data First, we generated data from a variety of product manifolds by uniformly sampling from either the n-dimensional sphere $S^n \subset \mathbb{R}^{n+1}$, represented as vectors with unit length, or the special orthogonal group $SO(n) \subset \mathbb{R}^{n \times n}$, represented as orthonormal matrices with positive determinant. We then concatenated the vectorized data to form a product manifold. As these manifolds are geodesically complete and have a metric structure inherited from their embedding in Euclidean space, we would expect GeoManCEr to discover their global product structure easily.

Due to sampling noise, no eigenvalues were exactly zero, but it can be seen in Fig. 5(a) that the spectrum has a large gap at the number corresponding to the number of submanifolds, which grows with increased data. In Fig. 5(b) we show the average angle between the learned subspaces and the ground truth tangent spaces of each submanifold. For all manifolds, a threshold is crossed beyond which the error begins to decline exponentially with more data. Unsurprisingly, the amount of data required grows for more complex high-dimensional manifolds, but in all cases GeoManCEr is able to discover the true product structure. Even for manifolds like $SO(n)$ that are not simply connected, GeoManCEr still learns the correct local coordinate system. Note that the most complex manifolds we are able to disentangle consist of up to 5 submanifolds, significantly outperforming other recent approaches to symmetry-based disentanglement which have not been applied to anything more complex than a product of two submanifolds [16, 17]. Moreover, our approach is fully unsupervised, while theirs requires conditioning on actions.

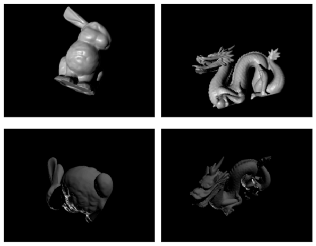


Figure 6: The Stanford Bunny and Dragon with different poses and illumination.

Stanford 3D Objects To investigate more realistic data, we applied GeoManCEr to renderings of the Bunny and Dragon object from the Stanford 3D Scanning Repository [71] under different pose and lighting conditions (Fig. 6). We chose to render our own data, as existing datasets for 3D objects are limited to changes in azimuth and a limited range of elevations [6, 34–38]. Instead, we sampled rotations of object pose uniformly from $SO(3)$, while the light location was sampled uniformly from the sphere S^2. In Table 1, we show the accuracy of GeoManCEr applied to several different embeddings using the same performance metric as in Fig. 5(b). When applied directly to the true latent state vectors, GeoManCEr performs exceptionally well, even if the state is rotated randomly. When individual dimensions are multiplied by a random scale factor, the performance degrades, and if a random linear transformation is applied to the state, performance is no better than chance. This shows that accurate metric information is necessary for GeoManCEr to work. We also applied our method to embeddings learned by Laplacian Eigenmaps using no information other than knowledge of the nearest neighbors in the true latent space. While not as accurate as working from the true latent state, it still performs far better than chance, showing that metric information alone is *sufficient* for GEOMANCER to work.

Trying to disentangle directly from pixels is more complicated. As the mapping from state to observation is highly nonlinear, GEOMANCER performs no better than chance directly from pixels. However, existing algorithms to disentangle directly from pixels fail as well [19] (See Supp. Mat., Sec. D.3). Even when applying GEOMANCER to the latent vectors learned by the β-VAE, the results are no better than chance. The poor performance of both GEOMANCER and β-VAE on the Stanford 3D Objects shows that disentangling full 3D rotations from pixels without side information remains an open problem.

5 Discussion

We have shown that GEOMANCER is capable of factorizing manifolds directly from unstructured samples. On images of 3D objects under changes in pose and illumination, we show that correct metric information is critical for GEOMANCER to work. There are several directions for improvement. As GEOMANCER is a nonparametric spectral method, it does not automatically generalize to held-out data, though there are extensions that enable this [72, 73]. While GEOMANCER scales well with the amount of data, the number of nonzero elements in $\overline{\Delta}^2$ grows as $\mathcal{O}(k^4)$ in the dimensionality of the manifold, meaning new approximations are needed to scale to more complex manifolds. The motivating mathematical intuition could also be implemented in a parametric model, enabling end-to-end training.

The missing ingredient for fully unsupervised disentangling is a source of correct metric information. We have shown that existing generative models for disentangling are insufficient for learning an embedding with the right metric structure. We hope that this will be a challenge taken up by the metric learning community. We are hopeful that this charts a new path forward for disentangling research.

Broader Impact

The present work is primarily theoretical, making its broader impact difficult to ascertain. We consider the algorithm presented here to be a potential core machine learning method, and as such it could have an impact in any area that machine learning can be applied to, but particularly in unsupervised learning, computer vision and robotic manipulation.

Acknowledgments and Disclosure of Funding

We would like to thank Mélanie Rey, Drew Jaegle, Pedro Ortega, Peter Toth, Olivier Hénaff, Malcolm Reynolds, and Kevin Musgrave for helpful discussions and Shakir Mohamed for support and encouragement.

References

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation Learning: A Review and New Perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[2] Karl Ridgeway and Michael C Mozer. Learning Deep Disentangled Embeddings with the F-Statistic Loss. *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[3] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[4] Tian Qi Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders. *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[5] Emilien Dupont. Learning Disentangled Joint Continuous and Discrete Representations. *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[6] Hyunjik Kim and Andriy Mnih. Disentangling by Factorising. *Proceedings of the Sixth Annual International Conference on Learning Representations (ICLR)*, 2018.

[7] Babak Esmaeili, Hao Wu, Sarthak Jain, Alican Bozkurt, N. Siddharth, Brooks Paige, Dana H. Brooks, Jennifer Dy, and Jan-Willem van de Meent. Structured Disentangled Representations. *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

[8] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations. *Proceedings of the Sixth Annual International Conference on Learning Representations (ICLR)*, 2018.

[9] Alessandro Achille, Tom Eccles, Loic Matthey, Christopher P Burgess, Nick Watters, Alexander Lerchner, and Irina Higgins. Life-Long Disentangled Representation Learning with Cross-Domain Latent Homologies. *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[10] Abdul Fatir Ansari and Harold Soh. Hyperprior Induced Unsupervised Disentanglement of Latent Representations. *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

[11] Emile Mathieu, Tom Rainforth, N. Siddharth, and Yee Whye Teh. Disentangling Disentanglement in Variational Autoencoders. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[12] Nicki S. Detlefsen and Soren Hauberg. Explicit Disentanglement of Appearance and Perspective in Generative Models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[13] Amir Dezfouli, Hassan Ashtiani, Omar Ghattas, Richard Nock, Peter Dayan, and Cheng Soon Ong. Disentangled Behavioral Representations. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[14] Dominik Lorenz, Leonard Bereska, Timo Milbich, and Bjorn Ommer. Unsupervised Part-Based Disentangling of Object Shape and Appearance. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[15] Wonkwang Lee, Donggyun Kim, Seunghoon Hong, and Honglak Lee. High-Fidelity Synthesis with Disentangled Representation. *arXiv preprint arXiv:2001.04296*, 2020.

[16] Hugo Caselles-Dupré, Michael Garcia-Ortiz, and David Filliat. Symmetry-Based Disentangled Representation Learning Requires Interaction with Environments". *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[17] Robin Quessard, Thomas D. Barrett, and William R. Clements. Learning Group Structure and Disentangled Representations of Dynamical Environments. *arXiv preprint arXiv:2002.06991*, 2020.

[18] Aditya Ramesh, Youngduck Choi, and Yann LeCun. A Spectral Regularizer for Unsupervised Disentanglement. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[19] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. *Proceedings of the Fifth Annual International Conference on Learning Representations (ICLR)*, 2017.

[20] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[21] Irina Higgins, David Amos, David Pfau, Sébastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a Definition of Disentangled Representations. *arXiv preprint arXiv:1812.02230*, 2018.

[22] William C Hoffman. The Lie algebra of visual perception. *Journal of Mathematical Psychology*, 3(1):65–98, 1966.

[23] Peter C Dodwell. The Lie transformation group model of visual perception. *Perception & Psychophysics*, 34(1):1–16, 1983.

[24] Rajesh PN Rao and Daniel L Ruderman. Learning Lie groups for invariant visual perception. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 810–816, 1999.

[25] Jascha Sohl-Dickstein, Ching Ming Wang, and Bruno A Olshausen. An unsupervised algorithm for learning Lie group transformations. *arXiv preprint arXiv:1001.1027*, 2010.

[26] Fabio Anselmi, Lorenzo Rosasco, and Tomaso Poggio. On invariance and selectivity in representation learning. *Information and Inference: A Journal of the IMA*, 5(2):134–158, 2016.

[27] David E Rumelhart and Adele A Abrahamson. A Model for Analogical Reasoning. *Cognitive Psychology*, 5(1):1–28, 1973.

[28] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.

[29] Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[30] Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep Visual Analogy-Making. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1252–1260, 2015.

[31] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *Proceedings of the Fourth Annual International Conference on Learning Representations (ICLR)*, 2016.

[32] Dawn Chen, Joshua C Peterson, and Thomas L Griffiths. Evaluating Vector-Space Models of Analogy. *Proceedings of the Thirty-Ninth Annual Conference on Cognitive Science (CogSci)*, 2017.

[33] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. *arXiv preprint arXiv:1912.04958*, 2019.

[34] Yann LeCun, Fu Jie Huang, and Leon Bottou. Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–104. IEEE, 2004.

[35] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[36] Sanja Fidler, Sven Dickinson, and Raquel Urtasun. 3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model. In *Advances in neural information processing systems*, pages 611–619, 2012.

[37] Mathieu Aubry, Daniel Maturana, Alexei A Efros, Bryan C Russell, and Josef Sivic. Seeing 3D Chairs: Exemplar Part-Based 2D-3D Alignment Using a Large Dataset of CAD Models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3762–3769, 2014.

[38] Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3D Face Model for Pose and Illumination Invariant Face Recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 296–301. IEEE, 2009.

[39] Kunihiko Fukushima. Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. *Biological Cybernetics*, 36(4):193–202, 1980.

[40] Robert Gens and Pedro M Domingos. Deep Symmetry Networks. In *Advances in Neural Information Processing Systems*, pages 2537–2545, 2014.

[41] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 2990–2999, 2016.

[42] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. *Proceedings of the Sixth Annual International Conference on Learning Representations (ICLR)*, 2018.

[43] Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch–Gordan Nets: a Fully Fourier Space Spherical Convolutional Neural Network. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10117–10126, 2018.

[44] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S Cohen. 3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 10381–10392, 2018.

[45] Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7090–7099, 2019.

[46] Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing Convolutional Neural Networks for Equivariance to Lie Groups on Arbitrary Continuous Data. *arXiv preprint arXiv:2002.12880*, 2020.

[47] Taco Cohen and Max Welling. Learning the Irreducible Representations of Commutative Lie Groups. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 1755–1763, 2014.

[48] Benjamin Culpepper and Bruno A Olshausen. Learning Transport Operators for Image Manifolds. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 423–431, 2009.

[49] Marissa Connor and Christopher Rozell. Representing Closed Transformation Paths in Encoded Network Latent Space. *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[50] Georges De Rham. Sur la réductibilité d'un espace de Riemann. *Commentarii Mathematici Helvetici*, 26(1):328–344, 1952.

[51] Mikhail Belkin and Partha Niyogi. Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. In *Advances in neural information processing systems*, pages 585–591, 2002.

[52] Ronald R Coifman and Stéphane Lafon. Diffusion Maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[53] Amit Singer and Hau-tieng Wu. Orientability and diffusion maps. *Applied and computational harmonic analysis*, 31(1):44–58, 2011.

[54] Amit Singer and Hau-Tieng Wu. Vector Diffusion Maps and the Connection Laplacian. *Communications on pure and applied mathematics*, 65(8):1067–1144, 2012.

[55] Yifeng Fan and Zhizhen Zhao. Multi-Frequency Vector Diffusion Maps. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[56] Tingran Gao, Jacek Brodzki, and Sayan Mukherjee. The Geometry of Synchronization Problems and Learning Group Actions. *Discrete & Computational Geometry*, pages 1–62, 2019.

[57] Tingran Gao. The Diffusion Geometry of Fibre Bundles: Horizontal Diffusion Maps. *Applied and Computational Harmonic Analysis*, 2019.

[58] Dian Gong, Xuemei Zhao, and Gérard Medioni. Robust Multiple Manifolds Structure Learning. *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

[59] Manfredo P Do Carmo. *Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition*. Courier Dover Publications, 2016.

[60] Shoshichi Kobayashi and Katsumi Nomizu. *Foundations of Differential Geometry*, volume 1. New York, London, 1963.

[61] John Nash. The Imbedding Problem for Riemannian Manifolds. *Annals of Mathematics*, pages 20–63, 1956.

[62] Zhenyue Zhang and Hongyuan Zha. Principal Manifolds and Nonlinear Dimensionality Reduction via Tangent Space Alignment. *SIAM Journal on Scientific Computing*, 26(1):313–338, 2004.

[63] Moshe Salhov, Guy Wolf, and Amir Averbuch. Patch-to-Tensor Embedding. *Applied and Computational Harmonic Analysis*, 33(2):182–203, 2012.

[64] Guy Wolf and Amir Averbuch. Linear-Projection Diffusion on Smooth Euclidean Submanifolds. *Applied and Computational Harmonic Analysis*, 34(1):1–14, 2013.

[65] Mikhail Belkin and Partha Niyogi. Towards a Theoretical Foundation for Laplacian-based Manifold Methods. *Journal of Computer and System Sciences*, 74(8):1289–1308, 2008.

[66] Peizhen Zhu and Andrew V Knyazev. Angles Between Subspaces and their Tangents. *Journal of Numerical Mathematics*, 21(4):325–340, 2013.

[67] Andreas Ziehe, Pavel Laskov, Guido Nolte, and Klaus-Robert Müller. A Fast Algorithm for Joint Diagonalization with Non-Orthogonal Transformations and its Application to Blind Source Separation. *Journal of Machine Learning Research*, 5(Jul):777–800, 2004.

[68] Raphael Suter, Đorđe Miladinović, Bernhard Schölkopf, and Stefan Bauer. Robustly Disentangled Causal Mechanisms: Validating Deep Representations for Interventional Robustness. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[69] Cian Eastwood and Christopher KI Williams. A Framework for the Quantitative Evaluation of Disentangled Representations. *Proceedings of the Sixth Annual International Conference on Learning Representations (ICLR)*, 2018.

[70] Sunny Duan, Loic Matthey, Andre Saraiva, Nick Watters, Chris Burgess, Alexander Lerchner, and Irina Higgins. Unsupervised Model Selection for Variational Disentangled Representation Learning. In *Proceedings of the Eighth Annual International Conference on Learning Representations (ICLR)*, 2020.

[71] Marc Levoy, J Gerth, B Curless, and K Pull. The Stanford 3D Scanning Repository. *URL http://www-graphics. stanford. edu/data/3dscanrep*, 5, 2005.

[72] Yoshua Bengio, Jean-françcois Paiement, Pascal Vincent, Olivier Delalleau, Nicolas L Roux, and Marie Ouimet. Out-of-sample Extensions for LLE, IsoMap, MDS, Eigenmaps, and Spectral Clustering. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 177–184, 2004.

[73] David Pfau, Stig Petersen, Ashish Agarwal, David GT Barrett, and Kimberly L Stachenfeld. Spectral Inference Networks: Unifying Deep and Spectral Learning. *Proceedings of the Seventh Annual International Conference on Learning Representations (ICLR)*, 2019.

[74] Prasanta Chandra Mahalanobis. On the Generalized Distance in Statistics. *Journal of the Asiatic Society of Bengal*, 26:541–588, 1936.

[75] Brian Kulis. Metric Learning: A Survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013.

[76] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A Physics Engine for Model-Based Control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.