Projective Shape Metamorphosis With Optimal Transport

VENKATARAM SIVARAM, University of California San Diego, USA

We address the problem of inverse rendering for discrete triangle meshes when only a binary silhouette of the target is available. In the absence of shading cues, traditional physically-based inverse rendering breaks down; likewise, differentiable rasterizers yield useful gradients only at silhouette boundaries and fail entirely when source and target silhouettes do not overlap. Humans, by contrast, effortlessly infer full shape motions from silhouette changes alone. Motivated by this cognitive ability, we cast silhouette matching as an optimal-transport problem in image space. We treat both source and target silhouettes as continuous mass distributions, compute an image-space velocity field that transports one distribution into the other, then lift those velocities back to the mesh vertices. An iterative loop alternates between silhouette transport and mesh deformation until convergence. Our experiments demonstrate that this simple, gradient-free approach can animate smooth shape metamorphoses and recover plausible vertex motions even when traditional silhouette gradients vanish. Additionally, our solution opens a new avenue for silhouette-driven animation.

1 INTRODUCTION

Reconstructing 3D geometry from visual observations is a long-standing pursuit in computer graphics and vision. Classical inverse rendering leverages shading, texture, and lighting cues to infer surface properties, material parameters, and scene geometry under physically based models. However, when the only available observation is a binary silhouette—common in applications ranging from silhouette-based morphing and object tracking to shape-from-silhouette methods—these rich photometric cues vanish. Standard physically based inverse rendering formulations deteriorate without shading information, and differentiable rasterization techniques contribute gradients only along silhouette edges, leading to vanishing or ill-posed updates when source and target outlines do not overlap.

On the other hand, humans routinely perceive and animate complex silhouette deformations: we effortlessly infer continuous shape motions from evolving outlines in stop-motion animations, shadow puppetry, or video. This cognitive ability suggests that silhouette matching may be better addressed by global shape correspondences rather than purely local boundary signals. Optimal transport provides a mathematically principled framework for computing minimal-effort mappings between two mass distributions, ensuring global consistency and smooth displacement fields even when support regions differ substantially.

In this work, we leverage optimal transport in image space to bridge the gap between cognitive intuition and computational practice for silhouette-only inverse rendering. We interpret both source and target silhouettes as continuous mass distributions over the image plane. By solving a Kantorovich transport problem, we obtain a dense image-space velocity field that prescribes how regions in the source silhouette should move toward the target. We then lift these velocities to the underlying triangle mesh by projecting onto corresponding vertices and iteratively deforming the mesh.

Our approach offers several key advantages:

Author's address: Venkataram Sivaram, ves223@ucsd.edu, University of California San Diego, USA.

- *Gradient-free global correspondence.* Unlike differentiable rasterizers that rely on local boundary gradients, our method produces a dense velocity field over the entire silhouette region, enabling shape updates even when silhouettes are disjoint.
- *Plausible vertex motions.* By lifting image-space transport velocities to mesh vertices, we generate smooth, coherent deformations that respect the original connectivity and topology of the mesh.
- *Iterative refinement.* An alternating loop between optimal transport computation and mesh deformation converges rapidly to a shape that matches the target silhouette with high fidelity.

We demonstrate the efficacy of our method on a variety of shape metamorphosis and surface animation tasks, including cases with significant silhouette displacement. By reframing silhouette-driven inverse rendering as an optimal transport problem, we open a new direction for silhouette-only animation and shape reconstruction pipelines.

In the following sections, we review related work (Section 2), detail our optimal-tranport-based methodologies (Section 3), and present experimental results and applications (Section 4).

2 BACKGROUND

Our approach is situated amongst existing literature in inverse and differentiable rendering, differentiable rasterization, silhouette-based methods, and optimal transport.

2.1 Inverse Rendering

Let $R(C, \theta)$ denote a forward renderer that, given camera parameters C and scene parameters θ (including geometry, materials, and lighting), produces an image. The inverse-rendering problem seeks to recover scene parameters θ that best explain a set of N observed images $\{Ii\}$ captured under known cameras $\{Ci\}$:

$$\theta^* = \arg\min_{\theta} \sum_{i=1}^{N} \mathcal{L}(\mathcal{R}(C_i, \theta), I_i) \tag{1}$$

Here, $\mathcal{L}$ is typically a photometric loss, such as an L_2 difference in pixel intensities or a perceptual metric, possibly augmented by regularization on θ to enforce smoothness or physical plausibility. Under rich input images I_i that contain shading, specular highlights, and texture, physically based renderers R supply ample gradients for recovering detailed surface geometry and material properties.

In contrast, when each I_i is reduced to a binary silhouette (i.e. a mask indicating only surface visibility) the renderer effectively becomes

$$\mathcal{R}_{\text{sil}}(C, \theta; x) = \mathcal{V}(x; C, \theta) = \begin{cases} 1 & \text{if ray from } x \text{ hits surface,} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

and the photometric loss collapses to matching two indicator functions. In this setting all shading and texture-based gradients vanish,

leaving only silhouette-edge signals, and the inverse problem becomes severely underconstrained. This loss of local surface information motivates our shift to a global, silhouette-driven correspondence framework.

2.2 Differentiable Rendering

Differentiable rasterization methods aim to back-propagate through the discrete visibility and interpolation steps of the GPU pipeline. Soft Rasterizer [Liu et al. 2019] "fuzzes" triangle boundaries by probabilistically blending contributions of all primitives to each pixel, thereby enabling gradients to propagate across occluded edges and into the interior of silhouettes arxiv.org. Building on approximate antialiasing, nvdiffrast [Laine et al. 2020] provides a high-performance, GPU-accelerated suite of modules, each equipped with exact derivative formulas, and serves as a state-of-the-art baseline for silhouette-driven optimization. Despite their efficiency, these rasterization-based gradients are inherently local and concentrate along silhouette edges; when source and target masks are disjoint, they vanish, impeding convergence.

To overcome discontinuities at geometric boundaries, physically-based differentiable rendering (PBDR) methods operate on the continuous light-transport integral rather than the rasterization step. Edge-sampling approaches [Li et al. 2018] explicitly sample paths at scene discontinuities, such as edges where visibility changes, to directly estimate boundary integrals. However, selecting and sampling these edges requires complex data structures and precomputation. Warped-Area Sampling [Bangaru et al. 2020] sidesteps explicit boundary detection by applying the divergence theorem to convert the boundary integral into an interior area integral, yielding an unbiased estimator of the derivative without edge enumeration.

Our approach replaces the narrow, edge-only gradients of rasterization with a dense, globally-aware motion field that provides wholistic coverage like physically-based methods.

2.3 Silhouette-Based Reconstruction

Silhouettes have long served as a foundation for 3D shape inference in computer vision. Early work on visual hulls reconstructs a coarse approximation of object geometry by intersecting back-projected silhouette cones from multiple calibrated views [Laurentini 1994]. This concept was extended through space carving, which refines the volume estimate by iteratively carving away voxels inconsistent with each silhouette, yielding exact volumetric reconstructions under suitable conditions [Kutulakos and Seitz 1999]. To capture finer detail, subsequent methods have combined silhouettes with sparse texture or shading cues, enabling more accurate image-based shape fitting [Matusik et al. 2002]. More recent advances leverage learned shape priors to perform single-view reconstruction: by constraining the deformation of a generic 3D template to match a silhouette, these approaches recover plausible geometry even from a single outline [Wu et al. 2018]. Collectively, these studies demonstrate that silhouette contours encode strong global shape information, an insight we exploit in our own iterative mesh-deformation framework.

2.4 Optimal Transport

Optimal transport provides a global, mathematically rigorous framework for computing correspondences between two mass distributions by minimizing the total "effort" required to transport one distribution into another. In the continuous Kantorovich formulation [Villani et al. 2008], this effort is expressed as

$$\mathcal{E}(\pi) \;=\; \int_X \int_Y c(x,y),\pi(x,y)\,dx\,dy, \tag{3}$$

where $c(x,y)$ denotes the cost of moving a unit of mass from x to y, and the transport plan $\pi(x,y) \geq 0$ must satisfy the marginal constraints

$$\int_X \pi(x,y)\,dx = \nu(y), \qquad \int_Y \pi(x,y)\,dy = \mu(x), \tag{4}$$

ensuring that π indeed redistributes the source density μ into the target density ν.

Directly solving the variational problem $\min_\pi \mathcal{E}(\pi)$ is computationally intractable at image scale. A practical remedy is to add an entropic penalty, yielding the strictly convex objective [Cuturi 2013]:

$$\begin{aligned}\mathcal{E}_\epsilon(\pi) = &\int_X \int_Y c(x,y)\,\pi(x,y)\,dx\,dy \\ &- \epsilon \int_X \int_Y \pi(x,y)\big(\log \pi(x,y) - 1\big)\,dx\,dy,\end{aligned} \tag{5}$$

which can be solved efficiently via the Sinkhorn–Knopp matrix-scaling iterations.

For binary silhouettes, we treat each mask as a probability density over the image plane and employ the p-Wasserstein distance,

$$W^p(\mu,\nu) = \min_\pi \mathcal{E}(\pi), \tag{6}$$

where $c(x,y) = \|x-y\|^p$. The resulting regularized transport plan yields smooth, dense correspondences even under large silhouette displacements, making it ideal for our silhouette-driven mesh deformation pipeline.

3 METHOD

In this section we present our framework for shape metamorphosis, which deforms a source triangle mesh to match a collection of target silhouettes. We treat the source mesh as a shape evolving in time, much like a continuous optimization, by repeatedly estimating per-vertex velocities and integrating these displacements. Concretely, each iteration consists of two stages: velocity estimation and velocity integration. This two-stage abstraction applies equally to our optimal-transport-based method and to differentiable-rasterization baselines, and we use this property for direct comparisons within a unified pipeline.

3.1 Velocity Estimation

Our velocity estimation formulates a p-Wasserstein flow between the source geometry and the target silhouette in image space. We represent the source and target as discrete point clouds $\mathcal{S}(\theta)$ and $\mathcal{T}(I)$, then compute vertex velocities by lifting gradients of the transport cost to the mesh parameters:

$$\frac{\partial \theta}{\partial t} = (\mathfrak{L} \circ \nabla) W^p\left(\mathcal{S}(\theta), \mathcal{T}(I)\right) \tag{7}$$

Here, we first compute the gradient of the p-Wasserstein distance with respect to the positions of the discrete source samples in $\mathcal{S}(\theta)$. These per-sample gradients capture how each point should move to minimize the transport cost. We then apply a lifting operator $\mathfrak{L}$ that aggregates and interpolates these sample-level motions onto the mesh vertices, yielding a consistent velocity field over θ.

To construct $\mathcal{T}(I)$, we map the center of each pixel within the binary mask to its 2D image-space coordinate, as illustrated in Figure 1. Formally, we denote this discrete distribution by

$$p \sim \mathfrak{D}(I), \quad \mathcal{T}(I) = \mathfrak{D}(I), \tag{8}$$

where $\mathcal{D}(I)$ is the empirical distribution over the set of transformed pixel centers inside the silhouette.

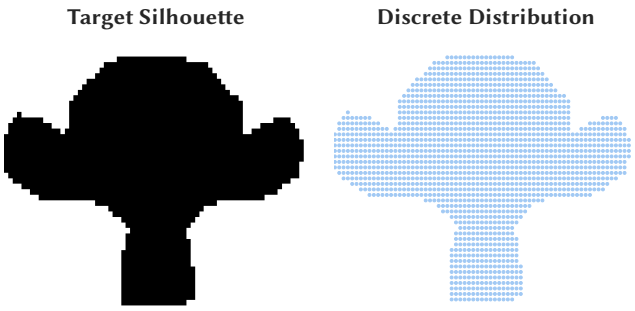


Fig. 1. **Forming the target distribution.** We convert a binary silhouette (left) into a discrete point distribution by mapping the center of each foreground pixel to its screen-space coordinates (center).

Defining $\mathcal{S}(\theta)$ and $\mathfrak{L}$ is more flexibility, since we can leverage various geometric cues to approximate the source's silhouette distribution. In the next section, we explore these sampling strategies.

3.2 Distribution Sampling Strategies

We introduce three distinct strategies for sampling the source distribution. For each strategy, we specify (1) how to construct the discrete distribution $\mathcal{S}(\theta)$ from the mesh geometry and (2) how to define the lifting operator $\mathfrak{L}$ that maps sample-level motions to vertex velocities. Note that, concretely, $\mathfrak{L}$ can be represented as a sparse weighting operator: for a discrete sample p and a mesh vertex v, the function $\mathfrak{L}(p, v)$ returns the weight by which the motion of p contributes to the velocity of v.

Pixels. Considering that we aim to match a target silhouette for each view, we find it reasonable to adopt an analysis-by-synthesis approach similar to differentiable rendering by comparing the silhouette of the source geometry. To apply optimal transport, we convert the source silhouette into a discrete distribution in the same manner as for the target distribution (see Figure 1). Formally, we define

$$\mathcal{S}(\theta) = \mathfrak{D}(\mathcal{R}_{\text{sil}}(C, \theta)) \tag{9}$$

The lifting operator must produce a complete set of vertex velocities from the given pixel samples. Our procedure first determines, for each pixel sample p, the set of triangles within its visual cone. We then transfer the backprojected pixel velocities to world-space triangle velocities. These triangle velocities are scattered to vertices and averaged over all adjacent triangles. Mathematically, the lifting operation is given by

$$\mathfrak{L}(p, v) = \frac{1}{|T|} \sum_{t \in T} \mathcal{V}(p; t) \cdot C^{-T} \cdot \nabla_p W^1 \tag{10}$$

where T denotes the set of triangles adjacent to the vertex v, $\mathcal{V}(p; t)$ is the indicator function for triangle-cone visibility, C^{-T} backprojects image-space velocities to world space, and $\nabla_p W^1$ is the sample gradient of the Wasserstein distance for $p = 1$.

Branching away from this silhouette comparison approach, the subsequent sampling strategies exploit explicit geometric information from the mesh.

Vertices. A natural choice for explicit geometric sampling is the set of mesh vertices. We project each vertex v_i under the camera C to form the discrete distribution

$$\mathcal{S}(\theta) = \{C \cdot v_i\}. \tag{11}$$

Since each sample corresponds one-to-one with a vertex, the lifting operator simplifies: the image-space gradient $\nabla_p W^2$ at the projected position $p = C \cdot v_i$ is directly mapped back to that vertex via the inverse-transpose of C. Formally, for any sample pp arising from vertex vv,

$$\mathfrak{L}(p, v) = C^{-T} \cdot \nabla_p W^2, \tag{12}$$

where we use the 2-Wasserstein distance ($p = 2$) in the transport computation.

Centroids. An alternative strategy uses triangle centroids to capture surface-area–weighted geometry. We project each triangle centroid c_i under C to obtain

$$\mathcal{S}(\theta) = \{C \cdot c_i\}. \tag{13}$$

To lift the image-space gradient at a centroid sample p to vertex velocities, one defines a weighting over the vertices adjacent to the corresponding triangle. Using a uniform combinatorial weight, the lifting operator for sample p and vertex v becomes:

$$\mathfrak{L}(p, v) = \frac{1}{|T|} C^{-T} \cdot \nabla_p W^2 \tag{14}$$

Here again the 2-Wasserstein distance is employed, and each centroid-derived gradient is backprojected and equally distributed among its triangle's vertices.

3.3 Velocity Integration

After estimating per-vertex velocities, we integrate them to update the mesh geometry. Rather than using a fixed time step, we adopt advanced adaptive integrators inspired by gradient-based optimizers. For example, Adam [Kingma and Ba 2014] can accelerate convergence, but it is prone to introducing artifacts in shape optimization. To mitigate this, we employ an intrinsically regularized optimizer [Nicolet et al. 2021] that filters raw vertex velocities through the mesh's differential coordinates. This filtering enforces smooth, coherent deformations and prevents the instabilities typically observed with standard adaptive methods.

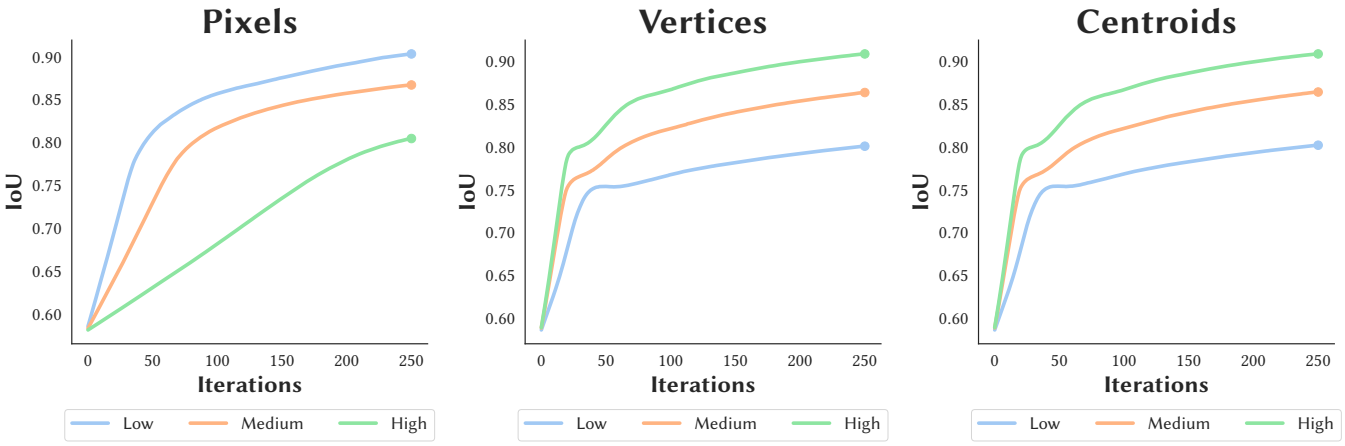


Fig. 2. **Scaling properties of sampling strategies.** We plot IoU over iterations for low, medium, and high resolution settings: for the pixel-centric strategy (silhouette rendered at 128 × 128, 256 × 256, 512 × 512), and for the vertex- and centroid-centric strategies (base mesh, one-level tessellated mesh, two-level tessellated mesh). The pixel-centric approach converges faster at lower silhouette resolution, likely because coarser pixels influence more vertices per update. In contrast, vertex- and centroid-centric strategies achieve faster convergence as mesh resolution increases, reflecting finer gradient information at higher geometric detail.

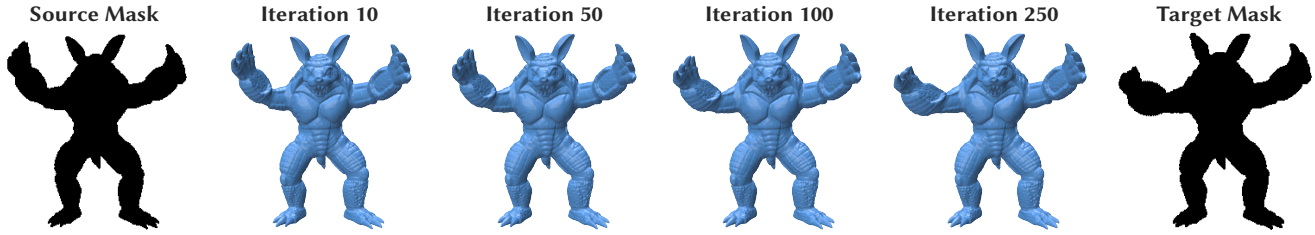


Fig. 3. **Silhouette-based animation.** A user edits the silhouette of a source mesh in image space, here the Armadillo's left arm, and our optimal-transport-based method deforms the mesh to match the modified silhouette. Over successive iterations, the arm moves smoothly and coherently to align with the edited outline, producing artifact-free animation.

4 RESULTS

We evaluate our silhouette-only inverse rendering approach on a variety of shape metamorphosis and surface animation scenarios. In these experiments, we demonstrate both qualitative and quantitative improvements over rasterization-based baselines.

4.1 Shape Metamorphosis

To evaluate our method for shape metamorphosis, we generate synthetic scenes in which both source and target are specified as triangle meshes. For each scenario, we render the target mesh from a chosen viewpoint to produce a binary silhouette mask at a fixed resolution of 1024 × 1024. During deformation, we render the evolving source mesh from the same view at each iteration and compute the Intersection over Union (IoU) between source and target silhouettes as our primary metric.

Baselines. We compare our optimal-transport-based method to a differentiable rasterization baseline using nvdiffrast [Laine et al. 2020]. Both approaches fit into the same iterative framework, comprising of velocity estimation and integration through the identical regularized optimizers. For the baseline, velocity estimation is obtained through vertex gradients directly from the rasterizer backpropagation.

Figure 4 demonstrates the metamorphosis from the Armadillo model to Suzanne. Although all methods converge to a high IoU, indicating successful silhouette matching, the intermediate deformations exhibit markedly different behaviors. The differentiable rasterization baseline often contracts or collapses regions of the mesh that initially do not overlap the target silhouette (for example, the limbs of the Armadillo), leading to local artifacts and loss of structural detail. In contrast, our optimal-transport-based approach induces globally coherent transformations, moving all parts of the geometry in a way that better preserves the original topology and overall shape. Moreover, IoU convergence is substantially accelerated: our method surpasses 0.8 IoU within 50 iterations, outperforming the baseline in both speed and qualitative fidelity.

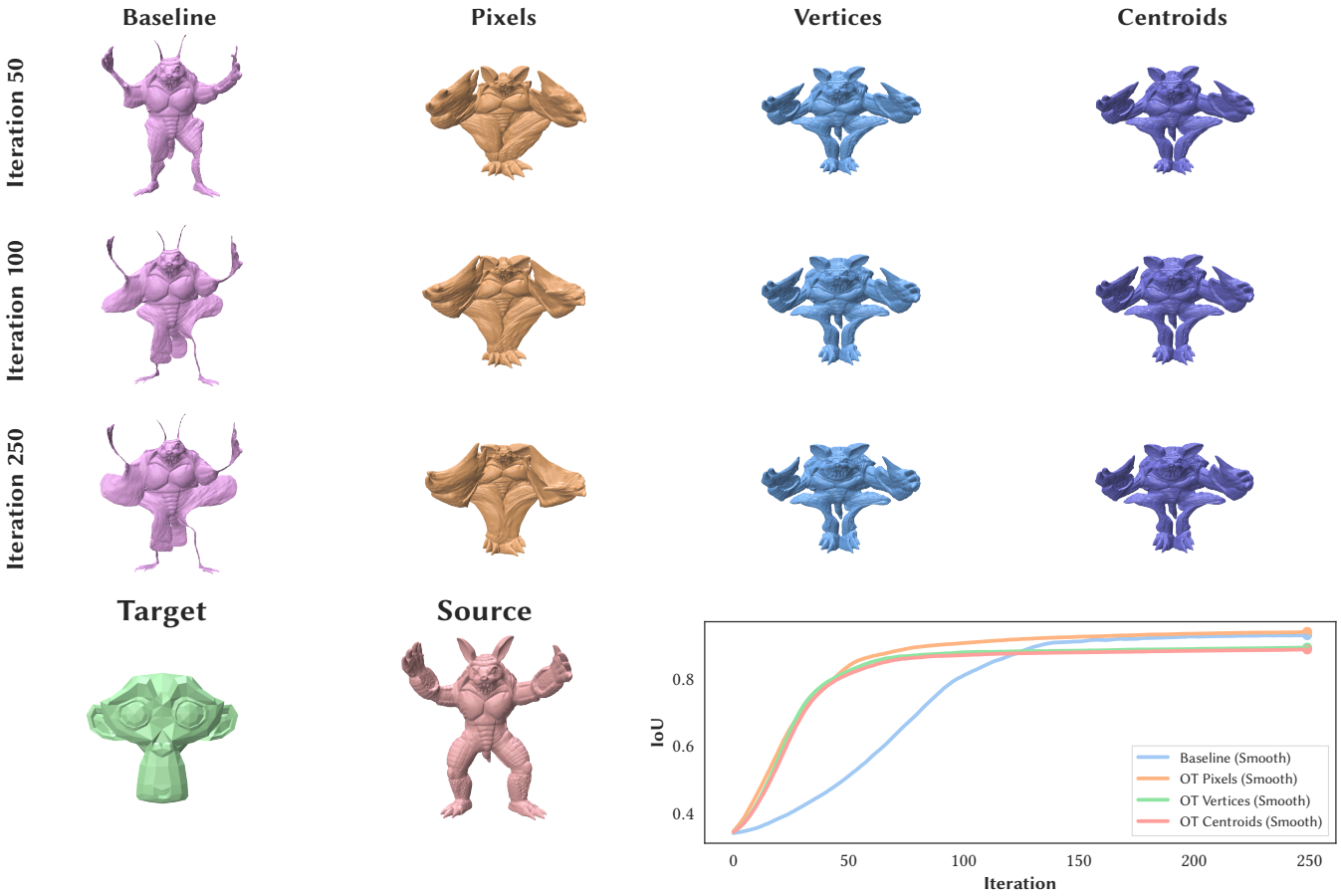
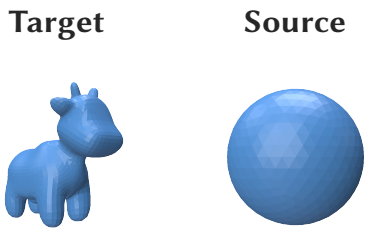


Fig. 4. **Evolution of geometry during shape metamorphosis.** While both differentiable rasterization and our optimal-transport-based methods achieve high IoU, the rasterization baseline often collapses non-overlapping regions. This can be seen above with the limbs of the Armadillo, which are rapidly compressed into thin, self-intersecting geometry. On the other hand, our optimal-transport-based approaches yields globally coherent deformations that better preserve topology and converges faster.

Integrators. Our optimal-transport-based method produces globally coherent velocity estimates even before applying the regularized integrator. Figure 5 illustrates the deformation of the Igea mesh into Lucy for each sampling strategy, both with and without the regularized integrator. Without gradient filtering, each strategy exhibits distinct artifacts: the pixel-based strategy shows jagged silhouette edges, indicating sensitivity to rendering resolution, while the vertex- and centroid-based strategies behave similarly but the vertex sampling often incurs pronounced issues such as self-intersections. After applying the regularized integrator's filtering, these artifacts are effectively eliminated, yielding smooth, stable deformations across all strategies.

Ablations. We examine how each sampling strategy's deformation quality depends on its resolution parameters. In the pixel-centric strategy, performance varies with the silhouette rendering resolution; in the vertex- and centroid-centric strategies, it depends on the geometric resolution of the source mesh. To evaluate these dependencies, we measure IoU over iterations for low, medium, and high settings in Figure 2. For the pixel-centric strategy, we render silhouettes at 128×128 (low), 256×256 (medium), and 512×512 (high). For the geometry-based strategies, we begin with the base mesh shown below as the low-resolution configuration; we obtain the medium-resolution mesh by applying one level of standard triangle tessellation to the low mesh, and the high-resolution mesh by tessellating the medium mesh.



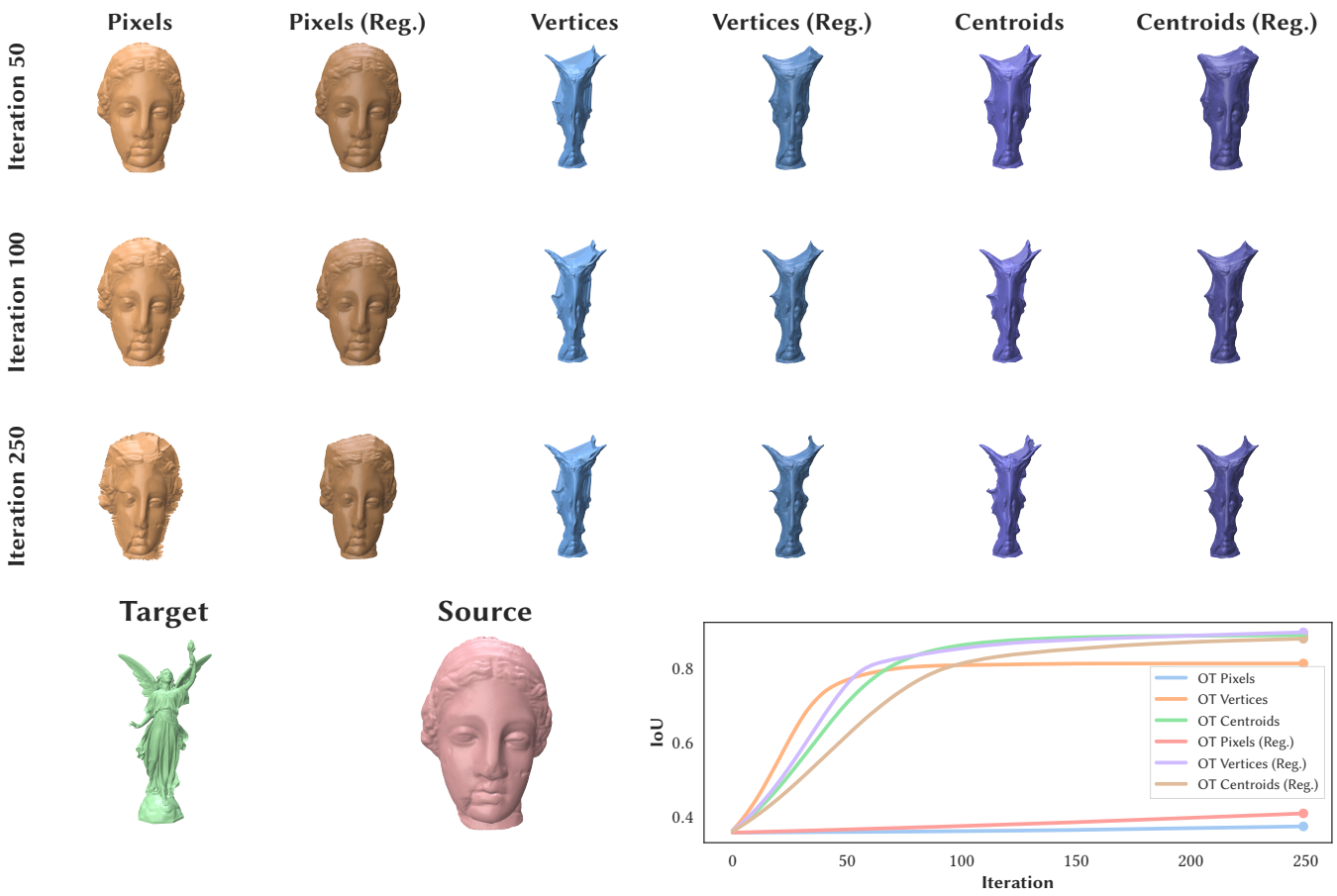


Fig. 5. **Impact of a regularized integrator.** Without gradient filtering, pixel-based sampling yields jagged silhouette edges and vertex-based sampling can introduce self-intersections, while centroid-based sampling shows moderate artifacts. Applying the regularized integrator removes these artifacts across all strategies, resulting in smooth, stable deformations of Igea into Lucy.

Counterintuitively, the pixel-centric strategy converges more rapidly at lower silhouette resolutions. We hypothesize that coarser pixel grids cause each boundary pixel to influence a larger set of vertices when lifted, effectively producing stronger update signals early in optimization. In contrast, the vertex- and centroid-centric strategies behave as expected: higher mesh resolution yields finer-grained gradient information and thus faster convergence. These ablations confirm that the choice of sampling resolution has a significant impact on deformation speed and suggest practical guidelines for balancing computational cost against convergence behavior in our silhouette-driven metamorphosis.

4.2 Surface Animation

Our optimal-transport-based method produces smooth deformations well suited for silhouette-driven animation. In this workflow, a user captures the silhouette of a source geometry, edits the silhouette in an image editor using operations such as warping or cage-based deformations, and then inputs the modified silhouette into our algorithm to generate the corresponding surface deformation. Figure 3 exemplified this process by deforming the silhouette of the Armadillo's left arm. Over successive iterations, our method incrementally moves the arm to align with the edited silhouette, yielding a coherent and artifact-free animation. We find that our pixel-centric sampling strategy is most appropriate for this task.

5 DISCUSSION

Our silhouette-only inverse rendering framework produces dense, globally coherent deformations, yielding faster IoU convergence and better structural preservation than boundary-only rasterization baselines. In shape metamorphosis experiments, it avoids collapse of non-overlapping regions. In silhouette-driven animation, it delivers smooth, artifact-free updates that faithfully track edited outlines.

Topology changes. The current pipeline assumes fixed mesh connectivity and cannot by itself create or remove handles or components when silhouettes imply topological differences. Incorporating topology-aware operations, such as mesh surgery or level-set updates guided by silhouette cues and topological sensitivity, would

be necessary to handle cases where the target silhouette requires new holes or disconnected parts. Defining robust criteria to trigger such modifications from 2D outlines remains nontrivial.

Silhouette ambiguity. Silhouettes inherently discard depth and fine surface detail, so distinct 3D shapes may share the same outline. As a result, our optimization may converge to any geometry within the silhouette equivalence class. To mitigate this ambiguity, one could integrate additional constraints or priors (e.g., photometric cues, learned shape priors) alongside silhouette transport, steering the solution toward more plausible 3D configurations without compromising the core optimal-transport-driven deformation.

Multi-view integration. Although our experiments focus on single-view settings, extending to multiple silhouettes requires aggregating image-space transport signals from different cameras in an occlusion-aware manner. One must weight contributions by visibility or confidence, and reconcile potentially conflicting updates to ensure consistent 3D deformation. A unified multi-view Wasserstein formulation or adaptive per-view weighting scheme could allow simultaneous silhouette alignment across viewpoints while preserving the coherent, global deformation behavior observed in single-view scenarios.

REFERENCES

Sai Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased Warped-Area Sampling for Differentiable Rendering. *ACM Trans. Graph.* 39, 6 (2020), 245:1–245:18.

Marco Cuturi. 2013. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *Advances in Neural Information Processing Systems*, C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger (Eds.), Vol. 26. Curran Associates, Inc.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). https://api.semanticscholar.org/CorpusID:6628106

K.N. Kutulakos and S.M. Seitz. 1999. A theory of shape by space carving. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Vol. 1. 307–314 vol.1. https://doi.org/10.1109/ICCV.1999.791235

Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. 2020. Modular Primitives for High-Performance Differentiable Rendering. *ACM Transactions on Graphics* 39, 6 (2020).

A. Laurentini. 1994. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16, 2 (1994), 150–162. https://doi.org/10.1109/34.273735

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable Monte Carlo Ray Tracing through Edge Sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 37, 6 (2018), 222:1–222:11.

Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. 2019. Soft Rasterizer: A Differentiable Renderer for Image-based 3D Reasoning. *The IEEE International Conference on Computer Vision (ICCV)* (Oct 2019).

Wojciech Matusik, Hanspeter Pfister, Remo Ziegler, Addy Ngan, and Leonard McMillan. 2002. Acquisition and rendering of transparent and refractive objects. In *Proceedings of the 13th Eurographics Workshop on Rendering* (Pisa, Italy) *(EGRW '02)*. Eurographics Association, Goslar, DEU, 267–278.

Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. 2021. Large Steps in Inverse Rendering of Geometry. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). https://doi.org/10.1145/3478513.3480501

Gabriel Peyré and Marco Cuturi. 2019. Computational Optimal Transport. *Foundations and Trends in Machine Learning* 11, 5-6 (2019), 355–607.

Cédric Villani et al. 2008. *Optimal transport: old and new*. Vol. 338. Springer.

Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T. Freeman, and Joshua B. Tenenbaum. 2018. Learning Shape Priors for Single-View 3D Completion and Reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*.