Faraday Cage Estimation of Normals for Point Clouds and Ribbon Sketches

DANIEL SCRIVENER, Boston University, USA
DANIEL CUI, Boston University, USA
ELLIS COLDREN, Boston University, USA
S. MAZDAK ABULNAGA, Massachusetts Institute of Technology / Harvard Medical School, USA
MIKHAIL BESSMELTSEV, Université de Montréal, Canada
EDWARD CHIEN, Boston University, USA

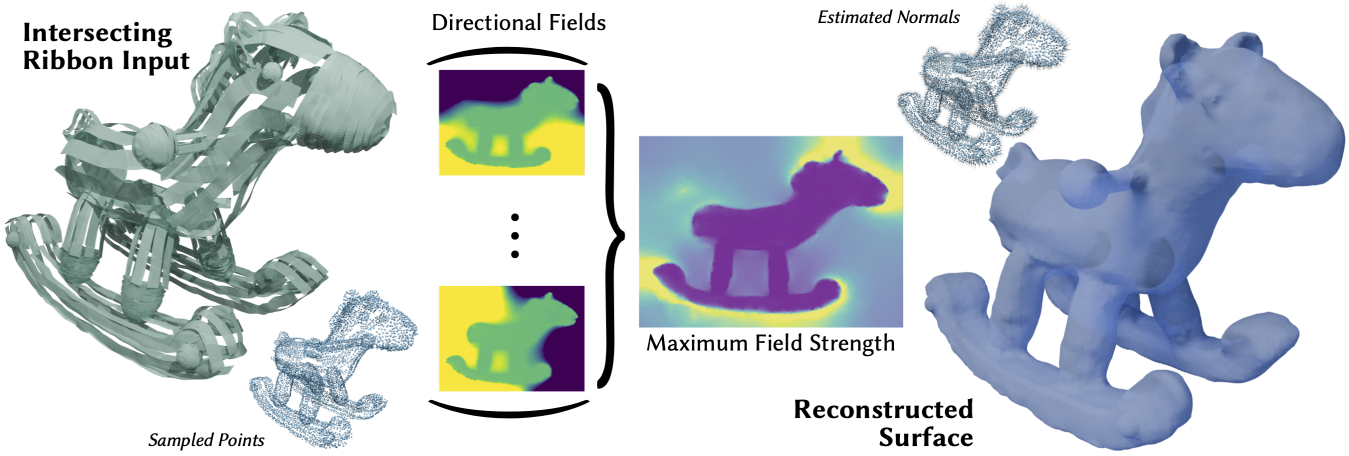


Fig. 1. Pipeline outline and demonstration on sparse intersecting ribbon input. Left: A VR ribbon sketch [Rosales et al. 2021, 2019] is sparsified to mimic a simpler, plausible user input, and points are evenly sampled from the ribbons. Middle: Points form a Faraday cage around the interior. Electric potentials under various linear external fields, and the maximum electric field strength over these scenarios are shown. Right: Gradient information from maximum field strength is used to estimate normals and filter interior parts of ribbons. Poisson Surface Reconstruction [Kazhdan and Hoppe 2013] is used to generate the surface free of interior structures and concavities at intersecting points, as seen in comparison methods (see Fig. 10). Input drawing: © Jafet Rodriguez.

We propose a novel method (FaCE) for normal estimation of unoriented point clouds and VR ribbon sketches that leverages a modeling of the Faraday cage effect. Input points, or a sampling of the ribbons, form a conductive cage and shield the interior from external fields. The gradient of the maximum field strength over external field scenarios is used to estimate a normal at each input point or ribbon. The electrostatic effect is modeled with a simple Poisson system, accommodating intuitive user-driven sculpting via the specification of point charges and Faraday cage points. On inputs sampled from clean, watertight meshes, our method achieves comparable normal quality to existing methods tailored for this scenario. On inputs containing interior structures and artifacts, our method produces superior surfacing output when combined with Poisson Surface Reconstruction. In the case of ribbon sketches, our method accommodates sparser ribbon input while maintaining an accurate geometry, allowing for greater flexibility in the artistic process. We demonstrate superior performance to an existing approach for surfacing ribbon sketches in this sparse setting.

CCS Concepts: • **Computing methodologies** → **Point-based models**; *Virtual reality*.

Additional Key Words and Phrases: Faraday cage, winding number, point cloud orientation, surface reconstruction

ACM Reference Format:
Daniel Scrivener, Daniel Cui, Ellis Coldren, S. Mazdak Abulnaga, Mikhail Bessmeltsev, and Edward Chien. 2025. Faraday Cage Estimation of Normals for Point Clouds and Ribbon Sketches. *ACM Trans. Graph.* 44, 4 (August 2025), 13 pages. https://doi.org/10.1145/3731212

Authors' Contact Information: Daniel Scrivener, dscriv@bu.edu, Boston University, Boston, MA, USA; Daniel Cui, dcui@bu.edu, Boston University, Boston, MA, USA; Ellis Coldren, ecoldren@bu.edu, Boston University, Boston, MA, USA; S. Mazdak Abulnaga, abulnaga@mit.edu, Massachusetts Institute of Technology / and Harvard Medical School, Boston, MA, USA; Mikhail Bessmeltsev, bmpix@iro.umontreal.ca, Université de Montréal, Montreal, Quebec, Canada; Edward Chien, edchien@bu.edu, Boston University, Boston, MA, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM 1557-7368/2025/8-ART
https://doi.org/10.1145/3731212

1 Introduction

A *Faraday cage* [Faraday 1832] is a conductive enclosure that is used in many engineering contexts to block some electromagnetic fields. It frequently takes the form of a mesh or cage to allow other fields through, e.g. visible light. A common example is the mesh on the

front of a microwave door, which protects users outside from the radiation within while allowing the user to see inside. This shielding effect is achieved by the movement of positive and negative charges in the cage to counteract and greatly dampen the imposed field.

Drawing inspiration from this effect, we propose a novel method for estimating normals of unoriented point clouds ("Faraday cage estimation," or FaCE). Points are modeled as part of a Faraday cage, shielding the interior from external electrical fields. By inducing a series of linear external fields, we are able to probe the geometry of the underlying shape. The maximum field strength over this set of external fields defines a scalar function whose value provides a notion of interiority for unoriented point clouds. Moreover, the gradient of said function yields an effective estimate of the normal at the input points. To further extend this model, we introduce prototype post-processing operations (§3.6). These take the form of adding external fields imposed by point charges (for a subtractive effect) and/or additional Faraday cage points (for an additive effect).

An (ideal) Faraday cage in the electrostatic setting is modeled with a simple Poisson problem, requiring a harmonic potential function with an equipotential constraint on the cage surface. The gradient of the potential yields the electric field; its norm indicates field strength under each imposed external field. Due to the well-investigated structure of Poisson problems, we are able to leverage fast numerical linear algebra solvers to find the resulting fields, resulting in an efficient pipeline for normal estimation.

FaCE differs from existing normal orientation methods, representing the geometry of surfaces resulting from intersecting components more faithfully. In particular, it tends to orient each component consistently, recovering normals that align with the negative gradient of the winding number (§4.1). This improves subsequent surface reconstructions by (1) preventing unwanted concave artifacts near intersection points (concavities) and (2) reducing surface artifacts where proximate surfaces run parallel to one another (pitting). We encounter both scenarios in misaligned partial point clouds and demonstrate superior quantitative and qualitative results on several synthetic scans (§4.3). Additionally, our method exhibits strong performance on CAD objects with interior components and artifacts from modeling (§4.5). On clean models, FaCE achieves competitive results and shorter running times than several state-of-the-art methods. (§4.6).

Interior structures also arise commonly in artistic or design modeling contexts in which users are not necessarily concerned with maintaining closed, watertight surfaces. We consider the application of virtual reality (VR) sketches drawn with the "ribbon brush" technique of [Rosales et al. 2021, 2019]. These sketches are created by tracing (unoriented) ribbons with a multi-axis wand peripheral. Each ribbon sketch intends to define a watertight surface: to recover it, we propose orienting point clouds evenly sampled from each sketch via our method and running Poisson Surface Reconstruction (PSR) [Kazhdan and Hoppe 2013]. Unlike the ribbon surfacing method of [Rosales et al. 2019], which frequently generates outputs with multiple components, ours produces a closed, watertight mesh that adheres to the sketch geometry. Moreover, the approach of [Rosales et al. 2019] assumes tightly constructed ribbon sketches with virtually no gaps between adjacent strokes, increasing the labor of the artist. Our pipeline appropriately handles sparser ribbon input and exhibits better performance than existing point cloud orientation methods in this scenario (§4.4). We hope that our work will promote easier and more widespread use of ribbon brush tools.

2 Related Work

Shen et al. [2010] are among the first to employ the Faraday cage model in computer graphics, using electric field strength as a measure of shielding to generate cleaned volumetric representations of triangle soups. Their setup models the equipotential constraint by constraining the domain boundary and input geometry to fixed function values. In contrast, our work leverages this same information for a different application: orientation and surfacing of point clouds. In addition to the field strength itself, our method uses the spatial variation of this electric field strength and shows that it reliably offers orientation information even in concave (Figs. 3, 14) and interior (§4.1) regions when field strength is relatively uniform. Other contributions of our work include extensive validation across multiple categories of input, Dirichlet "sculpting" of the electromagnetic field (§3.6), and the formulation of the equipotential constraint (from Chapman et al. [2015]) which allows the cage value to float according to the exterior boundary conditions.

Elsewhere, our work draws inspiration from two main families of methods: surface reconstruction and point cloud orientation.

2.1 Surface reconstruction from point clouds

Some of the classical methods in point cloud reconstruction either require oriented normals or compute them separately. The early work by Hoppe et al. [1992] computes an SDF by fitting local planes into a neighborhood of every point and measures a signed distance to those planes. Generalizing this linear approach, an implicit function can be defined as the sum of nonlinear radial basis functions (RBFs) centered at the points [Carr et al. 2001; Muraki 1991], unfortunately leading to dense and ill-conditioned systems. Perhaps the two most well-known methods in this category are Poisson Surface Reconstruction (PSR) and Screened PSR [Kazhdan et al. 2006; Kazhdan and Hoppe 2013]. In particular, Kazhdan et al. [2006] observe that normals are the gradient of an indicator function. They minimize an L_2 norm of the difference between the indicator gradient and the normals, leading to a Poisson equation. Kazhdan and Hoppe [2013] add a data term, equivalent to solving a Screened Poisson equation, yielding more precise reconstructions. We demonstrate surface reconstructions with our normals using these methods in §4. Recent work has aimed to incorporate uncertainty information into these methods [Sellán and Jacobson 2023; Sellán and Jacobson 2022]. Alternatively, Feng and Crane [2024] propose a robust method to compute an SDF directly from noisy oriented geometry. The 0-level set of the SDF yields a reconstruction of the surface.

2.1.1 Without orientations, or partial information. Earlier surface reconstruction methods, including Crust [Amenta et al. 1998, 2001], tight cocone [Dey and Goswami 2003], and α-shapes [Bajaj et al. 1995; Bernardini et al. 1999; Edelsbrunner and Mücke 1994] rely on classical computational geometry approaches that provide theoretical guarantees but do not specifically address input noise. Dey & Goswami [Dey and Goswami 2004] are among the first to tackle the problem of noise, yet their method often produces a noisy surface

when the assumed noise model is not satisfied [Kazhdan et al. 2006]. A more recent method, VIPSS [Huang et al. 2019], reconstructs an implicit surface via minimizing a variational smoothness energy with a data term and a unit gradient norm constraint on the input points. Their method is robust yet computationally inefficient for dense point clouds and does not handle interior structures.

Some recent methods address a domain-specific problem of reconstructing approximately developable surfaces [Dong et al. 2024; Wang et al. 2023]. Our method addresses a generic problem, only assuming an appropriate degree of smoothness of the input surface, and thus has a wider range of applications.

Recent progress in implicit surface representation via deep networks brings new tools for surface reconstruction. For instance, Occupancy Networks [Mescheder et al. 2019; Peng et al. 2020] learn a binary inside-outside predicate. Some of the well-known approaches like DeepSDF, Neural Pull, and others [Baorui et al. 2022; Chabra et al. 2020; Ma et al. 2021; Park et al. 2019] learn a full signed distance function (SDF), or an unsigned distance function [Lu et al. 2024; Ren et al. 2023]: see Farshian et al. [2023] for a recent review. While these methods can be used for point cloud reconstruction, they typically excel within only a single category of shapes and have trouble generalizing to other categories [Farshian et al. 2023]. Similarly, these methods do not address the issue of interior points and therefore are not directly applicable to our problem.

For surfacing ribbon sketches, the current state of the art is SurfaceBrush [Rosales et al. 2019]. They reconstruct the connectivity of ribbon sketches via a discrete constrained optimization. We demonstrate that due to the variational nature of our framework, our method outperforms theirs on a variety of VR ribbon inputs; the difference is especially noticeable for sparse VR sketches (§ 4).

2.2 Point cloud orientation

In the case of unoriented point clouds, many works have aimed to solve for normals and applied PSR as a practical indicator of the normal estimation quality.

Following the work of Hoppe et al. [1992], classical approaches to normal estimation tend to compute the tangent plane for each point individually and perform global alignment to obtain a consistent orientation. Mitra and Nguyen [2003] optimize local neighborhood size for least squares. Pauly et al. [2003] apply the moving least squares surface model of Levin [2004] to develop an implicit normal representation. Xie et al. [2003] propose a global alignment strategy involving multi-seed propagation for sharp corner detection that König and Gumhold [2009] refine with Hermite splines. Schertler et al. [2017] formulate an objective for the general orientation propagation problem and assess the performance of numerous solvers. Metzer et al. [2021] orient normals via greedily aligning each consistently oriented patch with the electric field gradient.

Recently, focus has shifted from estimating normals using an approximation of the local neighborhood toward global computation. Curiously, Hou et al. [2022] show that starting with random normal orientations, then iterating between PSR and recomputing normals, converges in many practical scenarios. Xu et al. [2023] (GCNO) optimize the generalized winding number evaluated at Voronoi poles [Amenta et al. 2001], guiding the winding number towards binary values of 0 or 1. Gotsman and Hormann [2024] observe that for a correct orientation of normals, a surface integral of a curl of any vector field should be always zero. As such, they find a family of test functions and solve a linear system for the normal signs. Liu et al. [2024] maximize an energy derived from Dirichlet energy; their method empirically converges to the correct normal orientations. Lin et al. [2022] (PGR) considers products of normals and surface element areas as variables. They find an indicator function — parameterized via these unknowns via Gauss formula — that is equal to 1/2 at the query points. The indicator function then defines the normals. Lin et al. [2024] optimize normals until they are aligned with the winding number field gradient. Liu et al. [2025] improve their performance and introduce a noise-resilient *screened* generalized winding number. We compare with some of these methods in §4. In general, these methods all assume input points to be on or near the surface and thus create artifacts near interior points.

Finally, several learning-based approaches for normal estimation of point clouds have been developed [Boulch and Marlet 2016; Guerrero et al. 2018; Li et al. 2023, 2022; Zhu et al. 2021]. In contrast to these, our method is purely geometric and does not make use of training data. Moreover, to our knowledge, no learning-based method has been designed for normal estimation on point clouds with complex interior structures.

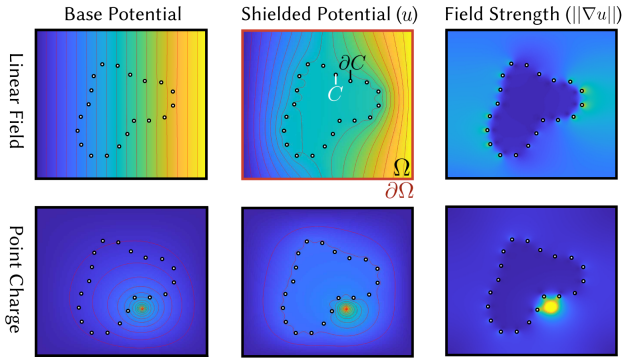


Fig. 2. 2D example of Faraday cage shielding under a linear external field (Top Row) and a point-charge–induced field (Bottom Row). Left: the electric potential in the absence of shielding. Middle: the resulting potential u in the presence of shielding. Right: The field strength $||\nabla u||$.

3 Method

3.1 Faraday cage model

The electrostatic effect of an (ideal) Faraday cage may be modeled by finding a harmonic electric potential $\Delta u = 0$ subject to an equipotential constraint on the surface of the conductive cage [Chapman et al. 2015]. The direction and strength of the electric field is then given by the gradient vector field ∇u.

As a physical analogue, such a Poisson system may also be used to model an elastic sheet at equilibrium as represented by its height map, subject to the same Dirichlet and equipotential boundary conditions. This is a reasonable approximation for small z-deflections and is called *antiplane strain* within linear elasticity [Slaughter 2002].

The potential u may be seen as the solution to a quadratic variational problem with linear constraints. Let $\Omega \subsetneq \mathbb{R}^3$ denote a bounded domain of interest, with boundary $\partial\Omega$ far from the cage $C \subsetneq \Omega$, and let ∂C denote the boundary of the cage as schematically illustrated in the red outlined image in Fig. 2.

$$\min_{u:\Omega\setminus C\to\mathbb{R},\, u_C\in\mathbb{R}} \quad \int_\Omega ||\nabla u||^2 \tag{1a}$$

$$\text{such that} \quad u|_{\partial C} = u_C \tag{1b}$$

$$u|_{\partial\Omega} = g. \tag{1c}$$

In the above, u_C is a variable denoting the potential on the cage surface. The function g denotes a boundary Dirichlet constraint that imposes an external field by capturing the potential values at these points in the absence of shielding. In our method, we consider the imposition of both linear and point-charge–induced fields.

3.1.1 Linear external fields. We use the term *linear external fields* to refer to those induced by linear potentials on the boundary: $g(\vec{x}) = \vec{v}^T\vec{x} + b$ for some $v \in \mathbb{R}^3$ and $b \in \mathbb{R}$. Note that, in the absence of any shielding, the field would be constant and equal to $\vec{v}$ everywhere. In Supp. §5, we argue for symmetries that allow us to parameterize the space of relevant linear fields by just a unit vector $\vec{v}$ from half of $\mathbb{S}^2$ (a hemisphere) or equivalently, by a choice of 1-dimensional subspace (element of $\mathbb{RP}^2$).

In the elastic sheet example, imposing an external linear field with direction $\vec{v}$ yields Dirichlet boundary constraints $u_B(x) = \vec{v}^T x$ on $\partial\Omega$ and pins points on ∂C to the same z-value (u_c, an unknown variable).

3.1.2 Point-charge-induced fields. The most accurate way to model point-charge–induced fields is to puncture the domain Ω at the point charge locations and enforce a $1/r$ limiting behavior as the distance r to the point charge approaches 0 [Chapman et al. 2015]. Additionally, one should impose a similar $1/r$ limiting behavior as $r \to \infty$. For simplicity in modeling the problem, we simply assign a high value $M \gg 0$ to $u_{\partial\Omega}$ at the point charge positions and $u_{\partial\Omega} = 0$ otherwise (on the outer boundary of Ω).

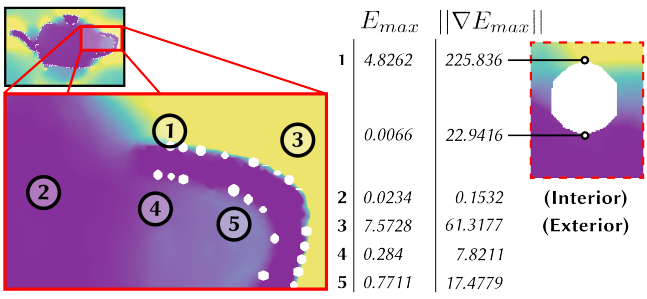


Fig. 3. Close-up view of concave region of `walking_teapot`, where white circles represent cage spheres. We measure $E_{\max}$ (linear external fields only) and $||\nabla E_{\max}||$ at various sample points. Both are several orders of magnitude smaller for interior points (2) compared to exterior points (3). Across the cage sphere (1), $||\nabla E_{\max}||$ is largest in the direction of the desired normal. Points (4) and (5) are located in a region of the exterior with greater shielding from nearby features; nonetheless, $E_{\max}$ remains an order of magnitude larger than the interior. Input drawing: © Jafet Rodriguez.

3.2 Discretization details

To model the conductive cage C, we instantiate spheres about the input points of the point cloud. As the shielding is positively correlated with the "thickness" of the cage bars [Chapman et al. 2015], we aim to make the spheres large relative to the local density of the input without creating intersections. To accomplish this, we construct a 1-NN graph and set each radius to be 45% of the distance to the nearest neighboring point. This ensures a sensibly varying "cage radius" and strengthens the shielding effect.

Given such an Ω and C, we solve the Poisson problem at hand with a mesh-based FEM discretization. In 2D, we use Triangle [Shewchuk 1996], and in 3D we use Tetgen [Si 2015] to generate constrained conforming Delaunay triangulations of the domain $\Omega\setminus C$. The balls that constitute C are approximated by icosahedra. Further refining this approximation yields diminishing returns: see Supp. §6.

On this mesh, the standard cotangent Laplacian $L = d_0^T *_1 d_0$ is used for the Dirichlet energy (expressed with DEC operators [Crane et al. 2013]). Imposing Dirichlet boundary constraints on $\partial\Omega$ and equipotential constraints on ∂C would result in a quadratic optimization with linear constraints. However, we incorporate the known Dirichlet boundary conditions into the objective and gather the potential values along ∂C into a single scalar variable to obtain an unconstrained quadratic optimization, which reduces to a linear solve. A brief derivation is outlined below.

We assume that we have ordered the mesh vertices with interior vertices first (index set **I**), boundary ($\partial\Omega$) vertices second (index set **B**), and cage vertices third (index set **C**): $\mathbf{u}^T = \begin{bmatrix}\mathbf{u_I}^T & \mathbf{u_B}^T & \mathbf{u_C}^T\end{bmatrix}$. The equipotential constraints are specified with a single variable $\mathbf{u_C} = u_C\mathbf{1}$. Dirichlet boundary conditions $\mathbf{u_B}$ are given by $\vec{v}^T\vec{x}$ for a linear field scenario (§3.1.1) or simply $M \gg 0$ at the point charge positions and 0 elsewhere for a point-charge-induced field (§3.1.2). The Dirichlet energy objective $\mathbf{u}^T L\mathbf{u}$ may then be considered in block form and simplified.

$$\begin{bmatrix}\mathbf{u_I}^T & \mathbf{u_B}^T & \mathbf{u_C}^T\end{bmatrix}\begin{bmatrix}L_{\mathbf{II}} & L_{\mathbf{IB}} & L_{\mathbf{IC}}\\ L_{\mathbf{IB}}^T & L_{\mathbf{BB}} & L_{\mathbf{BC}}\\ L_{\mathbf{IC}}^T & L_{\mathbf{BC}}^T & L_{\mathbf{CC}}\end{bmatrix}\begin{bmatrix}\mathbf{u_I}\\ \mathbf{u_B}\\ \mathbf{u_C}\end{bmatrix} =$$
$$\mathbf{u_I}^T L_{\mathbf{II}}\mathbf{u_I} + \mathbf{u_B}^T L_{\mathbf{BB}}\mathbf{u_B} + \mathbf{u_C}^T L_{\mathbf{CC}}\mathbf{u_C} +$$
$$2\mathbf{u_I}^T L_{\mathbf{IB}}\mathbf{u_B} + 2\mathbf{u_I}^T L_{\mathbf{IC}}\mathbf{u_C} + 2\mathbf{u_B}^T L_{\mathbf{BC}}\mathbf{u_C} \tag{2}$$

The second of the diagonal terms is a constant, with $\mathbf{u_B}$ known, and the third diagonal term simplifies to $u_C^2\mathbf{1}^T L_{\mathbf{CC}}\mathbf{1}$. Similar substitutions for off-diagonal terms result in an equivalent expression in terms of the unknown variables $\tilde{\mathbf{u}}^T = \begin{bmatrix}\mathbf{u_I}^T & u_C\end{bmatrix}$:

$$\begin{bmatrix}\mathbf{u_I}^T & u_C\end{bmatrix}\begin{bmatrix}L_{\mathbf{II}} & L_{\mathbf{IC}}\mathbf{1}\\ \mathbf{1}^T L_{\mathbf{IC}}^T & \mathbf{1}^T L_{\mathbf{CC}}\mathbf{1}\end{bmatrix}\begin{bmatrix}\mathbf{u_I}\\ u_C\end{bmatrix} + \begin{bmatrix}\mathbf{u_B}^T L_{\mathbf{IB}}^T & \mathbf{1}^T L_{\mathbf{BC}}^T\mathbf{u_B}\end{bmatrix}\begin{bmatrix}\mathbf{u_I}\\ u_C\end{bmatrix}. \tag{3}$$

Taking a derivative with respect to $\tilde{\mathbf{u}}$ results in the linear system:

$$\begin{bmatrix}L_{\mathbf{II}} & L_{\mathbf{IC}}\mathbf{1}\\ \mathbf{1}^T L_{\mathbf{IC}}^T & \mathbf{1}^T L_{\mathbf{CC}}\mathbf{1}\end{bmatrix}\begin{bmatrix}\mathbf{u_I}\\ u_C\end{bmatrix} = -\begin{bmatrix}L_{\mathbf{IB}}\mathbf{u_B}\\ \mathbf{u_B}^T L_{\mathbf{BC}}\mathbf{1}\end{bmatrix} \tag{4}$$

This reduces to a sparse positive-definite linear system, which we solve with the CHOLMOD suite of algorithms for sparse Cholesky

factorization [Chen et al. 2008]. With the factorization in hand, we may efficiently solve over the various field scenarios i.

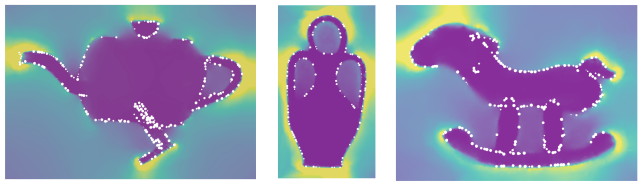


Fig. 4. Slices of the three-dimensional max field strength $E_{\max}$ function (linear external fields only) for a few models. Left to right: `walking_teapot` ribbon sketch, huapen manifold input, `toyhorse` ribbon sketch. Note that even with linear fields, the field strength reaches into concavities and handles due to the global nature of Faraday cage shielding. `walking_teapot`, `toyhorse`: © Jafet Rodriguez.

3.3 Max. field strength

A simple way to gather the shielding information is via a function that measures the maximum electric field strength $||\nabla u||$ under all external field scenarios. Under the elastic sheet analogy, points surrounded by the cage will have very low gradient norm (field strength) for all directions $\vec{v}$, while points outside the cage will have high gradient norm for some directions. Thus, taking a maximum of field strengths over all directions results in outward-pointing gradients near the cage. For notational brevity, let the index set $\mathcal{L} = \{1, \ldots, n\}$ label the n external linear field scenarios, each given by a unit vector $\vec{v_i} \in \mathbb{S}^2$. Let u_i denote the resulting potential. If the user opts to consider point-charge–induced external fields, we gather all specified point charge positions $p_1, \ldots, p_m$ into a single field scenario and denote the resulting potential as u_{pts}. If no such point-charge–induced field is given, let $u_{\text{pts}} = 0$. Then, the *max. field strength* is given by

$$E_{\max}(x) = \max\left(\max_{i \in \mathcal{L}} ||\nabla u_i(x)||,\ ||\nabla u_{\text{pts}}(x)||\right). \tag{5}$$

3.3.1 Discretization. We discretize $E_{\max}$ as a piecewise-linear function over the triangle/tet mesh, obtained via the following averaging steps:

(1) Average triangle/tet-wise gradient vectors ∇u_i onto vertices, via an area/volume-weighted average
(2) Take their norm to obtain $||\nabla u_i||$ as a vertex-valued function
(3) Take their maximum over i in order to determine $E_{\max}$

The $E_{\max}$ values at vertices are linearly interpolated over the the triangle/tet elements to obtain a function that we use for normal estimation and classification of interior points. Fig. 3 and Fig. 4 illustrate that the global nature of shielding allows for probing of handles and concavities with just linear fields.

3.3.2 Linear field sampling. By Lemmas 1 and 2 in Supp. §5, we need only sample from a hemisphere or from 1-dimensional subspaces. In practice, we found that remarkably few directions suffice to elicit reasonable output from $E_{\max}$, and that diminishing returns are quickly reached. We show this in Supp. §2. For three-dimensional scenarios, we end up sampling just 6 unit vectors from half of the icosahedron.

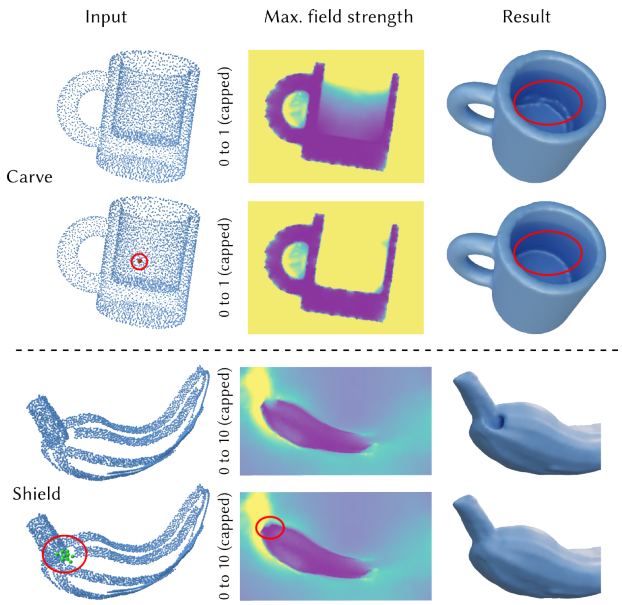


Fig. 5. Example of subtractive and additive Faraday cage sculpting operations. Top: The addition of a point charge field source at the red point refines the concave interior of `mug` by locally increasing the value of $E_{\max}$. Bottom: The addition of several additional cage points (green) near a large gap in `chili_decimated` improves nearby normal estimates and filters more interior points, repairing an unwanted hole. `chili`: © Jafet Rodriguez.

3.4 Normal estimation and subsequent surfacing

To estimate normals at input points, we average $\nabla E_{\max}$ from the triangle/tet. elements onto mesh vertices, weighting by area/volume. We then average the gradients on the cage vertices surrounding each input point and normalize the result to obtain our final normal estimates. Afterwards, Poisson Surface Reconstruction may be run on the input with or without screening to obtain a surface.

3.5 Interior point filtering

If the input contains structures/artifacts within the object's interior, we find that removing interior points improves surface reconstruction quality. As such, we use $E_{\max}$ to filter interior points as follows:

(1) Extract the α-isosurface of $E_{\max}$ via marching tetrahedra
(2) Classify points as inside/outside relative to the α-isosurface by thresholding on the Fast Winding Number [Barill et al. 2018] with value 0.9

With regard to linear external fields, there is a natural reference point for a set threshold α. In the absence of any shielding, the field strength is constant at $||\vec{v}|| = 1$; as such, any region for which $E_{\max}$ is less than 1 may be considered shielded. In practice, we opt for a more conservative filtering of interior points and find $\alpha = 0.01$ and $\alpha = 0.05$ to be effective for dense/complete inputs and sparse inputs (e.g. ribbon sketches with large missing areas) respectively.

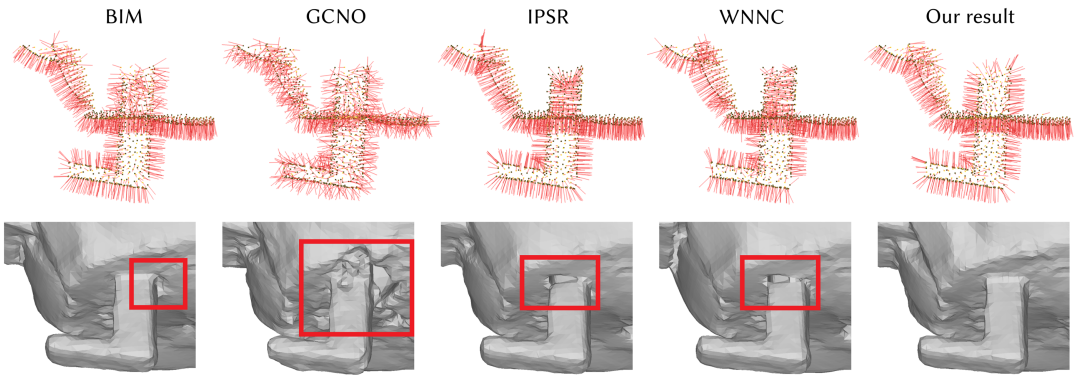


Fig. 6. Example of consistent orientation of intersecting components. A closeup comparison of the intersection between a leg of the `walking_teapot` model and the body. Top row: a slice of the resulting normals for various methods; bottom row: A view of the screened PSR reconstruction (without interior point filtering for our method). Our method is the only one that achieves correctly outward-pointing normals for both components and has the most faithful reconstruction near the surface intersection. Input drawing: © Jafet Rodriguez.

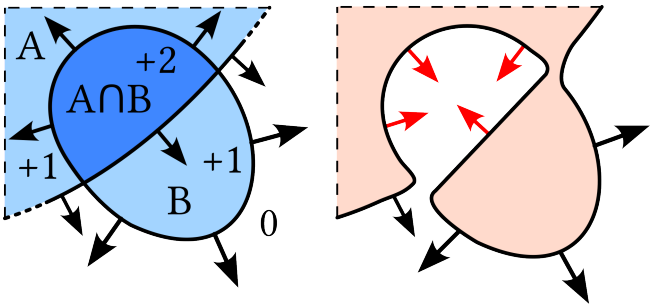


Fig. 7. Schematic of winding-number-consistent normals. Left: with two intersecting domains A and B, the normals produced by FaCE align with the negative gradient of the winding number; these are labelled in their respective regions, and would become differentiable over ∂A and ∂B if sampled as point clouds. Right: Other methods implicitly assume $w \in [0, 1]$ and tend to resolve such intersections by giving inward-point normals for $A \cap B$ resulting in large intrusive concavities.

3.6 Faraday cage sculpting

Although this work focuses primarily on fully automatic normal estimation, we introduce prototypes for post-processing tools to demonstrate our method's flexibility. First, normal estimation methods such as ours may struggle with sharp concavities. As such, we allow users to specify point charges, each of which imposes another external field (§3.1.2). Point charges *subtract* from the inferred interior and sharpen nearby boundaries, improving nearby normal estimates. Second, the user may achieve an *additive* effect by placing additional cage vertices. This is useful in the sparse ribbon setting: gaps between ribbons may prevent interior structures from being properly shielded. Fig. 5 demonstrates each operation, depicting the user's input as well as the effect of each operation on $E_{\max}$ and the resulting surface reconstruction. See Supp. §3 for additional implementation details.

4 Results

We begin with a qualitative observation and then present results on point clouds of four different kinds: misaligned point cloud scans, samples from ribbon sketches, samples from intersecting ModelNet/Thingi10k meshes, and samples from single-component watertight meshes, hereafter termed "clean" inputs. Existing methods to which we compare are fully automatic. As such, only linear external fields are used to compute $E_{\max}$ in the results that follow. We forgo use of our post-processing tools (§3.6) entirely to provide a more balanced comparison.

4.1 Intersecting surfaces and winding number

An interesting empirical observation is that FaCE robustly produces correct outwardly-pointing normals on intersecting components, even for points sampled from "within" another surface (see Fig. 6). All other methods that we have tested against do not reproduce this behavior: iPSR [Hou et al. 2022] and WNNC [Lin et al. 2024] consistently flip orientations on the parts within the other surface, GCNO [Xu et al. 2023] struggles to converge in a reasonable amount of time on such input, and BIM [Liu et al. 2024] handles orientations inconsistently in these regions, getting some right and some wrong. This difference is key to the superior Screened PSR reconstructions that we obtain in these scenarios, producing fewer concavity artifacts at these regions of intersection.

Intuitively, FaCE produces normals that better reflect the negative gradient of the ground truth winding number w [Barill et al. 2018]. Regions surrounded by multiple components are shielded to a greater extent; as such, the gradient of $E_{\max}$ should align roughly with $-\nabla w$. Fig. 7 illustrates the comparative behavior of our method with others in this scenario. While FaCE produces normals consistent with the winding number, other methods do not, leading to the intruding concavities shown. This behavior derives from implicit assumptions that surfaces are non-intersecting and watertight, resulting in $w = 0, 1$ over the entire domain.

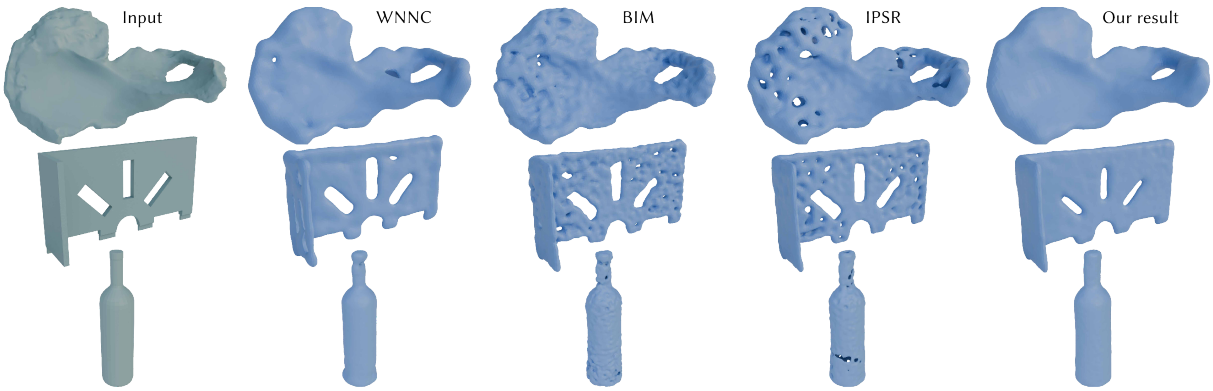


Fig. 8. Results on misaligned, non-uniformly sampled point cloud scans generated by the synthetic scanning pipeline of [Huang et al. 2024]. Compared with existing normal orientation methods, ours achieves better treatment of thin structures. Holes and deep concavities can result from inward facing normals on misaligned points in other methods.

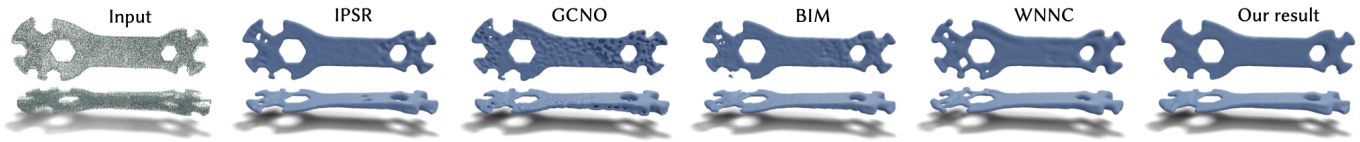


Fig. 9. Example of a real-world misaligned scan of a wrench and reconstructions after normal estimate. By preserving outward normal orientations for interior points, ours is the only one that reproduces the surface without unwanted holes or missing features.

4.2 Experimental setup

Below, we perform experiments on the categories of inputs outlined in §4. For each mesh/ribbon input, we obtain a point cloud via Poisson disk sampling except in §4.3 which follows a synthetic scanning pipeline. We compare to several state-of-the-art methods: iPSR [Hou et al. 2022], GCNO [Xu et al. 2023], BIM [Liu et al. 2024], and WNNC [Lin et al. 2024] for normal estimation and SurfaceBrush [Rosales et al. 2019] for surfacing ribbon sketches. iPSR was run with default parameters (max. 30 iterations). As noted in other works [Lin et al. 2024; Liu et al. 2024; Xu et al. 2023], the running time of GCNO may be quite long on inputs exceeding around 5000 points: we stop GCNO after 1 hour on each input (obtaining ≥ 50 iterations for each) and find this sufficient for most small-scale models. For, BIM we set an internal "scale" parameter to 0.05 for inputs scaled to the $[0, 1]$ AABB as recommended by the authors. WNNC incorporates a smoothing parameter: we empirically determine the best value for each category of input. Timing experiments were performed on a 2021 MacBook Pro (M1 Pro, 16 GB LPDDR5). For general geometry processing algorithms, we use MeshLab [Cignoni et al. 2008].

For quantitatively assessing normal estimates, we consider alignment percentage (positive dot product with ground truth), mean angle error, and standard deviation of angle error. For assessing surface reconstruction, we consider symmetric Hausdorff (max. of both directed Hausdorff distances) and Chamfer distances. Broadly speaking, these metrics do not always capture obvious qualitative differences in output. In particular, we find that Chamfer distance does not always capture undesirable oscillations e.g. bumpiness. Hausdorff distance roughly quantifies the severity of the output's largest defect: we find it to be more effective (though still imperfect) for detecting concavities and surface oscillation.

4.3 Comparison on misaligned point cloud scans

Composition of point clouds from multiple scanning viewpoints is a challenging problem; errors in registration often lead to misalignment between surface points. The resulting "doubling" effect, in which the point cloud contains slightly offset copies of the target surface, poses a compelling application for FaCE and its handling of interior structures. Using the method of Huang et al. [2024], we generate 14 examples of misaligned point clouds from a synthetic scanning pipeline. In the synthetic scans, we specify nonuniform sampling, a misalignment angle of 3°, and noise intensity of 0.01 (relative to the unit bounding box). We then downsample the synthetic scans to roughly 20,000 points via PDS and run our method as well as those of [Hou et al. 2022; Lin et al. 2024; Liu et al. 2024].

FaCE effectively filters interior points that otherwise create unwanted artifacts with PSR, especially its screened variant. Sample reconstructions are shown in Fig. 8. Other methods struggle to maintain a consistent orientation on the interior "doubled" surface, exacerbating the negative effect of these points and leading to pitting (`abc`, Middle row) and holes (`pelvis`, Top row) in thin regions. Average Chamfer and Hausdorff distances for each method are provided in Table 1a, with full results in Supp. §4.1; our method achieves better Hausdorff on the greatest number of inputs. Furthermore, we provide a representative example of a real-world misaligned scan in Fig. 9 and find that FaCE is the only one that faithfully reproduces the input with no holes.

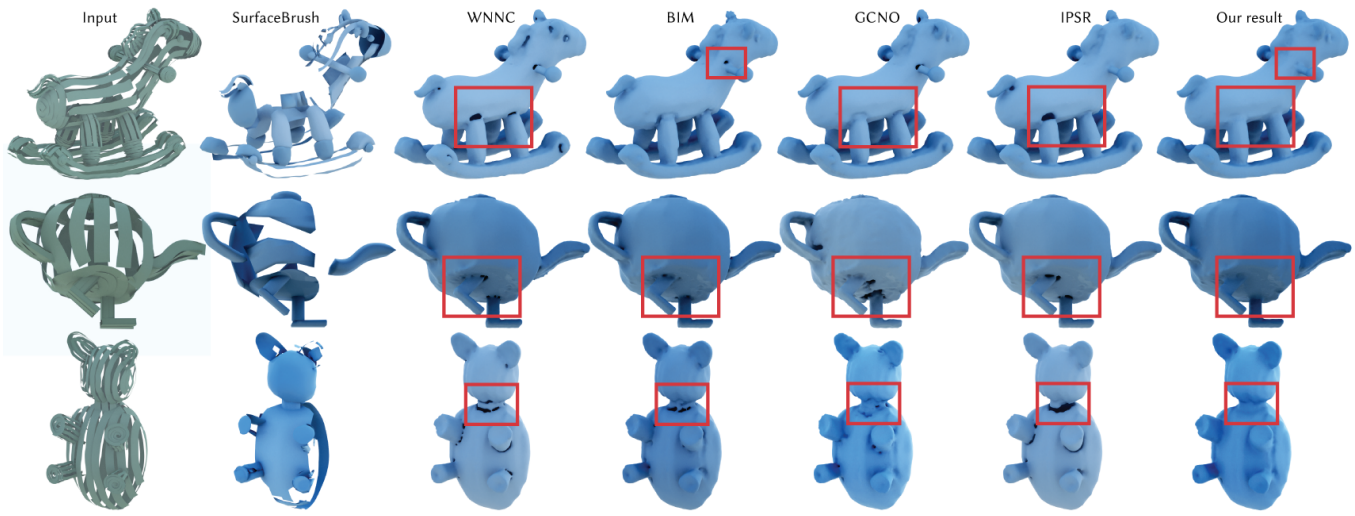


Fig. 10. Results on sparsified ribbon inputs from [Rosales et al. 2019]. Compared to the alternative point cloud normal orientation methods and SurfaceBrush, ours (right) correctly treats internal structures typical for VR ribbon sketches. `walking_teapot`, `toyhorse`: © Jafet Rodriguez. `teddy_bear`: © Elinor Palomares.

4.4 Comparison on ribbon sketches

In this section, we focus on the ribbon sketches of [Rosales et al. 2019]. In addition to using 12 of the original sketches, we generate sparsified versions to model a simpler workflow in which the user's strokes need not cover the entirety of the intended surface, allowing for gaps. Removing strokes at random would completely eliminate key features of the sketches that are drawn as one continuous ribbon, so we instead follow a heuristic of removing alternating "stripe" ribbons that surround an intended component. We compare to the normal estimation methods of [Hou et al. 2022; Lin et al. 2024; Liu et al. 2024; Xu et al. 2023] as well as the surfacing algorithm of [Rosales et al. 2019]. Given the lack of a true "ground truth" for surfaced sketches, we take the boolean union of the results in [Rosales et al. 2019] to obtain a single connected component as our reference mesh. This removes unwanted intrusions into the interior. One non-manifold result is alpha wrapped as in §4.5.

Sample reconstructions on sparsified inputs are shown in Fig. 10. Of particular interest are the regions in which accessory components (e.g. legs, handle of `toyhorse`) intersect the interior. Other methods are prone to sculpting out concavities in these areas due to their inability to maintain a consistent outward orientation as described in §4.1. Such concavities are difficult to repair; in contrast, our method maintains a clean transition between components in both unmodified and sparse ribbon settings. Quantitatively, FaCE surpasses existing methods in all areas, achieving better normal metrics as well as lower Chamfer and Hausdorff distance (Tables 1b, 1c). Full results are in Supp. §4.2 and §4.3.

4.5 Comparison on CAD meshes with interior structures

In order to further assess the suitability of FaCE for processing interior structures, we consider triangle soups in CAD settings sampled from ModelNet [Wu et al. 2015] (20 models) and Thingi10k [Zhou and Jacobson 2016] (5 models). To approximate a ground truth surfacing, we cannot use boolean operations, which depend on an input's winding number being piecewise constant [Zhou et al. 2016]. As an alternative, we perform alpha wrapping with offset with $\alpha = 0.005, \delta = 0.001$. Although alpha wraps are unsuited for general surfacing of point clouds with noise and large gaps [Portaneri et al. 2022], we find that they are effective for representing the surface geometry of these triangle soups, which contain no exterior gaps.

Sample reconstructions from Screened PSR are shown in Fig. 11, with additional reconstructions done using the unscreened variant in 12. Similar to the synthetic scans of §8, other methods suffer from misaligned normals on interior structures, producing large interior voids (such where the base meets the body of `train_monitor`) and surface oscillations. Rippling artifacts are seen on the cover of `test_bed`, sideboard of `test_chair`, and screen of `train_monitor`. By filtering interior points and maintaining more consistent orientations on interior points, FaCE avoids these defects and recovers the intended surface. We achieve better Chamfer and Hausdorff distance than existing methods (Table 1d). Full quantitative results are provided in Supp. §4.4.

4.6 Comparison on clean inputs

For completeness and parity with previous works, we consider inputs that are uniformly sampled from watertight, manifold surfaces. We run existing methods [Hou et al. 2022; Lin et al. 2024; Liu et al. 2024; Xu et al. 2023] on 18 inputs provided by Xu et al. [2023]. Overall, we find all methods to be highly capable: every method achieves an average alignment across inputs exceeding 99%. FaCE achieves better running times than GCNO and BIM in the general case and iPSR on noisier inputs where convergence is slow (Table 2). Metrics are presented in Supp. §4.5.

4.7 Additional experiments

In Supp. §1, we test our method's robustness to two different types of noise: perturbation of the input points with Gaussian noise and addition of white noise points. We compare to [Lin et al. 2024]

and find that our method's performance is strong even in the presence of heavy noise. In Supp. §2, we show that the performance of FaCE does not improve significantly by sampling more directions, justifying our use of few external field scenarios. In Supp. §6, we demonstrate that refining the cage sphere approximation yields minimal performance gains at the cost of increased meshing time.

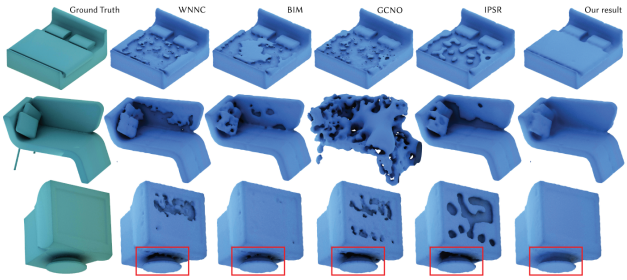


Fig. 11. CAD objects sampled and re-surfaced by combining normal estimation methods with screened PSR. These inputs contain interior structures left over from the modeling process. Our ability to filter interior points results in smooth, detail-preserving reconstructions and avoids concavities surface pitting.

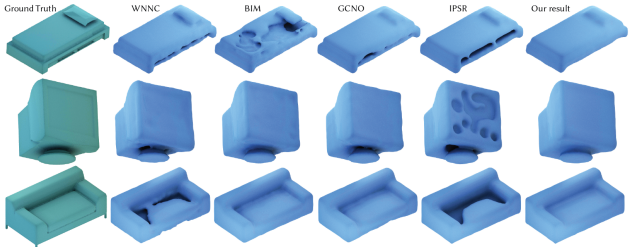


Fig. 12. CAD objects sampled and re-surfaced with a combination of various normal estimation methods and PSR without screening. Although unscreened PSR is less likely to preserve sharp features, we find that the advantages of our method in preventing concavities and surface pitting persist. Areas like the lower region of `test_bed_0611` (Top row), the underside of `train_monitor_0465` (Middle row), and the crevice in `train_sofa_0060` (Bottom row) may produce inconsistent normals using other methods.

5 Conclusion

We propose a novel method (FaCE) for point cloud orientation which draws inspiration from the Faraday cage shielding effect. The gradient of the max. field strength over a handful of external field scenarios is used to estimate point normals. Each scenario is efficiently modeled with a simple Poisson system that results in a sparse positive-definite linear system.

After obtaining normals with FaCE, applying standard Poisson Surface Reconstruction provides an indication of normal estimation quality. We find that this pipeline is superior to existing methods in surfacing misaligned/misregistered point clouds and sparse ribbon brush sketches. In particular, our results avoid sharp concavities due to component intersection and surface pitting resulting from slightly offset surfaces. We hypothesize that this is due to the fact that FaCE recovers normals that are better aligned with the negative gradient of the winding number. Our pipeline is competitive on "clean" input sampled from connected watertight meshes.

For ribbon brush inputs, our pipeline reduces the labor required of the artist by accommodating sparser input. Finally, we propose tools for sculpting $E_{\max}$ via the addition of point charges (subtractive effect) and Faraday cage points (additive effect), leading to refinement of normal estimates and interior classification.

5.1 Limitations and future work

A fundamental limitation of FaCE is handling domains with enclosed interior cavities, e.g., an annulus in 2D or a spherical shell in 3D. With only linear fields, FaCE interprets the interior cavities as another domain and reverses the ground truth normals. However, these scenarios are unlikely to arise in our applications, and could be corrected with user placement of point charges within the cavities.

Next, we note several challenges that we hope to address in future work. First, robust reconstruction of open/nearly-flat surfaces requires further investigation. Fig. 13 illustrates the limitations of Poisson Surface Reconstruction in processing thin structures with little appreciable volume, regardless of normal estimation quality. Second, lack of sample density further complicates reconstruction of normals in thin, concave regions. PSR introduces breaks in filament-like features when sampling is too sparse regardless of normal estimation quality; artifacts resulting from improper normal estimation are also observed. (Fig. 14). Third, slight concavities result in sparsified ribbon inputs where ribbons are removed. Different reconstruction algorithms such as [Feng and Crane 2024] or a tailored method may alleviate these artifacts.

On a technical level, the current implementation of our method is unoptimized: the majority of the running time is spent meshing the input (Fig. 15). Newer tet. meshing libraries (e.g. [Hu et al. 2020]) may offer improved performance: we use TetGen in this work for its robustness. Alternatively, it would be possible to accelerate the method significantly by adopting multigrid solvers, as is done in Poisson Surface Reconstruction [Kazhdan et al. 2006]. This could potentially enable real-time Faraday cage sculpting.

Lastly, we suspect that our method offers rich information regarding point cloud interiority that is not captured by $E_{\max}$ alone. As such, we encourage further exploration of the distribution of field strengths over all external field scenarios and look forward to variations on our approach that make use of such information.

Acknowledgments

We would like to thank the anonymous reviewers for their feedback as well as Liu et al. [2024] for assistance running their method. Furthermore, we thank Andrew Wood[1], Erick Jimenez Berumen[2], and Adrish Dey[3] for (1) lending computing equipment (2) designing Fig. 8 and (3) providing technical assistance.

The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant No.: RGPIN-2024-04968 ("Modelling and animation via intuitive input"), the NSERC - Fonds de recherche du Québec - Nature et technologies (FRQNT) under NOVA Grant No. 314090, and the National Institutes of Health under Grant No. UM1MH130981, R01 AG064027.

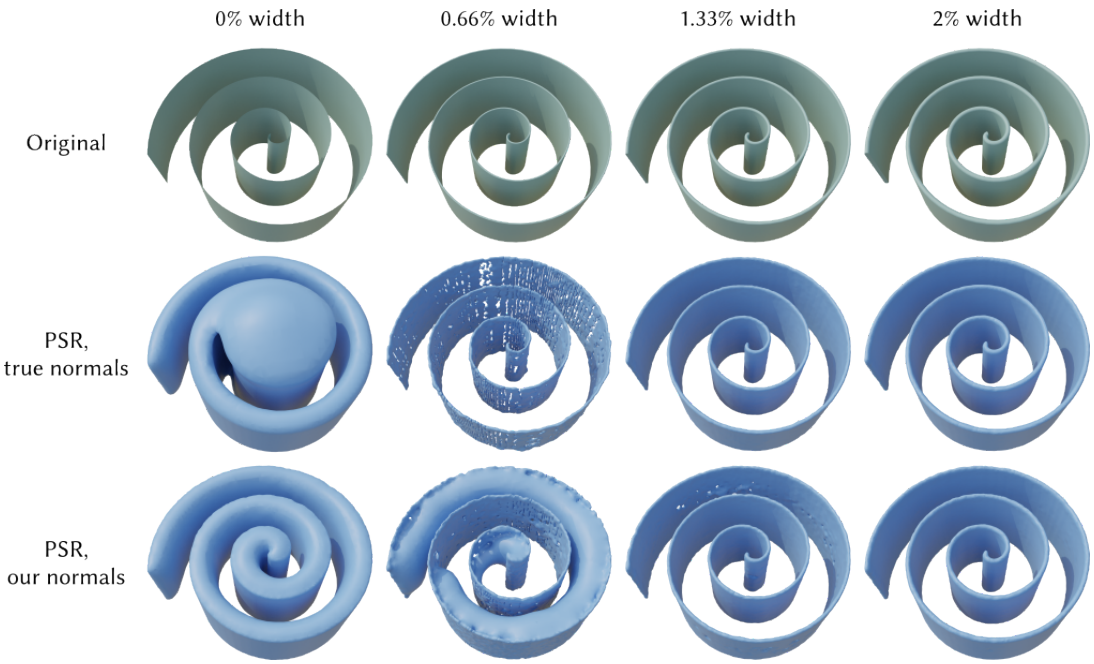


Fig. 13. We run FaCE on Archimedean spiral "ribbons" of varying thickness to investigate our method's behavior on thin, concave structures, sampling 20,000 points from each original mesh. The thickness of each ribbon is stated as a percentage of the spiral's endpoint radius. Poisson Surface Reconstruction fails on vanishingly thin volumes even if the ground truth normals are supplied. As thickness increases, FaCE normals remain competitive relative to the ground truth in terms of reconstruction quality.

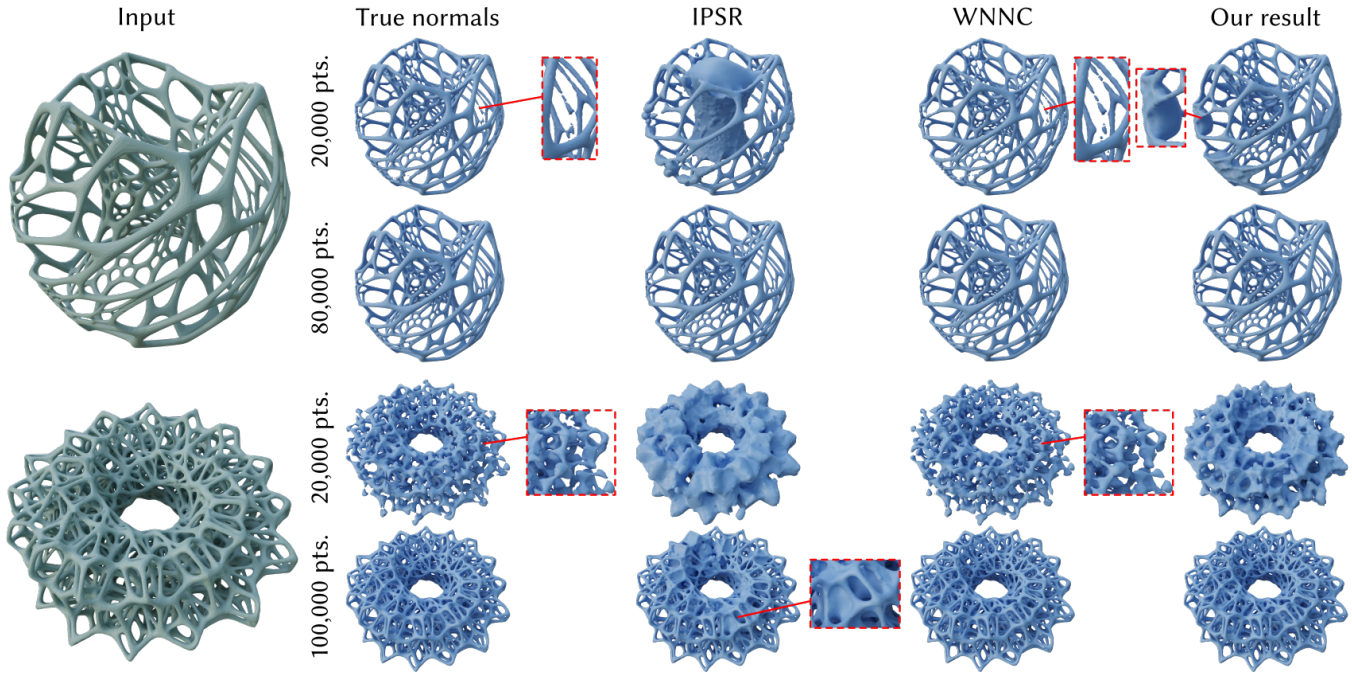


Fig. 14. Results on complex inputs from Xu et al. [2023]. If sampled sparsely (top row for each input), Poisson Surface Reconstruction struggles to reconstruct thin filaments even when supplied with ground truth normals. Furthermore, with sparse sampling, FaCE and Hou et al. [2022] improperly estimate normals in highly shielded "interior" regions of these concave structures, producing blob-like artifacts. With more dense sampling (bottom row for each input), FaCE successfully recovers the target shape.

Table 1. Abridged results of quantitative experiments. For normal estimation, we consider the percentage of normals that align with the ground truth (positive dot product), the average error, and the standard deviation of the error. For surface reconstruction, we consider Chamfer distance and symmetrized Hausdorff distance. We report averages across each dataset as well as the number of models for which each method achieved the lowest distance.

(a) Surface reconstruction metrics on synthetic scans in §4.3.

Method	iPSR (2022)		BIM (2024)		WNNC (2024)		Ours	
Metric	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR
Avg. Chamfer	0.0001092	0.0001282	0.0000984	0.0001323	0.0000870	0.0000969	0.0000900	0.0000999
Avg. Hausdorff	0.0038414	0.0037226	0.0019987	0.0025345	0.0018770	0.0023778	0.0024578	0.0021429
# Best, Chamfer	8		0		1		5	
# Best, Hausdorff	3		0		3		8	

(b) Normal estimation metrics on sparsified ribbon sketches in §4.4. GCNO did not converge on two inputs: metrics are reported across remaining models.

Metric	iPSR (2022)	GCNO (2023)*	BIM (2024)	WNNC (2024)	Ours
Avg. align %	95.84	89.45	95.93	94.01	98.47
Avg. ME (deg.)	17.75	32.53	17.70	21.96	15.35
Avg. SE (deg.)	25.90	29.29	25.01	31.94	18.42

(c) Surface reconstruction metrics on sparsified ribbon sketches in §4.4. SurfaceBrush and GCNO did not terminate/converge on two inputs: metrics are reported across remaining models.

Method	S.B. (2019)*	iPSR (2022)		GCNO (2023)*		BIM (2024)		WNNC (2024)		Ours	
Metric	–	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR
Avg. Chamfer	0.0023428	0.0001195	0.0001375	0.0002445	0.0002194	0.0002022	0.0001809	0.0001733	0.0001925	0.0001978	0.0000926
Avg. Hausdorff	0.0323616	0.0083068	0.0103586	0.0067215	0.0087957	0.0072071	0.0110549	0.0092783	0.0119257	0.0068092	0.0054694
# Best, Chamfer	0	1		1		2		0		8	
# Best, Hausdorff	0	0		1		3		1		7	

(d) Surface reconstruction metrics on CAD models in §4.5.

Method	iPSR (2022)		GCNO (2023)		BIM (2024)		WNNC (2024)		Ours	
Metric	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR	PSR	Sc. PSR
Avg. Chamfer	0.0003133	0.0002326	0.0026134	0.0018425	0.0004234	0.0002357	0.0003420	0.0002743	0.0003499	0.0001917
Avg. Hausdorff	0.0115380	0.0089797	0.0316620	0.0317204	0.0160205	0.0105053	0.0143459	0.0108921	0.0118015	0.0085078
# Best, Chamfer	0		2		2		11		8	
# Best, Hausdorff	4		4		3		5		7	

Table 2. Timing (in seconds) of methods on three models: M1 = `torus`, M2 = huapen, M3 = `test_bed_0523`. On tested inputs, FaCE generally performs favorably relative to iPSR, which may take longer to converge on noisy inputs. Accurate timings for BIM and GCNO could not be obtained on our hardware due to architectural issues with the reference implementations; see [Lin et al. 2024; Liu et al. 2024; Xu et al. 2023] for their results, typically much slower than FaCE even on more powerful systems. For detailed runtime analysis, refer to §5 "Runtime" and Supplementary §3 in [Liu et al. 2024].

Method	iPSR (2022)			Ours			WNNC (2024)		
No. points	M1	M2	M3	M1	M2	M3	M1	M2	M3
1,000	46.73	149.45	109.51	5.36	7.25	6.45	1.95	1.69	2.10
5,000	52.09	72.71	202.02	28.80	29.10	27.49	2.17	2.78	2.76
10,000	64.80	84.14	373.17	61.76	61.93	60.60	4.35	4.08	4.02
20,000	64.58	85.01	421.19	134.52	132.87	135.07	6.21	6.15	5.85

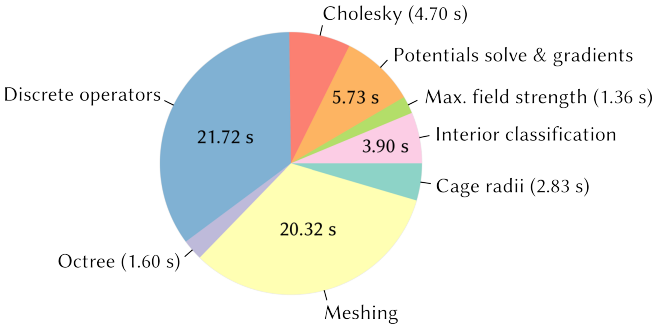


Fig. 15. Breakdown of running time for our method on `torus` input (10,000 samples, total 62.16 s). Meshing and computation of discrete operators pose the main performance bottleneck.

References

Nina Amenta, Marshall Bern, and Manolis Kamvysselis. 1998. A New Voronoi-based Surface Reconstruction Algorithm. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. 415–421.

Nina Amenta, Sunghee Choi, and Ravi Krishna Kolluri. 2001. The Power Crust. In *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications (Sma '01)*. Association for Computing Machinery, New York, NY, USA, 249–266.

Chandrajit L Bajaj, Fausto Bernardini, and Guoliang Xu. 1995. Automatic Reconstruction of Surfaces and Scalar Fields from 3D Scans. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*. 109–118.

Ma Baorui, Liu Yu-Shen, and Han Zhizhong. 2022. Reconstructing Surfaces for Sparse Point Clouds with On-Surface Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Gavin Barill, Nia Dickson, Ryan Schmidt, David I.W. Levin, and Alec Jacobson. 2018. Fast Winding Numbers for Soups and Clouds. *ACM Transactions on Graphics* (2018).

F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. 1999. The Ball-Pivoting Algorithm for Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics* 5, 4 (1999), 349–359.

Alexandre Boulch and Renaud Marlet. 2016. Deep learning for robust normal estimation in unstructured point clouds. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 281–290.

J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. 2001. Reconstruction and Representation of 3D Objects with Radial Basis Functions. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (Siggraph '01)*. Association for Computing Machinery, New York, NY, USA, 67–76.

Rohan Chabra, Jan E. Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. 2020. Deep Local Shapes: Learning Local SDF Priors for Detailed 3D Reconstruction. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX*. Springer-Verlag, Berlin, Heidelberg, 608–625.

S. Jonathan Chapman, David P. Hewett, and Lloyd N. Trefethen. 2015. Mathematics of the Faraday Cage. *SIAM Rev.* 57, 3 (2015), 398–417.

Yanqing Chen, Timothy A. Davis, William W. Hager, and Sivasankaran Rajamanickam. 2008. Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate. *ACM Trans. Math. Softw.* 35, 3, Article 22 (Oct. 2008), 14 pages.

Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. 2008. MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference*, Vittorio Scarano, Rosario De Chiara, and Ugo Erra (Eds.). The Eurographics Association.

Keenan Crane, Fernando de Goes, Mathieu Desbrun, and Peter Schröder. 2013. Digital Geometry Processing with Discrete Exterior Calculus. In *ACM SIGGRAPH 2013 courses* (Anaheim, California) *(SIGGRAPH '13)*. ACM, New York, NY, USA, 126 pages.

Tamal K Dey and Samrat Goswami. 2003. Tight cocone: a water-tight surface reconstructor. In *Proceedings of the eighth ACM symposium on Solid modeling and applications*. 127–134.

Tamal K Dey and Samrat Goswami. 2004. Provable Surface Reconstruction from Noisy Samples. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry*. 330–339.

Qiujie Dong, Rui Xu, Pengfei Wang, Shuangmin Chen, Shiqing Xin, Xiaohong Jia, Wenping Wang, and Changhe Tu. 2024. NeurCADRecon: Neural Representation for Reconstructing CAD Surfaces by Enforcing Zero Gaussian Curvature. *ACM Trans. Graph.* 43, 4, Article 51 (July 2024).

Herbert Edelsbrunner and Ernst P Mücke. 1994. Three-Dimensional Alpha Shapes. *ACM Transactions On Graphics (TOG)* 13, 1 (1994), 43–72.

Michael Faraday. 1832. V. Experimental researches in electricity. *Philosophical transactions of the Royal Society of London* 122 (1832), 125–162.

Anis Farshian, Markus Götz, Gabriele Cavallaro, Charlotte Debus, Matthias Nießner, Jón Atli Benediktsson, and Achim Streit. 2023. Deep-Learning-Based 3-D Surface Reconstruction—a Survey. *Proc. IEEE* 111, 11 (2023), 1464–1501.

Nicole Feng and Keenan Crane. 2024. A Heat Method for Generalized Signed Distance. *ACM Trans. Graph.* 43, 4, Article 92 (jul 2024), 19 pages.

Craig Gotsman and Kai Hormann. 2024. A Linear Method to Consistently Orient Normals of a 3D Point Cloud. In *ACM SIGGRAPH 2024 Conference Papers (Siggraph '24)*. Association for Computing Machinery, New York, NY, USA, Article 55.

Paul Guerrero, Yanir Kleiman, Maks Ovsjanikov, and Niloy J. Mitra. 2018. PCPNet: Learning Local Shape Properties from Raw Point Clouds. *Computer Graphics Forum* 37, 2 (2018), 75–85.

Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. 1992. Surface Reconstruction from Unorganized Points. *SIGGRAPH Comput. Graph.* 26, 2 (July 1992), 71–78.

Fei Hou, Chiyu Wang, Wencheng Wang, Hong Qin, Chen Qian, and Ying He. 2022. Iterative Poisson Surface Reconstruction (iPSR) for Unoriented Points. *ACM Trans. Graph.* 41, 4, Article 128 (July 2022).

Yixin Hu, Teseo Schneider, Bolun Wang, Denis Zorin, and Daniele Panozzo. 2020. Fast tetrahedral meshing in the wild. *ACM Transactions on Graphics* 39, 4 (Aug. 2020).

Zhiyang Huang, Nathan Carr, and Tao Ju. 2019. Variational Implicit Point Set Surfaces. *ACM Trans. Graph.* 38, 4, Article 124 (July 2019).

ZhangJin Huang, Yuxin Wen, ZiHao Wang, Jinjuan Ren, and Kui Jia. 2024. Surface Reconstruction From Point Clouds: A Survey and a Benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* 46, 12 (Dec. 2024), 9727–9748.

Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson Surface Reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, Vol. 7.

Michael Kazhdan and Hugues Hoppe. 2013. Screened Poisson Surface Reconstruction. *ACM Transactions on Graphics (ToG)* 32, 3 (2013), 1–13.

Sören König and Stefan Gumhold. 2009. Consistent Propagation of Normal Orientations in Point Clouds.. In *VMV*, Vol. 83. 92.

David Levin. 2004. Mesh-independent surface interpolation. In *Geometric modeling for scientific visualization*. Springer, 37–49.

Qing Li, Huifang Feng, Kanle Shi, Yi Fang, Yu-Shen Liu, and Zhizhong Han. 2023. Neural Gradient Learning and Optimization for Oriented Point Normal Estimation. In *SIGGRAPH Asia 2023 Conference Papers* (Sydney, NSW, Australia) *(SA '23)*. Association for Computing Machinery, New York, NY, USA, Article 122, 9 pages.

Qing Li, Yu-Shen Liu, Jin-San Cheng, Cheng Wang, Yi Fang, and Zhizhong Han. 2022. HSurf-Net: Normal estimation for 3D point clouds by learning hyper surfaces. *Advances in Neural Information Processing Systems* 35 (2022), 4218–4230.

Siyou Lin, Zuoqiang Shi, and Yebin Liu. 2024. Fast and Globally Consistent Normal Orientation based on the Winding Number Normal Consistency. *ACM Transactions on Graphics* 43, 6 (Nov. 2024), 1–19.

Siyou Lin, Dong Xiao, Zuoqiang Shi, and Bin Wang. 2022. Surface Reconstruction from Point Clouds without Normals by Parametrizing the Gauss Formula. *ACM Trans. Graph.* 42, 2, Article 14 (Oct. 2022), 19 pages.

Weizhou Liu, Jiaze Li, Xuhui Chen, Fei Hou, Shiqing Xin, Xingce Wang, Zhongke Wu, Chen Qian, and Ying He. 2025. Diffusing Winding Gradients (DWG): A Parallel and Scalable Method for 3D Reconstruction from Unoriented Point Clouds. *ACM Trans. Graph.* (April 2025). Just Accepted.

Weizhou Liu, Xingce Wang, Haichuan Zhao, Xingfei Xue, Zhongke Wu, Xuequan Lu, and Ying He. 2024. Consistent Point Orientation for Manifold Surfaces via Boundary Integration. In *ACM SIGGRAPH 2024 Conference Papers (Siggraph '24)*. Association for Computing Machinery, New York, NY, USA, Article 54.

Yujie Lu, Long Wan, Nayu Ding, Yulong Wang, Shuhan Shen, Shen Cai, and Lin Gao. 2024. Unsigned Orthogonal Distance Fields: An Accurate Neural Implicit Representation for Diverse 3D Shapes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. 2021. Neural-Pull: Learning Signed Distance Functions from Point Clouds by Learning to Pull Space onto Surfaces. arXiv:2011.13495 [cs.CV]

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy Networks: Learning 3D Reconstruction in Function Space . In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 4455–4465.

Gal Metzer, Rana Hanocka, Denis Zorin, Raja Giryes, Daniele Panozzo, and Daniel Cohen-Or. 2021. Orienting Point Clouds with Dipole Propagation. *ACM Trans. Graph.* 40, 4, Article 165 (July 2021).

Niloy J. Mitra and An Nguyen. 2003. Estimating surface normals in noisy point cloud data. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry* (San Diego, California, USA) *(SCG '03)*. Association for Computing Machinery, New York, NY, USA, 322–328.

Shigeru Muraki. 1991. Volumetric Shape Description of Range Data Using "Blobby Model". *SIGGRAPH Comput. Graph.* 25, 4 (July 1991), 227–235.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation . In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 165–174.

Mark Pauly, Richard Keiser, Leif P Kobbelt, and Markus Gross. 2003. Shape modeling with point-sampled geometry. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 641–650.

Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. 2020. Convolutional Occupancy Networks. arXiv:2003.04618 [cs.CV]

Cédric Portaneri, Mael Rouxel-Labbé, Michael Hemmer, David Cohen-Steiner, and Pierre Alliez. 2022. Alpha wrapping with an offset. *ACM Trans. Graph.* 41, 4, Article 127 (July 2022), 22 pages.

Siyu Ren, Junhui Hou, Xiaodong Chen, Ying He, and Wenping Wang. 2023. GeoUDF: Surface Reconstruction from 3D Point Clouds via Geometry-Guided Distance Representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14214–14224.

Enrique Rosales, Chrystiano Araújo, Jafet Rodriguez, Nicholas Vining, Dongwook Yoon, and Alla Sheffer. 2021. AdaptiBrush: adaptive general and predictable VR ribbon brush. *ACM Trans. Graph.* 40, 6, Article 247 (Dec. 2021), 15 pages.

Enrique Rosales, Jafet Rodriguez, and ALLA SHEFFER. 2019. SurfaceBrush: from virtual reality drawings to manifold surfaces. *ACM Trans. Graph.* 38, 4, Article 96 (July 2019), 15 pages.

Nico Schertler, Bogdan Savchynskyy, and Stefan Gumhold. 2017. Towards Globally Optimal Normal Orientations for Large Point Clouds. *Computer Graphics Forum* 36, 1 (2017), 197–208. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12795

Silvia Sellán and Alec Jacobson. 2023. Neural Stochastic Poisson Surface Reconstruction. In *SIGGRAPH Asia 2023 Conference Papers* (Sydney, NSW, Australia) *(SA '23)*. Association for Computing Machinery, New York, NY, USA, Article 74, 9 pages.

Silvia Sellán and Alec Jacobson. 2022. Stochastic Poisson Surface Reconstruction. *ACM Transactions on Graphics* (2022).

Chao-Hui Shen, Guo-Xin Zhang, Yu-Kun Lai, Shi-Min Hu, and Ralph R. Martin. 2010. Harmonic Field Based Volume Model Construction from Triangle Soup. *Journal of Computer Science and Technology* 25, 3 (2010), 562–571.

Jonathan Richard Shewchuk. 1996. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In *Applied Computational Geometry: Towards Geometric Engineering*, Ming C. Lin and Dinesh Manocha (Eds.). Lecture Notes in Computer Science, Vol. 1148. Springer-Verlag, 203–222. From the First ACM Workshop on Applied Computational Geometry.

Hang Si. 2015. TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator. *ACM Trans. Math. Softw.* 41, 2, Article 11 (Feb. 2015), 36 pages.

W.S. Slaughter. 2002. *The inearized Theory of Elasticity.* Birkhäuser.

Zixiong Wang, Yunxiao Zhang, Rui Xu, Fan Zhang, Peng-Shuai Wang, Shuangmin Chen, Shiqing Xin, Wenping Wang, and Changhe Tu. 2023. Neural-Singular-Hessian: Implicit Neural Representation of Unoriented Point Clouds by Enforcing Singular Hessian. *ACM Trans. Graph.* 42, 6, Article 274 (Dec. 2023).

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3D ShapeNets: A deep representation for volumetric shapes . In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 1912–1920.

Hui Xie, Jianning Wang, Jing Hua, Hong Qin, and Arie Kaufman. 2003. Piecewise C/sup 1/continuous surface reconstruction of noisy point clouds via local implicit quadric regression. In *IEEE Visualization, 2003. VIS 2003.* IEEE, 91–98.

Rui Xu, Zhiyang Dou, Ningna Wang, Shiqing Xin, Shuangmin Chen, Mingyan Jiang, Xiaohu Guo, Wenping Wang, and Changhe Tu. 2023. Globally Consistent Normal Orientation for Point Clouds by Regularizing the Winding-Number Field. *ACM Trans. Graph.* 42, 4, Article 111 (July 2023).

Qingnan Zhou, Eitan Grinspun, Denis Zorin, and Alec Jacobson. 2016. Mesh arrangements for solid geometry. *ACM Trans. Graph.* 35, 4, Article 39 (July 2016), 15 pages.

Qingnan Zhou and Alec Jacobson. 2016. Thingi10K: A Dataset of 10,000 3D-Printing Models. *arXiv preprint arXiv:1605.04797* (2016).

Runsong Zhu, Yuan Liu, Zhen Dong, Yuan Wang, Tengping Jiang, Wenping Wang, and Bisheng Yang. 2021. AdaFit: Rethinking learning-based normal estimation on point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6118–6127.