Homework 3
PhysGaussian

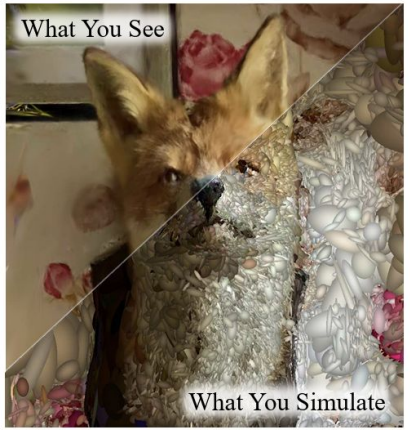
Embodied Vision
NTU CSIE, Spring 2025

Outline

- Problem
- Grading
- Environment
- Submission
- Delay Policy

Problem: Physics-Integrated 3DGS

PhysGaussian provides a unified representation for dynamic 3D scenes in which rendered motions and deformations strictly adhere to underlying physical laws. It generates high-fidelity animations across a spectrum of materials—elastic solids, plastic bodies, non-Newtonian fluids, and granular media—while maintaining interactive performance. By adhering to the “what you simulate is what you render” principle, it delivers consistent, physically accurate visualizations in both synthetic and real-world reconstructions.



[1] Xie, Tianyi, et al. "Physgaussian: Physics-integrated 3d gaussians for generative dynamics." *In CVPR*. 2024.

Part 1 – PhysGaussian Baseline

- Please select **any two** of the following materials—jelly, metal, sand, foam, snow, and plasticine—to simulate and produce a video.
- The primary script is **gs_simulation.py**, which, for each frame, drives the physical evolution via the **p2g2p()** function in the **mpm_solver**, involving the following parameters:

n_grid	Resolution of the MPM background grid per dimension
substep_dt	Number of p2g2p substeps per frame
grid_v_damping_scale	Grid velocity damping factor (>1 accelerates dissipation; <1 decays)
softening	Stress softening factor in the constitutive model

Part 2 – Exploring MPM Parameter Effects

- Please try adjusting the aforementioned physical parameters—including **n_grid**, **substeps**, **grid_v_damping_scale**, and **softening**—on the two baseline simulations introduced earlier.
- For each parameter adjustment, please record the changes in **PSNR values** along with the corresponding **simulation videos**.
- For these adjustments and their corresponding effects, please describe the changes you **observed** and the **insights** you gained.
- **BONUS (short answer)** – The paper notes that a key limitation of PhysGaussian is that most **material-related parameters are manually defined**, which confines its applicability to known materials and precludes use with unfamiliar ones. If we wanted to extend this method to automatically infer and generate the appropriate parameters for arbitrary target materials, how would you design the framework?

Grading

- **Part 1 (20%):**
  - Baseline simulation videos for the two materials (20%)

- **Part 2 (75%):**
  - Describe the adjustments you made to the physical parameters (10%)
  - Record the PSNR and simulation videos for each parameter variation (25%)
  - Share your key takeaways, findings encountered during the ablation study (25%)
  - upload your simulation results to YouTube and provide the link in your README.md (10%)
  - short answer for BONUS part (5%)

Environment

Please follow the instructions under the "*Python Environment*" section in the official `README.md` to set up the environment.

Submission

- Deadline: **2025/6/15 (Sun.) 23:59 (GMT+8)**
- Please click the link below and sign in to your GitHub account to access your submission repository: https://classroom.github.com/a/SdXSjEmH
- Kindly link your GitHub account to the classroom with your **student ID**.
- If your student ID is not listed, please contact TA by Discord.
- We will clone the **main branch** of your repository.
- Please submission your **GitHub classroom link** to **NTU COOL Homework 3**.

Submission

- Your GitHub repository should include the following files:
  - All codes
  - README.md

Delay Policy

- We offer a total of **5 free late days** within 3 homework submissions this semester. Beyond this, Your score will be **multiplied by 0.7 for each day** it is submitted late.
- If you wish to use your late day allowance or submit an earlier version, please complete and submit this form. You may submit the form **only once**, and **no edits** are allowed after submission.

Google Form: https://forms.gle/N94QYebnVHoAcb967

Final Reminder

- **Do NOT** send private messages to TAs via Facebook or spam TAs email.
- TAs will **NOT** debug for you (e.g., coding, environment, dependencies, etc. issues).
- As mentioned in the first class, we **DO NOT** provide any computation resources.
- TA is here to help. Discussions and Q&As are welcome. Question can be asked in Discord hw2 discussion group:

  https://discord.com/channels/1340225584550121504/1340225584550121512