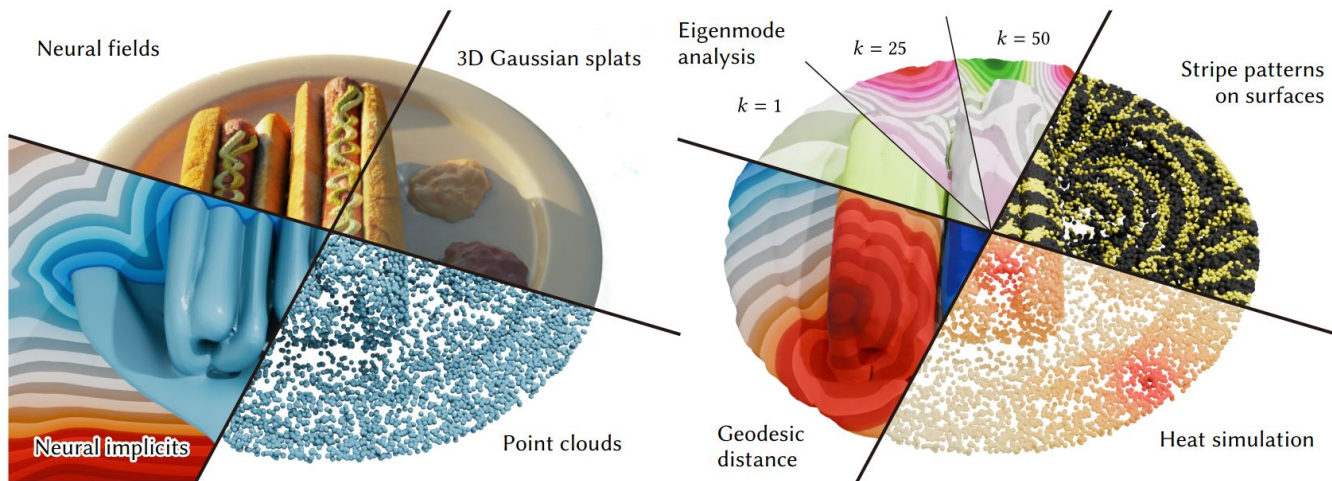


Mesh Processing Non-Meshes via Neural Displacement Fields

Yuta Noma¹ , Zhecheng Wang¹ , Chenxi Liu¹ , Karan Singh¹ , Alec Jacobson^{1,2} 

¹University of Toronto, ²Adobe Research

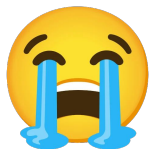


Publié à EG2026

Du traitement géométrique pour du streaming?



Principe du streaming en traitement géométrique

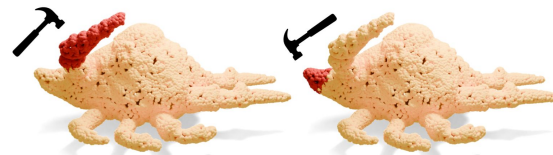


Les ressources et la
bande passante sont
limitées

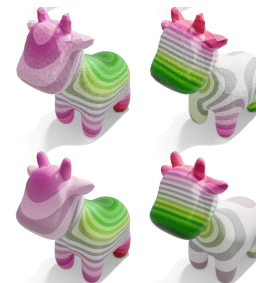


utilisateur

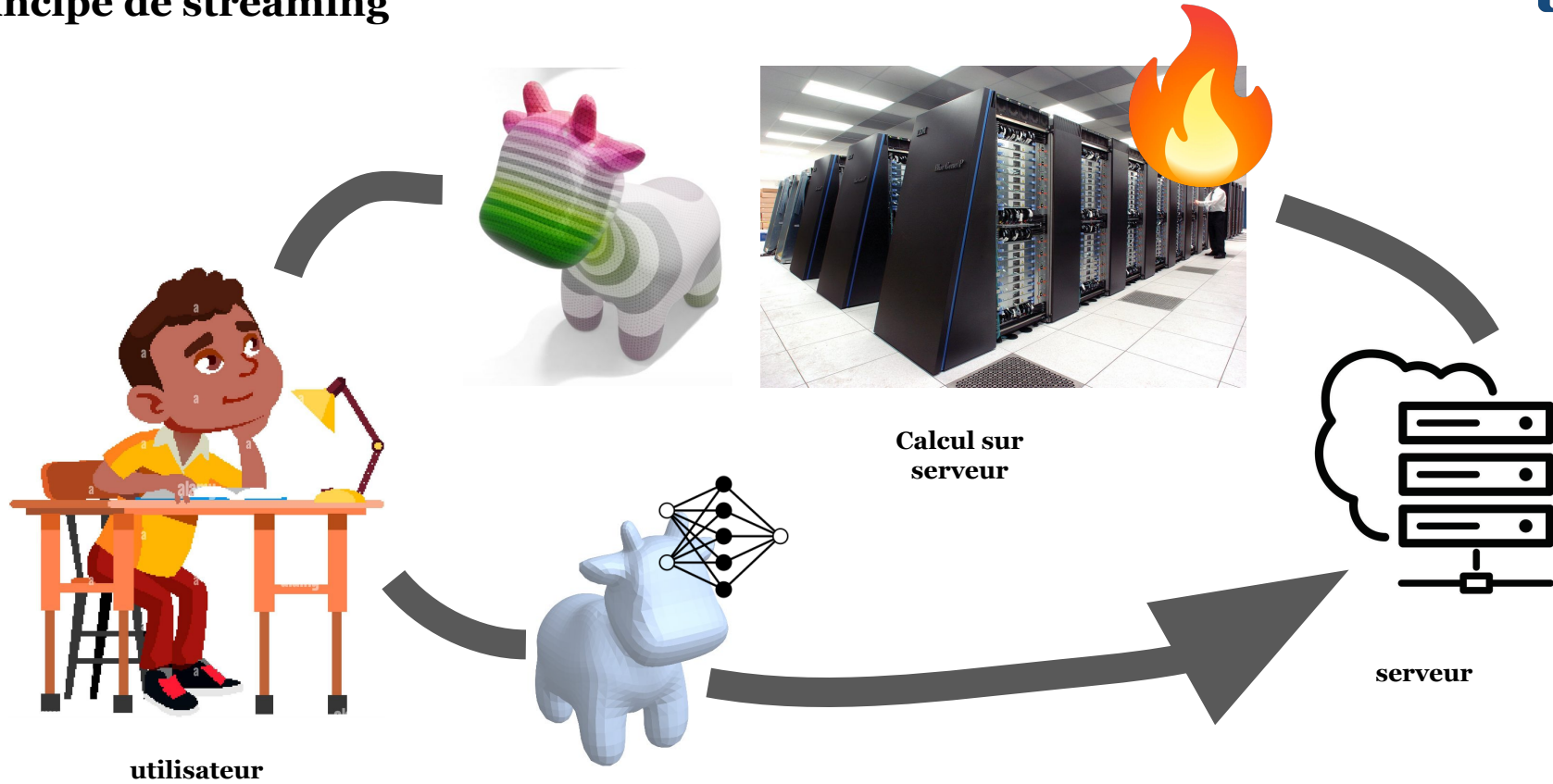
Simulation de fracture Temp réel



Visualiser
les valeurs
propres du
laplacien



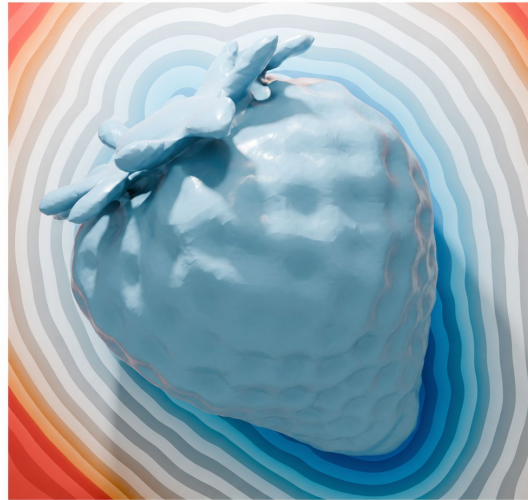
Principe de streaming



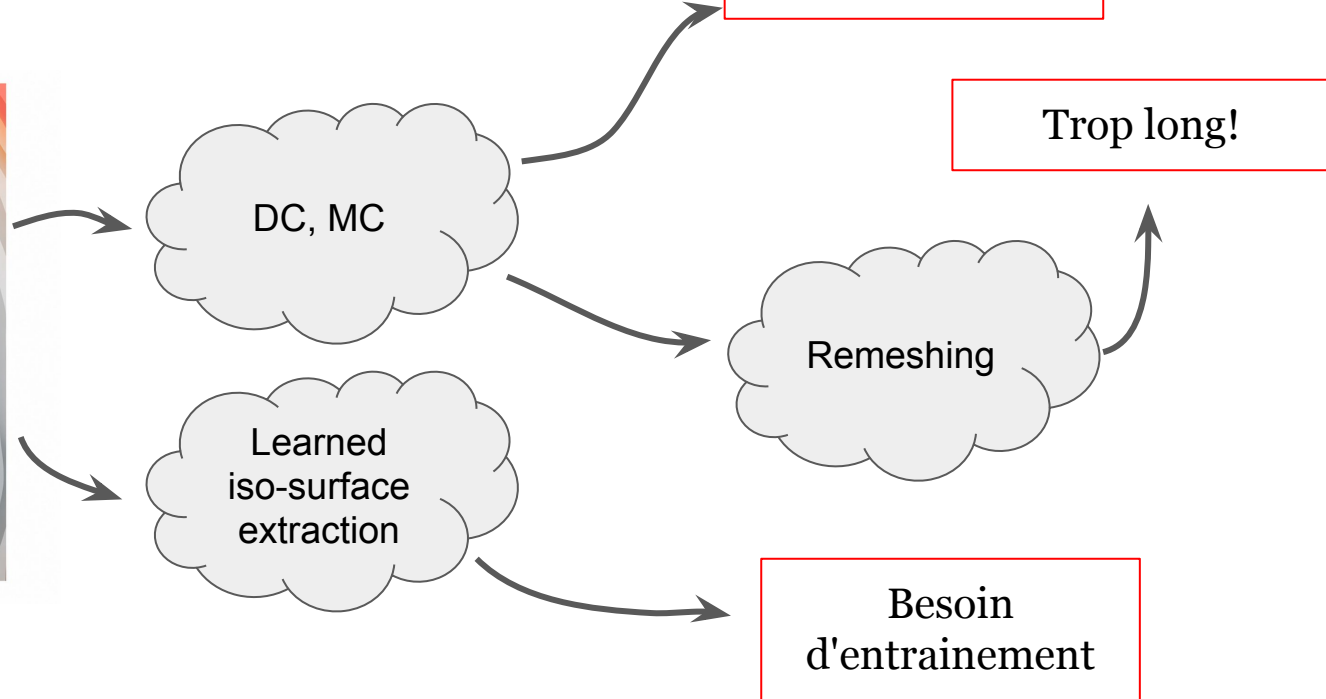
Critères: + **Vitesse** (téléchargement, rendu) + **compacité**

Que faire!

1ère possibilité: Le client extrait le maillage!



Neural
representation

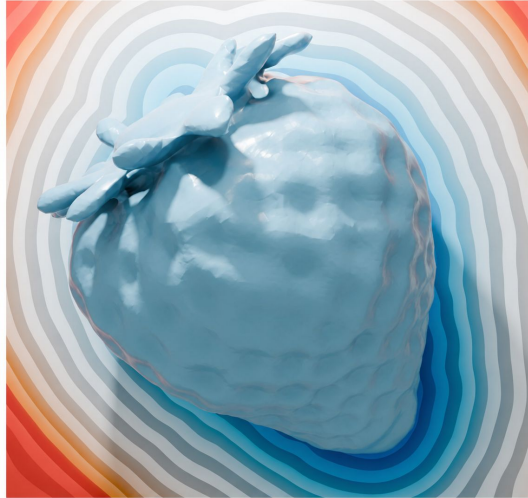


à temps équivalent, leur méthode est plus précise!

Preuve? plus tard!

Que faire!

2ème possibilité: Le serveur extrait le maillage et on compresse



**Neural
representation**

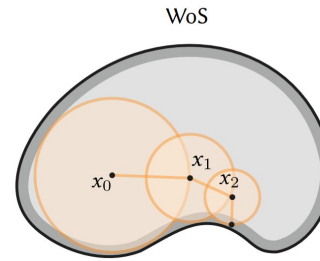


Compression

Méthodes de compression de maillage impactent la qualité de résultats.

Que faire!

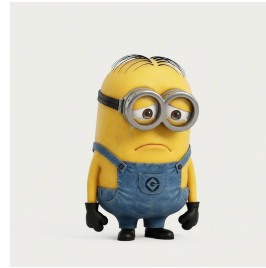
3ème possibilité: Méthodes Meshless



CPM ou méthodes Monte Carlo? Pas besoin de trianguler le domaine!

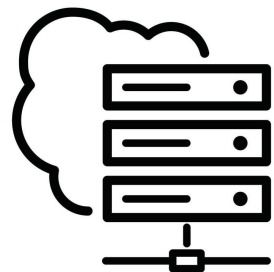


C'est trop lent!

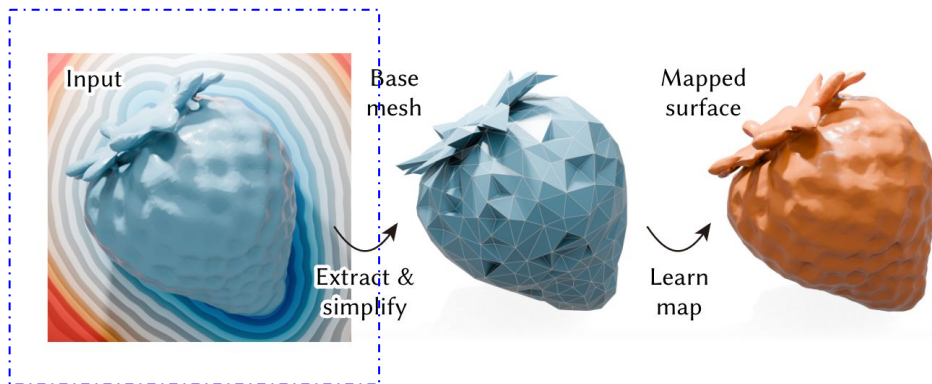


Qu'en est il des méthodes de simulation temps réelle sur GPU?

Méthode



serveur

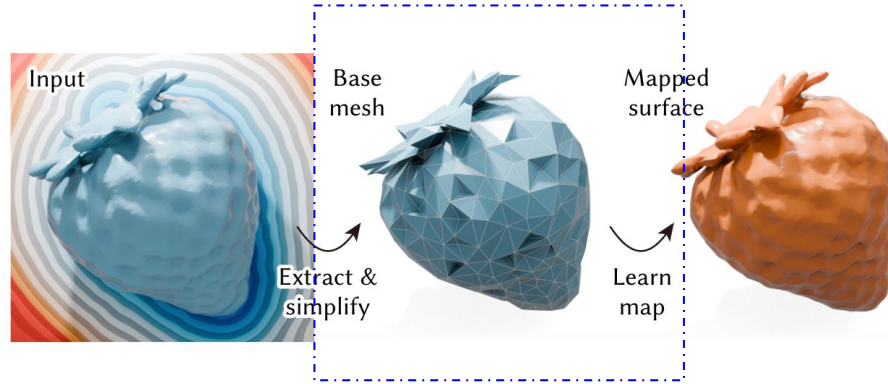


- Extraire un maillage haute résolution à partir du NDF
 - Implicite → MC
 - Nerf → MC avec l'occupancy!
 - Gaussian splats, nuages de points
 - Recaler une TSDF dessus
 - Post traiter le nuage de points et Delauney



utilisateur

Méthode



- Réduire la résolution

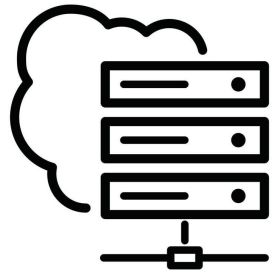
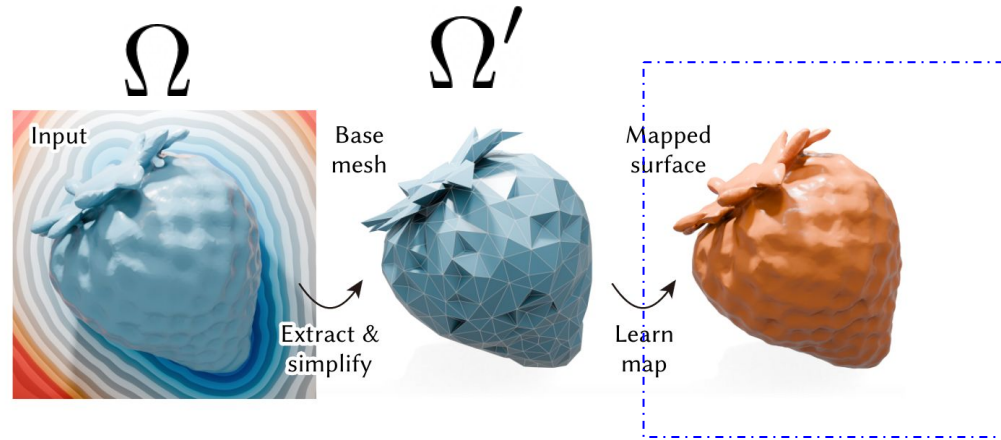


serveur



utilisateur

Méthode



serveur

- Apprendre un champs de déplacement avec un MLP léger
(2 hidden layers de 64 neurones, ReLU)*

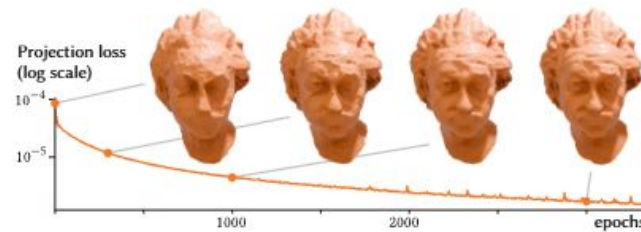


Figure 12: Ω_0 learns to approximate Ω as training proceeds.



utilisateur

*Des ablations sont faites sur ces choix là

Apprendre le displacement field

Les losses sont calculées à partir d'échantillons sur la surface coarse Ω'

$$\underset{\theta, \phi}{\operatorname{argmin}} \lambda_C \mathcal{L}_C(\theta, \phi) + \lambda_P \mathcal{L}_P(\theta),$$

$$\mathcal{L}_C(\theta, \phi) = \frac{1}{s} \sum_{i=1}^s \|g_\phi(g_\theta(\mathbf{x}_i)) - \mathbf{x}_i\|_2^2.$$

$$\mathcal{L}_P(\theta) = \frac{1}{s} \sum_{i=1}^s \|g_\theta(\mathbf{x}_i) - \mathcal{P}(g_\theta(\mathbf{x}_i))\|_2^2.$$

Assurer que le champs de déplacement approxime le neural field

$\mathcal{P}$ est un opérateur de projection sur la surface target (Nerf, une soupe de triangle ...)

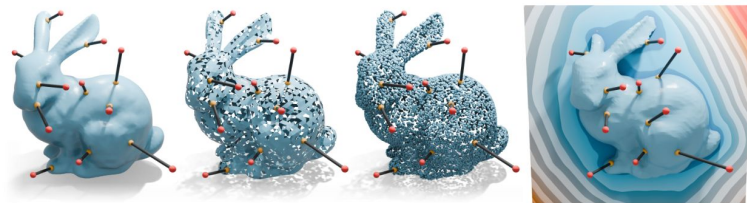


Figure 13: Our projection operator $\mathcal{P}$ supports various surface representations, such as (left to right) meshes, polygon soups, point clouds, or neural occupancy functions where the Eikonal equation is not necessarily met (right).

Et alors?

Extraction de mesh

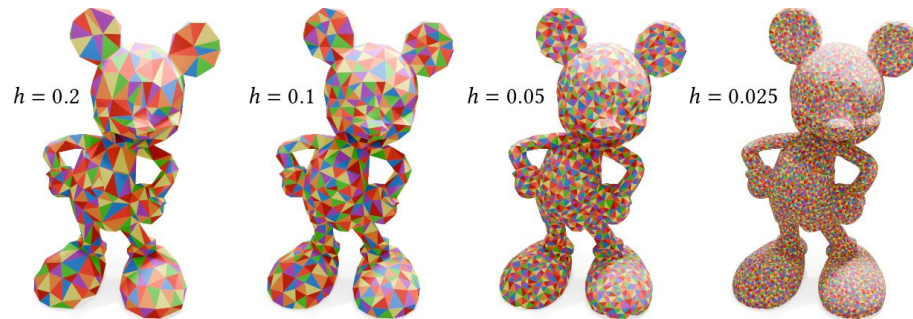


Figure 15: *Our neural displacement field can progressively refine the mesh, allowing users to trade off runtime and accuracy in their computation tasks.*

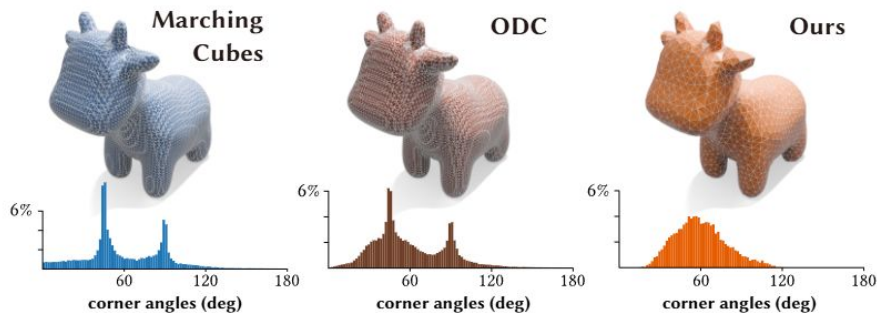


Figure 4: *Our mesh is free of thin sliver triangles unlike isosurfacing methods [LC87, HS24], leading to accurate and efficient computation on the mesh.*

Compression de champs scalaires

Apprendre un champs scalaire en jonction avec le déplacement field

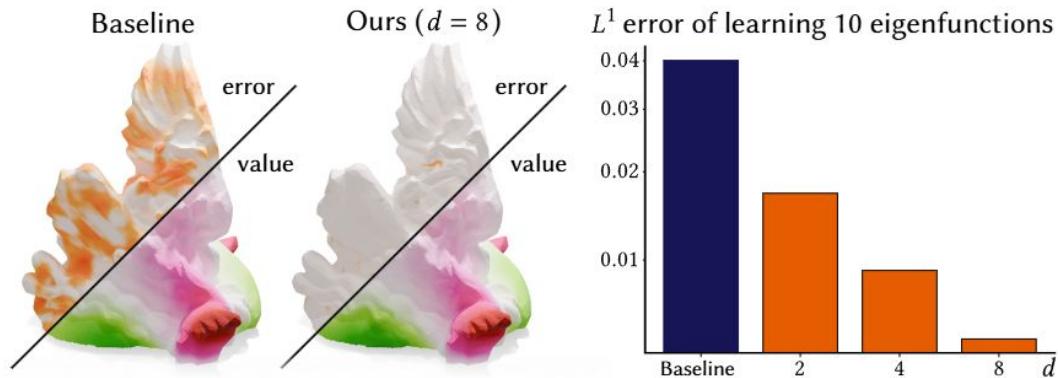
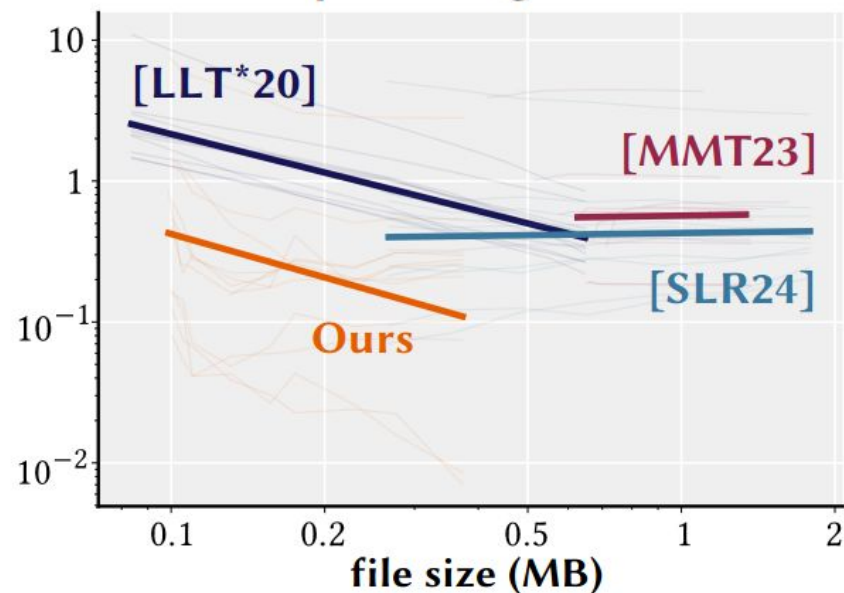


Figure 23: Using our map and our intrinsic encoding, one may train a neural network that represents a scalar function on a surface, like the eigenfunctions of the Laplacian (left). By increasing the dimension of the feature vectors d , our method can achieve lower error without changing the number of layers/neurons of the MLP (right).

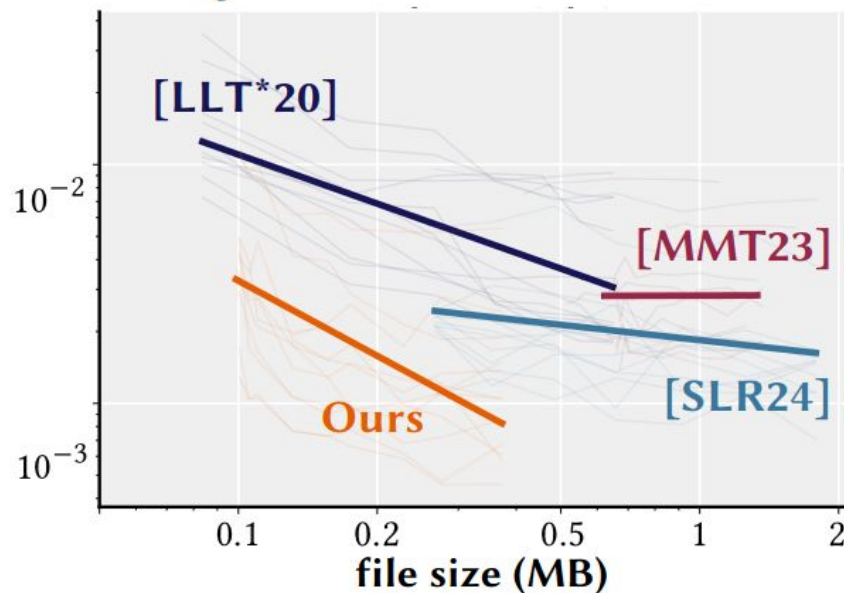
Résultats vs compression

Error vs. file size of geometry processing tasks vs. mesh compression methods

L^1 error of 20 Laplacian eigenvalues



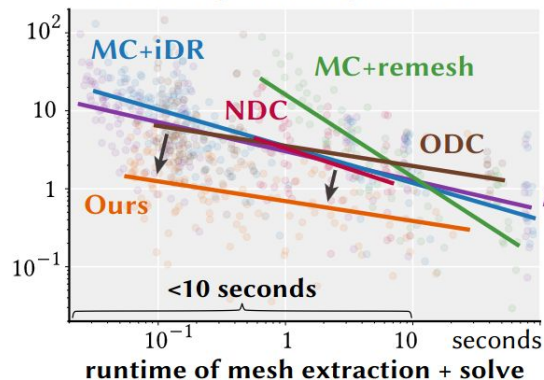
L^1 error of geodesic distance



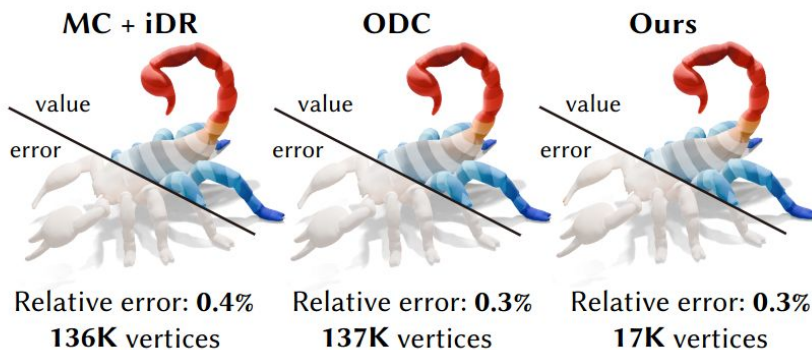
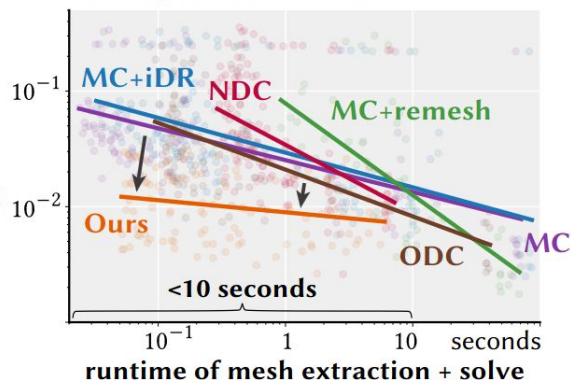
Résultats vs iso-surface extraction

Error vs. client side runtime of geometry processing tasks on **NeuS** [WLL*21]

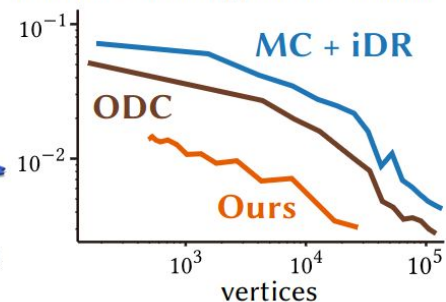
L^1 error of 20 Laplacian eigenvalues



L^1 error of geodesic distance



Relative error of geodesic distance



Résultats si le base mesh n'est pas top

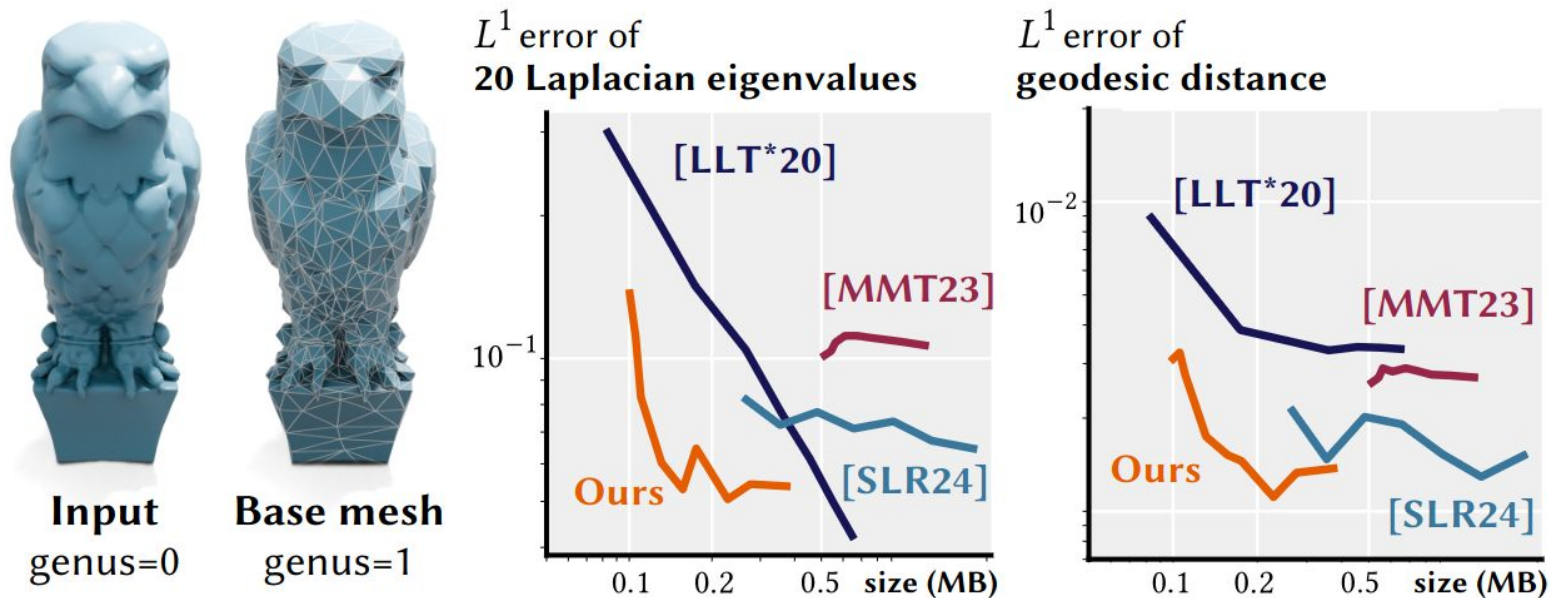


Figure 20: *Our method outperforms other methods, even when the base mesh does not match the topology of the input surface.*

“That’s all Folks!”