MeshUDF: Fast and Differentiable Meshing of Unsigned Distance Field Networks

Benoît Guillard, Federico Stella, Pascal Fua

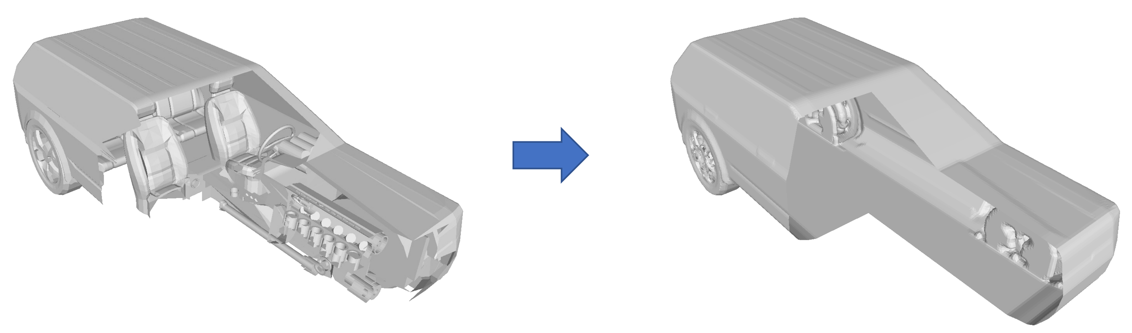
Computer Vision Laboratory, EPFL, Lausanne (Switzerland)

ECCV TEL AVIV 2022

CVLab EPFL

1. Motivation

Signed Distance [1] and Occupancy Field [2] Networks can represent watertight surfaces with any topology, thus requiring regularization and preprocessing in order to make raw meshes watertight. This removes internal details and is not suitable to represent open surfaces.

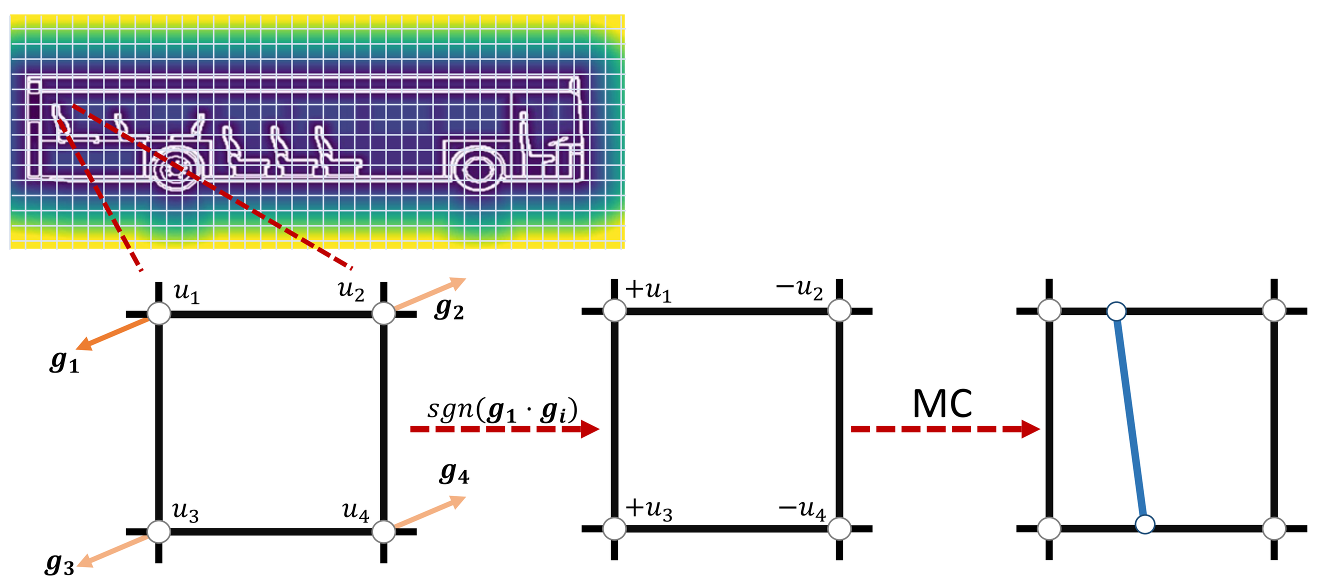


Unsigned Distance Fields (UDFs) can represent open surfaces, but they lack an efficient way to be converted into an explicit mesh, an issue we solve as our core contribution.

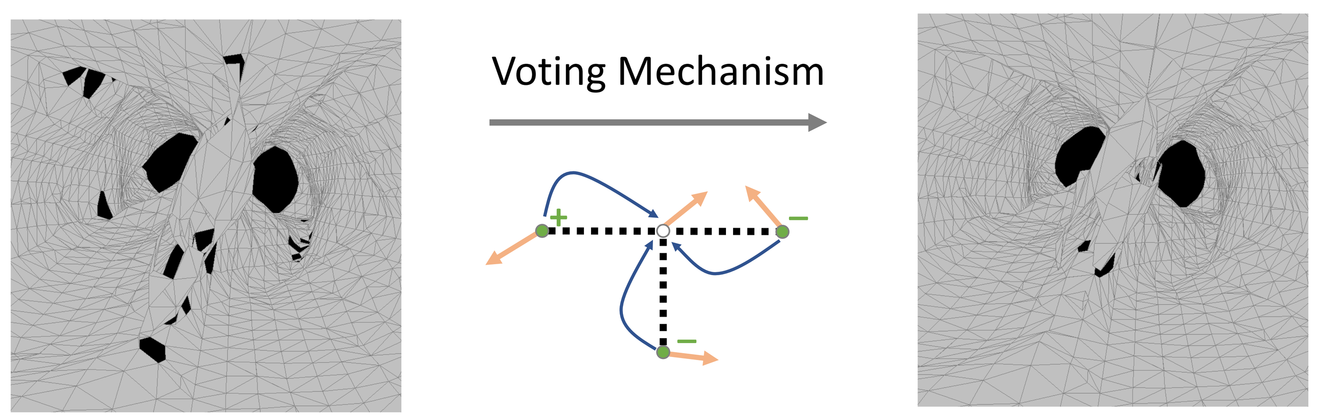
2. Method

UDFs do not have sign flips, thus marching cubes (MC [3]) does not apply directly. Instead:

1. At each grid cell we rely on gradient orientations to infer pseudo-signs, which are then processed using MC;



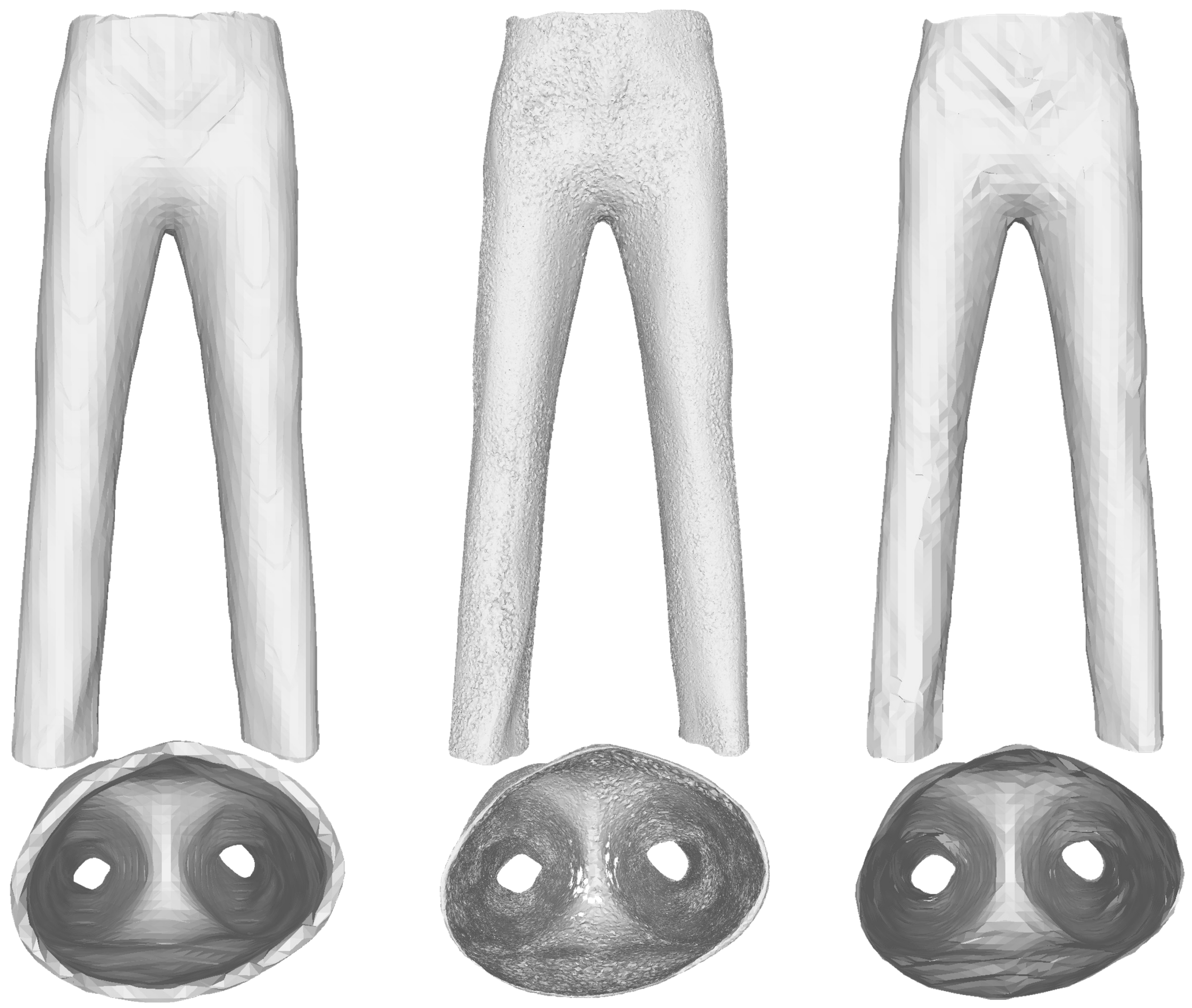
2. We employ a voting mechanism on pseudo-signs to reduce artifacts due to noisy gradients;



Lots of undesirable holes

Fewer holes

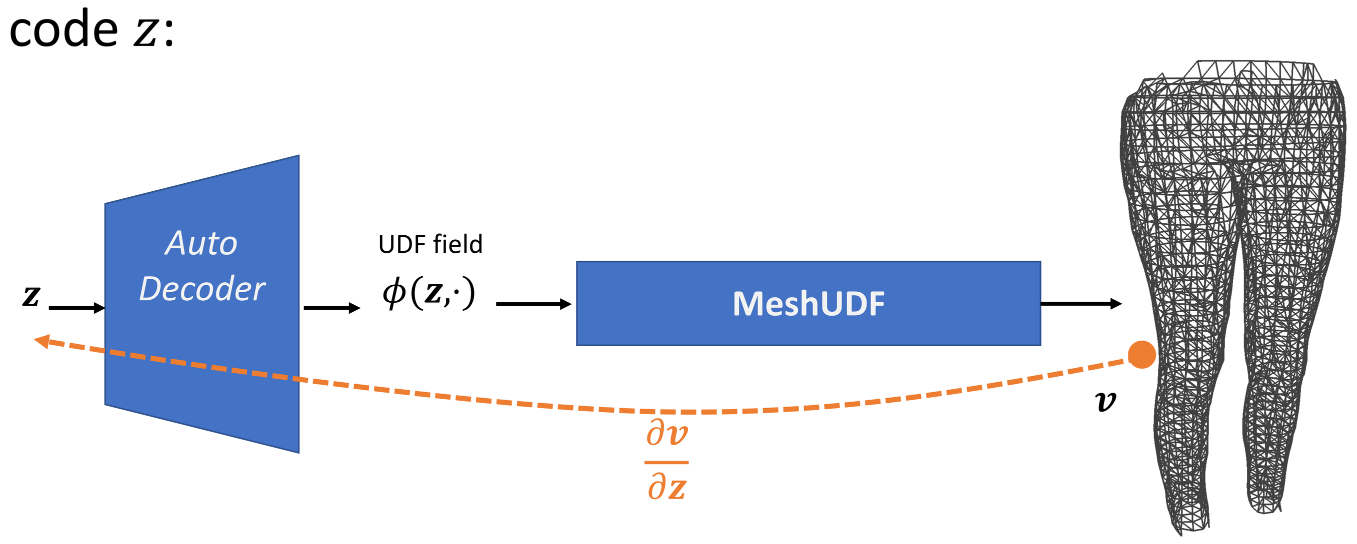
3. We explore the grid in a breadth-first manner to ensure consistency between neighboring cells.



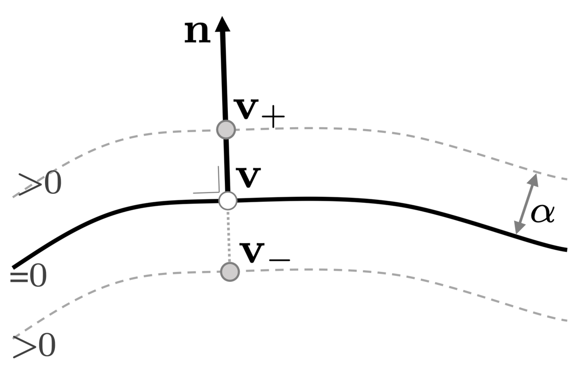
MC [3] on ϵ-levelset:	Ball Pivoting:	**Ours:**
✓ Fast	X Slow	✓ **Fast**
X Inaccurate	– Accurate but rough	✓ **Accurate**
X Inflated (watertight)	✓ Open surface	✓ **Open surface**

3. Differentiability

Surface points lie on a singularity of the UDF field. To achieve differentiability, we propose gradients for vertices v with w.r.t. the latent code z:



Intuitively, when the UDF increases at v_- and decreases at v_+, v moves in the direction of $\vec{n}$. Leveraging the equations in [4] we derive:



$$\frac{\partial \mathbf{v}}{\partial \mathbf{z}} \approx \frac{\mathbf{n}}{2}\left[\frac{\partial \phi}{\partial \mathbf{z}}(\mathbf{z}, \mathbf{v} - \alpha\mathbf{n}) - \frac{\partial \phi}{\partial \mathbf{z}}(\mathbf{z}, \mathbf{v} + \alpha\mathbf{n})\right]$$

4. Results

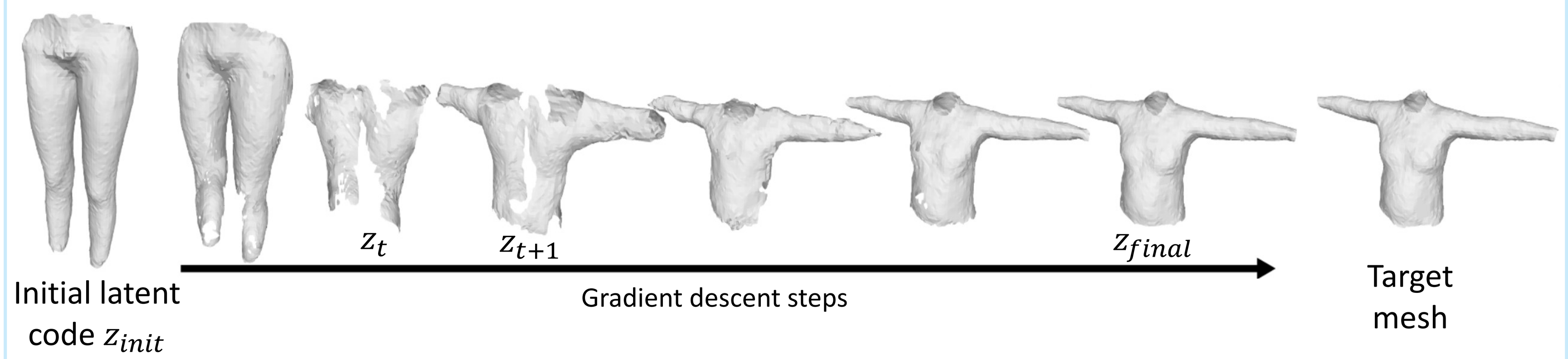
1. **UDF auto-decoder, MGN [5] garments**
Garments can be naturally represented as open surfaces, resulting in the perfect test scenario for our method, which results faster than Ball Pivoting and more accurate than inflating the shapes.

	Inflation	*Ball Pivoting*	*Ours*
Chamfer Distance (↓)	3.00	1.62	**1.51**
Image Consistency (%, ↑)	88.48	92.51	**92.80**
Normal Consistency (%, ↑)	94.16	89.50	**95.50**
Meshing Time (↓)	**1.0 sec.**	16.5s + 3000s	1.2 sec.

2. **Off-the-shelf NDF [6] network, ShapeNet cars**
Some topological cases are missing, i.e. T junctions, leading to artifacts. However, the previous conclusions still apply.

	Inflation	*Ball Pivoting*	*Ours*
Chamfer Distance (↓)	11.24	6.84	**6.63**
Image Consistency (%, ↑)	87.09	90.50	**90.87**
Normal Consistency (%, ↑)	**73.19**	61.50	70.38
Meshing Time (↓)	**4.8 sec.**	24.7s + 8400s	7.1 sec.

3. **Differentiability**
We can apply a differentiable loss function directly on the vertices (e.g., 3D Chamfer) and optimize the latent code with gradient descent. Here this results in a topology change.



References

[1] **DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation**, Park et al., CVPR 2019
[2] **Occupancy Networks: Learning 3D Reconstruction in Function Space**, Mescheder et al., CVPR 2019
[3] **Marching Cubes: a High Resolution 3D Surface Construction Algorithm**, Lorensen and Cline, SIGGRAPH 1987
[4] **MeshSDF: Differentiable Iso-Surface Extraction**, Remelli et al., NeurIPS 2020
[5] **Multi-Garment Net: Learning to Dress 3D People from Images**, Bhatnagar et al., ICCV 2019
[6] **Neural Unsigned Distance Fields for Implicit Function Learning**, Chibane et al., NeurIPS 2020

GitHub