Computational Biomimetics of Winged Seeds

QIQIN LE, Shanghai Qi Zhi Institute, China
JIAMU BU, Tsinghua University, China
YANKE QU, Peking University, China
BO ZHU, Georgia Institute of Technology, United States of America
TAO DU, Tsinghua University, China and Shanghai Qi Zhi Institute, China

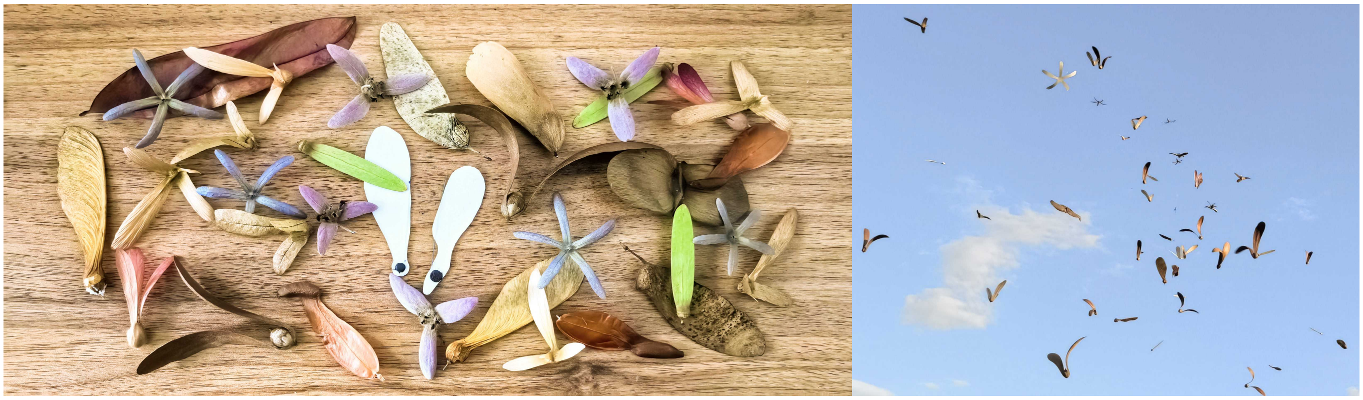

Fig. 1. *Natural and handmade winged-seeds*: Left: two white fabricated seeds (Sec. 9.5) surrounded by colorful natural seeds. Right: a photo of fabricated and natural seeds flying in the sky.

We develop a computational pipeline to facilitate the biomimetic design of winged seeds. Our approach leverages 3D scans of natural winged seeds to construct a bio-inspired design space by interpolating them with geodesic coordinates in the 3D diffeomorphism group. We formulate aerodynamic design tasks with probabilistic performance objectives and adapt a gradient-free optimizer to explore the design space and minimize the expectation of performance objectives efficiently and effectively. Our pipeline discovers novel winged seed designs that outperform natural counterparts in aerodynamic tasks, including long-distance dispersal and guided flight. We validate the physical fidelity of our pipeline by showcasing paper models of selected winged seeds in the design space and reporting their similar aerodynamic behaviors in simulation and reality.

CCS Concepts: • **Computing methodologies → Shape modeling**.

Additional Key Words and Phrases: Computational design, winged seeds

ACM Reference Format:
Qiqin Le, Jiamu Bu, Yanke Qu, Bo Zhu, and Tao Du. 2024. Computational Biomimetics of Winged Seeds. *ACM Trans. Graph.* 43, 6, Article 180 (December 2024), 13 pages. https://doi.org/10.1145/3687899

Authors' addresses: Qiqin Le, Shanghai Qi Zhi Institute, China, Shanghai, leqq@sqz.ac.cn; Jiamu Bu, Tsinghua University, China, Beijing, bjm21@mails.tsinghua.edu.cn; Yanke Qu, Peking University, China, Beijing, ykqu@stu.pku.edu.cn; Bo Zhu, Georgia Institute of Technology, United States of America, Atlanta, bo.zhu@gatech.edu; Tao Du, Tsinghua University, China, Beijing and Shanghai Qi Zhi Institute, China, Shanghai, taodu@tsinghua.edu.cn.

0730-0301/2024/12-ART180
https://doi.org/10.1145/3687899

1 INTRODUCTION

Winged seeds are biological structures that can passively fly through the air by utilizing lift generated by their rotating wing shapes. There are wide varieties of winged seeds in nature, including those from maple, tulip, and elm trees, among others, spanning a broad spectrum of geometric features and flying mechanisms (Fig. 1). The aerodynamic behavior of these winged seeds represents an intriguing natural phenomenon that has inspired extensive scientific inquiries. Through physical experiments, mathematical analyses, 3D scanning, and numerical simulations, scientists from different disciplines have dedicated their efforts to developing a comprehensive understanding of the aerodynamic interaction among winged seeds' shape, material, mass distribution, and surrounding airflow. This interaction underpins the stable, long-range flight behavior of wing structures heavier than air, providing profound inspiration for the biomimetic design of passive artificial flyers. For instance, the design principles derived from these studies have inspired the development of active (e.g., monocopters [Win et al. 2022]) and passive (e.g., airborne microrobots [Kim et al. 2021]) flying machines, among many other microscopic aerodynamic devices (e.g., RoboBee [Chen et al. 2019; Ma et al. 2013]).

In this paper, we aim to develop a computational pipeline to facilitate the biomimetic design of winged seeds. Our overarching goal is to create a design, simulation, and optimization pipeline that automatically discovers novel biomimetic winged seed designs with aerodynamic performances surpassing their natural counterparts. Our computational biomimetic design pipeline starts by collecting 55 3D-scanned winged seed examples from 14 diverse species with various morphology. One key challenge is constructing from the collected data a bio-inspired design space that explores intrinsically embedded natural principles. We address this problem by adopting geodesic coordinates in the 3D diffeomorphism group to interpolate and extrapolate between collected winged seeds. The shape space leads to a simple and fast generation of shapes based on geometric priors that can rigorously ensure shape design constraints such as smoothness and non-penetration. Furthermore, it effectively propagates the biomimetic principles underpinning natural winged seeds in an intuitive and controllable manner.

A second challenge in exploring bio-inspired, high-performing winged seed designs is the dependency on simulating the intricate and turbulent physical interaction between air and seeds, making standard search or optimization strategies sensitive to initial simulation conditions and computationally expensive. We propose formulating design tasks as expectation minimization problems with a probabilistic performance objective, encouraging optimized designs to have robust behavior under perturbation. We build a sample-efficient optimizer for our design tasks by adapting one of the state-of-the-art gradient-free optimizers [Fischer and Ritschel 2023] to accommodate probabilistic objectives, leading to the efficient discovery of novel winged seeds with robust performance.

Our pipeline differs significantly from existing biomimetic and computational design pipelines. Traditional biomimetic design processes heavily rely on human designers' domain-specific expertise and trial-and-error efforts. In contrast, our pipeline computationally and automatically discovers novel winged seeds from natural priors. Existing computational design pipelines for airborne objects (e.g., multicopters [Du et al. 2016], kites [Martin et al. 2015], and paper planes [Umetani et al. 2014]) primarily base their design space on a grammatical description of parametric shapes supported by computer-aided-design (CAD) tools [Schulz et al. 2014]. Compared with these works, we accommodate a novel design space spanned by organic surfaces that balances expressing flexible organic shapes and conforming to natural design principles.

To showcase the efficacy of our pipeline, we optimize the aerodynamic performance of winged seeds in bio-inspired tasks, including long-distance dispersal and directional flight. We report findings of various novel winged seed designs that are geometrically complex and intersection-free, characterized by the organic features typical of winged seeds, and outperform their natural bases in simulation. We validate the physical fidelity of our results by creating paper models for a selection of digital winged seeds and reporting qualitatively similar aerodynamic behaviors between simulation and reality.

We summarize our technical contributions as follows:

1. We provide a dataset of 55 digital winged-seed models scanned from 14 species of natural winged seeds with diverse morphology;
2. We develop a design space spanned by natural winged seeds and geodesic coordinate interpolation to facilitate new shape synthesis;
3. We present a set of design tasks formulated with probabilistic performance objectives that enable effective gradient-free optimization of winged seed designs;
4. We showcase a family of novel winged seed designs discovered by our computational pipeline with improved performance over their natural counterparts.

2 RELATED WORKS

Winged seed morphology. Winged seeds (Samaras) represent a fascinating adaptation in the evolutionary biology of plants, facilitating efficient seed dispersal through wind mechanisms [Nathan et al. 2002]. These botanical structures, characterized by their wing-like appendages, are found across a diverse range of species, notably in genera such as Acer (maples), Fraxinus (ashes), and Ulmus (elms) [Bonner and Karrfalt 2008]. Unlike animals that actively flap their wings, samaras passively adjust their orientation in the air solely through their unique shapes and mass distributions, interacting with the air to produce various intriguing descent and flight paths [Minami and Azuma 2003]. The aerodynamic properties of those samaras increase their dispersal distance from the parent plant, enhancing their chances of successful germination and colonization of new habitats. Many previous studies [Azuma and Yasuda 1989; Kim et al. 2021; Lentink et al. 2009; Vincent et al. 2006] have focused on helping humans understand existing samaras' structure and flight principles. Our work, however, takes a different perspective by exploring novel designs that do not exist in nature through computational methods, with the potential for superior performance in specific tasks.

Computational design. Computational design leverages computational methods for exploring, simulating, and optimizing design objects to improve their performance objectives [Matusik and Schulz 2019]. Examples include computational design pipelines for furniture [Schulz et al. 2014], interlocking structures [Chen et al. 2022], swimming characters [Ma et al. 2021], legged rigid robots [Ha et al. 2018], and plushies [Bern et al. 2017], to name a few. Closest to this work are computational design pipelines for airborne objects, e.g., unmanned flying vehicles [Du et al. 2016; Xu et al. 2019], airfoils [Viquerat et al. 2021], paper planes [Umetani et al. 2014], and kites [Martin et al. 2015]. The design emphasis in these works is primarily on man-made objects characterized by parametric shapes employed in CAD tools. In contrast, our work focuses on biomimetic designs represented by organic surfaces. Beyond the computational design of airborne objects, the design space most similar to ours is DiffAqua [Ma et al. 2021], which interpolates a small number of organic volumetric shapes (fishes) based on their Wasserstein distances. We differ from DiffAqua in that we collect real-world nature designs and that we focus on organic and open surfaces, which are more sensitive to self-collisions and warrant a different approach.

Generative shapes. A critical step in our pipeline is constructing a shape space spanned from a small number of seed designs. We classify related studies into two categories: geometry-based and learning-based shape generation.

Geometry-based approaches exploit geometric metrics to generate novel shapes from given seeds. A typical example is the functional map method [Huang et al. 2014; Ovsjanikov et al. 2012; Rustamov et al. 2013; Weng et al. 2013], which interpolates shapes based on their correspondences. However, as different winged seeds have various morphology, establishing a high-quality correspondence map is nontrivial. Another approach is diffeomorphic pattern matching, which involves finding a series of time-dependent diffeomorphisms that continuously transform a template geometry into a target geometry [Dupuis et al. 1998; Trouvé 1998]. This method has been thoroughly explored within the frameworks of Lie group theory and geometric mechanics [Holm et al. 2009; Younes 2010]. Practically, it is often structured as a variational problem, specifically as large deformation diffeomorphic metric mapping (LDDMM) [Beg et al. 2005]. This approach aims to identify a path that minimizes the metric between the template and the target while ensuring the path's regularity. Diffeomorphic pattern matching can accommodate a diverse array of geometric objects, including point sets [Joshi and Miller 2000], curves [Glaunes et al. 2008], distributions [Glaunes et al. 2004], currents [Vaillant and Glaunes 2005], and varifolds [Charon and Trouvé 2013; Kaltenmark et al. 2017]. The transformation paths derived in this manner align with the principles of morphology and evolution in biological structures [Holm et al. 2009; Thompson et al. 2010], sparking broad interest in the application across various medical imaging studies [Jiang et al. 2018; Miller et al. 2015b; Pan et al. 2016; Tang et al. 2015]. Our work explores the application of LDDMM in computational design and optimization of organic forms.

Learning-based approaches leverage data distribution to facilitate shape generation. A representative method in this category is variational autoencoders (VAE) for meshes [Ranjan et al. 2018; Tan et al. 2018; Yuan et al. 2020], which are capable of generating diverse and realistic 3D shapes, including faces and human bodies. Mesh VAEs typically expect meshes with homogeneous topologies, limiting their applicability to our collection of winged seeds with diverse shapes. Another family of approaches applies learning-based methods to signed distance fields (SDF) [Park et al. 2019] or unsigned distance fields (UDF) [Chibane et al. 2020; Christiansen et al. 2023; Juhl et al. 2021]. However, reconstructing surfaces from learned UDFs is prone to nonsmoothed results as the learning model does not strictly conform to properties required for UDFs, e.g., Eikonal equations. Finally, several recent works propose to learn descriptors for functional maps and apply flow-based methods for shape matching and interpolation [Eisenberger and Cremers 2020; Eisenberger et al. 2021; Huang et al. 2019; Litany et al. 2017]. One notable example is NeuroMorph [Eisenberger et al. 2021], which combines a feature extractor network to learn functional descriptors for shape correspondence and applies to non-isometric pairs from different object categories.

3 OVERVIEW

Our pipeline begins with collecting and scanning real-world winged seeds (Sec. 4) to generate a dataset of 55 digital winged seeds represented as triangle meshes (Fig. 3). We then construct a shape space spanned from the collected data with linear interpolation among their geodesic coordinates (Sec. 5). This shape space provides a low-dimensional parametrization of winged seeds with a compact encoding of natural design principles embedded in the collected data.

Next, for each shape sampled from the design space, we perform fluid-solid simulation to evaluate its performance under gravity and aerodynamic forces from surrounding air (Sec. 6). We choose the Lattice Boltzmann method (LBM) [Li et al. 2023] with cut-cell discretization on a translating grid as our simulation method because it balances the computational speed and physical fidelity needed in our design pipeline.

The last step in our pipeline is discovering high-performance winged seeds in the design space via numerical optimization (Sec. 7). As the fluid-solid simulation is sensitive to initial conditions, we perform multiple simulations under random initial conditions to estimate its performance expectation, which serves as the true optimization objective. The simulation sensitivity inhibits directly applying gradient-based optimization in our design tasks, and we adopt gradient-free optimization [Fischer and Ritschel 2023] to build a neural network surrogate model that fits the probabilistic objective landscape. Finally, we minimize the surrogate model to explore winged seed designs in the shape space. Fig. 2 summarizes our pipeline.

4 DATA COLLECTION

We created a dataset of 55 winged seeds across 14 species, showcasing diverse geometric features, e.g., size, flatness, and number of wings (Fig. 3). We scanned both sides of the seeds using a handheld laser scanner and obtained surface triangle meshes. To improve mesh quality, we used geodesic coordinate interpolation in Sec. 5.3, resulting in non-intersecting and smooth surface meshes. Our final dataset consists of 49 high-quality digital winged seeds with an average of 25K vertices (Fig. 3). The dataset is not only useful for our computational design pipeline but also beneficial for any related studies on modeling, simulating, and evaluating winged seeds.

5 GEOMETRIC DESIGN

While existing methods can span a generative shape space from a dataset (Sec. 2), many of these methods encounter inherent technical difficulties when applied to exploring organic and open surfaces in our problem setting. The most related approach from previous literature is the Wasserstein-distance interpolation [Ma et al. 2021], which is more suited to 3D voxel representation of volumetric shapes instead of open surfaces. Machine learning-based methods [Christiansen et al. 2023; Park et al. 2019; Ranjan et al. 2018] typically struggle to enforce geometric constraints within a neural network model, e.g., smoothness and no self-intersection. Violating these geometric constraints can compromise the numerical stability of fluid-solid simulations. Given these limitations, we adopt a method that utilizes geodesic coordinates in the 3D diffeomorphism group for shape interpolation, providing a robust and expressive shape space for exploration.

5.1 Background: Construction of a Linear Shape Space

The entire set of diffeomorphisms on $\mathbb{R}^3$ (smooth bijections) can be considered a naive shape space. By selecting an initial template shape, such as a triangular mesh disk named S, each diffeomorphism g corresponds to a shape $g(S)$. Broadly speaking, all diffeomorphisms on $\mathbb{R}^3$ constitute an infinite-dimensional Lie group G'

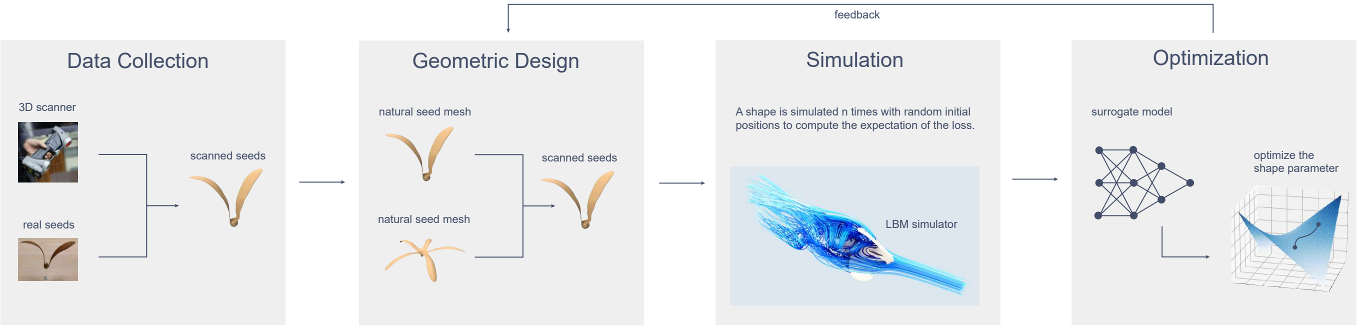

Fig. 2. *Overview of our pipeline:* Our pipeline consists of the following steps: a) 3D scanning of real samaras to generate thin-shell triangular meshes, b) identifying the corresponding geodesic coordinates in our linear shape space and solving for the corresponding shape based on given weights, c) performing fluid-solid coupling simulations using LBM and cut-cell methods, and calculating the empirical variance of the original loss from multiple simulations under random initial conditions as the true loss. d) We then perform local random perturbations of the shape parameters, calculate multiple loss values, and use a neural network to fit the loss function locally. Finally, shape parameters are optimized using gradient-based methods such as Adam.

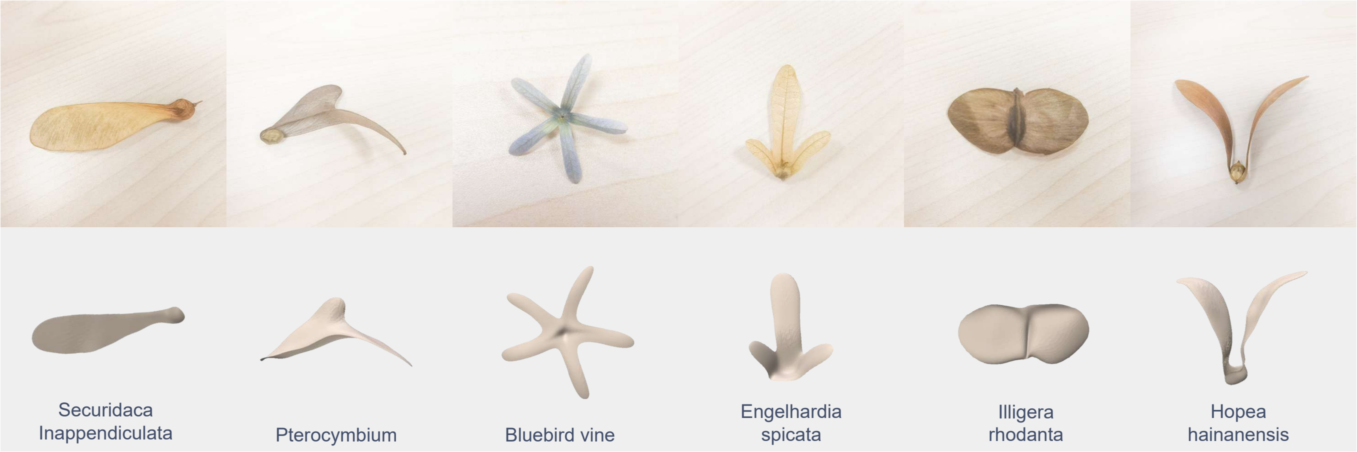

Fig. 3. *Dataset gallery (Sec. 4):* Top row: a selection of six representative winged seeds with various morphology; Bottom row: the corresponding surface triangle meshes generated from smoothing the 3D scanned data with geodesic coordinates (Sec. 5.3).

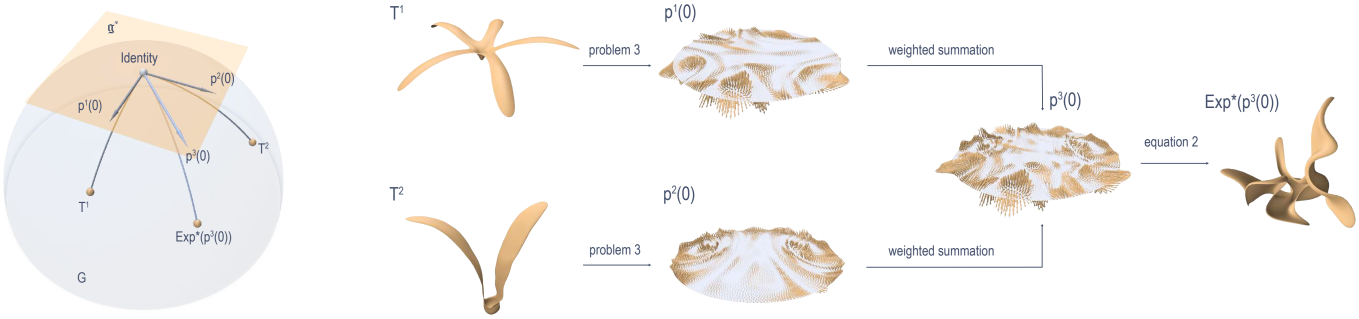

Fig. 4. *Interpolation of geodesic coordinates:* Given two seed meshes T^1 and T^2, we compute their corresponding geodesic coordinates $\boldsymbol{p}^1(0)$ and $\boldsymbol{p}^2(0)$, which are 3D vector arrays defined on the vertices of S (the light disk). Since these are vectors in $\mathfrak{g}^*$, we can directly add them with weights, e.g., $(0.7, 0.7)$, to obtain $\boldsymbol{p}^3(0)$. Finally, we derive $\mathrm{Exp}^*(\boldsymbol{p}^3(0))$ as our new shape.

[Holm et al. 2009], which is both a group and a manifold, with the identity transformation as its identity element.

Directly studying this infinite-dimensional, nonlinear manifold is highly challenging; therefore, we shift our focus to its Lie algebra. The Lie algebra $\mathfrak{g}'$ of G' is the tangent space at the identity element, consisting of all smooth vector fields $\boldsymbol{u}$ in $\mathbb{R}^3$. The exponential map from the Lie algebra to the Lie group, $\mathrm{Exp} : \mathfrak{g}' \to G'$, maps $\boldsymbol{u}$ to the point reached by moving along the geodesic for unit time with $\boldsymbol{u}$ as the initial velocity. This exponential map locally defines a bijective correspondence between G' and $\mathfrak{g}'$. As a linear space, $\mathfrak{g}'$ possesses more favorable statistical properties [Holm et al. 2009].

However, $\mathfrak{g}'$ remains an infinite-dimensional space of all vector fields on the entire $\mathbb{R}^3$. As we are only interested in studying deformations of S, we only consider vector fields that can be represented as an array of 3D vectors $\boldsymbol{p}_1, \cdots, \boldsymbol{p}_N$ defined on the locations $\boldsymbol{q}_1, \cdots, \boldsymbol{q}_N$ of the N vertex of S, denoted as momentum:

$$\boldsymbol{u}(\cdot) = \sum_{i=1}^{N} K(\boldsymbol{q}_i, \cdot)\boldsymbol{p}_i, \tag{1}$$

where $K(\cdot,\cdot) : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ is a kernel function. Moreover, the geodesic equation on G can now be explicitly written in its Hamiltonian formulation with momentum [Miller et al. 2015a], after lifting both $\boldsymbol{p}_i$ and $\boldsymbol{q}_i$ to be time-dependent:

$$\begin{cases} \dfrac{d\boldsymbol{q}_i(t)}{dt} = \boldsymbol{u}(t, \boldsymbol{q}_i(t)), \\ \dfrac{d\boldsymbol{p}_i(t)}{dt} = -\displaystyle\sum_{j=1}^{N} \nabla_1 K(\boldsymbol{q}_i(t), \boldsymbol{q}_j(t))\boldsymbol{p}_i(t)^\top \boldsymbol{p}_j(t). \end{cases} \tag{2}$$

We define our shape space as the linear space comprised of all 3D vector arrays defined on the vertices of S, denoted by $\mathfrak{g}^*$, which can be viewed as a subspace of $\mathfrak{g}'$. We also denote the subgroup of G' whose (co)tangent space is $\mathfrak{g}^*$ as G. Given any 3D vector arrays in $\mathfrak{g}^*$, we may find its corresponding shape in G by solving Eqn. (2) with the array as its initial momentum. We denote the initial momentum as the geodesic coordinates of the shape. We define $\mathrm{Exp}^* : \mathfrak{g}^* \to G$ as the function that maps a set of geodesic coordinates to its corresponding shape and Log^* as its inverse.

The collection forms an expansive and high-quality set. Importantly, because a diffeomorphism is by definition a bijective smooth mapping, self-intersection—meaning two points being mapped to the same location—is not allowed. Therefore, these shapes rarely intersect themselves, except in cases caused by spatial discretization errors, and can maintain smoothness. The smoothness of the shape can also be adjusted by the kernel function. Notably, the process of morphological evolution and growth in organisms is consistent with Eqn. (2), where the initial momentum can be considered the trend of evolution.

5.2 Preserving topology

The assumption of diffeomorphism implies all shapes in the shape space constructed above share the same topology (disk in our case). This simple topology is sufficient for exploring a wide range of meaningful applications and is consistent with most natural seed shapes.

5.3 Geodesic Coordinates of a Scanned Seed

Given a scanned mesh T^1, we aim to find its geodesic coordinates $\boldsymbol{p}^1(0) := \mathrm{Log}^*(T^1)$. However, since T^1 may not belong to G due to potentially noisy scan data and error-prone mesh processing, we instead solve an optimization problem

$$\min_{\boldsymbol{p}^1(0)} e(\mathrm{Exp}^*(S), T^1) + \frac{1}{2}\sum_{i,j} K(\boldsymbol{q}_i(0), \boldsymbol{q}_j(0))\boldsymbol{p}_i^1(0)^\top \boldsymbol{p}_j^1(0), \tag{3}$$

with $\boldsymbol{p}_i(t)$s and $\boldsymbol{q}_i(t)$s satisfying Eqn. (2) and the error term e penalizing the discrepancy between two shapes $\mathrm{Exp}^*(S)$ and T based on their position and normal differences from each face [Kaltenmark et al. 2017]. Note that this way of finding geodesic coordinates is consistent with the shooting method of LDDMM [Miller et al. 2015a] and updates the scanned mesh to a similar one in G but with improved smoothness and free from self-intersection due to diffeomorphism.

5.4 Interpolation with Geodesic Coordinates

We end this section by extending the geodesic coordinates to interpolation between multiple shapes $T^1, \cdots, T^n$. One option is to compute the Karcher mean [Zimmermann 2021] of these shapes, but it is inefficient since each interpolation during optimization typically requires solving a problem similar to Prob. (3), which is time-consuming. Therefore, we apply normal interpolation [Zimmermann 2021]. We first obtain for each shape T^i their corresponding geodesic coordinate $\boldsymbol{p}^i(0)$ by solving Prob. (3). Note that this step needs to be performed only once and can be considered as an additional data preprocessing step. Since geodesic coordinates are all 3D vector arrays defined on S, we can view them as vectors in $\mathfrak{g}^*$. We add them with assigned weights to obtain a new geodesic coordinate $\boldsymbol{p}^{\mathrm{new}}(0)$ and the interpolated shape $\mathrm{Exp}^*(\boldsymbol{p}^{\mathrm{new}}(0))$ (Fig. 4).

6 SIMULATION

For each shape obtained from the shape space, we simulate its falling trajectory under gravity and aerodynamic forces from the surrounding air. As a computational design pipeline frequently simulates intermediate designs, we favor simulators with low computational cost to avoid prohibitively slow optimization. However, the simulator needs to maintain moderate physical fidelity to capture the intricate fluid-solid coupling critical for the aerodynamic performance of winged seeds. With these constraints, we choose to adopt an LBM simulator [Li et al. 2023] due to its balance between computing speed and physics plausibility.

To better accommodate the simulation of free-falling winged seeds, we adapt the LBM simulator with a standard translating grid centered at the seed. This way, the simulator focuses on the air-seed interaction in the local region and drastically reduces computational cost when the seed travels long distances. When handling solid-fluid interactions, after determining the combined effects of gravity and aerodynamic forces with the surrounding air on the linear and angular acceleration of the seed, only the angular acceleration is applied to the seed, while the linear acceleration is treated as an inertial force and applied to the air as an external force. This treatment allows us to simultaneously capture the accurate falling trajectory of the seed and observe its rotation within the translating grid.

7 OPTIMIZATION

Optimizing the aerodynamic performance of winged seeds faces several challenges. First, the intricate coupling between air and seeds often leads to intrinsically chaotic aerodynamics sensitive to the initial configuration of the fluid-solid system. This challenge persists after numerically discretizing the system and simulating it with LBM. The explicit nature and weak coupling in the LBM simulator in our pipeline necessitate small time steps that exacerbate the sensitivity to perturbations.

Therefore, our pipeline deviates from optimizing *deterministic* performance objectives as in several existing computational design pipelines of airborne objects [Du et al. 2016; Martin et al. 2015; Umetani et al. 2014]. Instead, we propose optimizing the *expectation* of performance objectives sampled with random initial configurations (positions, orientations, and velocities). This design decision is also partially inspired by the profound usage of random rewards in reinforcement learning (RL) [Sutton and Barto 2018] to optimize robust controllers in computational design [Xu et al. 2019], and we adopt this idea to optimize organic shapes.

Formally, let $f(\boldsymbol{s}, \boldsymbol{\alpha})$ be the performance objective evaluated with an initial configuration $\boldsymbol{\alpha}$ of the winged seed generated from interpolation weights $\boldsymbol{s}$. The evaluation of f involves executing the black-box LBM simulator and encodes users' preferences on the simulation results, e.g., establishing desired spinning rate or flight trajectories. Given a probability distribution of $\boldsymbol{\alpha}$ characterized by its probability density function p, we propose minimizing the expectation $\mathbb{E}_{\boldsymbol{\alpha}\sim p}[f(\boldsymbol{s}, \boldsymbol{\alpha})]$:

$$\min_{\boldsymbol{s}} \int p(\boldsymbol{\alpha}) f(\boldsymbol{s}, \boldsymbol{\alpha}) d\boldsymbol{\alpha}, \tag{4}$$

and employ gradient-free methods for optimization due to the non-smooth and chaotic nature of the objective (Sec. 8.4). Specifically, we adopt and modify the surrogate model in ZeroGrads [Fischer and Ritschel 2023] to obtain a local approximation of $\mathbb{E}_{\boldsymbol{\alpha}\sim p}[f(\boldsymbol{s}, \boldsymbol{\alpha})]$:

$$l(\boldsymbol{s}_i, \boldsymbol{\phi}) = \int \kappa(\boldsymbol{s} - \boldsymbol{s}_i)\left(\left[\int p(\boldsymbol{\alpha}) f(\boldsymbol{s}, \boldsymbol{\alpha}) d\boldsymbol{\alpha}\right] - h(\boldsymbol{s}, \boldsymbol{\phi})\right)^2 d\boldsymbol{s}, \tag{5}$$

where $h(\cdot, \boldsymbol{\phi})$ is a neural network surrogate model with parameters $\boldsymbol{\phi}$, κ a zero-centered Gaussian kernel, and $\boldsymbol{s}_i$ a given shape parameter. The loss l encourages the surrogate model to approximate the local behavior of $\mathbb{E}_{\boldsymbol{\alpha}\sim p}[f(\boldsymbol{s}, \boldsymbol{\alpha})]$ around $\boldsymbol{s}_i$ with the region of interest controlled by the covariance of κ, a tune-able hyperparameter in ZeroGrads.

We apply gradient-based optimizers to train h with l as the training loss. To derive $\nabla_{\boldsymbol{\phi}} l$, we note that $\int p(\boldsymbol{\alpha}) d\boldsymbol{\alpha} = 1$ and

$$\begin{aligned} l(\boldsymbol{s}_i, \boldsymbol{\phi}) &= \int \kappa(\boldsymbol{s} - \boldsymbol{s}_i)\left(\left[\int p(\boldsymbol{\alpha}) f(\boldsymbol{s}, \boldsymbol{\alpha}) d\boldsymbol{\alpha}\right] - h(\boldsymbol{s}, \boldsymbol{\phi})\right)^2 d\boldsymbol{s} \\ &= \int \kappa(\boldsymbol{s} - \boldsymbol{s}_i)\left(\int p(\boldsymbol{\alpha})(f(\boldsymbol{s}, \boldsymbol{\alpha}) - h(\boldsymbol{s}, \boldsymbol{\phi})) d\boldsymbol{\alpha}\right)^2 d\boldsymbol{s} \\ &= \int \kappa(\boldsymbol{s} - \boldsymbol{s}_i)\mathbb{E}^2_{\boldsymbol{\alpha}\sim p}[f(\boldsymbol{s}, \boldsymbol{\alpha}) - h(\boldsymbol{s}, \boldsymbol{\phi})] d\boldsymbol{s}. \end{aligned} \tag{6}$$

We further notice that $\mathbb{E}^2[f(\boldsymbol{\alpha}) - h]$ can be written as $\mathbb{E}[(f(\boldsymbol{\alpha}) - h)^2] - \mathrm{Var}[f(\boldsymbol{\alpha}) - h]$ (we dropped irrelevant variable names for brevity). Since $\mathrm{Var}[f(\boldsymbol{\alpha}) - h] = \mathrm{Var}[f]$ as h is a constant offset subtracted from f in the variance, and $\mathrm{Var}[f]$ is irrelevant to $\boldsymbol{\phi}$, we can safely replace $\mathbb{E}^2[f(\boldsymbol{\alpha}) - h]$ with $\mathbb{E}[(f(\boldsymbol{\alpha}) - h)^2]$ in l without affecting $\nabla_{\boldsymbol{\phi}} l$:

$$\begin{aligned} l &= \int \kappa(\boldsymbol{s} - \boldsymbol{s}_i)\mathbb{E}_{\boldsymbol{\alpha}\sim p}[(f(\boldsymbol{s}, \boldsymbol{\alpha}) - h(\boldsymbol{s}, \boldsymbol{\phi}))^2] d\boldsymbol{s} \\ &= \int\int \kappa(\boldsymbol{s} - \boldsymbol{s}_i) p(\boldsymbol{\alpha})(f(\boldsymbol{s}, \boldsymbol{\alpha}) - h(\boldsymbol{s}, \boldsymbol{\phi}))^2 d\boldsymbol{\alpha} d\boldsymbol{s} \end{aligned} \tag{7}$$

$$\frac{\partial l}{\partial \boldsymbol{\phi}} = \int\int 2p(\boldsymbol{\alpha})\kappa(\boldsymbol{s} - \boldsymbol{s}_i)\frac{\partial h(\boldsymbol{s}, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}(h(\boldsymbol{s}, \boldsymbol{\phi}) - f(\boldsymbol{s}, \boldsymbol{\alpha})) d\boldsymbol{\alpha} d\boldsymbol{s}. \tag{8}$$

Similar to ZeroGrads, the above definition enables a Monte-Carlo estimation based on jointly sampling $\boldsymbol{\alpha}$ and $\boldsymbol{s}$ from a probability density function $p(\boldsymbol{\alpha}, \boldsymbol{s})$:

$$\begin{aligned} \frac{\partial l}{\partial \boldsymbol{\phi}} &\approx \frac{1}{N}\sum_j \frac{2p(\boldsymbol{\alpha}_j)\kappa(\boldsymbol{s}_j - \boldsymbol{s}_i)}{p(\boldsymbol{\alpha}_j, \boldsymbol{s}_j)}\frac{\partial h(\boldsymbol{s}_j, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}(h(\boldsymbol{s}_j, \boldsymbol{\phi}) - f(\boldsymbol{s}_j, \boldsymbol{\alpha}_j)) \\ &= \frac{\partial}{\partial \boldsymbol{\phi}}\frac{1}{N}\sum_j \frac{p(\boldsymbol{\alpha}_j)\kappa(\boldsymbol{s}_j - \boldsymbol{s}_i)}{p(\boldsymbol{\alpha}_j, \boldsymbol{s}_j)}(h(\boldsymbol{s}_j, \boldsymbol{\phi}) - f(\boldsymbol{s}_j, \boldsymbol{\alpha}_j))^2 \end{aligned} \tag{9}$$

Specifically, if we consider independent samples $\boldsymbol{\alpha} \sim p(\boldsymbol{\alpha})$ and $\boldsymbol{s} \sim \kappa(\boldsymbol{s} - \boldsymbol{s}_i)$, we obtain a concise Monte-Carlo estimation of gradients:

$$\frac{\partial l}{\partial \boldsymbol{\phi}} \approx \frac{\partial}{\partial \boldsymbol{\phi}}\frac{1}{N}\sum_j (h(\boldsymbol{s}_j, \boldsymbol{\phi}) - f(\boldsymbol{s}_j, \boldsymbol{\alpha}_j))^2, \tag{10}$$

which can be used to train h. After h is optimized, we run gradient-based optimization with $\nabla_{\boldsymbol{s}} h$ to update the shape parameter. The iteration is repeated until convergence or a maximal allowable computational budget is reached.

Remarks. The above algorithm description extends the framework of ZeroGrads to probabilistic objectives, and we make a few remarks on our difference from ZeroGrads. While the probability density function p shares similarity with the smoothness kernel in ZeroGrads, the fact that $\int p = 1$ offers a probabilistic perspective (Eqn. (5)) for deriving the same joint sampling scheme originally proposed in ZeroGrads. It reveals that the rewriting actually adds a variance term into the original loss l, whose implication on variance reduction of the Monte-Carlo sampling may warrant further discussion. Furthermore, $\int p = 1$ also lifts the assumption of a unit-sized design space in the original ZeroGrads method. In particular, our result becomes identical to ZeroGrad when p is a uniform distribution.

8 EVALUATION

This section evaluates the technical method employed by each major step in our computational design pipeline. This section evaluates the technical methods in our computational design pipeline. We assess our shape space formulation by comparing it with typical generative models and traditional CAD. We then evaluate the physical fidelity of our LBM simulation by comparing its results with real-world winged seeds and paper seeds. Finally, we justify the use of gradient-free optimization with probabilistic performance objectives by analyzing the performance landscape.

8.1 Shape Generation Alternatives

We compare our shape space (Sec. 5) with alternative shape generation methods. We consider the following baselines: MeshPooling [Yuan et al. 2020], a VAE method on meshes; SSDF [Christiansen et al. 2023] and UDF [Juhl et al. 2021], encoder-decoder methods on distance fields; NeuroMorph [Eisenberger et al. 2021], an interpolation method based on shape correspondences. We evaluate these methods on the tasks of interpolating and extrapolating a small number of winged seeds and compare the generated shapes with metrics including smoothness, connectivity, and visual plausibility.

Training data. The size of our 3D-scanned winged-seed dataset is relatively small (49 seeds) for training standard learning-based baselines. To ensure the above baselines have reasonable performances, we augment our dataset into a dataset with 763 seeds generated by randomly deforming scanned winged seeds with the as-rigid-as-possible energy [Jacobson et al. 2018]. Note that the augmented seeds are used for training baselines only and are not included in the construction of our shape space. Despite accessing much less data, our method remains competitive and outperforms the baselines in the interpolation and extrapolation tasks.

Comparisons with MeshPooling. We train the MeshPooling model on the augmented dataset as above and obtain an encoder-decoder pair that maps mesh vertex features into a latent space. We interpolate and extrapolate winged seeds by first mapping them to the latent space. We then decode the linearly combined feature vectors in the latent space to obtain interpolation or extrapolation results.

We first consider an interpolation task among four representative winged seeds and report the results in Fig. 5 (left and middle). Note that these four basis seeds are slightly different in MeshPooling as the encoding-decoding module does not reconstruct the input shape perfectly. We observe that interpolations from MeshPooling are less smooth and tend to be flat, whereas geodesic shooting in our shape space generates smoother surfaces that preserve interesting biological structures from the four bases.

Next, we consider extrapolating the same four basis seeds with a uniform weight of 0.5, leading to a total weight sum of 2. The organic surface interpolated using our method (Fig. 6) is smooth and free from intersection by construction, whereas MeshPooling returns a non-smooth surface with noticeable self-intersections as indicated by the different coloring on the front and back side of the surface (Fig. 6). The result underscores the challenge of enforcing geometric constraints with a neural network model.

Comparisons with SSDF and UDF. We train SSDF and UDF with the augmented dataset. SSDF reconstructs an open surface by first getting a closed surface from its decoder and then removing the faces covering hole locations. The hole-removing method produces plausible open surfaces for garment datasets [Bhatnagar et al. 2019; Yin et al. 2006] where the surfaces are nearly closed. However, winged seeds are typically far from closed surfaces, and we report failure from SSDF when reconstructing training data through its encoder-decoder module (Fig. 6). Regarding UDF [Juhl et al. 2021], we follow the implementation in Christiansen et al. [2023] and train a Neural Dual Contouring (NDC) model [Guillard et al. 2022] to extract meshes from UDFs. Using it to interpolate existing seeds often leads to non-smooth or disconnected surfaces (Fig. 5 right), as the learned UDF is not under strict geometric constraints to be a valid distance field and may be prone to mesh extraction errors [Christiansen et al. 2023].

Comparisons with NeuroMorph. NeuroMorph finds correspondences between two input shapes and generates an interpolation flow, which limits its application to mostly interpolations or extrapolations between a pair of shapes. We trained NeuroMorph with a subset of 14 representative seeds from our scanned data until overfitting. We consider using NeuroMorph to interpolate between the top-left and top-right basis seeds in Fig. 5 and observe non-smooth results (Fig. 7) as a meaning correspondence between seeds with different numbers of wings is nontrivial to establish. Although previous works have demonstrated the success of NeuroMorph in interpolating shapes of mammals (e.g., camels and dogs in the SHREC20 dataset [Dyke et al. 2020]), it is more difficult to obtain the needed correspondences between winged seeds due to their substantially different morphology.

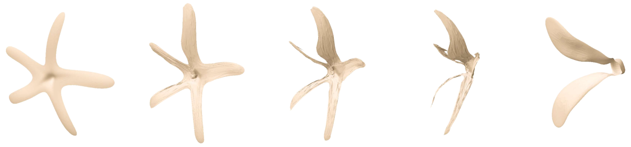

Fig. 7. *NeuroMorph:* Interpolating two scanned seeds (leftmost and rightmost) with evenly distributed weights in between.

8.2 Traditional Shape Space in Computational Design

By extracting a shape space from dozens of different winged seeds and employing a carefully constructed low-dimensional space, we harness strong natural priors in these high-performance designs optimized by natural evolution. In comparison with commonly used parametric spaces in traditional computational design, our shape space reduces reliance on manual design and avoids abundant sub-optimal samples in high-dimensional spaces. To illustrate this idea, we consider the following two spaces: 1) a shape space generated by randomly morphing the best sample in the dataset. The sample is deformed using a deformation cage with $6 \times 6 \times 6$ control points within a proper range; 2) a CAD-designed propeller template space with the number of blades, angle of attack, and major and minor axes as parameters. We compare the minimum and average falling loss, which we will define in Sec. 9.2, of 1000 random samples in our space and these spaces. As shown in Table 1, the mean and minimum loss values of our space are significantly lower.

Table 1. The minimal and mean losses of our space and two traditional design spaces. Lower losses are better.

	Ours	Deformation cage	Propeller
mean	**138.24**	153.56	146.34
min	**168.48**	204.17	237.55

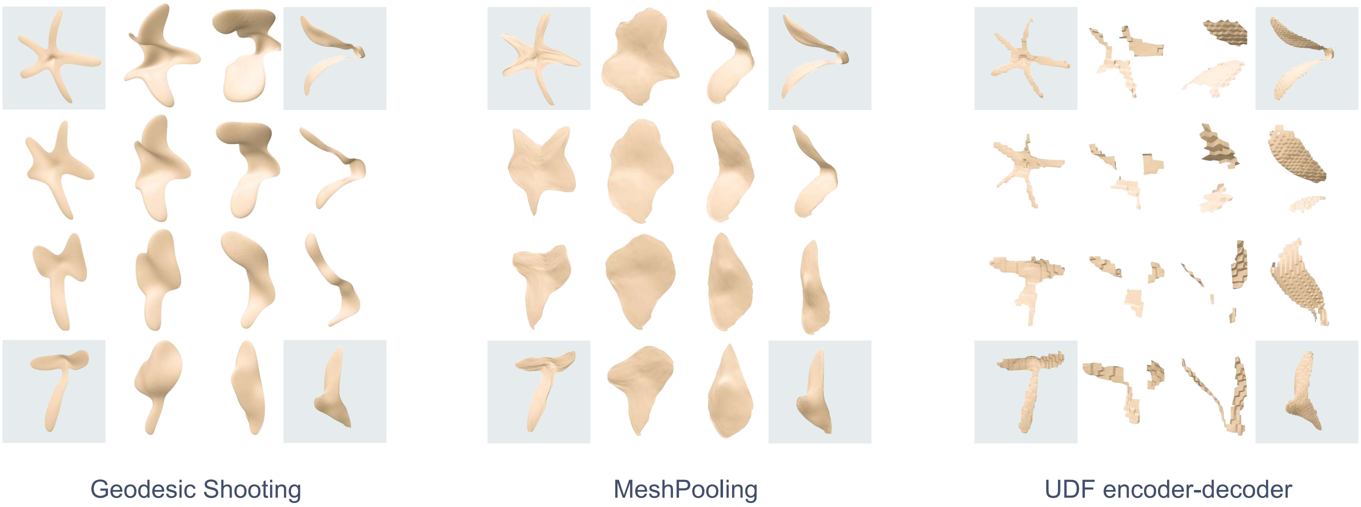

Fig. 5. Interpolation among four scanned seeds (blue boxes) with evenly distributed bilinear weights with the method in our shape space (left) and two baselines (middle and right).

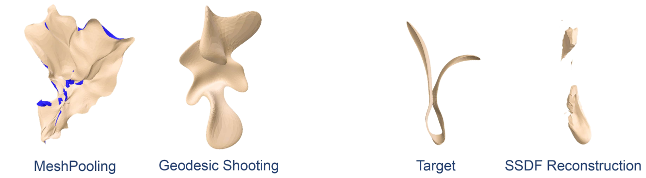

Fig. 6. Left: Extrapolating among four basis seeds from Fig. 5 with a uniform weight of 0.5 using two methods; Right: a target shape (middle right) sent to the trained SSDF and its reconstructed surface (rightmost).

8.3 Simulation Fidelity

We now investigate whether the motion patterns of real-world winged-seeds can be replicated in fluid-structure interaction simulations. We also examine whether these patterns can be reproduced in artificially created objects with similar shapes. We considered the seed shapes shown in Fig. 8, which are three types of seeds with different appearances but each capable of forming unique motion postures through their distinctive shapes. For example, type *c* descends in a stable oscillatory manner, type *b* spins symmetrically around its own center, while type *a* spins asymmetrically around the seed. It is noteworthy that these fascinating motion postures generally do not strongly correlate with the initial states; they simply appear when the seeds fall from a stationary position. We filmed the falling process of real seeds to show these motion postures. Subsequently, we used LBM to simulate on a similar scale and replicated these patterns. Finally, we approximated these shapes using a crude method, by merely using paper, wire, and modeling clay as weights, which allowed us to qualitatively reproduce these patterns. This proves that the qualitative laws of these interesting motion postures are quite robust, and are not affected by minor perturbations in shape and mass distribution. This observation encourages us to study the optimization methods for winged-seed shapes and apply them to enhance and optimize the aerodynamic properties of artificial objects made with rough, non-precision manufacturing techniques.

8.4 Optimization Choices

We simulate a randomly selected winged seed in our LBM implementation and consider a deterministic performance objective computed from the simulation output. By interpolating between two shapes, we obtain the objective landscape and slice it on one dimension of the initial condition (Fig. 9). The objective landscape is noisy and sensitive to perturbations of the input data. The intrinsically high-frequent variation of the objective comes as no surprise due to the chaotic nature of air-seed coupling in the fluid-solid environment. This result necessitates our proposal of probabilistic performance objectives to counteract high-frequent variations. More importantly, it justifies the usage of gradient-free optimizers in our pipeline, as gradient-based methods will inevitably suffer from poor gradient quality of the objective.

9 EXPERIMENTS

Our simulator was built using Taichi [Hu et al. 2019], with reference to Yang et al. [2022]. We utilized PyTorch and KeOps [Charlier et al. 2021] to develop our shape space and optimizer. Additionally, we employed Polyscope [Sharp et al. 2019] for visualizing select results.

9.1 Descent Experiment

In this study, we aim to optimize a specific seed shape to achieve a higher positional altitude after descending through air under the influence of gravity. Specifically, we solve 10,000 steps of LBM simulations on a 40×40 grid, and utilize the negative value of the mass center's z-coordinate (vertical direction) at the final position as our objective in optimization. We interpolate between eight initial seed shapes using barycentric interpolation (ensuring the sum of weights equals 1.0 and arbitrary weight combinations, with an additional constraint that the sum of weights must not exceed 2.0 to prevent

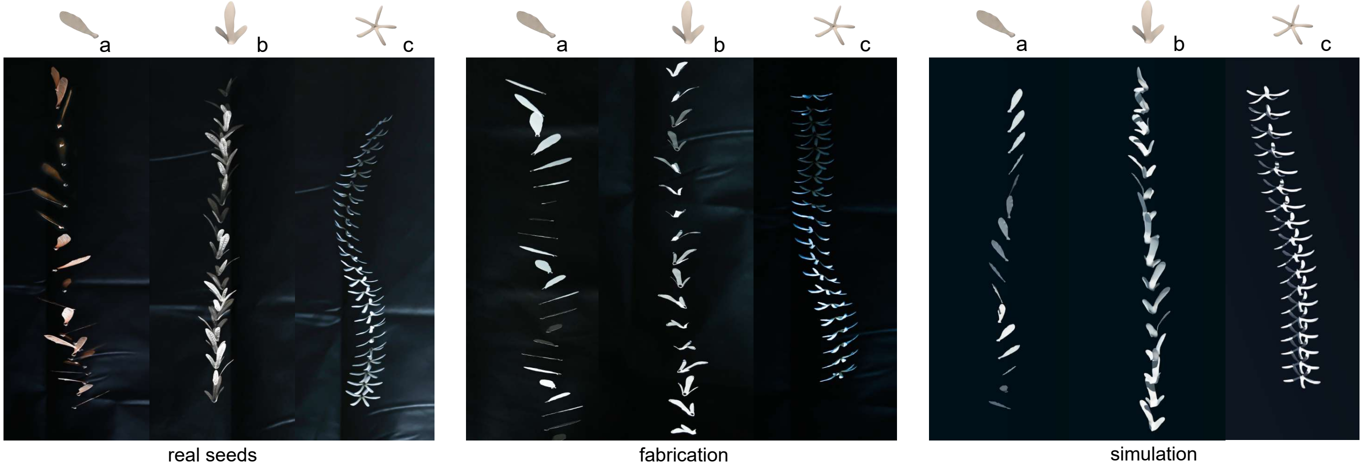

Fig. 8. *Simulation fidelity:* Flight trajectories of seeds with three different shapes. Left: trajectories of natural seeds. Middle: trajectories of fabricated seeds. Right: trajectories of simulated seeds. Real and fabricated seeds are recorded by stroboscopic photography.

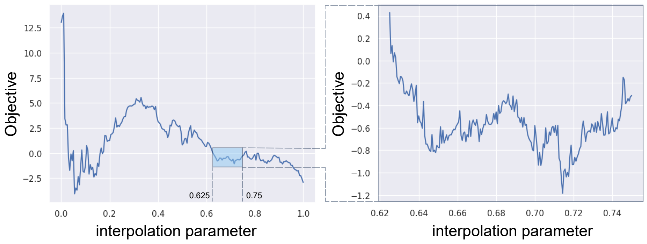

Fig. 9. *Objective landscape:* Left: a motion-based objective computed from simulating a winged seed as a function of linear interpolation parameter between two shapes; Right: a zoomed-in view of the objective in the blue box.

excessively large sizes. Among these, the shape with the lowest defined objective from the eight seeds is chosen as the initial shape for further optimization. Both arbitrary combination and barycentric methods successfully yield shapes with reduced objective values. Notably, the arbitrary combination approach achieved lower objectives due to its ability to explore a broader parameter space. Fig. 10 illustrates the shapes before optimization, as well as the optimized shapes using both barycentric and arbitrary combinations, and their flow profiles at the 10,000-th step. It reveals that the pre-optimized shape exhibits the highest descent rate, whereas the shape optimized via arbitrary combination shows the lowest, demonstrating the efficacy of our pipeline.

9.2 Rotational Acceleration Experiment

This experiment optimizes seed shapes for higher angular velocity along the z-axis (vertical direction). Similarly, we solve 10,000 steps of LBM simulations on a 40×40 grid, using the negative value of the angular velocity's z-component at the final step as the loss for optimization. We interpolate between eight initial seed shapes using settings similar to those in Sec. 9.1, performing both arbitrary combination and barycentric interpolation optimization tasks. We select the seed with the highest initial z-axis angular velocity as the starting shape. In each category of task, the optimization results in approximately double the angular velocity of the initial shape. Fig. 11 displays the optimized shapes and trajectories. The optimized shapes have a propeller-like appearance and motion trajectory, achieving one and a half more rotations than before optimization. We find that both categories of tasks produce similar trajectories and shapes, optimizing into forms resembling propellers to accelerate the rotation of the seeds.

9.3 Expectation Comparison Experiment

In the next experiment, we assess the differences between optimizing the original objective function directly and optimizing based on the expected value of the original objective. We launch seeds in the x-direction with an initial velocity of 10 meters per second, conducting simulations over 40,000 steps. The final mass center position of each simulation is recorded as (x, y, z), with z as the vertical axis. The original objective function is defined as $-x - z + |y|$, aiming to maximize the height and distance achieved by the seed while minimizing lateral deviation along the y-axis.

The optimization process begins with a randomly chosen seed shape. For one set of optimizations, we sample the objective function without altering the initial pose; for another set, we introduce perturbations to the three Euler angles of the seed's initial pose. Each angle is perturbed according to a uniform distribution in $[-0.6, 0.6]$.

The pre- and post-optimization shapes are tested by launching them in 12 random poses (Fig. 12). The initial seeds fail to sustain long flights, and the trajectory varies greatly between different poses. Optimization without expectation improves the distance the seeds can travel, yet the variability in their trajectories remains significant. Conversely, optimization with expectation produces a dart-like shape that not only flies significantly farther than the others but also shows remarkably consistent trajectories across different launching poses.

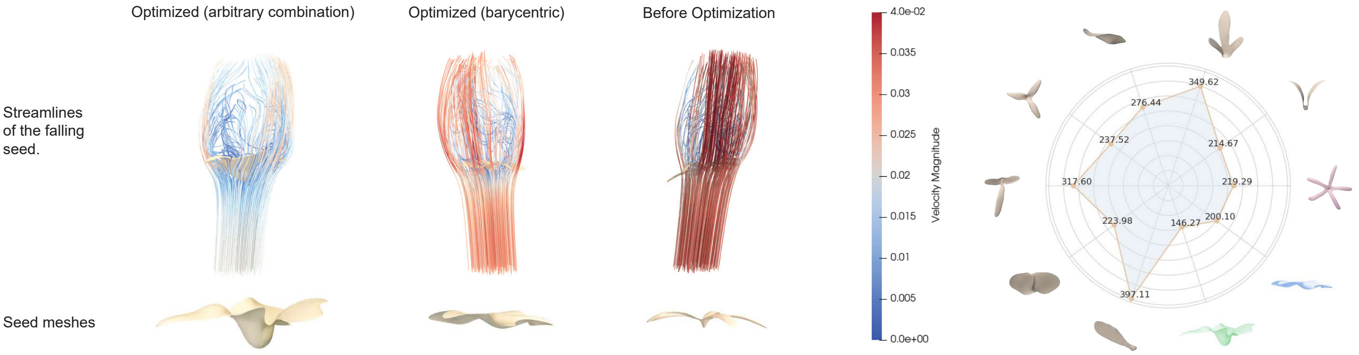

Fig. 10. *Descent experiment (Sec. 9.1)*: Left bottom: shapes of optimized and unoptimized seeds. Left top: streamlines of falling seeds, where colors represent magnitudes of velocities. Right: radar plot of eight natural seeds for interpolation and optimized seeds (pink: unoptimized, blue: barycentric, green: arbitrary combination), with radius being proportional to corresponding losses.

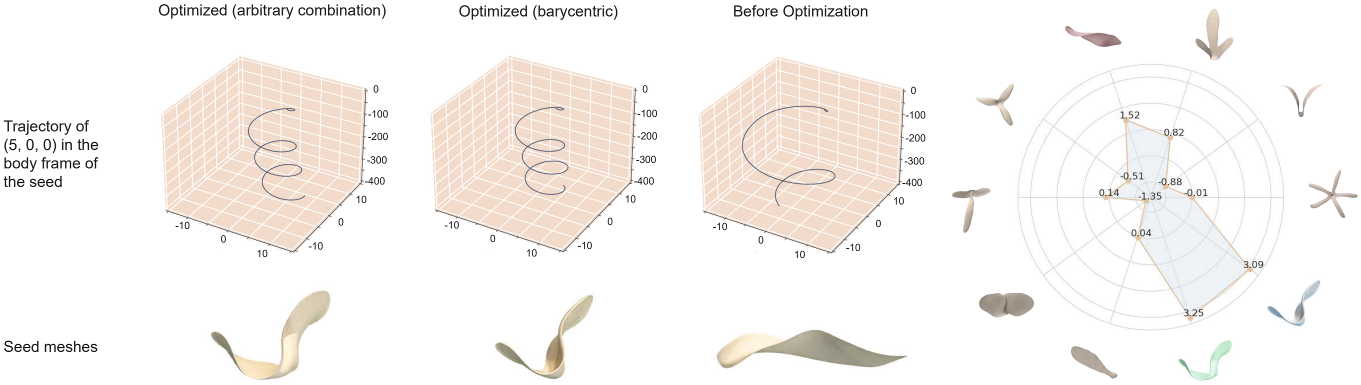

Fig. 11. *Rotational acceleration experiment (Sec. 9.2)*: Left bottom: shapes of optimized and unoptimized seeds. Left top: trajectories of (5,0,0) in body frames. Right: radar plot of eight natural seeds for interpolation and optimized seeds (pink: unoptimized, blue: barycentric, green: arbitrary combination), with radius being proportional to corresponding losses.

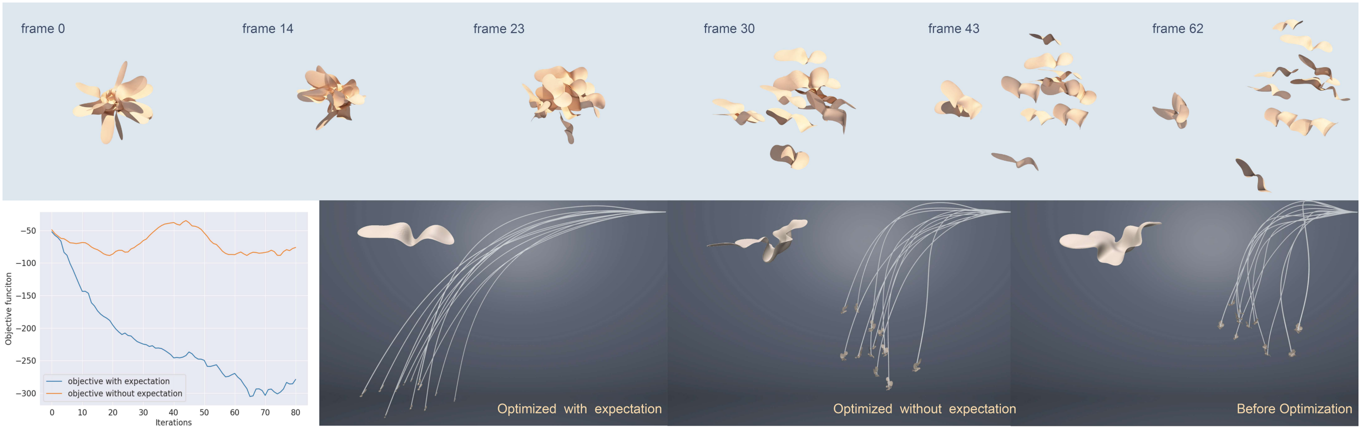

Fig. 12. *Expectation comparison experiment (Sec. 9.3):*Bottom row far left: objective curves of optimization. Bottom row middle left(right): shape and throwing trajectory of optimized seed with(out) expectation. Bottom row far right: unoptimized seeds. Top row: close-up of trajectories of seeds optimized with expectation.

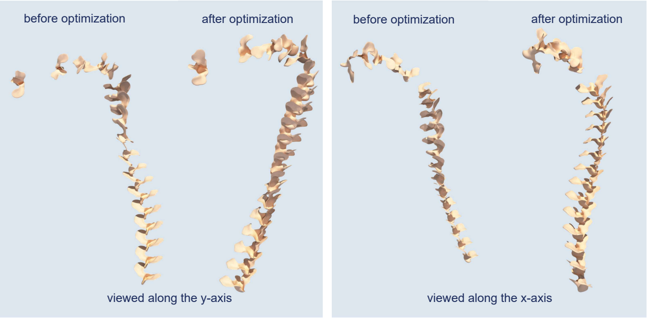

Fig. 13. *Regression experiment (Sec. 9.4):* Two lateral views (left and right) of the trajectories of winged seeds before and after optimization.

9.4 Regression Experiment

In this experiment, we launch a seed with an initial velocity of $(5, 0, 10)$ meters per second and an initial angular momentum of $(0, 1e3, 0)$, executing simulations over 10,000 steps. Inspired by the aerodynamics of a boomerang, the goal of this experiment is to encourage a seed design that returns to its original xy coordinates at the end of the simulation. We define the objective function as the distance from the z-axis at the final timestep. Fig. 13 demonstrates that through optimization, seeds which initially would have traveled straightforwardly or even diverged from the start point, can be optimized to drift back towards their starting xy coordinates. It can be observed that the optimized results, regardless of the viewing angle, are closer to the initial xy coordinates at the final step.

9.5 Fabrication Results

Finally, we validate the optimized winged shape design by fabricating and evaluating its performances in real-world experiments. Fabricating lightweight yet resilient organic surfaces is highly nontrivial, and our preliminary attempts with 3D printing and thermoforming resulted in several unsuccessful fabrications of optimized organic surfaces due to the relatively heavy or brittle fabrication material. Inspired by our preliminary validation of the simulation fidelity (Sec. 8.3), we use paper to create physical prototypes of our seed designs.

For the convenience of manufacturing, we conduct rotational acceleration experiments (Sec. 9.2) in the shape space of five types of single-winged samaras (Fig. 14 top row $[a]$). The wing density is set to match the parameters of the real material (double-layered paper). We add steel wires and tungsten putty to adjust the mass density during the simulation. We select a pair of seeds before and after optimization. The optimized seed differs mildly from its unoptimized counterpart ((Fig. 14 top row $[b]$) and is reported to rotate around 29% faster than the initial design in simulation (Fig. 14 bottom row $[a]$).

We then create paper models for both winged seeds and record their trajectories after release. We print their planar outlines on paper, and then cut them out with submillimeter errors (Fig. 14 top row $[c]$ and $[d]$). Finally, we assemble the seeds by combining the steel wires, tungsten putty, and paper pieces (Fig. 14 top row $[e]$). The seeds manufactured through the above process are flat, ignoring their 3D structure such as the ridged protrusion at the root of the optimized shape. We also create 3D versions using a rougher method (see supplemental video).

We record the falling trajectories of the paper models, both before and after optimization, using stroboscopic photography. The flash intervals are set to a fixed duration, allowing us to compare the period lengths by counting how many times the seeds were captured within each cycle. We repeat the experiment five times which shows about 27.6% - 50% increase in the rotational speed from the optimized design (Fig. 14 bottom row $[b]$, $[c]$, $[d]$, $[e]$, $[f]$). We also observe that the falling patterns of unoptimized shapes are more sensitive to uncontrollable factors such as air flow, hand posture, or shaking. In the experiments, two distinct falling patterns of unoptimized seeds appear randomly: one resembles the optimized seed ($[b]$, $[e]$), while the other seemed to lack aerodynamic advantages, resulting in a straight drop ($[c]$, $[d]$, $[f]$). The repeated real-world experiments imply that our optimized design performs consistently better than the unoptimized seed, which demonstrates the efficacy of our computational design pipeline.

10 CONCLUSIONS AND LIMITATIONS

We presented a computational biomimetic design pipeline for winged seeds characterized by organic surfaces. We collected and 3D-scanned a diverse set of natural winged seeds to build a high-quality mesh dataset of 49 wing seeds from 14 species. Our shape space spanned by these scanned seeds generated bio-inspired, smooth, and intersection-free winged seed samples with a low-dimensional parametrization from geodesic coordinate interpolation. Furthermore, we proposed probabilistic performance objectives and extended the state-of-the-art gradient-free optimizer to facilitate an efficient and effective discovery of novel winged seeds. Our pipeline produced winged seed designs surpassing their natural counterparts in tasks optimizing their aerodynamic performances.

Our work has several limitations. One major limitation is the lack of comprehensive physical validation of the optimized winged seeds due to their fabrication challenges. Therefore, we resorted to creating paper models, which prevented us from validating more organic and expressive winged seeds. Another limitation is the current shape space does not support topologically different winged seeds due to its dependency on a diffeomorphism group. Finally, the current design tasks do not explore performance objectives regarding agile aerodynamic motions. We leave these limitations as future work.

ACKNOWLEDGMENTS

Tsinghua authors thank Yuchen Sun, Jingyuan Hu, Professor Li Yi, Professor Weixin Huang, and Professor Xitong Liang for their suggestions on implementation, application, and fabrication of winged seed models. Tao Du acknowledges support from Tsinghua University and Shanghai Qi Zhi Institute. Bo Zhu thanks Side Effects Software Inc. (SESI) for providing a Houdini educational license.

REFERENCES

Akira Azuma and Kunio Yasuda. 1989. Flight performance of rotary seeds. *Journal of Theoretical Biology* 138, 1 (1989), 23–53.

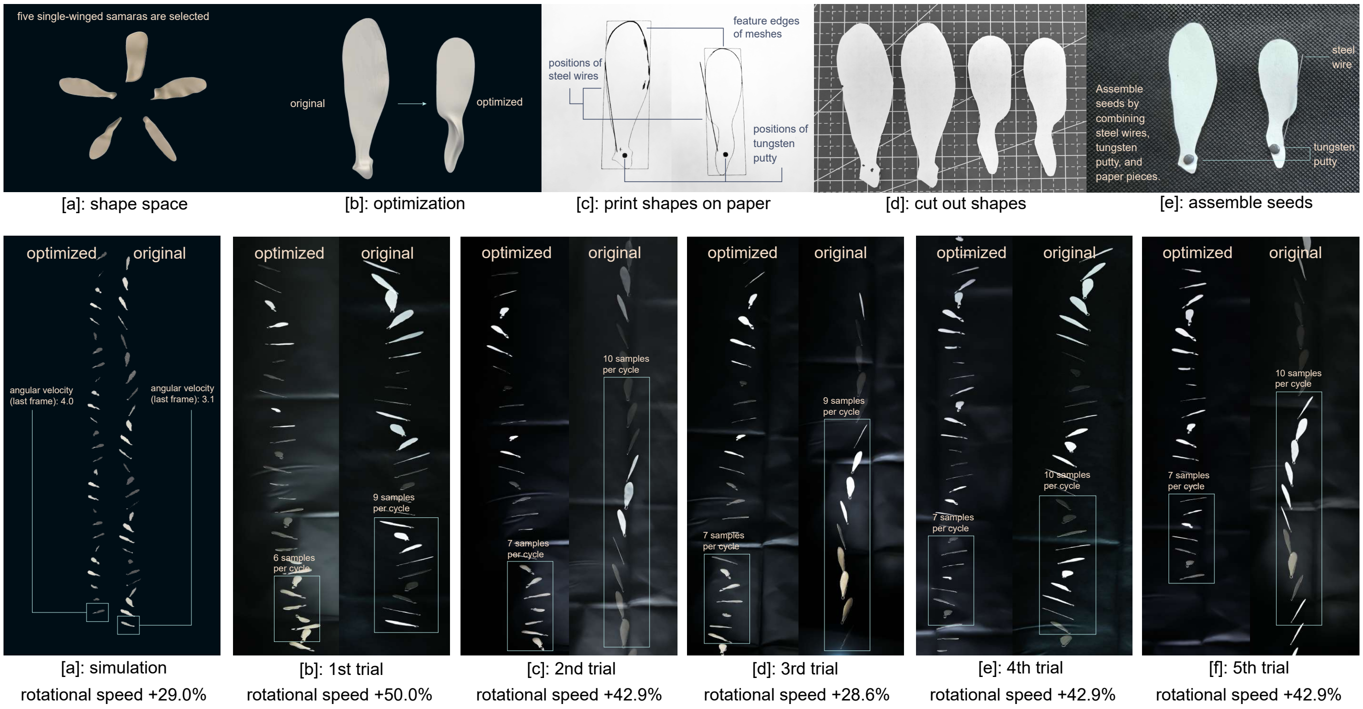

Fig. 14. *Fabrication results (Sec. 9.5):* Top: the process of fabrication. Bottom: falling trajectories of simulated seeds ($[a]$) and fabricated seeds (tried five times from $[b]$ to $[f]$). We also compute relative changes of rotational speed of simulated seeds based on their angular velocities at the last frame. Counterparts of fabricated seeds are computed based on the counts of samples every cycle, which are captured by stroboscopic photography every 1/40 second. These changes are displayed at the bottom of the figure.

Mirza Faisal Beg, Michael I. Miller, Alain Trouvé, and Laurent Younes. 2005. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision* 61 (2005), 139–157.

James M. Bern, Kai-Hung Chang, and Stelian Coros. 2017. Interactive design of animated plushies. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–11.

Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. 2019. Multi-Garment Net: Learning to Dress 3D People from Images. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE.

Franklin T. Bonner and Robert P. Karrfalt. 2008. *The woody plant seed manual.* Number 727. Forest Service.

Benjamin Charlier, Jean Feydy, Joan Alexis Glaunès, François-David Collin, and Ghislain Durif. 2021. Kernel Operations on the GPU, with Autodiff, without Memory Overflows. *Journal of Machine Learning Research* 22, 74 (2021), 1–6. http://jmlr.org/papers/v22/20-275.html

Nicolas Charon and Alain Trouvé. 2013. The varifold representation of nonoriented shapes for diffeomorphic registration. *SIAM journal on Imaging Sciences* 6, 4 (2013), 2547–2580.

Rulin Chen, Ziqi Wang, Peng Song, and Bernd Bickel. 2022. Computational design of high-level interlocking puzzles. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–15.

Yufeng Chen, Huichan Zhao, Jie Mao, Pakpong Chirarattananon, E. Farrell Helbling, Nak-seung Patrick Hyun, David R. Clarke, and Robert J. Wood. 2019. Controlled flight of a microrobot powered by soft artificial muscles. *Nature* 575, 7782 (2019), 324–329.

Julian Chibane, Aymen Mir, and Gerard Pons-Moll. 2020. Neural Unsigned Distance Fields for Implicit Function Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Thor V. Christiansen, Jakob Andreas Bærentzen, Rasmus R. Paulsen, and Morten R. Hannemose. 2023. Neural Representation of Open Surfaces. *Computer Graphics Forum* (2023). https://doi.org/10.1111/cgf.14916

Tao Du, Adriana Schulz, Bo Zhu, Bernd Bickel, and Wojciech Matusik. 2016. Computational multicopter design. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–10.

Paul Dupuis, Ulf Grenander, and Michael I. Miller. 1998. Variational problems on flows of diffeomorphisms for image matching. *Quarterly of applied mathematics* (1998), 587–600.

Roberto M. Dyke, YuKun Lai, Paul L. Rosin, Stefano Zappalà, Seana Dykes, Daoliang Guo, Kun Li, Riccardo Marin, Simone Melzi, and Jingyu Yang. 2020. SHREC'20: Shape correspondence with non-isometric deformations. *Computers & Graphics* 92 (2020), 28–43. https://doi.org/10.1016/j.cag.2020.08.008

Marvin Eisenberger and Daniel Cremers. 2020. Hamiltonian dynamics for real-world shape interpolation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16.* Springer, 179–196.

Marvin Eisenberger, David Novotny, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. 2021. NeuroMorph: Unsupervised Shape Interpolation and Correspondence in One Go. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 7473–7483.

Michael Fischer and Tobias Ritschel. 2023. Zero Grads Ever Given: Learning Local Surrogate Losses for Non-Differentiable Graphics. *arXiv preprint arXiv:2308.05739* (2023).

Joan Glaunes, Anqi Qiu, Michael I. Miller, and Laurent Younes. 2008. Large deformation diffeomorphic metric curve mapping. *International journal of computer vision* 80 (2008), 317–336.

Joan Glaunes, Alain Trouvé, and Laurent Younes. 2004. Diffeomorphic matching of distributions: A new approach for unlabelled point-sets and sub-manifolds matching. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, Vol. 2. Ieee, II–II.

Benoit Guillard, Federico Stella, and Pascal Fua. 2022. Meshudf: Fast and differentiable meshing of unsigned distance field networks. In *European conference on computer vision.* Springer, 576–592.

Sehoon Ha, Stelian Coros, Alexander Alspach, James M. Bern, Joohyung Kim, and Katsu Yamane. 2018. Computational design of robotic devices from high-level motion specifications. *IEEE Transactions on Robotics* 34, 5 (2018), 1240–1251.

Darryl D. Holm, Tanya Schmah, and Cristina Stoica. 2009. *Geometric mechanics and symmetry: from finite to infinite dimensions.* Vol. 12. Oxford University Press.

Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. 2019. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 201.

Qixing Huang, Fan Wang, and Leonidas Guibas. 2014. Functional map networks for analyzing and exploring large shape collections. *ACM Transactions on Graphics*

(TOG) 33, 4 (2014), 1–11.

Ruqi Huang, Marie-Julie Rakotosaona, Panos Achlioptas, Leonidas J. Guibas, and Maks Ovsjanikov. 2019. Operatornet: Recovering 3d shapes from difference operators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 8588–8597.

Alec Jacobson, Daniele Panozzo, et al. 2018. libigl: A simple C++ geometry processing library. https://libigl.github.io/.

Zihan Jiang, Huilin Yang, and Xiaoying Tang. 2018. Deformation-based statistical shape analysis of the corpus callosum in mild cognitive impairment and Alzheimer's disease. *Current Alzheimer Research* 15, 12 (2018), 1151–1160.

Sarang C. Joshi and Michael I. Miller. 2000. Landmark matching via large deformation diffeomorphisms. *IEEE transactions on image processing* 9, 8 (2000), 1357–1370.

Kristine Aavild Juhl, Xabier Morales, Ole de Backer, Oscar Camara, and Rasmus Reinhold Paulsen. 2021. Implicit Neural Distance Representation for Unsupervised and Supervised Classification of Complex Anatomies. In *Medical Image Computing and Computer Assisted Intervention (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics))*, Marleen de Bruijne, Philippe C. Cattin, Stéphane Cotin, Nicolas Padoy, Stefanie Speidel, Yefeng Zheng, and Caroline Essert (Eds.). Springer, 405–415. https://doi.org/10.1007/978-3-030-87196-3_38

Irene Kaltenmark, Benjamin Charlier, and Nicolas Charon. 2017. A general framework for curve and surface comparison and registration with oriented varifolds. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 3346–3355.

Bong Hoon Kim, Kan Li, Jin-Tae Kim, Yoonseok Park, Hokyung Jang, Xueju Wang, Zhaoqian Xie, Sang Min Won, Hong-Joon Yoon, Geumbee Lee, et al. 2021. Three-dimensional electronic microfliers inspired by wind-dispersed seeds. *Nature* 597, 7877 (2021), 503–510.

David Lentink, William B. Dickson, Johan L. Van Leeuwen, and Michael H. Dickinson. 2009. Leading-edge vortices elevate lift of autorotating plant seeds. *Science* 324, 5933 (2009), 1438–1440.

Wei Li, Tongtong Wang, Zherong Pan, Xifeng Gao, Kui Wu, and Mathieu Desbrun. 2023. High-Order Moment-Encoded Kinetic Simulation of Turbulent Flows. *ACM Trans. Graph.* 42, 6, Article 190 (dec 2023), 13 pages. https://doi.org/10.1145/3618341

Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. 2017. Deep functional maps: Structured prediction for dense shape correspondence. In *Proceedings of the IEEE international conference on computer vision.* 5659–5667.

Kevin Y. Ma, Pakpong Chirarattananon, Sawyer B. Fuller, and Robert J. Wood. 2013. Controlled Flight of a Biologically Inspired, Insect-Scale Robot. *Science* 340, 6132 (2013), 603–607. https://doi.org/10.1126/science.1231806 arXiv:https://www.science.org/doi/pdf/10.1126/science.1231806

Pingchuan Ma, Tao Du, John Z. Zhang, Kui Wu, Andrew Spielberg, Robert K. Katzschmann, and Wojciech Matusik. 2021. Diffaqua: A differentiable computational design pipeline for soft underwater swimmers with shape interpolation. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–14.

Tobias Martin, Nobuyuki Umetani, and Bernd Bickel. 2015. OmniAD: data-driven omni-directional aerodynamics. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–12.

Wojciech Matusik and Adriana Schulz. 2019. Computational fabrication. In *ACM SIGGRAPH 2019 Courses.* 1–305.

Michael I. Miller, Alain Trouvé, and Laurent Younes. 2015a. Hamiltonian systems and optimal control in computational anatomy: 100 years since D'Arcy Thompson. *Annual review of biomedical engineering* 17 (2015), 447–509.

Michael I. Miller, Laurent Younes, J. Tilak Ratnanather, Timothy Brown, Huong Trinh, David S. Lee, Daniel Tward, Pamela B. Mahon, Susumu Mori, Marilyn Albert, et al. 2015b. Amygdalar atrophy in symptomatic Alzheimer's disease based on diffeomorphometry: the BIOCARD cohort. *Neurobiology of aging* 36 (2015), S3–S10.

Shizuka Minami and Akira Azuma. 2003. Various flying modes of wind-dispersal seeds. *Journal of theoretical biology* 225, 1 (2003), 1–14.

Ran Nathan, Gabriel G. Katul, Henry S. Horn, Suvi M. Thomas, Ram Oren, Roni Avissar, Stephen W. Pacala, and Simon A. Levin. 2002. Mechanisms of long-distance dispersal of seeds by wind. *Nature* 418, 6896 (2002), 409–413.

Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. 2012. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–11.

Yue Pan, Gary E. Christensen, Oguz C. Durumeric, Sarah E. Gerard, Joseph M. Reinhardt, and Geoffrey D. Hugo. 2016. Current-and varifold-based registration of lung vessel and airway trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.* 126–133.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. 2018. Generating 3D faces using convolutional mesh autoencoders. In *Proceedings of the European conference on computer vision (ECCV).* 704–720.

Raif M. Rustamov, Maks Ovsjanikov, Omri Azencot, Mirela Ben-Chen, Frédéric Chazal, and Leonidas Guibas. 2013. Map-based exploration of intrinsic shape differences and variability. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.

Adriana Schulz, Ariel Shamir, David I.W. Levin, Pitchaya Sitthi-Amorn, and Wojciech Matusik. 2014. Design and fabrication by example. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–11.

Nicholas Sharp et al. 2019. Polyscope. www.polyscope.run.

Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement learning: An introduction.* MIT press.

Qingyang Tan, Lin Gao, YuKun Lai, and Shihong Xia. 2018. Variational autoencoders for deforming 3d mesh models. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 5841–5850.

Xiaoying Tang, Dominic Holland, Anders M. Dale, Laurent Younes, Michael I. Miller, and Alzheimer's Disease Neuroimaging Initiative. 2015. The diffeomorphometry of regional shape change rates and its relevance to cognitive deterioration in mild cognitive impairment and A lzheimer's disease. *Human brain mapping* 36, 6 (2015), 2093–2117.

D'Arcy Wentworth Thompson et al. 2010. *On growth and form.* Cambridge university press.

Alain Trouvé. 1998. Diffeomorphisms groups and pattern matching in image analysis. *International journal of computer vision* 28 (1998), 213–221.

Nobuyuki Umetani, Yuki Koyama, Ryan Schmidt, and Takeo Igarashi. 2014. Pteromys: Interactive design and optimization of free-formed free-flight model airplanes. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–10.

Marc Vaillant and Joan Glaunes. 2005. Surface matching via currents. In *Biennial international conference on information processing in medical imaging.* Springer, 381–392.

Julian F.V. Vincent, Olga A. Bogatyreva, Nikolaj R. Bogatyrev, Adrian Bowyer, and Anja-Karina Pahl. 2006. Biomimetics: its practice and theory. *Journal of the Royal Society Interface* 3, 9 (2006), 471–482.

Jonathan Viquerat, Jean Rabault, Alexander Kuhnle, Hassan Ghraieb, Aurélien Larcher, and Elie Hachem. 2021. Direct shape optimization through deep reinforcement learning. *J. Comput. Phys.* 428 (2021), 110080.

Yanlin Weng, Menglei Chai, Weiwei Xu, Yiying Tong, and Kun Zhou. 2013. As-Rigid-As Possible Distance Field Metamorphosis. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 381–389.

Shane Kyi Hla Win, Luke Soe Thura Win, Danial Sufiyan, Gim Song Soh, and Shaohui Foong. 2022. An Agile Samara-Inspired Single-Actuator Aerial Robot Capable of Autorotation and Diving. *IEEE Transactions on Robotics* 38, 2 (2022), 1033–1046. https://doi.org/10.1109/TRO.2021.3091275

Jie Xu, Tao Du, Michael Foshey, Beichen Li, Bo Zhu, Adriana Schulz, and Wojciech Matusik. 2019. Learning to fly: computational controller design for hybrid uavs with reinforcement learning. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.

Jianhui Yang, Yi Xu, and Liang Yang. 2022. Taichi-LBM3D: A Single-Phase and Multi-phase Lattice Boltzmann Solver on Cross-Platform Multicore CPU/GPUs. *Fluids* 7, 8 (2022). https://doi.org/10.3390/fluids7080270

Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and M.J. Rosato. 2006. A 3D facial expression database for facial behavior research. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06).* 211–216. https://doi.org/10.1109/FGR.2006.6

Laurent Younes. 2010. *Shapes and diffeomorphisms.* Vol. 171. Springer.

YuJie Yuan, YuKun Lai, Jie Yang, Qi Duan, Hongbo Fu, and Lin Gao. 2020. Mesh variational autoencoders with edge contraction pooling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.* 274–275.

Ralf Zimmermann. 2021. Manifold interpolation. *Model Order Reduction* 1 (2021), 229–274.