OReX: Object Reconstruction from Planar Cross-sections Using Neural Fields

Haim Sawdayee
The Blavatnik School of Computer Science
Tel Aviv University

haimsawdayee@mail.tau.ac.il

Amir Vaxman
School of Informatics
The University of Edinburgh

avaxman@inf.ed.ac.uk

Amit H. Bermano
The Blavatnik School of Computer Science
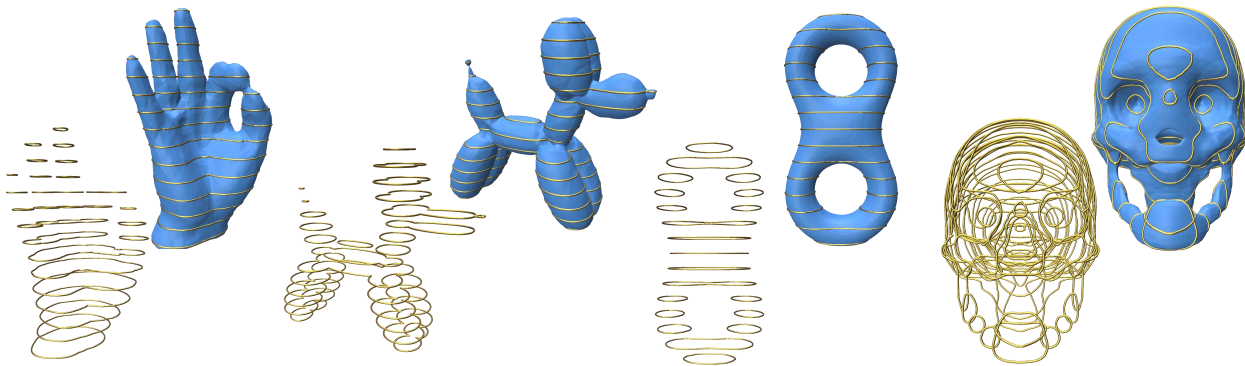Tel Aviv University

amberman@tauex.tau.ac.il

Figure 1. OReX reconstructs smooth 3D shapes (right) from input planar cross-sections (left). Our neural field-based prior allows smooth interpolation between slices while respecting high-frequency features and self-similarities.

Abstract

Reconstructing 3D shapes from planar cross-sections is a challenge inspired by downstream applications like medical imaging and geographic informatics. The input is an in/out indicator function fully defined on a sparse collection of planes in space, and the output is an interpolation of the indicator function to the entire volume. Previous works addressing this sparse and ill-posed problem either produce low quality results, or rely on additional priors such as target topology, appearance information, or input normal directions. In this paper, we present OReX, a method for 3D shape reconstruction from slices alone, featuring a Neural Field as the interpolation prior. A modest neural network is trained on the input planes to return an inside/outside estimate for a given 3D coordinate, yielding a powerful prior that induces smoothness and self-similarities. The main challenge for this approach is high-frequency details, as the neural prior is overly smoothing. To alleviate this, we offer an iterative estimation architecture and a hierarchical input sampling scheme that encourage coarse-to-fine training, allowing the training process to focus on high frequencies at later stages.

In addition, we identify and analyze a ripple-like effect stemming from the mesh extraction step. We mitigate it by regularizing the spatial gradients of the indicator function around input in/out boundaries during network training, tackling the problem at the root. Through extensive qualitative and quantitative experimentation, we demonstrate our method is robust, accurate, and scales well with the size of the input. We report state-of-the-art results compared to previous approaches and recent potential solutions, and demonstrate the benefit of our individual contributions through analysis and ablation studies. [1]

1. Introduction

Reconstructing a 3D object from its cross-sections is a long-standing task. It persists in fields including medical imaging, topography mapping, and manufacturing. The typical setting is where a sparse set of arbitrary planes is given, upon which the "inside" and "outside" regions of the de-

---

[1]Code and data available at https://github.com/haimsaw/OReX

picted domain are labeled, and the entire shape in 3D is to be estimated (see Fig. 1). This is a challenging and ill-posed problem, especially due to the sparse and irregular nature of the data. Classical approaches first localize the problem by constructing an arrangement of the input planes, and then introduce a local regularizer that governs the interpolation of the input to within each cell. While sound, these approaches typically involve simplistic regularization functions, that only interpolate the volume within a cell bounded by the relevant cross-sections; as a consequence, they introduce over-smoothed solutions that do not respect features. In addition, finding a cellular arrangement of planes is a computationally-heavy procedure, adding considerable complexity to the problem and rendering it quickly infeasible for large inputs (see Sec. 4). As we demonstrate (Sec. 4), recent approaches that reconstruct a mesh from an input point cloud are not well suited to our setting, as they assume a rather dense sampling of the entire shape. In addition, these methods do not consider the information of an entire cross-sectional plane, but rather only on the shape boundary.

In this paper, we introduce *OReX*—a reconstruction approach based on neural networks that estimates an entire shape from its cross-sectional slices. Similar to recent approaches, the neural network constitutes the prior that extrapolates the input to the entire volume. Neural networks in general have already been shown to inherently induce smoothness [17], and self-similarities [12], allowing natural recurrence of patterns. Specifically, we argue that Neural Fields [25] are a promising choice for the task at hand. Neural Fields represent a 3D scene by estimating its density and other local geometric properties for every given 3D coordinate. They are typically trained on 2D planar images, and are required to complete the entire 3D scene according to multiview renderings or photographs. This neural representation is differentiable by construction, and hence allows native geometric optimization of the scene, realized via training. We pose the reconstruction problem as a classification of space into "in" and "out" regions, which are known for the entire slice planes, and thus generate the indicator function which its decision boundary is the output shape.

The main challenge with applying neural fields to this problem is high-frequency details. Directly applying established training schemes [19] shows strong spectral bias, yielding overly smoothed results and other artifacts (Fig. 12). Spectral bias is a well-known effect, indicating that higher frequency is effectively learned slower [22]. To facilitate effective high-frequency learning, avoiding the shadow cast by the low frequency, we introduce two alterations. First, we sample the planar data mostly around the inside/outside transition regions, where the frequencies are higher. This sampling is further ordered from low to high-frequency regions (according to the distance from the inside/outside boundary), to encourage a low-to-high-frequency training progression.

In addition, we follow recent literature and allow the network to iteratively infer the result, where later iterations are responsible for finer, higher-frequency corrections [1, 23].

Finally, we consider another high-frequency artifact, also found in other neural-field-based works [9]. The desired density (or indicator) function dictates a sharp drop in value at the shape boundary. This is contradictory to the induced neural prior, causing sampling-related artifacts in the downstream task of mesh extraction (Sec. 3.4). To alleviate this, we penalize strong spatial gradients around the boundary contours. This enforces a smoother transition between the in and out regions, allowing higher-quality mesh extraction.

As we demonstrate (see Fig. 1), our method yields state-of-the-art reconstructions from planar data, both for woman-made and organic shapes. The careful loss and training schemes are validated and analyzed through quantitative and qualitative experimentation. Our method is arrangement-free, and thus both interpolates all data globally, avoiding local artifacts, and scales well to a large number of slices (see Sec. 4).

2. Related work

Reconstruction from cross-sections Reconstruction from planar cross-section has been a long-standing challenge in geometry processing and computational geometry. The problem was mostly studied in the parallel planes setting (e.g., [2,3,5]), where the 3D object was reconstructed per two such "keyframes". Later work offered general solutions for any orientation or distribution of planes [4, 16]. The general approach was based on tessellating the planes into convex cells by the planes of the arrangement and reconstructing the object in each cell, by defining some interpolant inside it. These methods suffer from several issues: the construction of the arrangement is computationally expensive (with at least cubic asymptotic times), and the local reconstruction introduces continuity artifacts (see Sec. 4). Furthermore, the interpolants were usually designed for smoothness or proximity, and failed to capture more global trends in the geometry of the slices, such as twists or extrusions (see Fig. 6). Some works provide topological guarantees [14, 26] or limited the solution to given templates [13], but do not provide a general solution without these priors. Our work is arrangement-free and provides a flexible high-parametric model, using neural networks, to capture the details of the reconstructed object.

Neural reconstruction Most recent reconstruction works employing neural networks address reconstruction from dense point clouds. Our problem could be cast as a point cloud reconstruction one, after sampling the planes. In this setting, the high-frequency problem is pronounced as in our case, and even more so since more information regarding the target shape is available. Almost all recent ap-
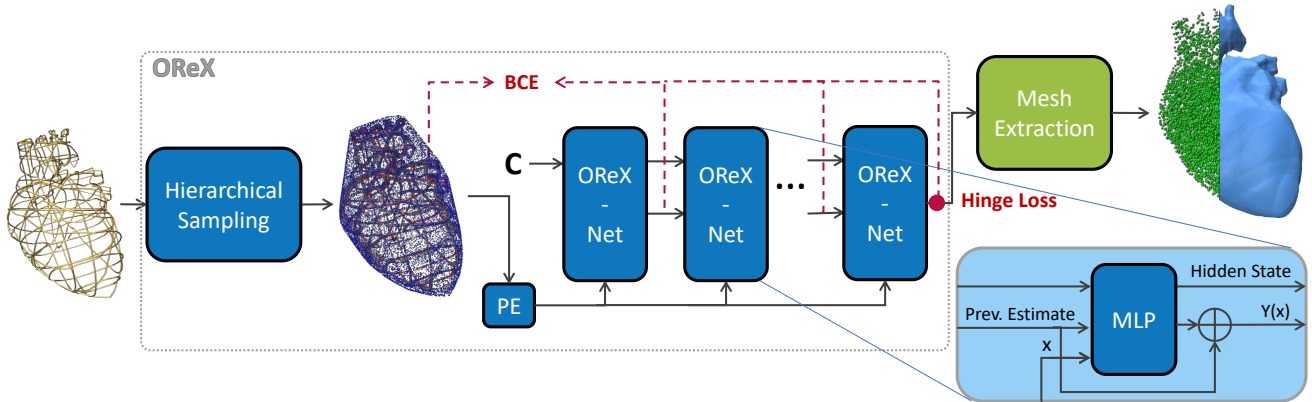
Figure 2. OReX method overview. The input (left) is a set of planar cross-sections with inside/outside information on them. The planes are sampled into points in a hierarchical scheme, and are fed to our iterative Neural Field for training. After training, the final shape is estimated using an off-the-shelf mesh extractor.
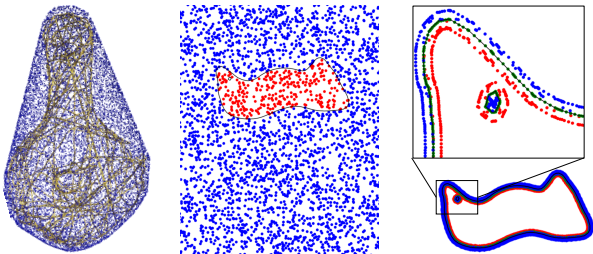

Figure 3. We train OReX on three types of sampling distributions of the input: A scaled-up version of the 3D convex hull bounds the reconstruction volume (left); Uniform sampling within each plane helps stabilize the learning (middle); most of the samples are concentrated where accuracy matters most—on and around the boundary contours on the input planes (right).

proaches employ an implicit representation of the shape. Some works populate 3D (sometimes adaptive) grids to handle fine-details [7, 8, 18, 24], which is challenging to scale. Other works [6, 21] perform local operations, which rely on a rather dense neighborhood. To alleviate this, a mesh can be directly optimized to match the input [12]. Other neural fields have been employed as well in this context [17], without additional information such as normals; see a recent survey for more approaches [25]. As we demonstrate (Sec. 4), this general line of works employs weaker priors, which do not fit the sparse nature of our problem well.

Perhaps a more similar setting is the one of reconstruction from projections or photographs. Here the reconstructions are from 2D data as well, which is also of sparser nature, albeit using appearance information that we do not have. Literature in this field is deep and wide, with several surveys [10, 11], including the usage of neural fields. We argue our work is orthogonal to that listed here, as it can be plugged in to replace our basic neural-field baseline. We leave for future work to inspect the most performing architecture. Addressing the most similar problem setting, concurrent work by Ostonov et al. [20] employs reinforcement learning and orthographic projections to ensure proper reconstruction. Except for requiring additional information during inference, they also require a training phase and are restricted to the domain trained on. Our approach, in contrast, requires no additional information besides the planes.

3. Method

We next lay out the details for effective high-quality shape reconstruction from slices. Our approach is based on a neural field, *OReXNet*. OReXNet outputs $Y(x)$, an extrapolation of the inside/outside indicator function for a query point $x \in \mathbb{R}^3$ (Sec. 3.1). Given a single set of input cross-sections, the network undergoes training to approximate the target function on the input. After training is complete, Y is sampled on the entire volume, and a resulting shape is extracted using the Dual Contouring (DC) approach [15]. This pipeline is depicted in Fig. 2.

As motioned in Sec. 1, the main challenge in reconstruction quality is high-frequency details. The straightforward approach to our problem would be to train a neural field for the desired indicator function by uniformly sampling all planes, and subsequently training a network using a *Binary Cross Entropy* (BCE) loss on all sampled points. Typical approaches also represent the input coordinates using Positional Encoding (PE) [19]. As it turns out, this approach yields overly smooth results failing even to interpolate the input (Fig. 12). To improve reconstruction fidelity, and allow higher frequency details in the resulting shape, we introduce two alterations to the aforementioned training scheme. We present our hierarchical input sampling scheme in Sec. 3.2, and our iterative-refinement architecture in Sec. 3.3.

Finally, we also address a common artifact in implicit mesh extraction [9]. Repeating patterns can be seen that correlate to the sampling pattern of the mesh extraction phase, as demonstrated in Fig. 8. We describe how we design the loss function to mitigate this artifact in Sec. 3.4. See the Supplementary material and code for implementation details and hyper-parameter values.

3.1. Problem Setting

We consider a set $\mathcal{P} = \{P_1, \cdots, P_k\}$ of 2D planes embedded in $\mathbb{R}^3$, with arbitrary offsets and orientations. Each plane P_i contains an arbitrary set of (softly) non-intersecting oriented contours $C_i = \{c_{i,1}, \cdots, c_{i,l_i}\}$ that consistently partition the plane into regions of "inside" and "outside" of an unknown domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$ (Fig. 2). The target output of our method is an indicator function $Y : \mathbb{R}^3 \to \mathbb{R}$, defining Ω as:

$$Y(x) = \begin{cases} 1 & x \notin \Omega \\ 0 & x \in \Omega \\ 0.5 & x \in \partial\Omega \end{cases} \quad (1)$$

In practice, we approximate Y using a function f, such that $Y(x) \approx \sigma(f(x))$, where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the sigmoid function.

3.2. Input Sampling

Regularly or randomly sampling the input planes for training yields inaccurate and overly smooth results (see Fig. 12). Instead, we define three types of relevant point distributions to sample from, as depicted in Fig. 3:

1. We bound the reconstructed volume by the 3D convex hull of all the input contours, and scale it up by 5%. We consequently sample it uniformly, where the sampled points are all outside.

2. We compute the bounding box (aligned to the principal axes) of the contours in each plane, and sample it with uniform distribution.

3. Most samples are taken around the boundaries of the contours, as these are the regions where accuracy matters most. We then sample every contour edge evenly, and further add off-surface points of varying distances for each edge sample. Off-surface locations are found by moving away from the contour on the plane (i.e., in the direction of the plane normal). We further sample off-surface points in a circle around each vertex.

Each sampling point x_i is matched with a label Y_i according to its position on the slice, and the pairs $\{x_i, Y_i\}$ constitute the input to our training. See supplementary material for the relevant hyperparameters.

Frequency-Oriented Sampling. In order to encourage better high-frequency learning, we sample the 3rd type of points around a set of varying distances, from 0.1% of the bounding box away from the contours to three orders of magnitude closer. In every epoch, we only use points sampled around three consecutive distance ranges. For early training iterations, we use the three largest distances, since further away points translate to lower frequency information about the shape. As the training progresses, and the lower frequencies are assumed to stabilize, we train with points that grow closer and closer to the actual contour, thus focusing the learning process on higher and higher frequencies (see Fig. 11). See supplementary material for more details and exact scheduling.

3.3. Architecture

OReXNet is simply an MLP that takes a 3D coordinate as input, represented using positional encoding, and produces the function Y (Sec. 3.1) at any query point. In order to encourage high-frequency details, we introduce an iterative refinement mechanism. Inspired by recent work [1], we allow the network to refine its own results by running them through the network for a small number of iterations. This process was previously shown to produce a coarse-to-fine evolution in the realm of 2D images [1], and we argue it applies to our case as well. As demonstrated in Fig. 2, ORexNet is hence a *residual* MLP that is fed the result of the previous iteration, and a small hidden state code, along with the query point. These iterations (10 in our experiments) are performed both during training and inference; at inference time, only the last result is taken, and during training, the loss is applied to the results of all steps. The first iteration starts from a learned constant C. We show that this process indeed sharpens features and allows the incorporation of higher-frequency details in Fig. 13.

3.4. Loss Function and Inference

In order to present our loss function, we must first attend to an issue in the final stage of our pipeline. In this stage, we extract an explicit mesh for the boundary of the indicator function using Dual Contouring [15]. This step uses a discretely sampled version of $Y(x)$ and $\nabla Y(x)$ on a regular 3D grid to extract the mesh. As can be seen in Fig. 8 (and as witnessed other works [9]), this creates a ripples-like artifact, that correlates to the 3D grid resolution. This results from an aliasing of the sampling in these regions where the gradient magnitudes exhibit high variance. This effect could be mitigated via higher resolution grid sampling, which is expensive, or a post-process operation that may compromise the geometry. Instead, we propose revisiting the loss function of the training process, and incorporating regularization to reduce this sharp drop in value at the shape boundary. First, we omit the activation—in other words, we use the
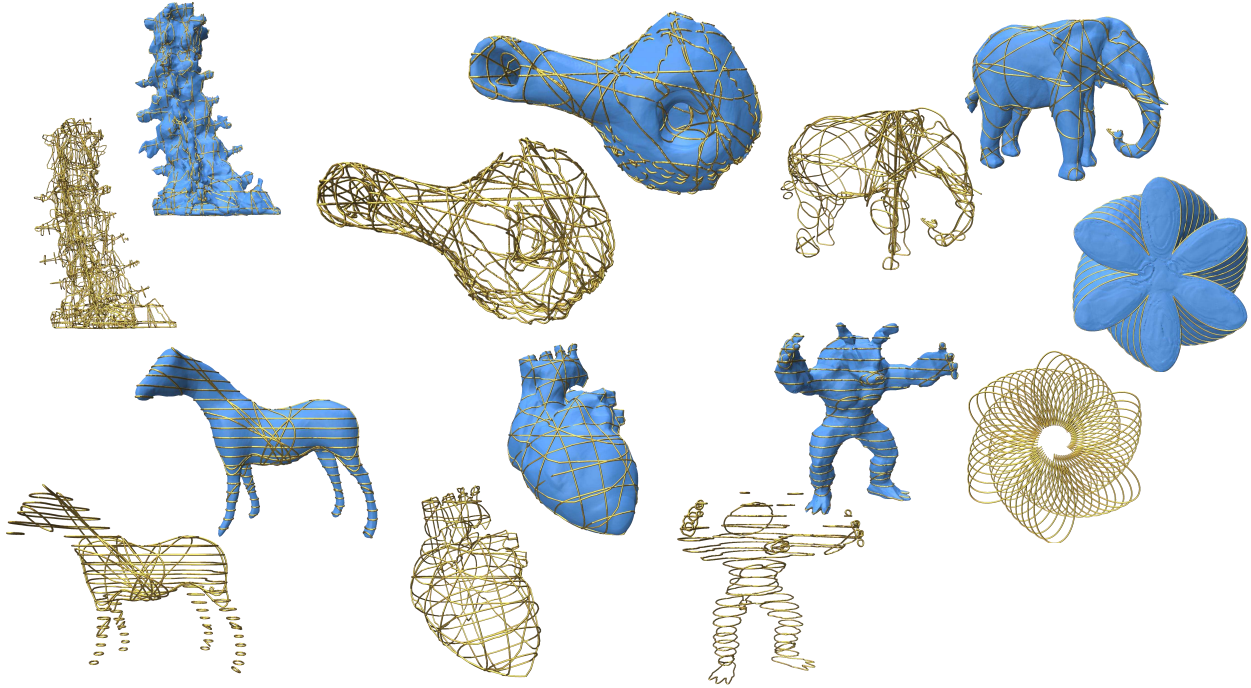
Figure 4. Results gallery. A qualitative demonstration of a collection of man-made and medical inputs. Zoomed-in viewing is recommended.
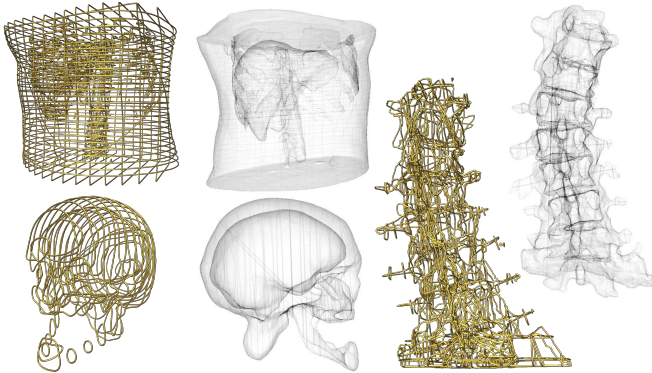

Figure 5. Additional qualitative results, where transparency reveals the reconstruction of internal cavities and tunnels. Zoomed-in viewing is recommended.

point x is then

$$\mathcal{L}(x, \theta) = \sum_{i=0}^{N-1} \text{BCE}(Y_i(x)) + \lambda \max\left(0, ||\nabla f_{N-1}(x)|| - \alpha\right)$$

(2)

Where σ is the sigmoid function, θ are the parameters of the network, $BCE()$ is the binary cross-entropy loss applied to all outputs of our iterative scheme, $max(0, x)$ is a hinge loss applied only to the last OReXnet iteration, N the number of iterations, and α and λ are hyperparameters. Fig. 8 demonstrates the correction effect of the hinge loss.

4. Experiments

To evaluate our method, we developed a prototype, and ran it on a NVIDIA GeForce GTX 1080 Ti. See the Supplementary Material for exact timing and memory consumption statistics. Generally speaking, all training processes took less than four hours on the single GPU.

4.1. Results and Comparisons

We first qualitatively demonstrate the result of our method on a variety of slice inputs, from both the medical and graphics worlds (Fig. 4). This demonstrates the versatility and generality of our method. We further show our our algorithm correctly reproduces the internal cavities and details of reconstructed objects in Fig. 5. Note how intricate details are learned along side a smooth interpolation between slices,

function $f(x)$ for contouring, while using $Y(x)$ only for training (Sec 3.1). This de-radicalizes the function values, already offering a softer transition between inside and outside regions. In addition, we explicitly penalize strong transitions by using a *hinge loss* (Eq. 2). With this, we reduce the gradient magnitude variance considerably by limiting them close to the input shape boundary. Our loss function for a single
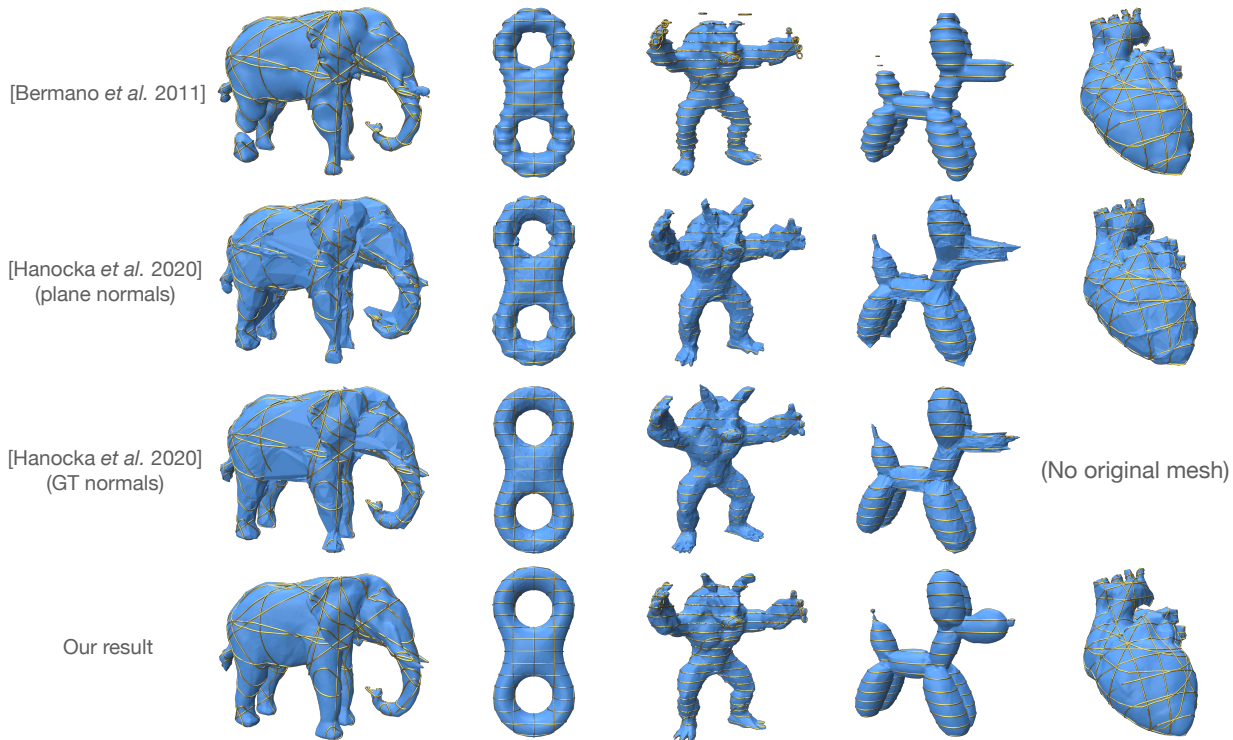
Figure 6. Qualitative comparisons to other approaches. Our result is smoother, more aligned to features, and with less artifacts. Note how the reconstructed shape silhouettes does not suffer imprints from the input planes. Check Table 1 for quantitative comparison.

leaving no slice transition artifacts on the resulting shape.

In terms of other methods, we qualitatively (Fig. 6) and quantitatively (Table 1) compare our method to Bermano et al. [4], which is a reconstruction from cross-section method with the same input-output as ours. We further compare to general state-of-the-art reconstruction methods that target point clouds [12]. Hanocka et al. [12] expect point normals as input as well. Hence we compare our result to the latter work with Ground Truth (GT) normals at input, and with ones coming from the input plane normals for a fairer comparison. We use the following metrics: 1) intersection-over-union volume of inside regions in reconstructed volumes and 2) GT planes, and 3) symmetric Hausdorff distance. These metrics demonstrate our superiority over the compared methods. See Supplementary Material for more comparisons.

Scalability We show that our method does not suffer the computational cost that arrangement-based methods [4, 14, 16] must bear, since we do not tessellate the space, and hence scale well with the input. In addition, we show (Fig. 7) how our method converges with the addition of slices.

4.2. Ablations

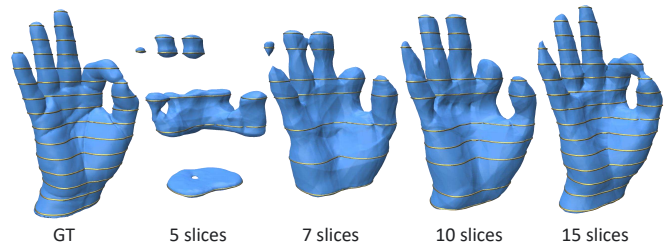We perform ablation testing to evaluate important aspects of our method and our design choices.

Figure 7. Increasing number of slices. As can be expected, introducing more slices converges consistently converges to the solution.

Gradient magnitude regularization (Fig. 8) We show the effect of the hinge-loss regularizer on gradient magnitudes, with increasing λ values. It is evident the hinge loss effectively filters out the high variance, and smooths the ripple artifacts. As can be seen (bottom row), without our hinge loss, geometry vertices tend to snap to discrete locations during the mesh extraction phase. This is because a large gradient induces a large bias in the magnitude of the indicator function, encouraging vertices to remain close to the grid points sampled during the extraction step. Regularizing the gradients around the boundry regions (top row) decreases the gap between the positive and negative sides of the function, allowing a more uniform, and hence continuous, distribution

of the mesh vertices. With this bias removed, the resulting geometry is better sampled and hence the ripple artifact is mitigated (middle).
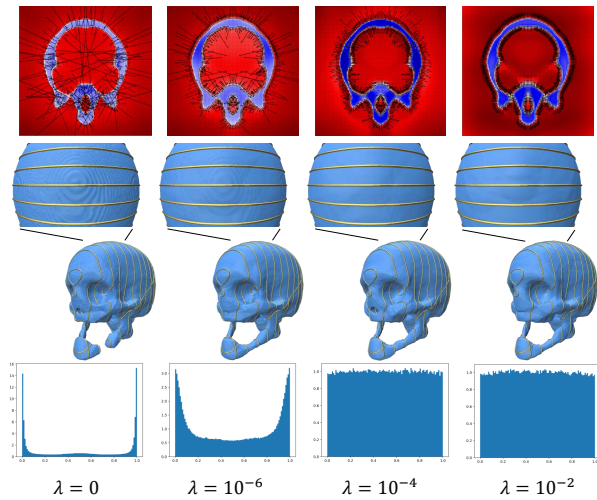

Figure 8. Left to right: our Hinge loss effect for mitigating ripple-like aliasing artifacts. Top: output values of $f(x)$ on a single slice (arrows show gradients). Middle: The reconstructed mesh with a closer view on the top of the skull. Bottom: distribution of vertices along grid edges during mesh extraction. It is evident the hinge loss effectively controls gradient magnitudes at the zero set, curbing the aliasing artifacts. The hinge regularization further allows to place more vertices in the middle of the sampled grid cells, and not only on their edges.

Iterative architecture (Fig. 9)   This ablations examines the effect of the number of refinement iterations on the accuracy of the mesh produced. As can be expected, the results show that as the number of iterations increases, high-frequency details are better represented.
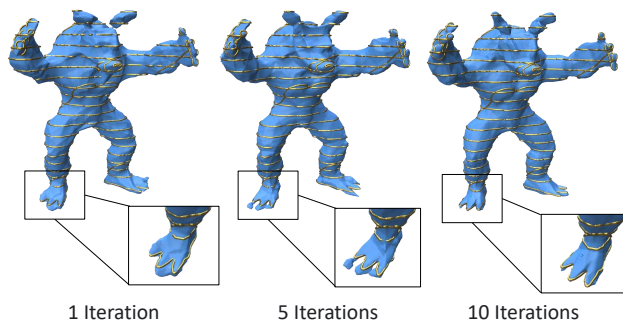

Figure 9. Iterative refinement experiment. Training and inferring with more OReXNet refinement iterations allows the network to perform smaller scale corrections and reduces spectral bias.

Sampling method   To evaluate our importance sampling scheme, we first replace it with regular grid sampling in two resolutions (Fig. 10). As can be seen, using each of the resolutions alone yields overly smoothed results due to spectral bias. Training with a gradually increasing resolution allows better handling of high-frequency details. In addition, we have demonstrate the effects on quality when using our importance scheme in a non-hierarchical manner, i.e. using all points uniformly without consideration of distance from the boundary (Fig. 11). As can be seen, our scheme improves in detail quality upon both experiments, demonstrating that our sampling scheme produces more accurate results with less spectral bias while using fewer samples.
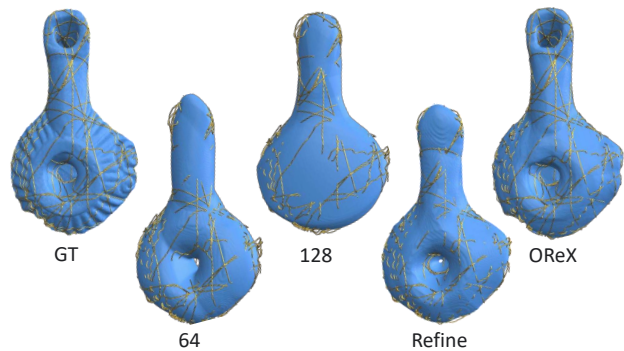

Figure 10. Sampling scheme Ablations. Left: ground truth. Middle: grid sampling with different resolutions, and a gradually increasing sampling resolution over the training process. Right: our method.
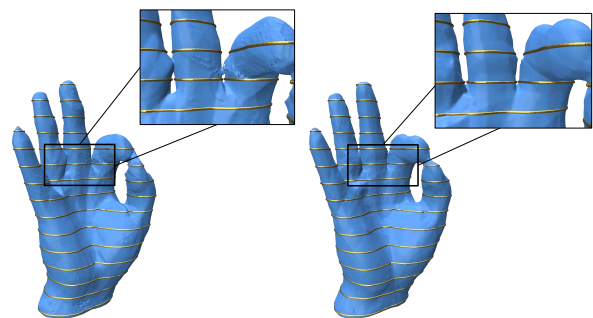

Figure 11. Left: non-hierarchical sampling, using points from all off-surface distances uniformly. Right: using our hierarchical sampling scheme.

Choice of architecture and sampling (Fig. 12)   We justify our design choices by training a baseline architecture with uniform grid sampling and no iteration refinement. It is evident such a model does not preserve details nor interpolate well, even for a relatively simple case.

Input	Hausdorff distance				IoU in $3d$				IoU in $2d$			
	OReX	Bermano et. al	point2mesh plane normals	point2mesh GT normals	OReX	Bermano et. al	point2mesh plane normals	point2mesh GT normals	OReX	Bermano et. al	point2mesh plane normals	point2mesh GT normals
Eight (S)	**0.018**	0.065	0.219	0.046	**0.984**	0.865	0.842	0.961	**0.988**	0.984	0.795	0.980
Eight (M)	**0.006**	0.033	0.045	0.014	**0.987**	0.893	0.915	0.974	**0.986**	0.976	0.961	0.971
Elephant	**0.056**	0.081	0.100	0.086	**0.966**	0.908	0.885	0.935	**0.975**	0.969	0.850	0.908
Balloon dog	**0.049**	0.194	0.078	0.086	**0.957**	0.868	0.897	0.928	**0.988**	0.977	0.926	0.956
Hand OK	**0.063**	0.177	0.195	0.135	**0.955**	0.860	0.921	0.931	**0.987**	0.968	0.908	0.882
Armadillo	**0.050**	0.121	0.057	0.059	**0.939**	0.891	0.891	0.921	**0.964**	0.776	0.850	0.868

Table 1. Quantitative comparisons. We measure performance using Hausdorff distance, IoU of the inner volume compared to the GT shape, and IoU of the inner surface on the input cross-sections. We compare our result to a dedicated cross-section based reconstruction method [4], and to two flavors of a pointcloud reconstruction method [12]. See more comparisons in the Supplementary Material.

work balances the smoothness of a neural prior with high-frequency features. We show our approach successfully produces smooth interpolation between contours while respecting high-frequency features and repeating patterns. In addition, we believe some of the analysis and insights presented here can be applied to neural fields in similar tasks.

The advantage of our method is also its disadvantage: using only binary data, much of the information is lost in the process. For example, for medical imaging an interesting future direction would be to directly use the raw grayscale density input of a planar probe such as that comes from an ultrasound. Another interesting direction for research is extending this work for multi-labeled volumes (e.g., the reconstruction of several organs simultaneously from a scan), and using partial or noisy data.
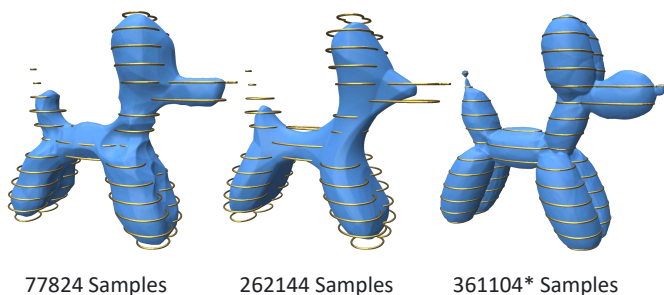

77824 Samples   262144 Samples   361104* Samples

Figure 12. Left and middle: the baseline experiment with regular grid samples of 64 and 128 resolution over the slices. Right: our result for the same amount for samples (our variable number of samples is provided at its mean).

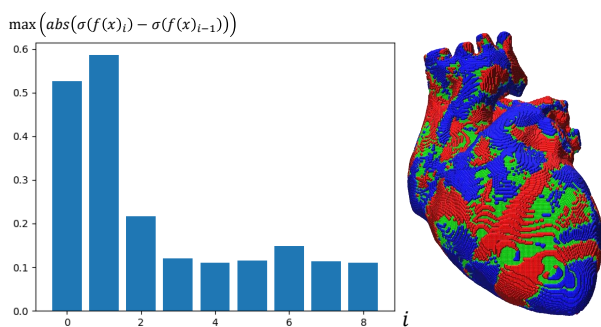$\max\left(abs(\sigma(f(x)_i) - \sigma(f(x)_{i-1}))\right)$

Figure 13. Refinement progression throughout the iteration process. The maximum change in the output value decreases along the iterations. Right: points that were added to the shape (Red) and that were removed from it (blue) in the last refinement iteration. These are concentrated on the level set, whereas most other points (green, covering all internal regions) have already converged.

5. Conclusions

We have presented OReX, a state-of-the-art method for the long-standing problem of shape reconstruction from planar cross-sectional indicator data. Free of datasets and training requirements, OReX is simple and intuitive to use. Our

References

[1] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6711–6720, 2021. 2, 4

[2] Chandrajit L Bajaj, Edward J Coyle, and Kwun-Nan Lin. Arbitrary topology shape reconstruction from planar cross sections. *Graphical models and image processing*, 58(6):524–543, 1996. 2

[3] Gill Barequet and Micha Sharir. Piecewise-linear interpolation between polygonal slices. In *Proceedings of the tenth annual symposium on Computational geometry*, pages 93–102, 1994. 2

[4] Amit Bermano, Amir Vaxman, and Craig Gotsman. Online reconstruction of 3d objects from arbitrary cross-sections. *ACM Transactions on Graphics (TOG)*, 30(5):1–11, 2011. 2, 6, 8

[5] Jean-Daniel Boissonnat and Pooran Memari. Shape reconstruction from unorganized cross-sections. In *Symposium on geometry processing*, pages 89–98, 2007. 2

[6] Alexandre Boulch and Renaud Marlet. Poco: Point convolution for surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6302–6314, 2022. 3

[7] Zhang Chen, Yinda Zhang, Kyle Genova, Sean Fanello, Sofien Bouaziz, Christian Häne, Ruofei Du, Cem Keskin, Thomas Funkhouser, and Danhang Tang. Multiresolution

deep implicit functions for 3d shape representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13087–13096, 2021. 3

[8] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3d shape reconstruction and completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6970–6981, 2020. 3

[9] Philipp Erler, Paul Guerrero, Stefan Ohrhallinger, Niloy J Mitra, and Michael Wimmer. Points2surf learning implicit surfaces from point clouds. In *European Conference on Computer Vision*, pages 108–124. Springer, 2020. 2, 4

[10] Kyle Gao, Yina Gao, Hongjie He, Denning Lu, Linlin Xu, and Jonathan Li. Nerf: Neural radiance field in 3d vision, a comprehensive review. *arXiv preprint arXiv:2210.00379*, 2022. 3

[11] Hanry Ham, Julian Wesley, and Hendra Hendra. Computer vision based 3d reconstruction: A review. *International Journal of Electrical and Computer Engineering*, 9(4):2394, 2019. 3

[12] Rana Hanocka, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. Point2mesh: A self-prior for deformable meshes. *arXiv preprint arXiv:2005.11084*, 2020. 2, 3, 6, 8

[13] Michelle Holloway, Cindy Grimm, and Tao Ju. Template-based surface reconstruction from cross-sections. *Computers& Graphics*, 58:84–91, 2016. Shape Modeling International 2016. 2

[14] Zhiyang Huang, Ming Zou, Nathan Carr, and Tao Ju. Topology-controlled reconstruction of multi-labelled domains from cross-sections. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017. 2, 6

[15] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 339–346, 2002. 3, 4

[16] Lu Liu, Chandrajit Bajaj, Joseph O Deasy, Daniel A Low, and Tao Ju. Surface reconstruction from non-parallel curve networks. In *Computer Graphics Forum*, volume 27, pages 155–163. Wiley Online Library, 2008. 2, 6

[17] Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Neural-pull: Learning signed distance function from point clouds by learning to pull space onto surface. In *International Conference on Machine Learning*, pages 7246–7257. PMLR, 2021. 2, 3

[18] Julien NP Martel, David B Lindell, Connor Z Lin, Eric R Chan, Marco Monteiro, and Gordon Wetzstein. Acorn: Adaptive coordinate networks for neural scene representation. *arXiv preprint arXiv:2105.02788*, 2021. 3

[19] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3

[20] Azimkhon Ostonov. Cut-and-approximate: 3d shape reconstruction from planar cross-sections with deep reinforcement learning. *arXiv preprint arXiv:2210.12509*, 2022. 3

[21] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 3

[22] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 2

[23] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[24] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11358–11367, 2021. 3

[25] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. In *Computer Graphics Forum*, volume 41, pages 641–676. Wiley Online Library, 2022. 2, 3

[26] Ming Zou, Michelle Holloway, Nathan Carr, and Tao Ju. Topology-constrained surface reconstruction from cross-sections. *ACM Transactions on Graphics (TOG)*, 34(4):1–10, 2015. 2