Variational Graph-based Normal Integration

Lixiong Chen[1] Bohan Yu[2,3] Victor Adrian Prisacariu[1] Imari Sato[3]
[1] Active Vision Lab, University of Oxford [2]Peking University [3]National Institute of Informatics
{lchen, victor}@robots.ox.ac.uk, ybh1998@pku.edu.cn, imarik@nii.ac.jp

Abstract

We present a general optimization-based framework for depth-preserving normal integration. Unlike existing methods that operate on surface orientations defined over regular grids, our approach introduces a unified graph-based formulation capable of integrating semi-differentiable surfaces on unstructured domains. Given a set of points uniformly sampled from a surface, we construct a directed, weighted graph that jointly parameterizes the surface geometry and pairwise point correlations. Surface depth is recovered by minimizing projected point-to-plane distances across the graph, and we attain this objective through variational inference. In our formulation, estimated surface normals serve as latent variables that encode local geometry via the posterior probabilities of a two-component Gaussian mixture, allowing depth discontinuities to be explicitly inferred from sampled triplet configurations. The unknowns are estimated in an alternating fashion, and we provide a geometric interpretation of this inference process by relating it to shape deformation. Experimental results show that the proposed method not only delivers superior performance on regularly gridded data, but also generalizes effectively to scattered points, which existing approaches do not directly support.

1. Introduction

Normal integration establishes an inverse mapping from a surface normal map to its depth, and plays a critical role across fundamental low-level 3D vision tasks—including photometric stereo [31], shape-from-shading [16], learned image-to-normal prediction [3], and dense single-view surface reconstruction [14]. Existing approaches typically model the surface as a scalar height field and operate on gradients stored on regular grids [26]. As depth discontinuities introduce spatial irregularities that grid-based numerical differentiation cannot properly capture, extending depth-preserving normal integration to scattered oriented points, an increasingly common data representation in today's data-rich environment, remains largely under-explored.

We address two primary challenges in recovering surfaces with irregular spatial representations. Regarding formulation, we require a flexible geometric model that parameterizes surface shape using randomly distributed points while explicitly modeling depth discontinuities. Operationally, we design an optimizer that processes sampled surface normals to reliably estimate these model parameters. These challenges are fundamentally coupled: because model complexity scales inversely with solver complexity, we must balance model parameterization against optimization constraints. For instance, field conservation is often imposed via separate heuristic routines [1], while depth discontinuities are handled through per-pixel, axis-aligned comparators [7].

This work presents a framework that offers a coherent formulation, systematic analysis, and effective depth estimation for semi-differentiable surfaces with flexible representations. We introduce a graph-based formulation that not only enables expressive surface parameterization in the presence of depth discontinuities, but also provides an interpretable and stable optimization strategy for surface depth recovery. Oriented points are treated as samples drawn from a surface and construct a weighted, directed graph in which each edge encodes the geometric compatibility between two nearby points. Correspondingly, projected point-to-plane distance is introduced to quantify this compatibility, which we use to measure the likelihood of existence of a depth discontinuity. Building on this, we derive an optimization problem that estimates surface depth by finding its most likely geometric configuration.

We solve this optimization problem iteratively within a variational inference framework. We construct per-point triplets, each consisting of two directed edges that share a common vertex, to analyze pairwise point interactions. Each triplet is modeled using a two-component Gaussian mixture, in which the shared vertex is treated as a sample drawn from a joint distribution by its two neighboring points. The resulting pairwise correlation can then be inferred as the posterior probability. At each iteration, the per-point normals of the estimated surface are updated as latent variables that govern the sampling process. This infer-

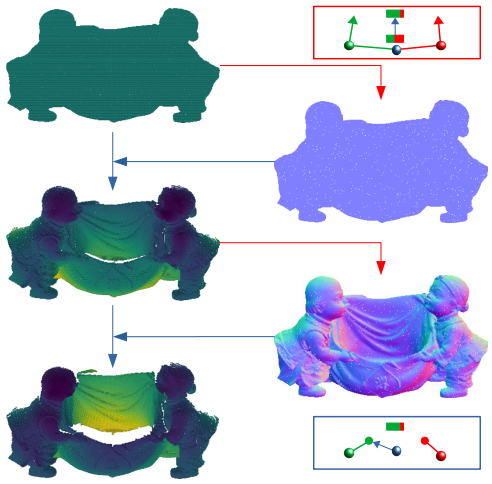


Figure 1. Graph-based normal integration pipeline. The optimizer operates within a variational inference framework and deforms an initial surface (e.g., a flat plane) toward the target shape by alternating between two stages: (1) Depth estimation (Sec. 4.1), where each point is repositioned to a new location along its viewing ray proposed based on the geometric compatibility of its local neighborhood; and (2) Shape inference (Sec. 4.2), where pairwise compatibilities are refined based on the current estimated geometry.

ence procedure yields a surface whose shape continuously deforms from its initial configuration toward the target geometry, as illustrated in Figure 1.

As a result, we propose an optimization-based framework for surface normal integration over randomly distributed oriented points. The contributions of this work are as follows:

1. We develop a graph-based framework that formulates normal integration as an optimization problem. Our approach derives a unified objective that enables a consistent treatment of both grid-based and scattered data representations.
2. We design an iterative optimizer that operates within a variational inference framework. Each iteration deforms the surface shape toward its target, where per-point estimated normals are the latent variables that guide this inference process.
3. We generalize the design of BiNI [7] by introducing a per-point triplet sampling scheme that effectively addresses depth discontinuities of flexible representations.

The remaining of this paper is organized as follows: Section 2 provides an overview of the existing literature; Section 3 derives a unified formulation for normal integration and introduces a graph-based model for object shapes in the presence of depth discontinuities. Section 4 presents an optimization routine for surface depth estimation derived from variational inference. Section 5 summarizes and analyzes the experiment results. Section 6 discusses future work and concludes this paper.

2. Related Work

We divide the related work into two categories: classical normal integration techniques and modern learning-based approaches for point cloud processing.

2.1. Normal Integration

Normal integration is an inverse problem usually solved by a PDE solver. Particularly, a valid solution is expected to minimize an energy-based functional [16]:

$$\min_{z} \int_{\Omega} w_u(\vec{x})E(\partial_u z - p(\vec{n})) + w_v(\vec{x})E(\partial_v z - q(\vec{n}))dudv, \tag{1}$$

where $p = -\frac{n_x}{n_z}$, $q = -\frac{n_y}{n_z}$, $\partial_u z$ and $\partial_v z$ are evaluated on a regular grid, and orthogonal constraint is introduced to enhance numerical stability [38]. Counterpart representation exists for perspective projections in log-depth space [12, 26]. w_u and w_v weighing the importance of the constraint accommodate depth discontinuities [13] whose parameterization is a design choice. Ideally, energy attains minimum when w_u and w_v vanishes at the depth discontinuity. The weighting function can be evaluated either statically or dynamically [2, 25]. In a static process depth edge detection is performed by directly analyzing normal map [32], handcrafting [34] or inferring from photometric cues [30], etc.. A dynamic processes depth estimation online and updates the weighting function iteratively in the form $w(\vec{n}, z)$[1, 27]. Bilateral Normal Integration (BiNI) and its variants [7, 19] represent the current state of the art, applies a semi-differentiable connectivity pattern in a local neighborhood to parameterize the weighting function. Our solution belongs to this category and is similar to a Poisson reconstruction problem [18] but z-value of each sample point is not provided.

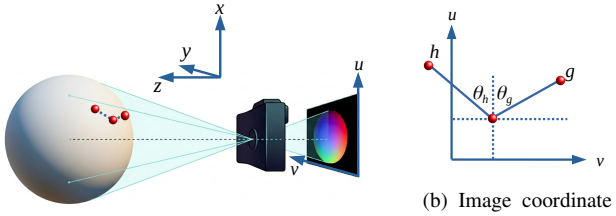
2.2. Point Cloud Processing and Learning

Learning point-to-point correlation to obtain a consistent per-point normal utilizing for point cloud processing is a well-defined task. Per-point normal of a scattered point can be estimated as a weighted average of its neighboring samples [4]. This can be parameterized using polynomials whose coefficients are to be determined [8]. A location-adaptive polynomial [39] improves this supervised learning process. Moreover, an implicit 0-level set can be integrated to promote the global orientation consistency of the normal estimation [21]. This formulation also applies to shading analysis [35]. In addition, graph Laplacian regularizes point

cloud filtering [9, 10] effectively. This line of work mainly focuses on denoising, where per-point depth of point cloud is provided [11]. In contrast, we are tackling a problem where neither the structure of the point cloud or the underlying graph is known as a-priori. Moreover, while recent development of 3D foundation models that generate fine-textured 3D shapes guided by images [22, 33, 37] has dominated in image-to-shape conversion, the advantage of normal integration persists in ensuring superior image-shape consistency.

3. Graph-based Shape Modeling

We formulate normal integration on a directed weighted graph. Given the normal map of a semi-differentiable surface, we first formulate the formation of a surface's normal map, based on which we introduce the projected point-to-plane distance in the object coordinates and use it to derive a per-edge objective function for a minimization problem. Furthermore, minimized distance is achieved by the most likely shape configuration obtained through variational inference. In particular, the minimizer is equipped with a flexible sampling scheme of triplet configurations where depth discontinuities in each per-point neighborhood are effectively handled.



(a) Normal map formation.

(b) Image coordinate conversion.

Figure 2. Projection of an object's surface normals onto the image plane. (a) Surface gradients (z_x, z_y) defined object space are transformed into image coordinate system (u,v). (b) For randomly distributed measurements, we establish a local coordinate system (h,g) based on point connectivity. These local coordinates are derived from the standard u-v coordinates via an affine transformation.

3.1. Normal Map Formation

A normal map represents the local orientation of an object's surface by storing the unit normal vector at each point, indexed by its corresponding image coordinates, as illustrated in Fig. 2a. Without loss of generality, let a semi-differentiable surface be parameterized as $z(h,g) = e^{d(x(h,g),y(h,g))}$, where w and g denote a pair of linearly-independent axes defining a local coordinate system relating to image coordinates as $u = g\cos\theta_g + h\cos\theta_h - o_u$, $v = g\sin\theta_g + h\sin\theta_h - o_v$ (Fig. 2b), the total derivative of a z taken with respect to h yields:

$$\frac{\partial z}{\partial h} = z_h = z_x x_h + z_y y_h, \tag{2}$$

where $z_h = zd_h$, $x_h = \frac{1}{f^x}(z_h u + z\cos\theta_h)$ and $y_u = \frac{1}{f^y}(z_h v + z\sin\theta_h)$ with f denoting the camera's focal length. Since $z > 0$ cancels on both sides, Eq. (2) can be reduced to:

$$(n_z + n_x\frac{u}{f^x} + n_y\frac{v}{f^y})\Delta_h d + n_x\frac{\Delta_h u}{f^x} + n_y\frac{\Delta_h v}{f^y} = 0, \tag{3}$$

by plugging in $n_x = -z_x n_z$, $n_y = -z_y n_z$. This equation indicates the following:

- $d(x,y)$ and $z(x,y)$ share the same normal map.
- When d_h is evaluated by a numerical depth difference, $\Delta_h d$, along segment h, Eq. (3) involves two samples of $z(h,g)$, coupling the geometry of two points of a differentiable surface.
- The equality defines a per-edge objective in a graph that minimizes a scaled distance of a point to a plane passing though its nearby surface.
- Two equations, derived for z_h and z_g, respectively, sufficiently parameterize the surface gradient at any point.

Furthermore, orthographic projection arises as $f \to \infty$ (*i.e.* $\frac{u}{f} \to 0$), $z \to \infty$, leading to $fd_u \approx z_u$, which yields

$$n_z\Delta_h z + n_x\Delta_h x + n_y\Delta_h y = 0. \tag{4}$$

This reduction motivates us to investigate shape modeling in the object space before extending it to perspective projections in the log-depth space, where n_z, $\Delta_h z$,$\Delta_h x$ and $\Delta_h y$ are replaced by their transformed counterparts $\bar{n}_z = f^x n_z + n_x u + \frac{f^x}{f^y}v$, $\Delta_h u$ and $\Delta_h v$, respectively.

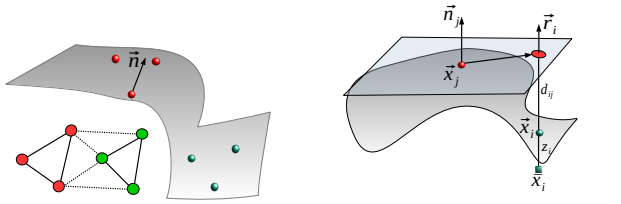
3.2. Object Shape as a Graph

Given a set of 3D points sampled from a surface with per-point surface normal as exemplified in Figure 3a, we construct a directed graph each of whose vertices represents a point, and the graph connectivity is established offline according to their x-y coordinates (*e.g.* KNN), and their z coordinates quantify the corresponding depth along the projection ray $\vec{r}$ as illustrated in Figure 3b, where $\bar{x}$ represents the optical center from which per-point relative depth is measured. In object coordinates, $\bar{x} = (x, y, 0)$, $\vec{r} = (0, 0, 1)$. Each directed edge encodes an equation involving the two vertices it connects. Let node i extend an edge that connects node j, and $\vec{n} = (n_x, n_y, n_z)$ denote its unit normal, then the following condition of orthogonality is satisfied:

$$(\vec{x}_i + d_{ij}\vec{r}_i - \vec{x}_j)\vec{n}_j = 0, \tag{5}$$

where d_{ij} is the distance of $\vec{x}_i$ onto the plane extending out from point $\vec{x}_j$ along ray $\vec{r}_i$. This can be decomposed as:

$$(\bar{x}_j - \bar{x}_i)\vec{n}_j + (z_j - z_i)\vec{r}_i \cdot \vec{n}_j = d_{ij}\vec{r}_i \cdot \vec{n}_j = d_p(i,j), \tag{6}$$



(a) A graph representation of a semi-differentiable surface.

(b) Projected point-to-plane distance is introduced to measure edge-wise correlation.

Figure 3. A semi-differentiable surface is represented by a directed weighted graphs. (a) Points in differentiable regions are strongly correlated, while points separated by depth edges are weakly correlated; (b) Each directed edge is parameterized using the projected point-to-plane distance defined in Eq. (6). $d_p(i,j)$ measures how $\vec{x}_j$ is correlated to $\vec{x}_i$. A triplet configuration (Sec. 3.3) consists of two of such per-edge settings. $d_p(ij)$ measures surface smoothness and by minimizing $\sum_{ij} |d_p(ij)|$.

where we define $d_p(i,j)$ to be the projected point-to-plane distance from $\vec{x}_i$ to $\vec{x}_j$, an asymmetric signed distance measure that vanishes when the two points are co-planar.

Depending on the projection type, $d_p(i,j)$ measures the discrepancy of Eq. (3) or Eq. (4), and this provides a geometric interpretation for our goal to minimize $\sum_{i,k} |d_p(i,j)|_2^2$ for all directed edges. However, if there is depth discontinuity between $\vec{x}_i$ and $\vec{x}_j$, Eq. (6) no longer holds, and this can be captured by a nonnegative unit edge weight, w_{ij}, with 0 edge weight indicating complete dissociation and 1 indicating complete correlation. In other words, we want to design a normal integrator that finds a depth configuration amounts to:

$$\arg\min_{\vec{z}, w_{ij}} \sum_{i,j} w_{ij} d_p(z_i, z_j), \tag{7}$$

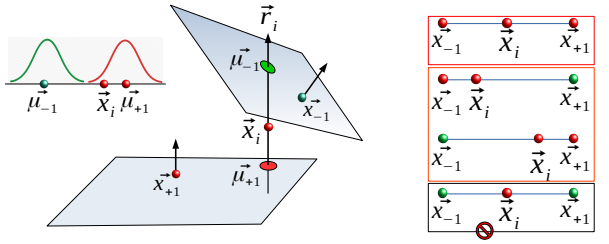
where $\vec{z}$ is replaced by $\vec{d}$ when the distance is measured in the log-depth space. In both projection settings it defines a unified weighted least square problem and each edge weight w_{ij} is captured by sampling triplets that parameterize a semi-differentiable surface.

3.3. Configuring Triplet

A triplet is formed by three oriented points connected by two edges in a per-point neighborhood as illustrated in Figure 4a: the neighborhood center $\vec{x}_0$ is joined by two edges connecting with its two neighbors $\vec{x}_{+1}$ and $\vec{x}_{-1}$, respectively. According to Sec. 3.2, this yields two coupled equations for $\vec{x}_0$. Specifically, $d_p(0,+1)$ and $d_p(0,-1)$ are obtained by $\vec{n}_{-1}$ and $\vec{n}_{+1}$ with three patterns of connectivity as depicted in Figure 4b:

1. No depth discontinuity exists in the neighborhood, where $\vec{x}_0$ is compatible with both $\vec{x}_{-1}$ and $\vec{x}_{+1}$, and $d_p(0,-1)$ and $d_p(0,+1)$ are approximately equal;
2. A depth discontinuity cuts in between $\vec{x}_0$ and one of $\vec{x}_{-1}$ and $\vec{x}_{+1}$, and one-sided compatibility leads to significantly large $|d_p(0,-1) - d_p(0,+1)|$;
3. A depth discontinuity cuts in between $\vec{x}_0$ and both of $\vec{x}_{-1}$ and $\vec{x}_{+1}$, where little about the depth of $\vec{x}_0$ can be inferred about $|d_p(0,-1) - d_p(0,+1)|$ due to complete detachment.

Because these discontinuity patterns are structured and sparse, case 3 arises often when $\vec{x}_{-1}$ and $\vec{x}_{+1}$ are co-located, but this scenario can be excluded by importance edge sampling explained in Sec. 4.3.



(a) A triplet represented by a two-component Gaussian mixture.

(b) Three pair-wise correlation scenarios.

Figure 4. Each point is associated with two neighbors to form a triplet to model pair-wise correlations. (a) Each triplet jointly weighs two per-edge distances in Figure 3b using a two-component Gaussian mixture; (b) Three scenarios exist for $\vec{x}_i$ to infer its correlations with $\vec{x}_{-1}$ and $\vec{x}_{+1}$. Only two are valid because we assume at least one of the two vertices is compatible with $\vec{x}_i$, we design an importance sampler to ensure configuration validity (Sec. 4.3).

The edge weight in the triplet configuration, $w_{0,-1}$ and $w_{0,+1}$, can be modeled using a two-component Gaussian mixture capturing the geometry of semi-differentiable surfaces. Specifically, $\vec{x}_0$ is treated as a sample jointly generated by the mixture of two Gaussian as:

$$p(\vec{x}_0) = \pi_{+1}\mathcal{N}(\vec{x}_0|\vec{x}_{+1}, \vec{n}_{+1}) + \pi_{-1}\mathcal{N}(\vec{x}_0|\vec{x}_{-1}, \vec{n}_{-1}), \tag{8}$$

where each individual Gaussian component is defined as:

$$\mathcal{N}(\vec{x}_0|\vec{x}_{+1}, \vec{n}_{+1}) = e^{-d_p^2(0,+1)\cdot k}. \tag{9}$$

The precision variable k serves as a perceptually subjective measure for geometric compatibility, and $\pi_{+1} + \pi_{-1} = 1$ are the prior probabilities for the two Gaussian components [23]. In this case, posterior probabilities in the closed form, $w_{0,+1} = 1 - w_{0,-1}$, are read as:

$$w_{0,+1} = p(\vec{x}_0 \sim \vec{x}_{+1}|\vec{x}_0) = \sigma_k(d_p^2(0,-1) - d_p^2(0,+1)), \tag{10}$$

where $\sigma_k(\cdot)$ is the sigmoid function tuned by k.

In comparison, BiNI [7] represents a special instance of our setting with more restrictive assumptions:

1. Triplets are configured deterministically for each per-point neighborhood, and they are axis-aligned;
2. Equal prior probabilities of the Gaussian components are implicitly assumed, and they are defined directly in terms of depth rather than geometry in Eq. (6).
3. Its precision variable k is fixed as opposed to being location adaptive in our design (Sec. 4.2).

3.4. Normal Integration as Optimization

Triplets introduced in Sec. 3.3 are obtained by sampling, where each sampled configuration yields a minimum distance problem to be solved by a gradient-based likelihood maximizer. Specifically, a single triplet yields two terms in Eq. (7), and given N nodes and K-connectivity, we obtain $2KN$ terms to be jointly optimized in terms of $\vec{z} \in \mathcal{R}^N$ (or $\vec{d} \in \mathcal{R}^N$ in log-depth space). As explained in Sec. 3.1, $K \geq 2$ is necessary to determine surface orientation. Let D_z be a $2KN$-by-N weighted incidence matrix whose $Ki+j$-th row stores $-n_{j,z}$ and $n_{j,z}$ (or $-\bar{n}_{j,z}$ and $\bar{n}_{j,z}$) in its i-th and j-th locations, respectively. Let $\vec{b}$ be a $2KN$-by-1 whose $Ki+j$-th row stores $n_x\Delta_h x + n_y\Delta_h y$ (or $n_x\Delta_h u + n_y\Delta_h v$), and W be a $2KN$-by-$2KN$ diagonal matrix whose $Ki+j$-th entry along its diagonal stores the corresponding edge weight w_{ij}, the optimality condition for the minimal total projected point-to-plane distance can be written as:

$$D_z^T W D_z \vec{z} = -D_z^T W \vec{b}. \tag{11}$$

In Sec. 4 we show that $\vec{z}$ (or $\vec{d}$) and W are treated as two separate sets of variables estimated alternatively in the framework of variational inference.

4. Graph-based Normal Integration

To solve both W and $\vec{z}$ in Eq. (11), we alternatively estimate one group of variables while keeping the other group fixed. We implement an iterative optimization routine using variational inference as $\vec{z}$ being samples generated by some specific shape configured by W. Two operations are executed consecutively in each iteration indexed by t:

- *depth estimation*: given W^t, solve for $\vec{z}^{t+1}$ from the Eq. (11) using conjugate gradient method [28].
- *shape inference*: given $\vec{z}^{t+1}$, infer the shape structure W^{t+1} through sampled triplets described in Sec. 3.3.

An identical procedure follows for $\vec{d}$. We relate these two stages to the learning of the structure of two-component Gaussian mixture to understand the optimization process.

4.1. Depth Estimation

Updating $\vec{z}^t$ to $\vec{z}^{t+1}$ (or $\vec{d}^t$ to $\vec{d}^{t+1}$) deforms the current (log-depth) shape to a state that is geometrically more compatible with the structure prescribed by W^{t+1}. The conjugate gradient designates the expected location for each surface point $\vec{x}_i$ to move to in terms of a distribution prescribed by its two neighbors $\vec{x}_{-1}$ and $\vec{x}_1$ in each sampled triplet according to Eq. (8). Because W^{t+1} determines the generation of $\vec{x}^{t+1}$, variational inference allows it to be directly inferred from the shape. In particular, in Sec. 4.2 we show that latent W^{t+1} can be obtained using the estimated normals of $\vec{z}^{t+1}$ according to Eq. (10).

4.2. Local Shape Inference

We estimate the gradient field of $\vec{z}$, $\vec{g}(x,y)$, to capture the dependencies caused by vertex sharing across multiple triplets to achieve graph-wise consistent inference. Equation 6, 10, and 12 together establish a consistent parameterization of W^t in terms of $\vec{z}^t$ through $d_p(i,j)$. This term involves $\vec{n}^t$, the surface normal estimated at iteration t.

Let the height function, $(x, y, z^t(x,y))$, denote the surface at t-th iteration. Correspondingly, its gradient field satisfies the first order approximation for a differential surface at point $\vec{x}_i$:

$$\delta z_{ij} = g_{i,x}\delta x_{ij} + g_{i,y}\delta y_{ij}, \tag{12}$$

where the gradient $g_{i,x}$ and $g_{i,y}$ taken at $\vec{x}_i$ are the unknowns to be estimated. Following a similar analysis described in Sec. 3.4, each sampled triplet yields two equations following Equation 12, let Δ_{xy} be a $2KN$-by-$2N$ matrix, and W represents the same $2KN$-by-$2KN$ diagonal matrix defined in Equation 11, Δ_z be the $2KN$-by-1 vector, we solve the $2N$-by-1 unknown $\vec{g}$ following:

$$\Delta_{xy}^T W \Delta_{xy} \vec{g} = -\Delta_{xy}^T W \Delta_z, \tag{13}$$

with conjugate gradients [28]. Consequently, each one of N surface points receives a gradient vector in 2-tuple, and the estimated normal, $\vec{n}^t$, can be obtained by rearranging $g_x n_z = n_x$, $g_y n_z = n_y$ and $n_z = \frac{1}{\sqrt{g_x^2+g_y^2+1}}$. Moreover, let $\vec{n}_{gt}$ and $\vec{n}_{est}$ be the ground truth and estimated per point surface normal, respectively. Because the cosine distance between this two quantities often increases at discontinuities, we use this metric to guide the the location-adaptive precision variable in Eq. (9):

$$k(\vec{n}_{gt}, \vec{n}_{est}) = 1 - \vec{n}_{gt}\vec{n}_{est}. \tag{14}$$

In other words, we let distance discrepancy be easily detected only when the vertex at the triplet center, $\vec{x}_0$, is highly likely to dissociate itself from one of its neighbors. In practice, we also implement a conservative updating strategy by clipping $d_p(i,j)$ caused by the high frequency component is removed $\vec{n}_{est}$ by $d_p(i,j) = \min(d_p(i,j), \delta z_{ij})$.

4.3. Sampling of Triplets

As discussed in Sec. 3.3, we want to avoid triplet configurations that $\vec{x}_0$ in the center is incompatible with both of its neighbors. To this end, we propose an importance edge

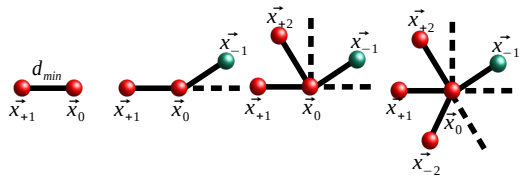


Figure 5. Triplet sampling for neighborhood centered at $\vec{x}_0$. Four neighbors are selected to form two triplets, which are necessary to characterize the geometry of a semi-differentiable surface. As discussed in Sec. 4.3, based on the projected point-to-plane distance and edge-wise angular separations on the image plane, the points $\vec{x}_{+1}$,$\vec{x}_{-1}$,$\vec{x}_{+2}$ and $\vec{x}_{-2}$ are selected from the neighborhood sequentially.

sampling scheme that produces a connectivity pattern in which edges are well-separated. Specifically, at each iteration the edge connecting the most compatible vertex (*e.g.* $\vec{x}_{0,+1}$) that produces the minimum distance is selected to ensure one-sided attachment. Subsequently, $\vec{x}_{0,-1}$ is sampled to make the angle between the two edges in the triplet closest to π.

The second triplet connecting $\vec{x}_{0,+2}$, $\vec{x}_{0,-2}$ and $\vec{x}_0$ following the same procedure. Specifically, to avoid colinearity, the edge connecting $\vec{x}_{0,+2}$ and $\vec{x}_0$ is selected that makes an angle closest to $\frac{\pi}{2}$ with the edge connecting $\vec{x}_{0,+1}$ and $\vec{x}_0$. Similarly, $\vec{x}_{0,-2}$ is sampled to make the angle between the two edges in the second triplet closest to π. This process is depicted in Fig. 5.

5. Experiment

Two types of experiments are conducted to validate our solution. First, to verify backward compatibility, we benchmark our method against existing approaches on datasets with grid representations to ensure a fair comparison. Subsequently, we apply our method to datasets with scattered point representations. We compare our approach against a baseline method constructed on a graph of identical topology where edge re-weighting through triplet sampling is disabled. It is worth noting that, the baseline method also utilizes the graph-based formulation presented in Sec. 3 to address unstructured representations.

Our experiment setup mainly focuses on orthographic projections in our experiments because both projection types share a similar formulation. Nevertheless, the benchmark results on grid-based data under perspective projection are reported in Tab 5. Notably, our method handles both input settings consistently, requiring no special configuration or mode switch during experiments.

DiLiGenT-MV [20] [29], a data set containing various shapes with ground truth, is the main source of our experiment input. Because the normal map of the same object varies significantly depending on the observer's viewing angle, to calibrate under orthographic projections, we use its mesh representation to synthesize the ground truth normal map [36] viewed from both the front and the side. We apply Halton sampler [5] to the synthesized normal maps to generate scattered data points where axis alignment is completely removed. The sampler keeps approximately 20% points from the points created on a dense grid. In addition, our method is also experimented against real world data originally used for implicit surface reconstruction [17].

We use Scipy to implement large scale sparse matrices [24]. Estimation results are evaluated using Mean Absolute Depth Error (MADE). Bilateral Normal Integration (BiNI) [7], inverse plane fitting (IPF) normal integration [6] and mesh decimation (MD) normal integration [15] (serving as the baseline) are used for comparison as their implementations are publicly available. BiNI represents a recent state of the art method that outperforms methods that precede it by a noticeable margin due to its success in depth discontinuity modeling. Our Graph-based Normal Integration is denoted as "GNI".

Front	bear	buddha	cow	pot	reading
GNI(ours)	**0.60**	**1.91**	5.62	**0.59**	**2.42**
BiNI [7]	0.74	2.07	**0.65**	1.30	14.73
IPF [6]	24.29	8.23	2.50	4.07	33.59
MD [15]	0.68	6.73	1.67	1.33	10.50

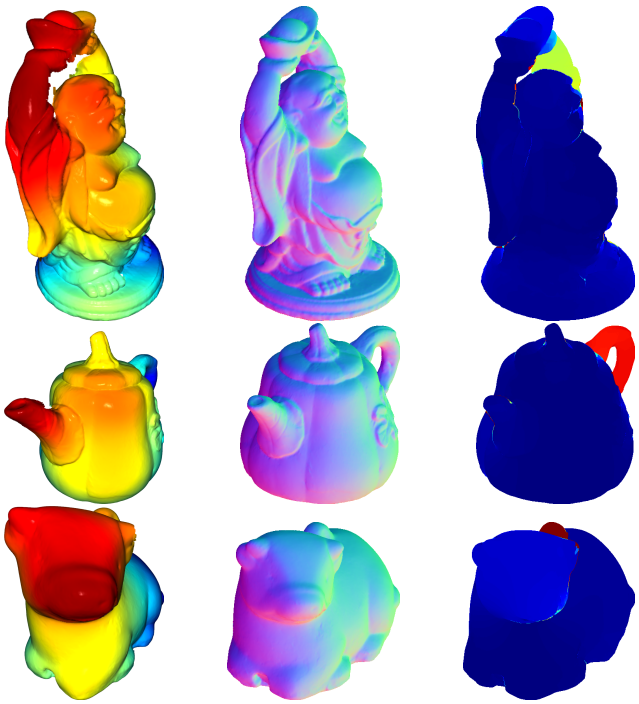
Table 1. Error of depth estimation evaluated in MADE on our data synthesized from DiLiGenT-MV [20]. Normal maps are synthesized from front viewing direction for each model. Our method outperforms BiNI on 4 of the 5 models.

Side	bear	buddha	cow	pot	reading
GNI(ours)	**1.24**	4.04	**0.36**	**7.2**	**2.87**
BiNI [7]	1.72	**3.81**	9.84	11.51	5.06
IPF [6]	29.29	10.79	38.66	17.35	10.67
MD [15]	13.46	20.99	42.44	19.84	31.23

Table 2. Error of depth estimation evaluated in MADE on our data synthesized from DiLiGenT-MV [20]. Normal maps are synthesized from side viewing direction for each model. Our method outperforms BiNI on 4 of the 5 models.

5.1. Summary of Benchmark Results

The performance of Graph-based Normal Integrator (GNI) and the other three methods on estimating the depth over grid data DiLiGenT-MV is compared in Tab 1 and 2, respectively. Each per-point neighborhood is generated through a



(a) est depth. (b) est normal. (c) error map.

Figure 6. Three objects (buddha (side), pot (side), cow (front)) with grid representations on which our solution delivers relatively large estimation error. As explained in Section 5.2, the primary source of error is the structural ambiguities due to single view measurements, where object shape with multiple disconnected parts caused by self-occlusions are fit into a singly-connected graph component. This scenario falls outside the scope of our problem definition. Numerical results are tabulated in Table 1 and 2. The maximum value set for the heat map is 100 evaluated in MADE.

K-nearest neighbor search with $K = 8$ from which two triplets are sampled as exemplified in Fig. 5. It can be seen from both tabulated results that our approach consistently outperforms BiNI on most models. Hence, it can be concluded that GNI is backward compatible in processing data with grid representations.

The performance of depth estimations for scattered oriented points on the same dataset is also summarized in Table 3 and 4. GNI with triplet sampling (w tri.) follows the same setting that is used to produce Table 1 and 2, whereas GNI without triplet sampling (w/o tri.) serves as a baseline method to verify the effectiveness of triplet configurations for depth preservation. In particular, we make three observations through per-table and cross table comparisons: (1) Table 1 and 2 shows that the formulation of point-to-plane distance is effective in terms of modeling general object shape in comparison with BiNI; (2) Table 3 and 4 provide convincing evidence that sampled triplets are also effective in addressing depth discontinuities in data with scattered representations, as prominent depth edges are

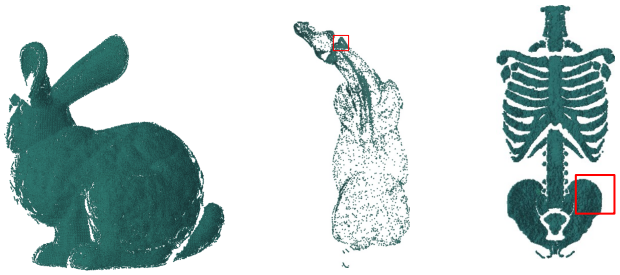
Front	bear	buddha	cow	pot	reading
GNI(w/ tri.)	1.26	6.05	18.72	1.29	5.08
GNI(w/o tri.)	1.91	7.71	35.70	1.73	10.14

Table 3. Error of depth estimation evaluated in MADE on DiLiGenT-MV [20] using the proposed graph-based formulation with and without triplet sampling. Triplet configuration significantly improves estimation accuracy.

Side	bear	buddha	cow	pot	reading
GNI(w/ tri.)	5.99	9.65	1.15	14.44	17.62
BS(w/o tri.)	17.74	18.48	2.21	19.35	31.23

Table 4. Error of depth estimation evaluated in MADE on DiLiGenT-MV [20] using the proposed graph-based formulation with and without triplet sampling. Triplet configuration significantly improves estimation accuracy through dedicated depth edge processing.

observed from the side view of "pot" and "reading"; (3) performance variations across different objects are consistent over both data representations (*i.e*. grid vs. scattered). By comparing Tab 1 with Tab 3 and Tab 2 with Tab 4, we see that the variation of error is "amplified" by scattered representations (*e.g*. performance degradation in "cow" (front)). Section 5.2 explains the cause of this.



(a) Bunny (GNI). (b) Horse (GNI). (c) Torso (GNI).

Figure 7. GNI assumes that surfaces are represented by uniformly sampled oriented points, but two cases demonstrate violations of this assumption. In contrast to the bunny dataset, where the points are largely uniform, portions of the horse exhibit irregular sampling, and a substantial fraction of the torso points are highly unevenly distributed. Consequently, the estimation accuracy degrades in these regions. This degradation arises because the uniform triplet sampler described in Section 4.3 does not adequately capture strong spatial irregularities.

5.2. Analysis

Figure 6 visualizes the estimated depth, estimated normal and the corresponding estimation error of the objects (buddha (side), pot (side) and cow (front)) on which our results

Model	bear	buddha	cat	cow	goblet	harvest	pot1	pot2	reading
BiNI [7]	0.49	0.77	0.10	0.06	8.51	3.34	0.62	0.21	0.32
GNI (ours)	0.47	0.73	0.11	0.06	9.75	2.83	0.59	0.20	0.26

Table 5. Performance comparison between our method and BiNI on the perspective DiLiGent dataset. Apart from "goblet", which presents prominent self-occlusions, our approach consistently delivers accurate reconstructions on real-world inputs. The point to plane distance is evaluated in the log-depth space.

are less accurate comparing with other objects as indicated in Tab 1 and 2. It is evident that the primary source of error arises from the structural ambiguities caused by self-occlusions that are inherent to single-view measurements. In particular, erroneous regions are completely detached from the main object body due to occlusions, and in this case their relative depths to the object cannot be inferred, despite that the per-point normals and local surface geometry can be provided. In the context of graph-based formulation, this amounts to two separate graphs rather than a single connected component. In fact, this form of ambiguity cannot be resolved by surface geometry in single view, and thus falls outside the scope of our problem definition.

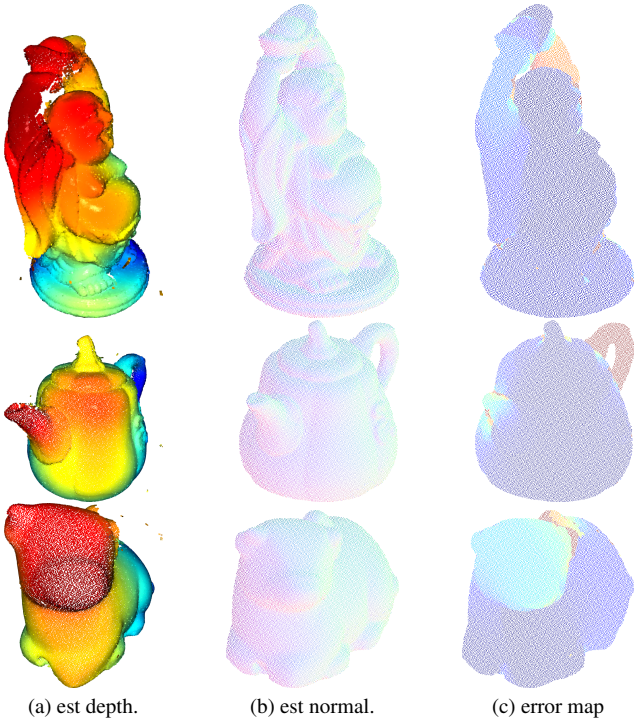
As suggested by Fig. 8, the errors caused by occlusion become more pronounced with scattered representations, because depth discontinuities often coincide with occluded regions, and identifying and handling these regions is inherently more challenging than when working with dense and structured measurements. Nevertheless, our reconstructions from sparse and unstructured inputs remain largely consistent with those obtained from dense grid data. Our graph-based design demonstrates strong resilience to distributional irregularities, and it also correctly handles depth discontinuities.

5.3. Future Work

The input points are assumed to be uniformly sampled from the surface whose depth is to be estimated. However, as illustrated in Fig. 7, this assumption does not always hold for real-world data. We apply our method to three objects commonly used for Poisson surface reconstruction [18]: "bunny", "horse", and "torso". Points in "bunny" are mostly uniformly distributed, so we deliver accurate depth estimates accordingly. Some points are irregularly distributed in a small region of "horse", and this leads to less accurate estimations in these areas. Comparatively, a substantial portion of "torso" consists of unevenly distributed points, which requires the triplet configuration to adapt to this pattern. We leave the potential improvement in triplet sampling for future investigation.

6. Conclusion

In this paper, we introduced a graph-based framework for depth-preserving normal integration. By formulating the problem within a variational inference framework, we developed a unified objective function that effectively handles both grid-based and scattered data representations. Our approach leverages point triplets modeled by a two-component Gaussian Mixture, allowing the optimizer to maintain high reconstruction accuracy even in the presence of depth discontinuities. Experimental results on both synthetic and real-world datasets demonstrate that our method provides a flexible and accurate solution for recovering geometry from surface normals. Future work will focus on extending this framework to enhance robustness when handling points with sparse and non-uniform distributions.



(a) est depth. (b) est normal. (c) error map

Figure 8. Three objects (buddha (side), pot (side), cow (front)) with scattered representations on which our solution delivers relatively large estimation error. GNI presents consistent performance over data in both grid and scattered representations. Despite distributional irregularities and sparsity, GNI delivers both accurate depth and normal estimations. Numerical results are tabulated in Table 3 and 4. The maximum value set for the heat map is 100 evaluated in MADE.

References

[1] Amit Agrawal, Ramesh Raskar, and Rama Chellappa. What is the range of surface reconstructions from a gradient field? In *ECCV*, 2006. 1, 2

[2] Hicham Badri, Hussein Yahia, and Driss Aboutajdine. Robust surface reconstruction via triple sparsity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 2

[3] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, 2024. 1

[4] Yizhak Ben-Shabat and Stephen Gould. Deepfit: 3d surface fitting via neural network weighted least squares. In *ECCV*, 2020. 2

[5] Michael Berblinger and Christoph Schlier. Monte carlo integration with quasi-random numbers: some experience. *Computer physics communications*, 1991. 6

[6] Xu Cao, Boxin Shi, Fumio Okura, and Yasuyuki Matsushita. Normal integration via inverse plane fitting with minimum point-to-plane distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 6

[7] Xu Cao, Hiroaki Santo, Boxin Shi, Fumio Okura, and Yasuyuki Matsushita. Bilateral normal integration. In *European Conference on Computer Vision*, 2022. 1, 2, 4, 6, 8

[8] Frédéric Cazals and Marc Pouget. Estimating differential quantities using polynomial fitting of osculating jets. *Computer aided geometric design*, 2005. 2

[9] Haolan Chen, Shitong Luo, Wei Hu, et al. Deep point set resampling via gradient fields. *IEEE TPAMI*, 2022. 3

[10] Chinthaka Dinesh, Gene Cheung, Ivan V. Bajić, and Cheng Yang. Local 3d point cloud denoising via bipartite graph approximation & total variation. In *International Workshop on Multimedia Signal Processing (MMSP)*, 2018. 3

[11] Chinthaka Dinesh, Gene Cheung, Saghar Bagheri, and Ivan V Bajic. Efficient signed graph sampling via balancing & gershgorin disc perfect alignment. 2024. 3

[12] Jean-Denis Durou and Frédéric Courteille. Integration of a normal field without boundary condition. In *Proceedings of the First International Workshop on Photometric Analysis For Computer Vision*, 2007. 2

[13] Jean-Denis Durou, Jean-François Aujol, and Frédéric Courteille. Integrating the normal field of a surface in the presence of discontinuities. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2009. 2

[14] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 1

[15] Moritz Heep, Sven Behnke, and Eduard Zell. Feature-preserving mesh decimation for normal integration. In *CVPR*, 2025. 6

[16] Berthold KP Horn and Michael J Brooks. The variational approach to shape from shading. 1986. 1, 2

[17] Misha Kazhdan. Adaptive multigrid solvers. https://www.cs.jhu.edu/~misha/Code/PoissonRecon. 6

[18] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, 2006. 2, 8

[19] Hyomin Kim, Yucheol Jung, and Seungyong Lee. Discontinuity-preserving normal integration with auxiliary edges. In *CVPR*, 2024. 2

[20] Min Li, Zhenglong Zhou, Zhe Wu, Boxin Shi, Changyu Diao, and Ping Tan. Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. *IEEE Transactions on Image Processing*, 2020. 6, 7

[21] Qing Li, Huifang Feng, Kanle Shi, Yue Gao, Yi Fang, Yu-Shen Liu, and Zhizhong Han. Neuralgf: unsupervised point normal estimation by learning neural gradient function. *NIPS*, 2024. 2

[22] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 3

[23] Geoffrey J McLachlan and David Peel. *Finite mixture models*. John Wiley & Sons, 2000. 4

[24] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 6

[25] Yvain Quéau and Jean-Denis Durou. Edge-preserving integration of a normal field: Weighted least-squares, tv and approaches. In *International Conference on Scale Space and Variational Methods in Computer Vision*, 2015. 2

[26] Yvain Quéau, Jean-Denis Durou, and Jean-François Aujol. Normal integration: a survey. *Journal of Mathematical Imaging and Vision*, 2018. 1, 2

[27] Yvain Quéau, Jean-Denis Durou, and Jean-François Aujol. Variational methods for normal integration. *Journal of Mathematical Imaging and Vision*, 2018. 2

[28] Jonathan Richard Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. 1994. 5

[29] Boxin Shi, Zhe Wu, Zhipeng Mo, Dinglong Duan, Sai-Kit Yeung, and Ping Tan. A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 6

[30] Yinting Wang, Jiajun Bu, Na Li, Mingli Song, and Ping Tan. Detecting discontinuities for surface reconstruction. In *Proceedings of the 21st International Conference on Pattern Recognition*, 2012. 2

[31] Robert J Woodham. Photometric stereo. 1978. 1

[32] Tai-Pang Wu and Chi-Keung Tang. Visible surface reconstruction from normals with discontinuity consideration. In *CVPR*, 2006. 2

[33] Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. Native and compact structured latents for 3d generation. *arXiv preprint arXiv:2512.14692*, 2025. 3

[34] Wuyuan Xie, Miaohui Wang, Mingqiang Wei, Jianmin Jiang, and Jing Qin. Surface reconstruction from normals: A robust dgp-based discontinuity preservation approach. In *CVPR*, 2019. 2

[35] Ying Xiong, Ayan Chakrabarti, Ronen Basri, Steven J Gortler, David W Jacobs, and Todd Zickler. From shading to local shape. *IEEE TPAMI*, 2014. 2

[36] Bohan Yu, Siqi Yang, Xuanning Cui, Siyan Dong, Baoquan Chen, and Boxin Shi. Milo: Multi-bounce inverse rendering for indoor scene with light-emitting objects. *IEEE TPAMI*, 2023. 6

[37] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 2024. 3

[38] Dizhong Zhu and William AP Smith. Least squares surface reconstruction on arbitrary domains. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, 2020. 2

[39] Runsong Zhu, Yuan Liu, Zhen Dong, Yuan Wang, Tengping Jiang, Wenping Wang, and Bisheng Yang. Adafit: Rethinking learning-based normal estimation on point clouds. In *ICCV*, 2021. 2