

Optimizing FMGS

Optimizing Open-Vocabulary-Embedded 3D Gaussian Splats

Background: What is FMGS?

“**F**oundation **M**odel embedded 3D **G**aussian **S**plats”

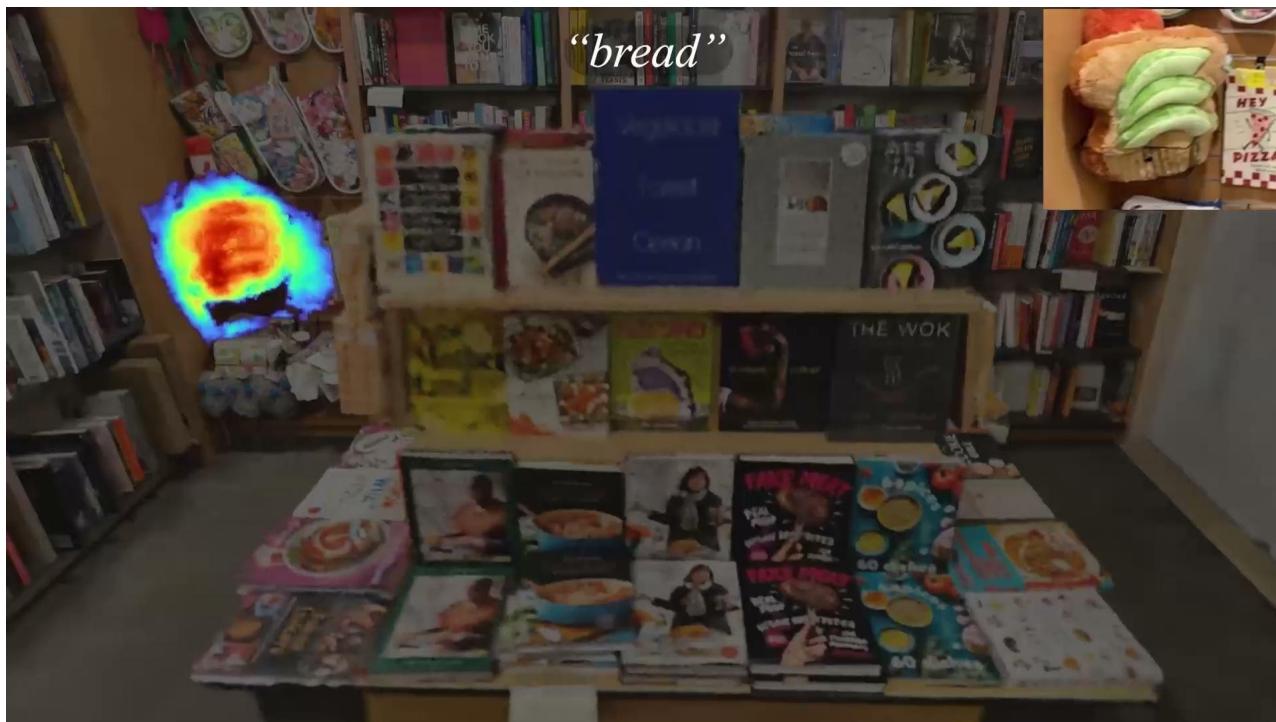
Essentially: outlines 3D scene content
most relevant to a given query.

Background: FMGS (2024)



Background: LERF (2023)

“Language Embedded Radiance Fields”



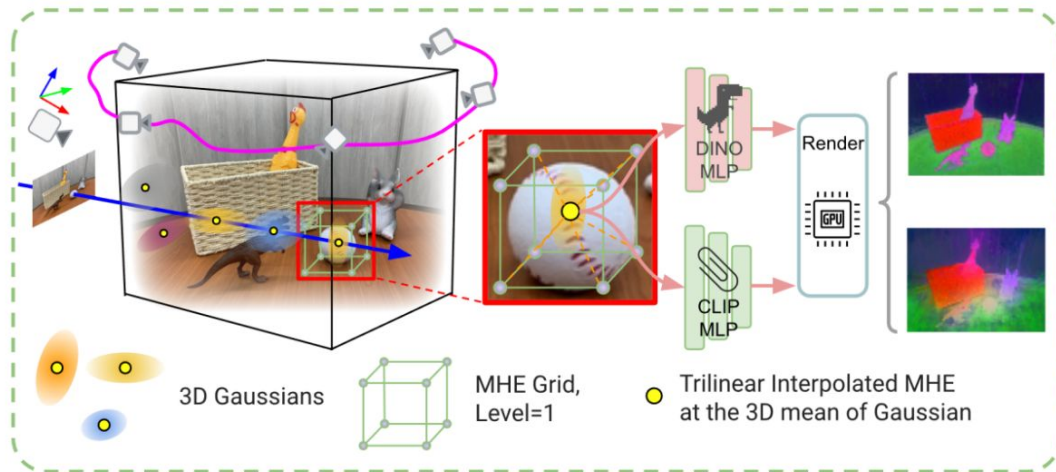
Motivations

- Large VRAM use
- Training time (can be improved)

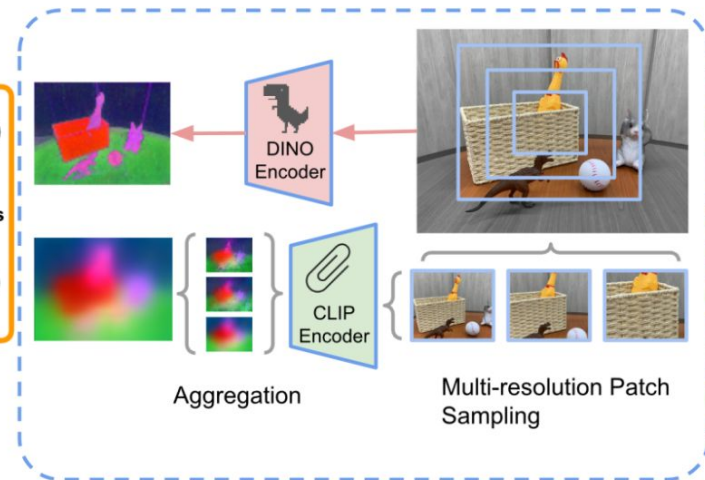
Goal: We want to extend FMGS such that it takes **less VRAM** and is **faster to train**.

Key Contribution

Feature Rendering with Gaussian Splatting

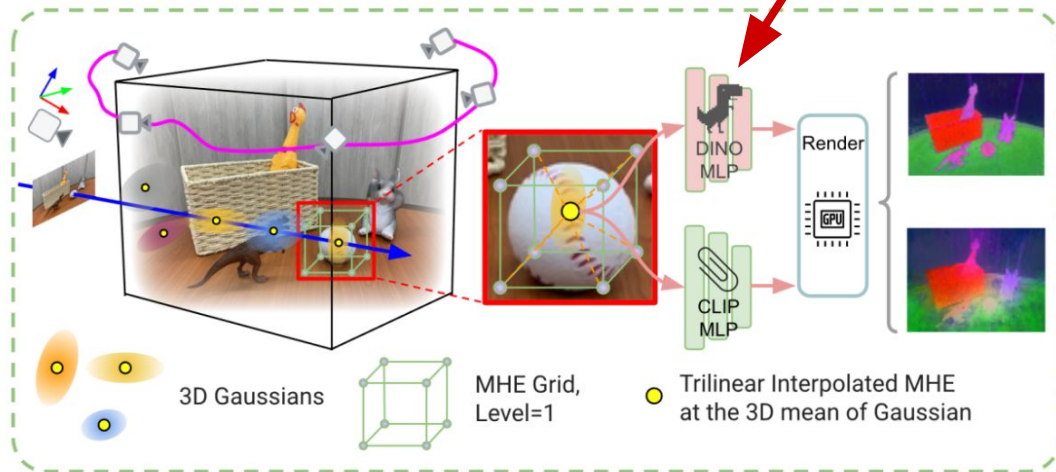


Supervision from Foundation Models

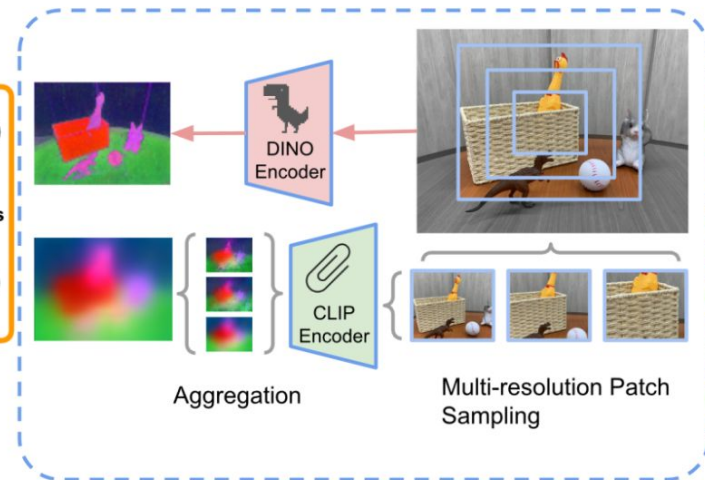


Key Contribution

Feature Rendering with Gaussian Splatting

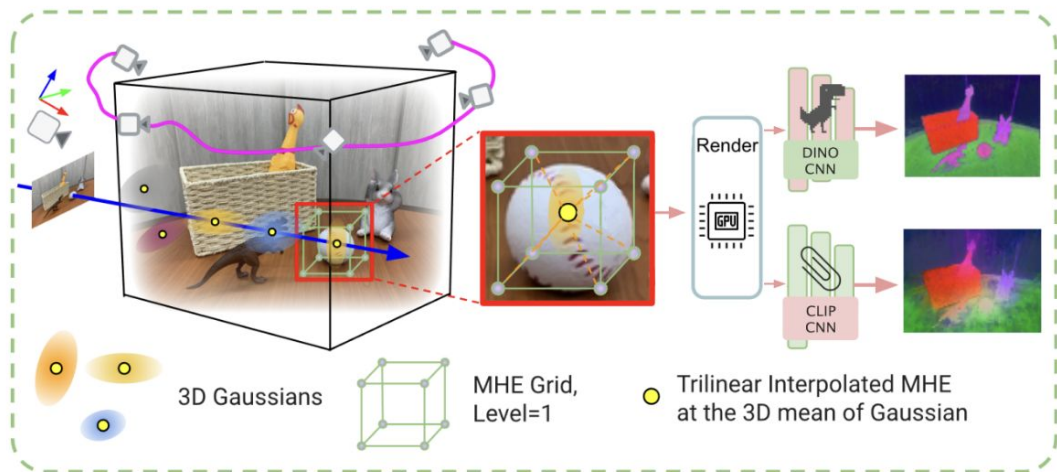


Supervision from Foundation Models

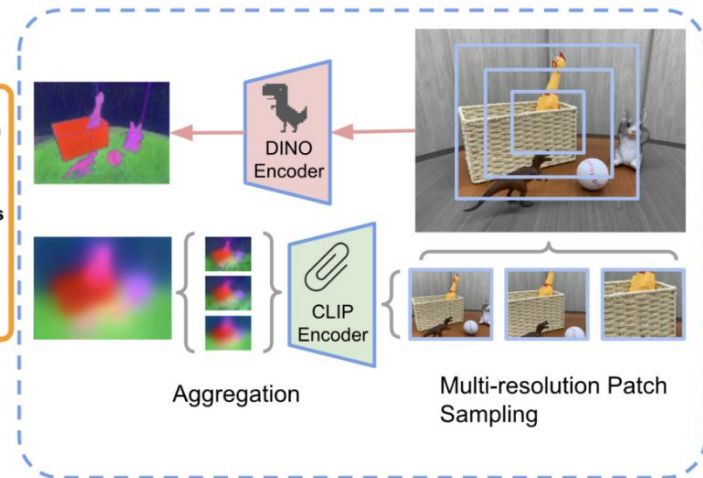


Key Contribution

Feature Rendering with Gaussian Splatting



Supervision from Foundation Models



Render from lower-dimensional MHE space, reduce parameters using CNNs.

Results on Object Detection

Scene	LERF [5]	FMGS	Ours
bouquet	83.3%	91.7%	100.0%
figurines	87.2%	79.5 %	79.5%
ramen	62.5%	80.0 %	82.5%
teatime	96.9%	87.5 %	90.6%
kitchen	85.2%	DNF	72.2%
Average Acc.	83.0%	84.68%	84.96%

(**DNF**: Did Not Finish/OOM)

	bouquet	fig	ramen	tea	waldo
FMGS	116	86.1	64	76	DNF
Ours	52	54	48	60	56

Table 2. Comparison of **Runtime** usage (min). Lower is better.

	bouquet	fig	ramen	tea	waldo
FMGS	20.685	20.165	15.687	23.473	>24
Ours	15.583	16.079	12.275	16.519	16.993

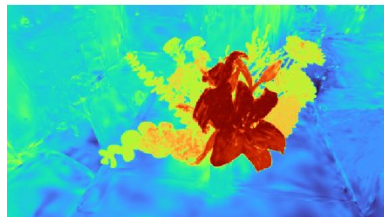
Table 3. Comparison of **VRAM** usage (GB). Lower is better.

FMGS

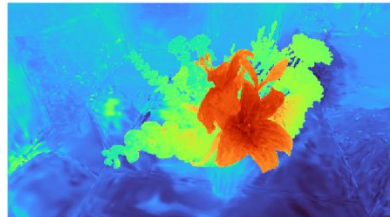
Ours

FMGS

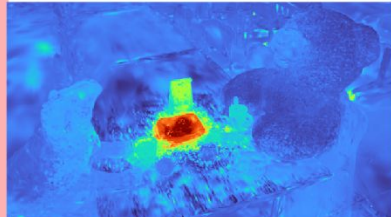
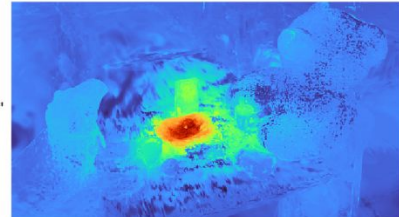
Ours



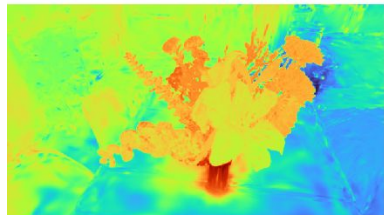
"lily"



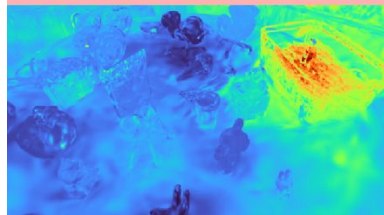
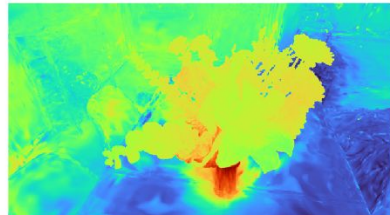
(bouquet)

"cookie"
bag

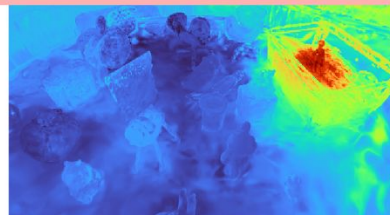
(teatime)



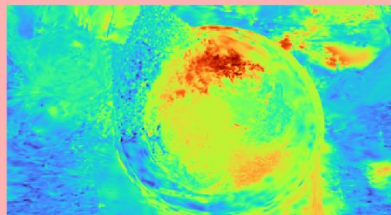
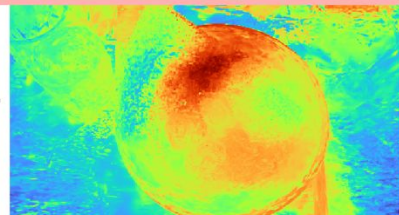
"vase"



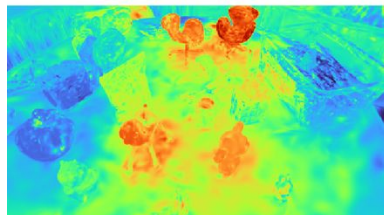
"waldo"



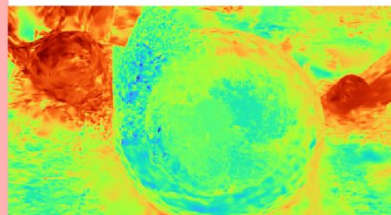
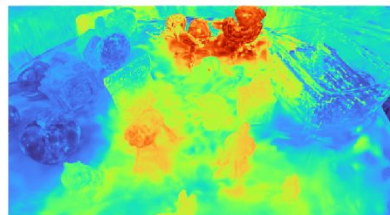
(figurines)

"green"
onion

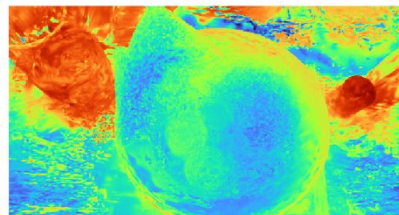
(ramen)



"chairs"



"sake"



FMGS

Ours



"waldo"