隐式场里的可微渲染与仿真

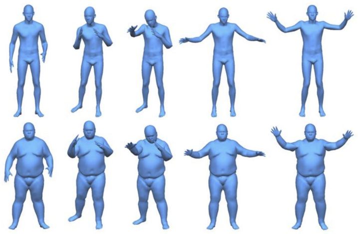
GAMES Webinar 几何专题

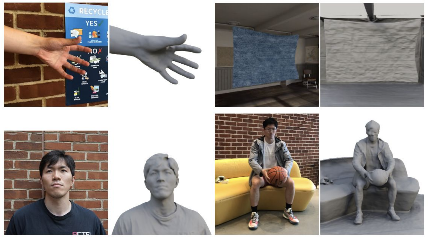
乔怿凌

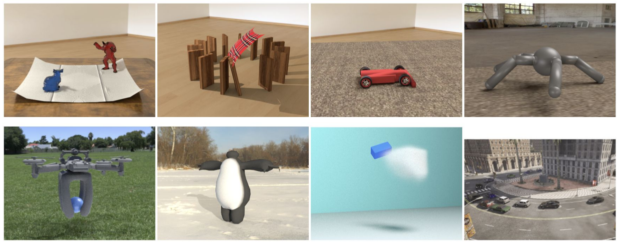
UNIVERSITY OF MARYLAND

个人介绍

- 乔怿凌 (ylqiao.net)
- 本科：中国科学院大学，2015 - 2019
  - 数学、计算机双学位
  - 中科院计算所科研，导师：高林老师，陈熙霖老师
  - 研究方向：3D geometry processing
- 博士：马里兰大学，2019 至今
  - 计算机专业
  - 导师：Ming Lin
  - 研究方向：Differentiable Simulation

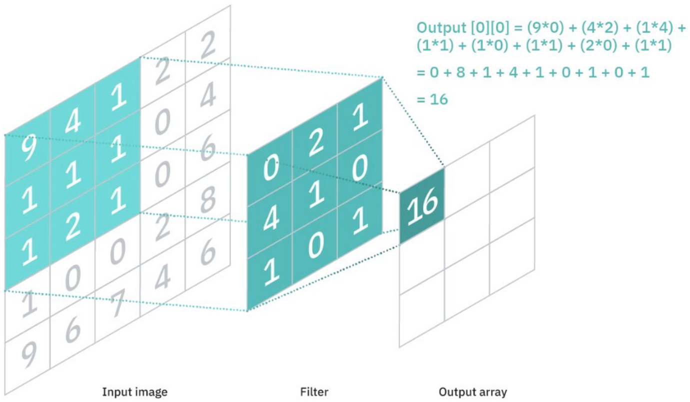


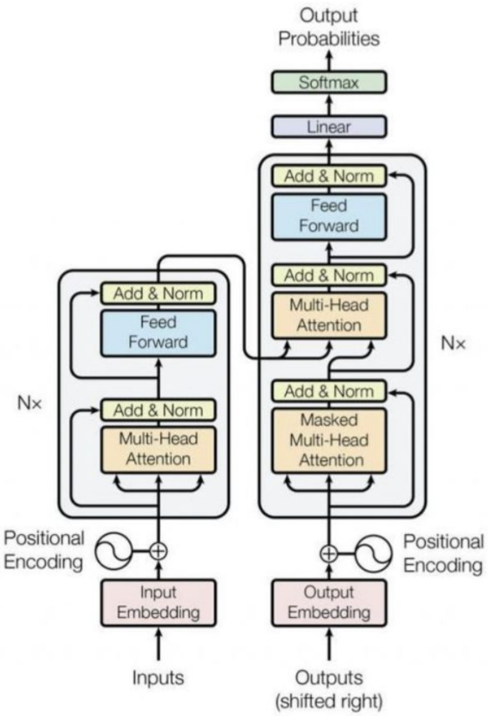




Deep Learning Is a Powerful Tool

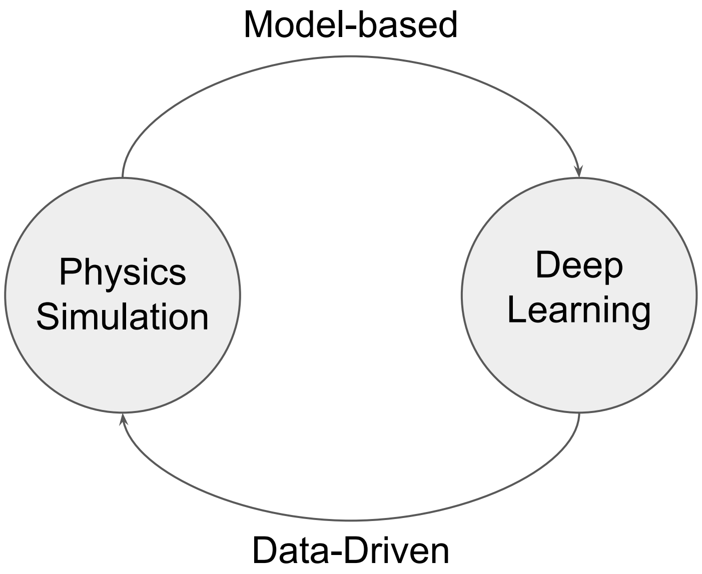
- In Vision, Language, etc…
- CNNs, Transformers, etc…



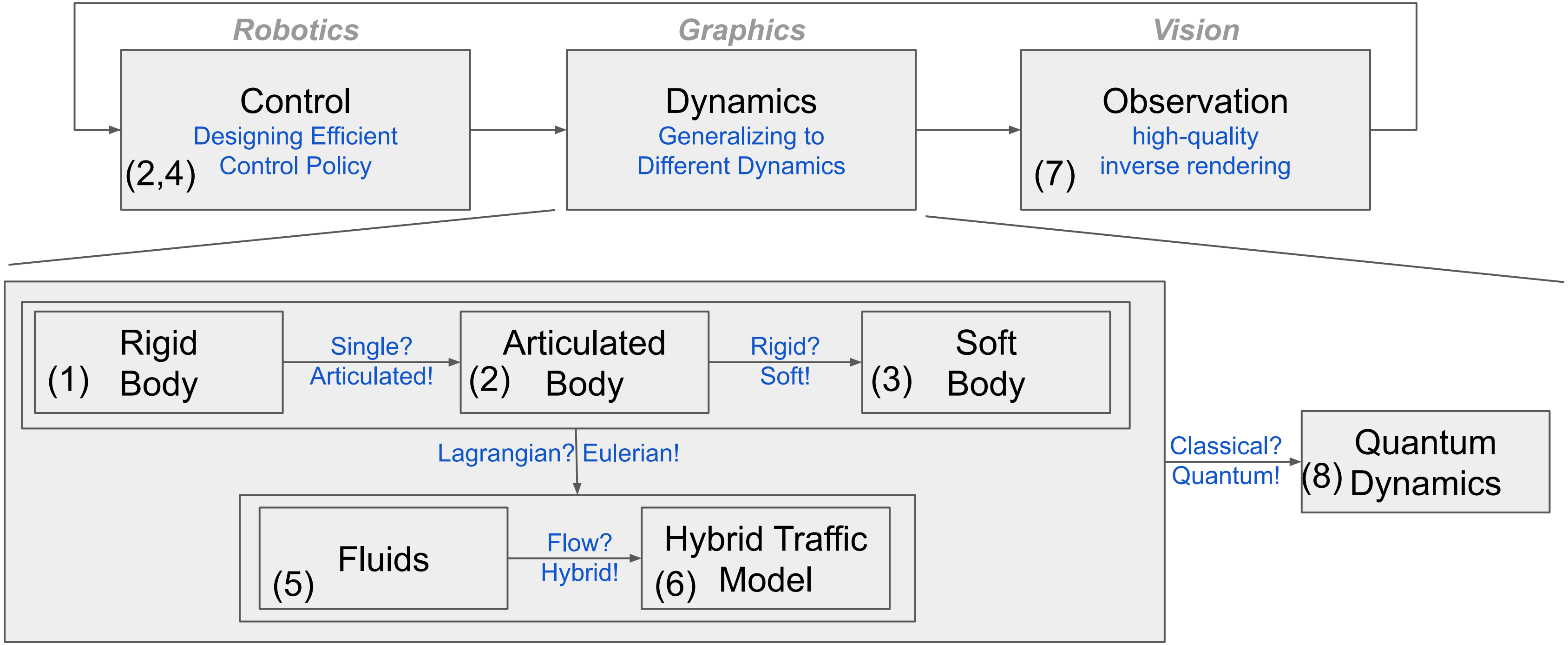


Physics Simulation

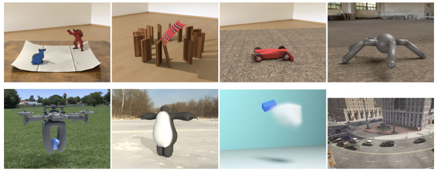
- Serve as an inductive bias for learning algorithms
- Utilize the power of deep learning to solve physics problems



My Research on Differentiable Simulation

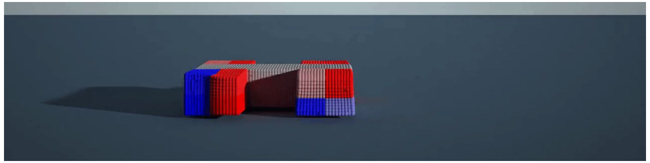


(1) Scalable Differentiable Physics for Learning and Control. *(ICML 2020)*
(2) Efficient Differentiable Simulation of Articulated Bodies. *(ICML 2021)*
(3) Differentiable Simulation of Soft Multi-body Systems. *(NeurIPS 2021)*
(4) OF-VO: Efficient Navigation among Pedestrians Using Commodity Sensors. *(ICRA/RAL 2021)*
(5) Differentiable Fluids with Solid Coupling for Learning and Control. *(AAAI 2021)*
(6) Differentiable Hybrid Traffic Simulation. *(Siggraph Asia 2022, Journal Track)*
(7) NeuPhysics: Editable Neural Geometry and Physics from Monocular Videos. *(NeurIPS 2022)*
(8) Differentiable Analog Quantum Computing for Optimization and Control. *(NeurIPS 2022)*

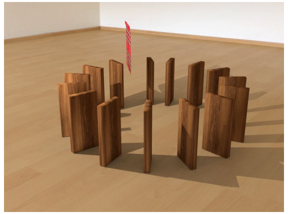


隐式场里的可微渲染与仿真

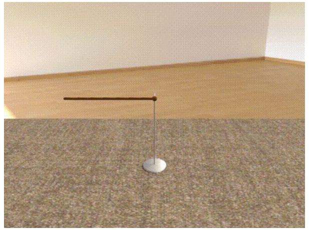
- 之前的可微仿真基本都是基于synthetic data
- 如何将可微仿真应用于真实的数据上？
- Real world video ← (?) → Physics Simulation
- 几何重建是关键！要想仿真，首先需要已知的几何模型

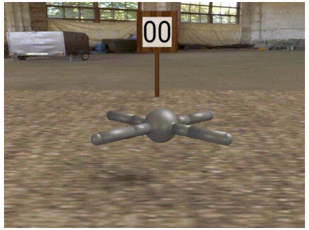


DiffTaichi



Warp





We Choose Neural Implicit Field [4]!

- Real world video ← NeuS [2] + Non-rigid NeRF [3] → Physics Simulation
- Reconstruct arbitrary geometry from real world data
- Image → geometry reconstruction → simulation → rendering → new Image
- Input:
    - A monocular RGB video
- Output:
    - Signed distance field for each frame
    - Physics parameters of the chosen object
    - Interfaces for video editing

NeuPhysics: Editable Neural Geometry and Physics from Monocular Videos

Yi-Ling Qiao*, Alexander Gao*, and Ming Lin

UNIVERSITY OF MARYLAND

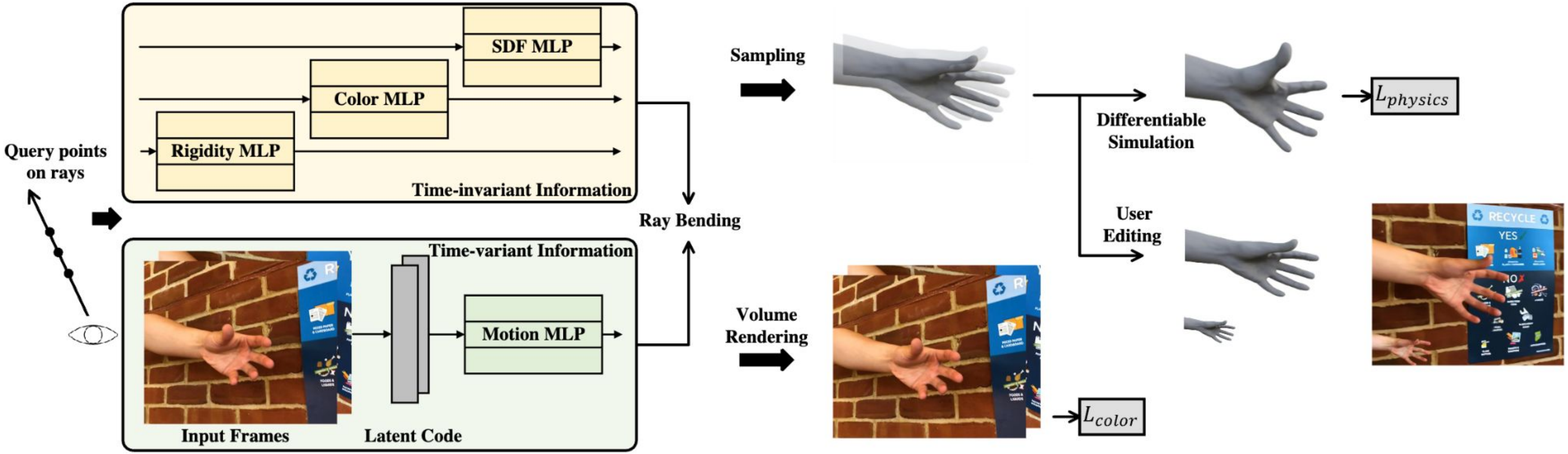
Contact: yilingq@umd.edu, awgao@umd.edu

Project Page (with code and data): https://sites.google.com/view/neuphysics

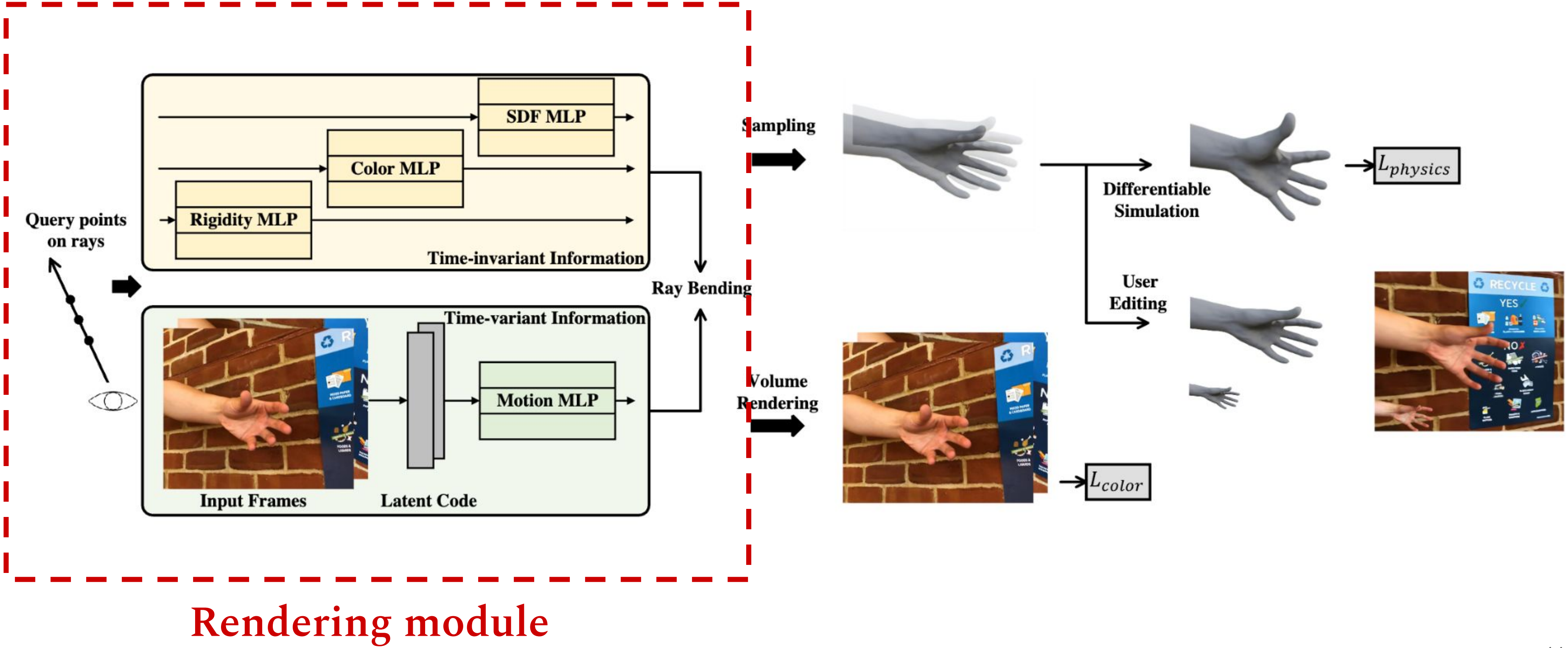
Motivation

- Reconstruct detailed 3D geometry from monocular videos
- Estimate physics parameters from videos
- Enable consistent and physically-based editing of the original video

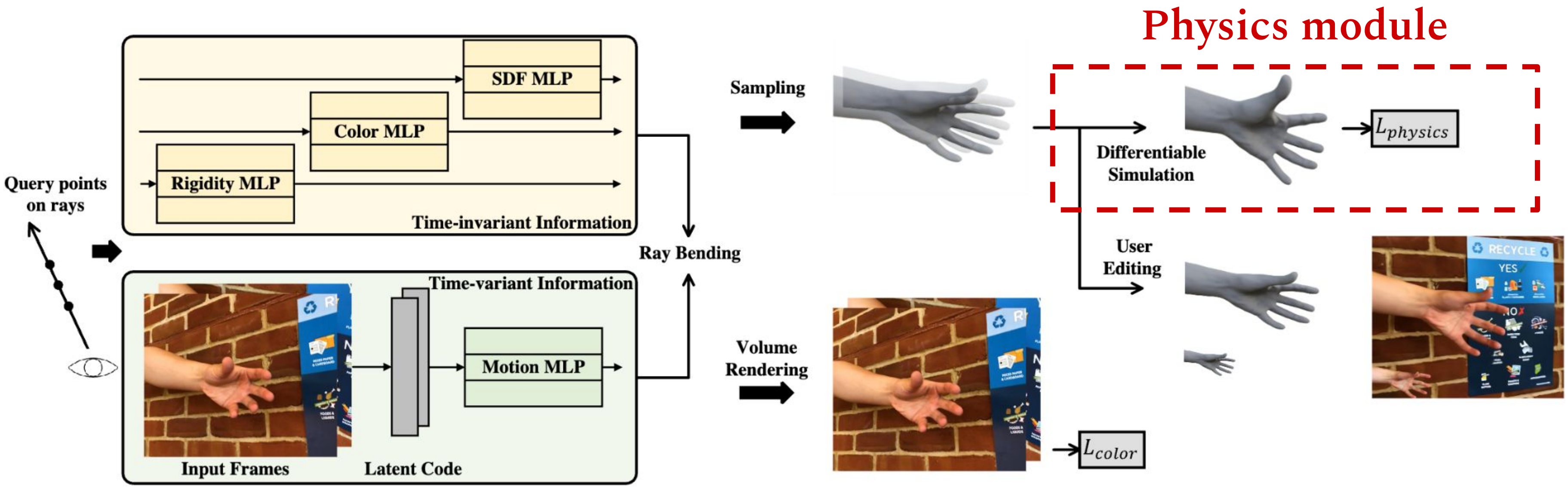
Pipeline


Query points
on rays
SDF MLP
Color MLP
Rigidity MLP
Time-invariant Information
Time-variant Information
Input Frames
Latent Code
Motion MLP
Ray Bending
Sampling
Volume
Rendering
Differentiable
Simulation
User
Editing
$L_{physics}$
L_{color}

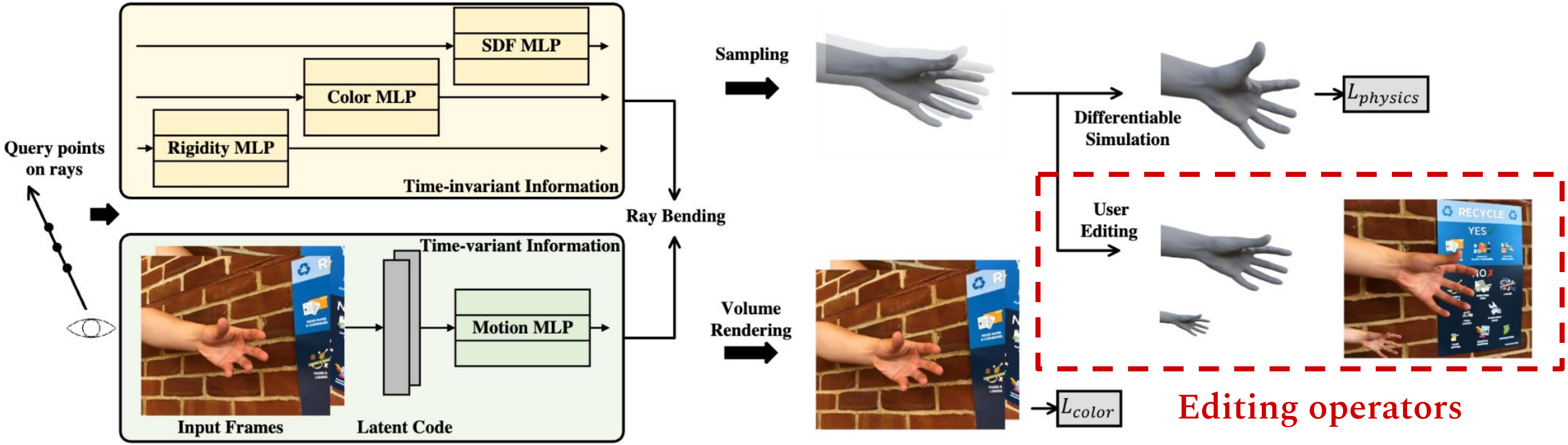
Pipeline: Rendering Module



Pipeline: Physics Module

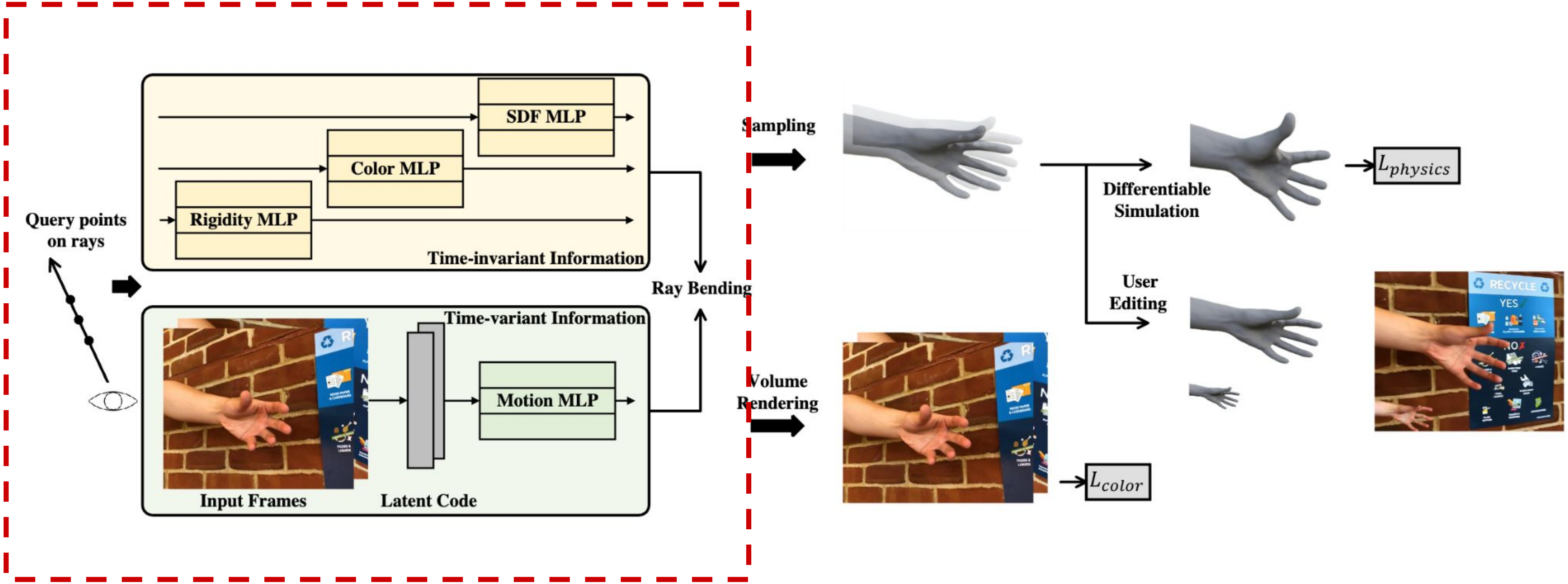


Pipeline: Editing Operators



Methods

Rendering



Rendering module

Rendering

- Pixels are rendered by volume rendering [**2**, **4**] with ray bending [**3**].

$$\mathbf{C}(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(s(\mathbf{p}(t))) \cdot \mathbf{c}(\mathbf{p}(t))\, \mathrm{d}t$$

  - $s_{\theta_{SDF}}$:= SDF Field
  - $\mathbf{c}_{\theta_{color}}$:= Color Field
  - $r_{\theta_{rigidity}}$:= Rigidity Field
  - $\mathbf{b}_{\theta_{motion}}$:= Motion Field

*More details can be found in our paper.

Rendering

- Pixels are rendered by volume rendering [**2**, **4**] with ray bending [**3**].

$$\mathbf{C}(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} w(s(\mathbf{p}(t))) \cdot \mathbf{c}(\mathbf{p}(t))\, \mathrm{d}t$$

- $s_{\theta_{SDF}}$:= SDF Field
- $\mathbf{c}_{\theta_{color}}$:= Color Field
- $r_{\theta_{rigidity}}$:= Rigidity Field

Defined on the canonical frame

- $\mathbf{b}_{\theta_{motion}}$:= Motion Field

*More details can be found in our paper.

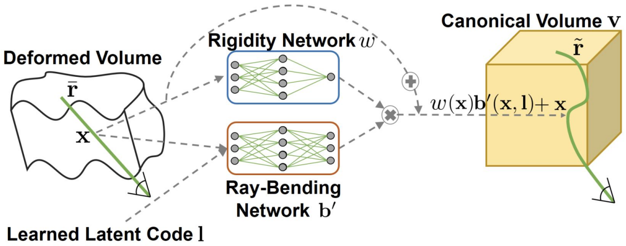
Rendering

- Pixels are rendered by volume rendering [**2, 4**] with ray bending [**3**].

$$\mathbf{C}(\mathbf{o},\mathbf{v}) = \int_0^{+\infty} w(s(\mathbf{p}(t)))\cdot \mathbf{c}(\mathbf{p}(t))\,\mathrm{d}t$$

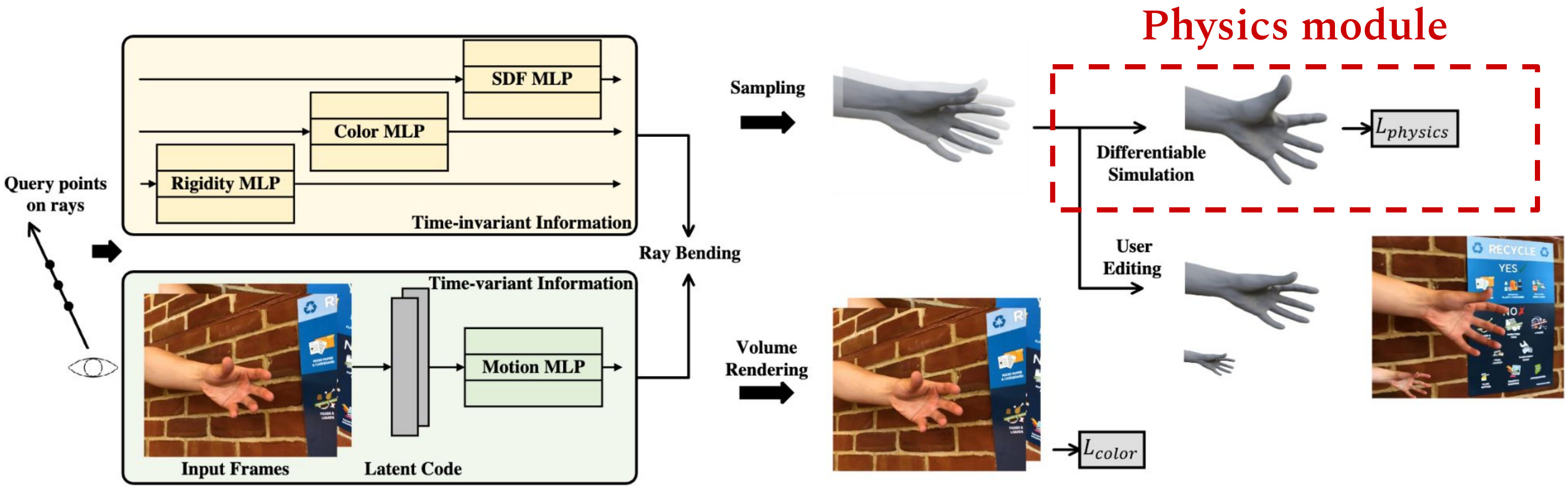
- $s_{\theta_{SDF}}$:= SDF Field
- $\mathbf{c}_{\theta_{color}}$:= Color Field
- $r_{\theta_{rigidity}}$:= Rigidity Field
- $\mathbf{b}_{\theta_{motion}}$:= Motion Field

Optimized per frame



*More details can be found in our paper.

Physics



Physics module
SDF MLP
Color MLP
Rigidity MLP
Query points on rays
Time-invariant Information
Time-variant Information
Motion MLP
Input Frames
Latent Code
Ray Bending
Sampling
Volume Rendering
Differentiable Simulation
$L_{physics}$
User Editing
L_{color}

Physics

- The simulation is integrated using Projective Dynamics [1]:

$$\mathbf{M}(\mathbf{Q}_{i+1} - \mathbf{Q}_i - h\dot{\mathbf{Q}}_i) = h^2(\nabla E(\mathbf{Q}_{i+1}, \theta_{physics}) + \mathbf{f}_{ext})$$

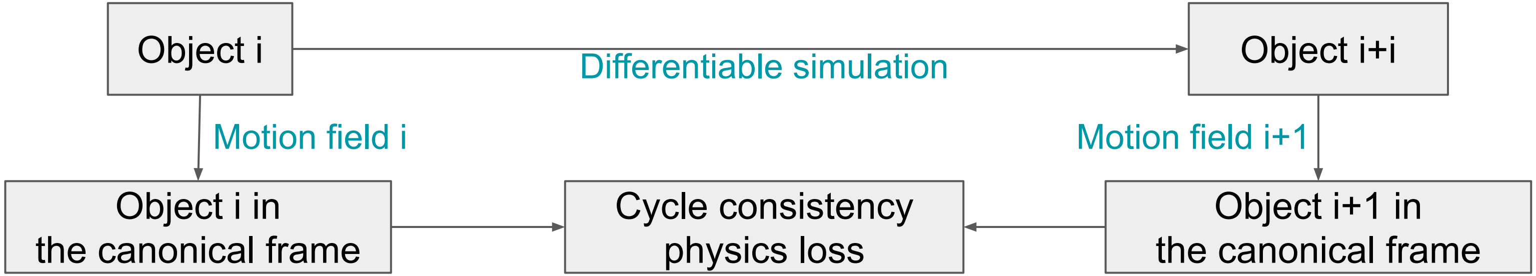
  - $\mathbf{M}$:= mass
  - $\mathbf{Q}$:= position
  - $\dot{\mathbf{Q}}$:= velocity
  - $\boldsymbol{h}$:= time step
  - $\boldsymbol{E}$:= Energy
  - $\boldsymbol{\theta}$:= physics parameters
  - $\mathbf{f}_{ext}$:= external forces

- We use the volume mesh projective dynamics implementation from DiffPD [5]

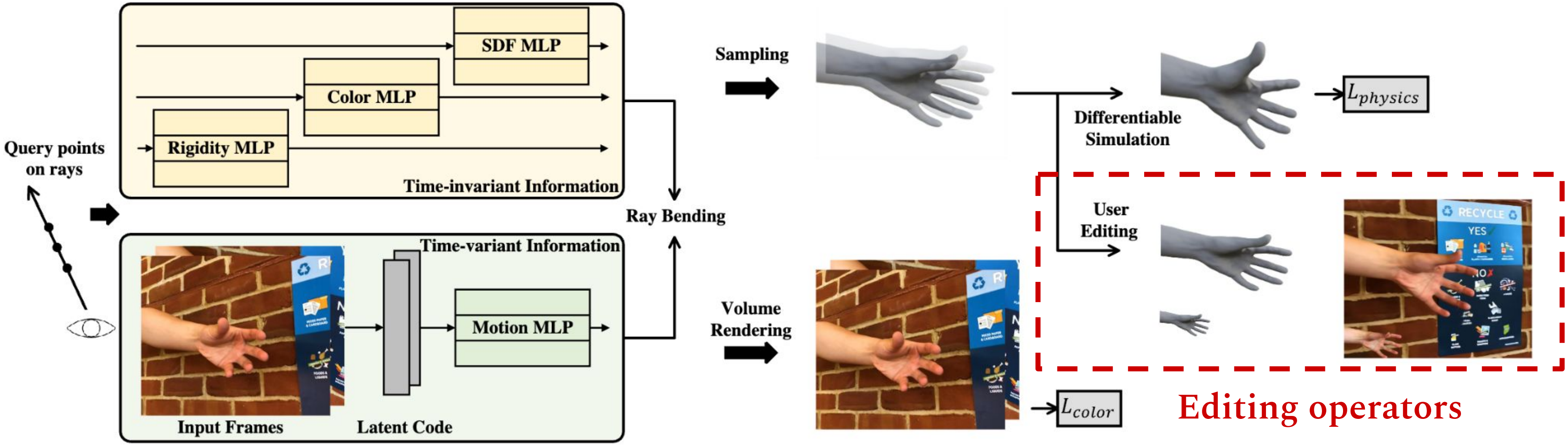
Physics

- The simulated objects are sampled from the learned SDF Field.
- How to find the correspondence of the simulated object between two frames?
- We optimize a cycle-consistency physics loss to get the physics parameters:

$$\mathcal{L}_{physics} = \left\| (\mathbf{m}_i + \mathbf{b}^{\theta_{rigidity}}_{\theta_{motion}}(\mathbf{l}_i, \mathbf{m}_i)) - (\mathbf{m}'_{i+j}(\theta_{physics}) + \mathbf{b}^{\theta_{rigidity}}_{\theta_{motion}}(\mathbf{l}_{i+j}, \mathbf{m}'_{i+j}(\theta_{physics}))) \right\|_2$$



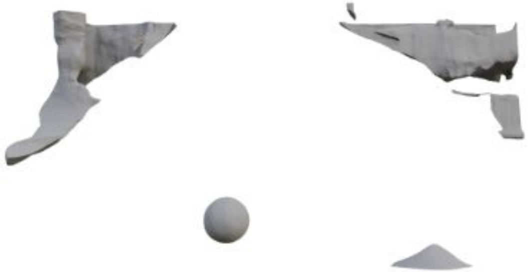
Editing



SDF MLP
Color MLP
Rigidity MLP
Time-invariant Information
Query points on rays
Time-variant Information
Motion MLP
Input Frames
Latent Code
Ray Bending
Sampling
Volume Rendering
Differentiable Simulation
$L_{physics}$
User Editing
L_{color}
Editing operators

Editing

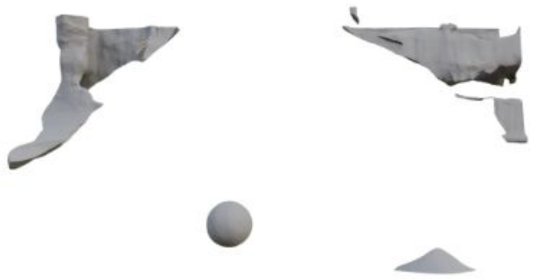
- Select Foreground:



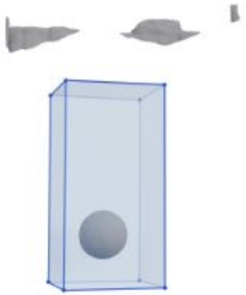
Original SDF Field

Editing

- Select Foreground:



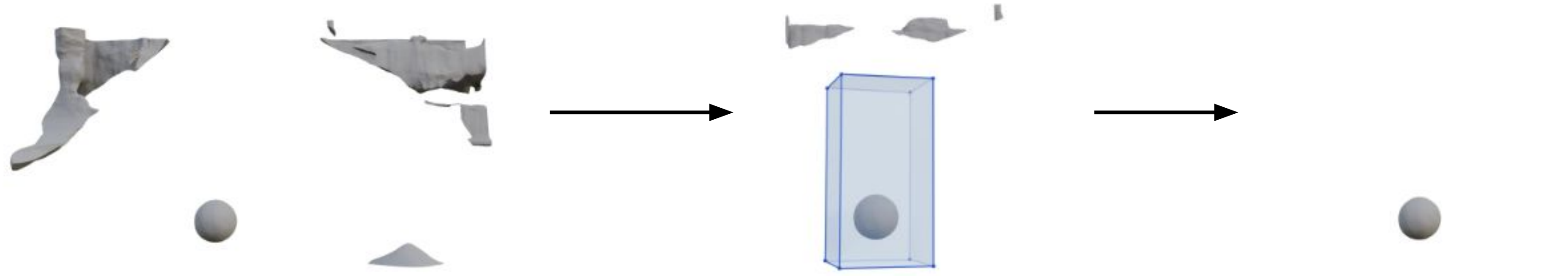
Original SDF Field



Rigidity Threshold + Bounding box

Editing

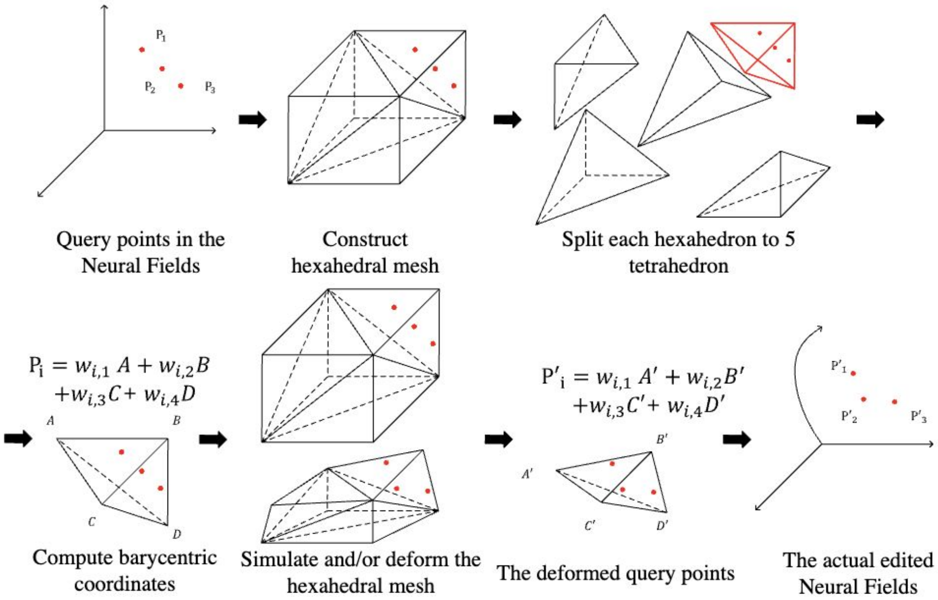
- Select Foreground:



Original SDF Field **Rigidity Threshold + Bounding box** **Extracted Foreground**

Editing

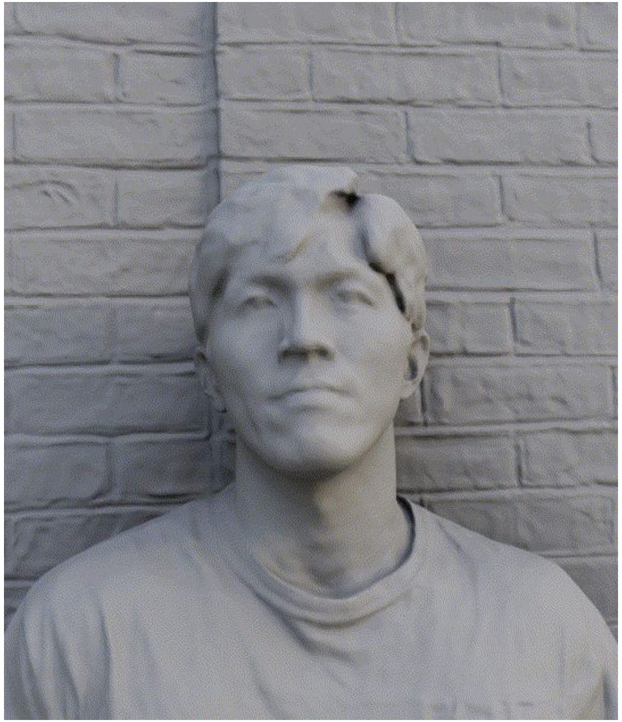
- Select foreground
- Sample hexahedral mesh
- Simulate/edit hexahedral mesh
- Compute deformation
    - Following NeRF-Editing [6]
- Rendering with ray bending



Results

Results: Dynamic Geometry Reconstruction

Face:



Ours

D-NeRF

NeuS

Hand:

Ours

D-NeRF

NeuS

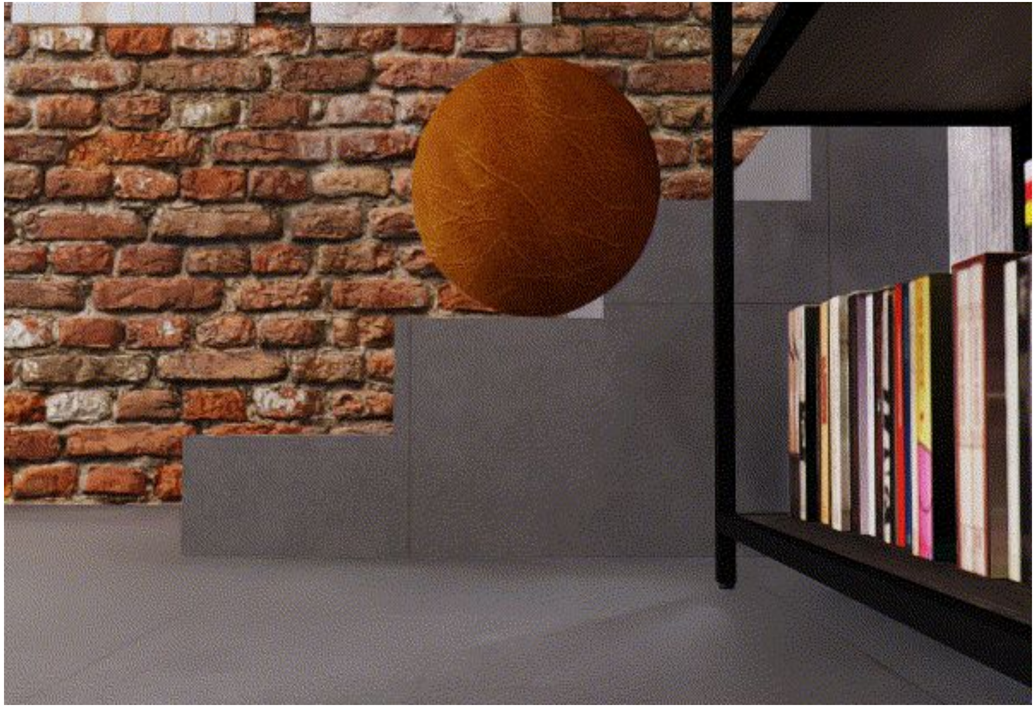
Human holding object:

Ours

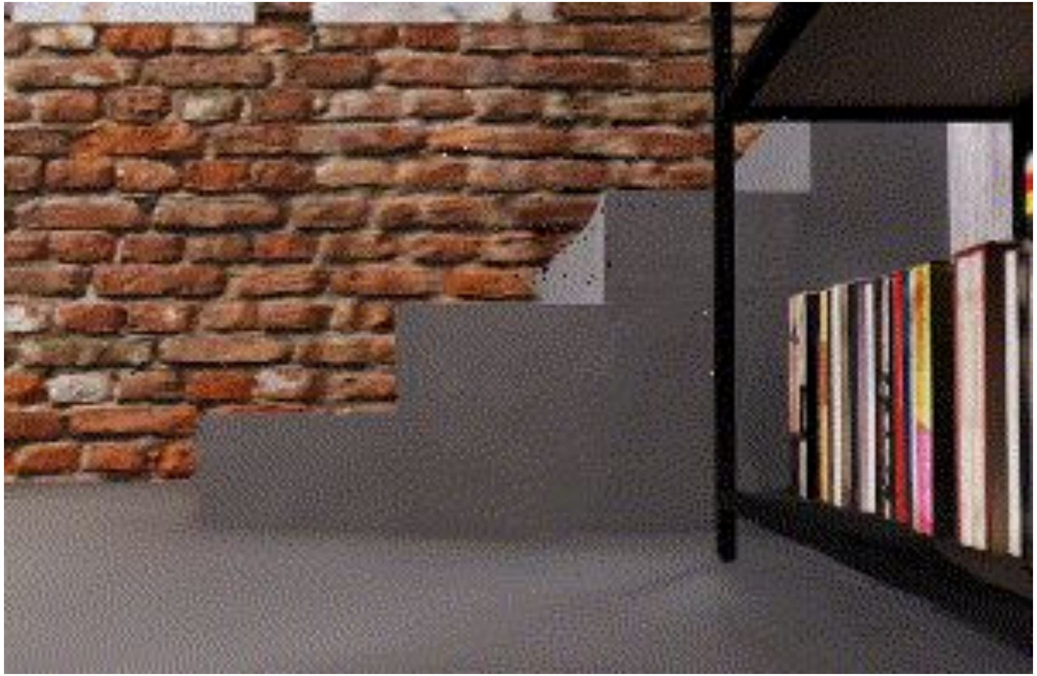
D-NeRF

NeuS

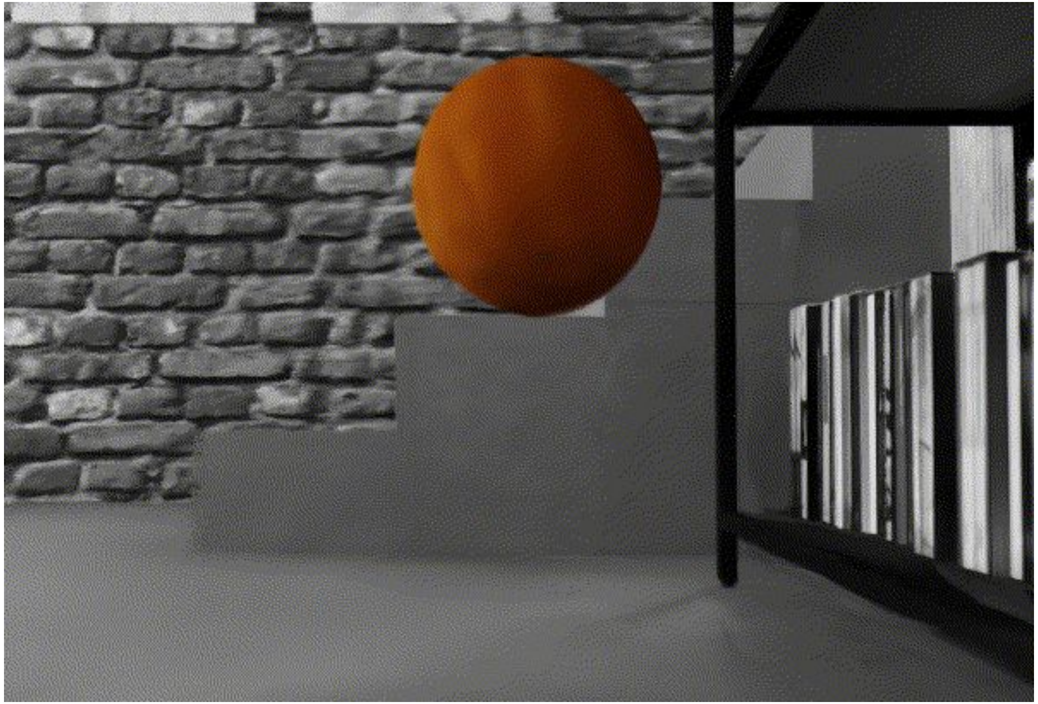
Editing: (Original Source Videos)



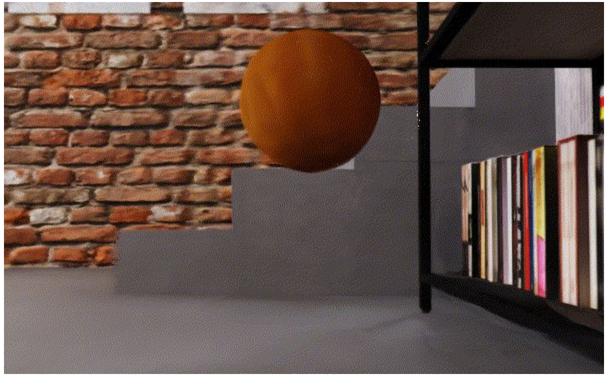
Editing: Remove Foreground



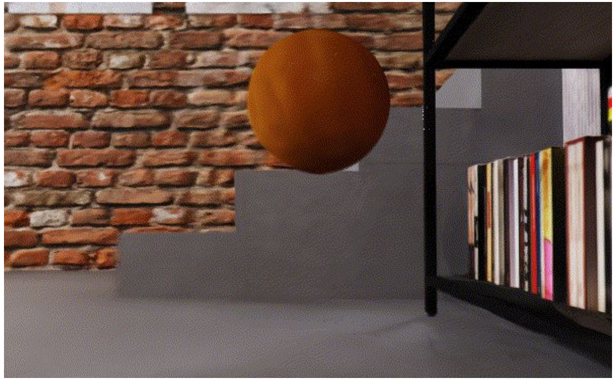
Editing: Highlight Foreground



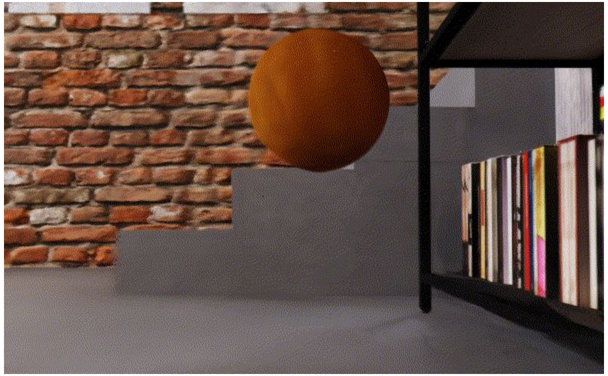
Editing: Change Velocity



Add velocity toward left



Ground Truth
(no horizontal velocity)



Add velocity toward right

Editing: Change Physical Material

Softer Material

Material Similar to Ground Truth

Harder Material

Future Work

- Integrate with other differentiable simulators to support more object types (e.g. articulated body and fluids).
- Incorporate stronger priors (e.g. 3D structure and semantic information).
- Improve editing quality (i.e. removing artifacts).
- Estimate textures and UV maps simultaneously with geometry.
- Improve training and inference speed.

Summary of Contributions

- An end-to-end differentiable simulation and rendering framework capable of *estimating and editing the geometry, appearance, and physics parameters from a monocular video*.

- A *bi-directional conversion between the implicit neural fields and explicit hexahedral mesh*, which allows users to *interact with the 3D scene* (add, delete, move, deform, simulate, etc.) and obtain high-quality rendering from novel camera views.

- By connecting our geometry reconstruction pipeline to the differentiable physics engine, we avoid manual construction of 3D models and poor initialization for the differentiable simulator, *enabling geometry and dynamics to be learned from scratch.*

References

- [**1**] Du, Tao, Kui Wu, Pingchuan Ma, Sebastien Wah, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. "Diffpd: Differentiable projective dynamics." ACM Transactions on Graphics (TOG) 41, no. 2 (2021): 1-21.
- [**2**] Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T. and Wang, W., 2021. "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction." arXiv preprint arXiv:2106.10689.
- [**3**] Tretschk, E., Tewari, A., Golyanik, V., Zollhöfer, M., Lassner, C. and Theobalt, C., 2021. "Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video." In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 12959-12970).
- [**4**] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R. and Ng, R., 2021. "NeRF: Representing scenes as neural radiance fields for view synthesis." Communications of the ACM, 65(1), pp.99-106.
- [**5**] Bouaziz, S., Martin, S., Liu, T., Kavan, L. and Pauly, M., 2014. "Projective dynamics: Fusing constraint projections for fast simulation." ACM transactions on graphics (TOG), 33(4), pp.1-11.
- [**6**] Yuan, Yu-Jie, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao. "NeRF-editing: geometry editing of neural radiance fields." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18353-18364. 2022.

Q & A

Thank you!