ENERGYMAP: UNRAVELING THE DATA MANIFOLD WITH ENERGY-BASED DIMENSIONALITY REDUCTION

Alec Helbling[1], **Parikshit Ram**[2], **Duen Horng (Polo) Chau** [1] **& Benjamin Hoover**[1,2]
[1]Georgia Institute of Technology
[2]IBM Research

ABSTRACT

Learning meaningful low-dimensional representations of high-dimensional data remains a central challenge in machine learning. Despite their popularity, dimensionality reduction methods like UMAP and t-SNE rely on heuristic assumptions about the local structure of data that scale poorly to complex modalities like images. In this work, we introduce ENERGYMAP, a dimensionality reduction method that leverages the implicit structure of data captured by an Energy-based model (EBM). Instead of relying on local Euclidean distance heuristics, ENERGYMAP computes geometrically faithful distances between points by finding energy-minimizing geodesics that traverse dense regions of the data distribution, and produces a low-dimensional embedding that preserves these geodesic distances. In order to efficiently optimize geodesics we develop an efficient algorithm for computing geodesics that combines discrete path finding with continuous gradient-based refinement, substantially improving over naive gradient-based optimization. We provide a promising proof of concept demonstrating the potential for ENERGYMAP to compute embeddings that capture semantically meaningful concepts in high dimensional data.

1 INTRODUCTION

Can we leverage the implicit geometric information captured by a generative model to unravel the manifold structure of data? A widely held hypothesis in machine learning is that high-dimensional data often lies on or near a low-dimensional manifold embedded in ambient space (Bengio et al., 2014). Uncovering this manifold structure is central to understanding, visualizing, and interpreting complex datasets. Classical dimensionality reduction methods such as PCA assume linear structure, while nonlinear methods such as Isomap (Tenenbaum et al., 2000), t-SNE (van der Maaten & Hinton, 2008), and UMAP (McInnes et al., 2018) attempt to preserve local neighborhood relationships. However, these nonlinear methods rely on heuristic assumptions about local Euclidean distances, which can break down for complex modalities such as images, where nearby points in ambient space may be far apart on the underlying manifold.

Energy-based models (EBMs) offer a compelling alternative perspective. By learning to assign low energy to regions of high data density, EBMs implicitly capture the geometric structure of the data manifold. The energy landscape naturally encodes which regions of space are "on-manifold" (low energy) and which are "off-manifold" (high energy). This observation suggests a principled approach to dimensionality reduction: rather than relying on Euclidean distance heuristics, we can use the energy function to define a Riemannian metric that makes movement through high-energy regions expensive, so that geodesic paths between data points naturally follow the data manifold.

We introduce ENERGYMAP, a dimensionality reduction method that leverages the implicit geometric information captured by an EBM to perform dimensionality reduction. Following (Béthune et al., 2025), we leverage an EBM to create a Riemannian metric which allows us to compute *geodesics* between pairs of points that lay in low-energy regions on the data manifold. We can then use the arc lengths of these geodesics to create a pairwise dissimilarity matrix that captures the structure of data and use methods like classical multidimensional scaling Kruskal (1964) to create a low dimensional embedding. We provide promising qualitative results showing the potential for ENERGYMAP to uncover semantically meaningful dimensions in image data (Figure 1). Additionally, we introduce

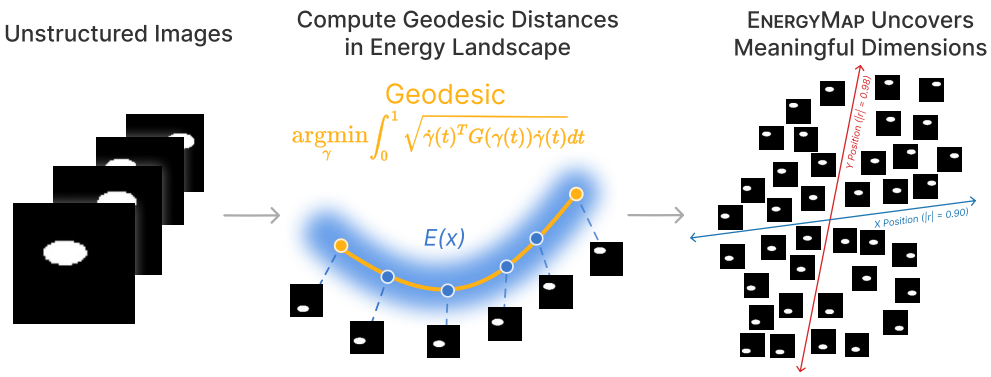

Figure 1: **ENERGYMAP embeds image data so that linear interpolation in the embedding captures continuous on-manifold transformations, by computing energy-minimizing geodesic distances and applying classical MDS.**

a novel algorithm for optimizing geodesics that combines discrete graph-based search with continuous gradient descent and demonstrate its ability to produce geodesics that better adhere to the data manifold.

2 BACKGROUND

Energy-Based Models An energy-based model (EBM) defines a probability distribution over data by associating each point $\boldsymbol{x} \in \mathbb{R}^D$ with a scalar energy $E(\boldsymbol{x}) \in \mathbb{R}$. The probability density is given by the Boltzmann distribution $p(\boldsymbol{x}) = \exp(-E(\boldsymbol{x}))/Z$ where Z is the partition function. Regions of high data density correspond to low energy, and vice versa. In practice, EBMs range from analytical forms such as Hopfield Networks (Hopfield, 1982) and Gaussian mixture models, all the way to neural network-parameterized energy functions trained via methods like equilibrium matching (Wang & Du, 2025).

Riemannian Geometry Riemannian geometry provides a mathematical framework for measuring distances and paths on curved spaces. Rather than assuming a fixed Euclidean notion of distance, a Riemannian manifold equips each point $\boldsymbol{x} \in \mathbb{R}^D$ with a local inner product that specifies how costly it is to move in different directions around that point. Formally, a *Riemannian metric* is a smoothly varying, positive-definite matrix-valued function $\boldsymbol{G}(\boldsymbol{x}) \in \mathbb{R}^{D\times D}$ which defines an inner product between tangent vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^D$ at $\boldsymbol{x}$ via $\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{\boldsymbol{x}} = \boldsymbol{u}^\top \boldsymbol{G}(\boldsymbol{x})\, \boldsymbol{v}$. Intuitively, the metric encodes the local geometry of space: directions associated with large metric values are expensive to traverse, while directions with small values are cheap.

Energy-based Riemannian Metrics Existing work Béthune et al. (2025) has demonstrated the ability to construct a Riemmannian metric using an EBM. They introduce a special case of Riemannian metric called a *conformal* metric, where the metric is a scalar multiple of the identity $\boldsymbol{G}(\boldsymbol{x}) = g(\boldsymbol{x})\, \boldsymbol{I}$, with $g(\boldsymbol{x}) > 0$. Under a conformal metric, all directions are treated isotropically, but the cost of movement varies spatially. This property makes conformal metrics particularly well suited for modeling data manifolds: by choosing $g(\boldsymbol{x})$ to be large in low-density regions and small in high-density regions, geodesics naturally follow the structure of the data distribution.

3 ENERGYMAP

Our goal is to create an embedding $f(\boldsymbol{x})$ of a set of high dimensional points $\{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_n\} \in \mathbb{R}^d$ in a way that is consistent with the geometry induced by an EBM $E_\theta : \mathbb{R}^D \to \mathbb{R}$. In order to do this we must first create a matrix $D \in \mathbb{R}^{n\times n}$ that captures the pairwise dissimilarity between points x_i. Given this matrix D we then need to infer an embedding $Z \in \mathbb{R}^{n\times z}$ for some small dimensionality z where dissimilarity between low dimensional points are consistent with matrix D.

Efficient Computation of Geodesics with Graph-based Routing Instead of training geodesic interpolant neural networks to approximate geodesics like in previous work (Kapuśniak et al., 2024;

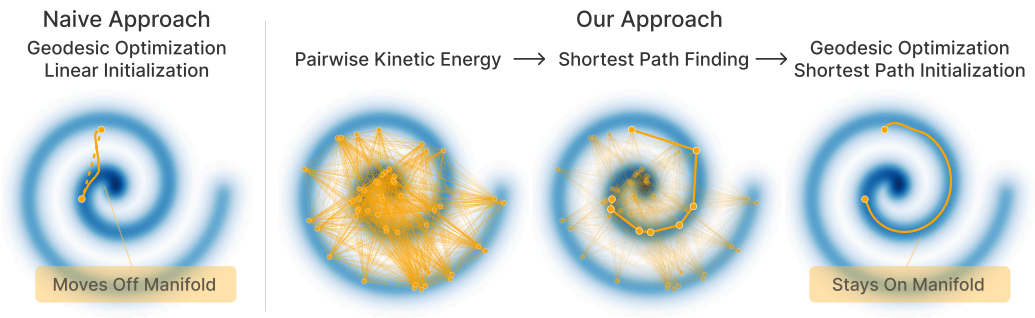

Figure 2: **Graph-based initialization yields much better on-manifold convergence than naive linear initialization.** Linear initialization leads to off-manifold convergence (left). Shortest paths through a K-NN graph weighted by Riemannian arc length provide on-manifold initializations (center) that, after continuous refinement, converge to high-quality geodesics (right).

Béthune et al., 2025) we choose to directly minimize the *action* of a discrete path with gradient-based optimization. Given T discrete samples from a curve parameterized by $\gamma(t)$ we minimize

$$\arg\min_{\gamma} \sum_{t=1}^{T} \sqrt{\dot{\gamma}(t)^\top \boldsymbol{G}(\gamma(t))\dot{\gamma}(t)}\,\Delta t, \tag{1}$$

which generalizes Euclidean arc length to our Riemannian metric. A *geodesic* between two points is a curve that minimizes this length functional, and thus represents the shortest path under the geometry induced by $\boldsymbol{G}$. See more details in Algorithm 2 in the Appendix.

A problem that arises is that Equation 1 is a non-convex problem and frequently converges to local minima corresponding to off-manifold paths when naively initialized with straight trajectories (Figure 2, left). We address this with a two-part algorithm (Algorithm 1). First, we construct a sparse K-nearest-neighbor graph over the data, where edge weights are Riemannian arc lengths under our energy-based metric. Shortest paths in this graph naturally penalize movement through high-energy regions, producing initializations that follow the data manifold. Second, we resample each shortest path to a fixed number of waypoints and refine it via continuous gradient-based optimization, which smooths the discrete path while preserving its manifold-following structure (Figure 2, right). Our experiments on the dSprites (Matthey et al., 2017) dataset show our approach produces smooth transformations between images that remain on the data manifold (Figure 3).

Non-linear Dimensionality Reduction with Geodesic Dissimilarity Once the pairwise geodesic distance matrix $\boldsymbol{D} \in \mathbb{R}^{m\times m}$ has been computed via Algorithm 1, we obtain low-dimensional coordinates via classical multidimensional scaling (MDS) (Kruskal, 1964). Classical MDS constructs the double-centered matrix $\boldsymbol{B} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{D}^{(2)}\boldsymbol{H}$, where $\boldsymbol{D}^{(2)}$ is the element-wise squared distance matrix and $\boldsymbol{H} = \boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ is the centering matrix, then extracts the top-d eigenvectors of $\boldsymbol{B}$ to produce coordinates $\boldsymbol{Y}^* \in \mathbb{R}^{m\times d}$ whose pairwise Euclidean distances best approximate the geodesic dissimilarities. This is equivalent to the embedding step of Isomap (Tenenbaum et al., 2000), but instead applied to our optimized geodesic distances rather than Euclidean nearest neighbor graph shortest-path distances. You can see preliminary result on the dSprites dataset in Figure 1.

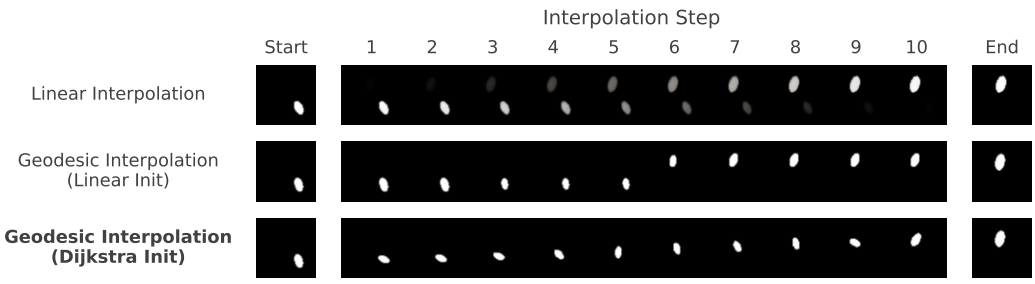

Figure 3: **Geodesic interpolations on dSprites.** We show naive linear interpolation in latent space (top), geodesic interpolation with linear initialization (center), and our graph-based intiailization (bottom). Our appraoch produces smooth transformations that remain on the data manifold.

REFERENCES

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives, 2014. URL https://arxiv.org/abs/1206.5538.

Louis Béthune, David Vigouroux, Yilun Du, Rufin VanRullen, Thomas Serre, and Victor Boutin. Follow the energy, find the path: Riemannian metrics from energy-based models, 2025. URL https://arxiv.org/abs/2505.18230.

J J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982. doi: 10.1073/pnas.79.8.2554. URL https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554.

Kacper Kapuśniak, Peter Potaptchik, Teodora Reu, Leo Zhang, Alexander Tong, Michael Bronstein, Avishek Joey Bose, and Francesco Di Giovanni. Metric flow matching for smooth interpolations on the data manifold, 2024. URL https://arxiv.org/abs/2405.14780.

Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv e-prints*, 2018.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Laurens van der Maaten and Geoffrey Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

Runqian Wang and Yilun Du. Equilibrium matching: Generative modeling with implicit energy-based models, 2025. URL https://arxiv.org/abs/2510.02300.

A Appendix

A.1 Experimental Details

Spiral Manifold We evaluate EnergyMap on a 2D spiral manifold, where the goal is to recover the 1D intrinsic parameterization of the spiral from the 2D ambient coordinates. To construct an energy function for the spiral, we place a Gaussian mixture model (GMM) along the curve: 200 equally-spaced Gaussian components are positioned at points along a parameterized spiral with isotropic covariance $\sigma^2 = 0.5$ and uniform mixture weights $w_k = 1/200$. The energy function is then defined as the negative log-likelihood of the GMM, $E(\boldsymbol{x}) = -\log p_{\mathrm{GMM}}(\boldsymbol{x})$, which is low on the spiral and high away from it. The spiral data and GMM means are normalized to unit standard deviation before computing geodesics.

dSprites We apply EnergyMap to the dSprites dataset (Matthey et al., 2017) of 64×64 grayscale ellipse images. We first encode the images into a 4-channel, 32×32 latent space using a pretrained Stable Diffusion VAE (Rombach et al., 2022), then train a small energy-based model in this latent space using equilibrium matching (Wang & Du, 2025). The energy model uses a transformer architecture (8 layers, hidden dimension 256, 4 attention heads, patch size 2) with adaptive layer normalization, where the energy is defined as the dot product between the transformer output and input. Geodesic distances are computed in the VAE latent space under the resulting energy landscape, and the final embedding is obtained by applying an embedding solver to the geodesic distance matrix.

A.2 Geodesic Computation with Graph-Based Initialization

Algorithm 1 Geodesic Computation with Graph-Based Initialization

Input: Search points $\mathcal{S} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n\}$, query points $\mathcal{Q} = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_m\}$, conformal metric $\boldsymbol{G}(\boldsymbol{x})$, number of waypoints T, number of nearest neighbors K

function ComputeGeodesicDistances($\mathcal{S}, \mathcal{Q}, \boldsymbol{G}, T, K$)

// Phase 1: Graph-based shortest paths

Merge all points into a single vertex set $\mathcal{V} \leftarrow \mathcal{S} \cup \mathcal{Q}$.

For each neighbor pair (i, j) with $\boldsymbol{v}_j \in \mathcal{N}_K(\boldsymbol{v}_i)$, set $\gamma_{ij} \leftarrow \textsc{StraightLinePath}(\boldsymbol{v}_i, \boldsymbol{v}_j, T)$.

Assign each edge its Riemannian arc length $\boldsymbol{W}[i, j] \leftarrow \sum_{t=1}^{T-1} \sqrt{\dot{\gamma}_{ij,t}^\top \boldsymbol{G}(\gamma_{ij,t})\, \dot{\gamma}_{ij,t}}\, \Delta t$.

Find the shortest path $\boldsymbol{p}_{ab} \leftarrow \textsc{ShortestPath}(\boldsymbol{W}, \boldsymbol{q}_a, \boldsymbol{q}_b)$ for every query pair (a, b).

// Phase 2: Continuous geodesic refinement

Resample each $\boldsymbol{p}_{ab}$ to T waypoints.

Refine via $\gamma^*_{ab} \leftarrow \textsc{OptimizeGeodesic}(\textsc{Resample}(\boldsymbol{p}_{ab}, T), \boldsymbol{G})$.

$\boldsymbol{D}[a, b] \leftarrow \sum_{t=1}^{T-1} \sqrt{\dot{\gamma}^{*\top}_{ab,t} \boldsymbol{G}(\gamma^*_{ab,t})\, \dot{\gamma}^*_{ab,t}}\, \Delta t$ for every pair (a, b).

return Pairwise geodesic distance matrix $\boldsymbol{D} \in \mathbb{R}^{m \times m}$

A.3 Geodesic Path Optimization

Algorithm 2 details the gradient-based geodesic optimization used within Algorithm 1. Given two endpoints, the path is discretized into T waypoints and initialized as a straight line (or from a Dijkstra shortest path). The energy functional $\mathcal{L}$, which measures the Riemannian arc length of the discretized path, is then minimized by gradient descent over the interior waypoints. This drives the path toward low-energy regions of the conformal metric, producing geodesics that follow the data manifold.

Algorithm 2 Geodesic Path Optimization via Finite Differences

Input: Endpoints $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{R}^D$, conformal metric $\boldsymbol{G}(\boldsymbol{x})$, waypoints T, step size η, iterations K
function OPTIMIZEGEODESIC($\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{G}, T, \eta, K$)
  $\Delta t \leftarrow 1/(T-1)$
  $\gamma_t \leftarrow (1 - \frac{t-1}{T-1})\,\boldsymbol{x}_0 + \frac{t-1}{T-1}\,\boldsymbol{x}_1 \quad$ for $t = 1, \ldots, T$ // Straight-line initialization
  for $k = 1$ to K **do**
    $\dot{\gamma}_t \leftarrow (\gamma_{t+1} - \gamma_t)/\Delta t \quad$ for $t = 1, \ldots, T-1$
    $\mathcal{L} \leftarrow \sum_{t=1}^{T-1} \dot{\gamma}_t^\top \boldsymbol{G}(\gamma_t)\,\dot{\gamma}_t\,\Delta t$
    $\gamma_t \leftarrow \gamma_t - \eta\,\nabla_{\gamma_t}\mathcal{L} \quad$ for $t = 2, \ldots, T-1$
  return γ

A.4 GEODESIC INTERPOLATION EXAMPLES

Figure 4 shows additional geodesic interpolation examples on the dSprites dataset. Each panel displays a single image pair with three interpolation methods: (top) linear interpolation in latent space, (middle) geodesic interpolation with linear initialization, and (bottom) geodesic interpolation with Dijkstra initialization. The middle and bottom rows show nearest-neighbor images from the dataset along the computed geodesic path. Dijkstra-initialized geodesics produce smoother transitions that better follow the data manifold.

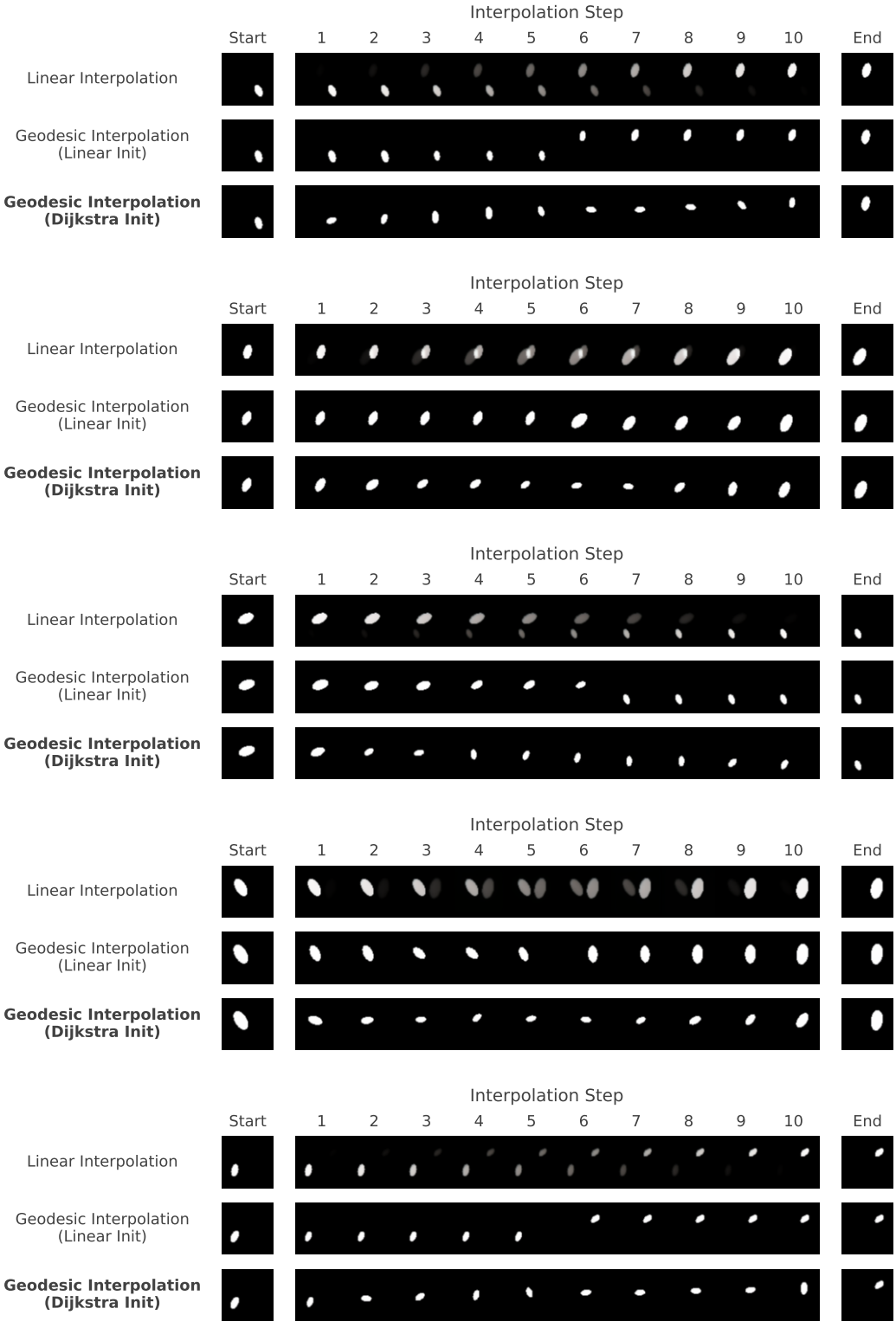

Figure 4: Geodesic interpolation comparisons on five dSprites image pairs. Each panel shows linear interpolation (top), geodesic interpolation with linear initialization (middle), and geodesic interpolation with Dijkstra initialization (bottom).