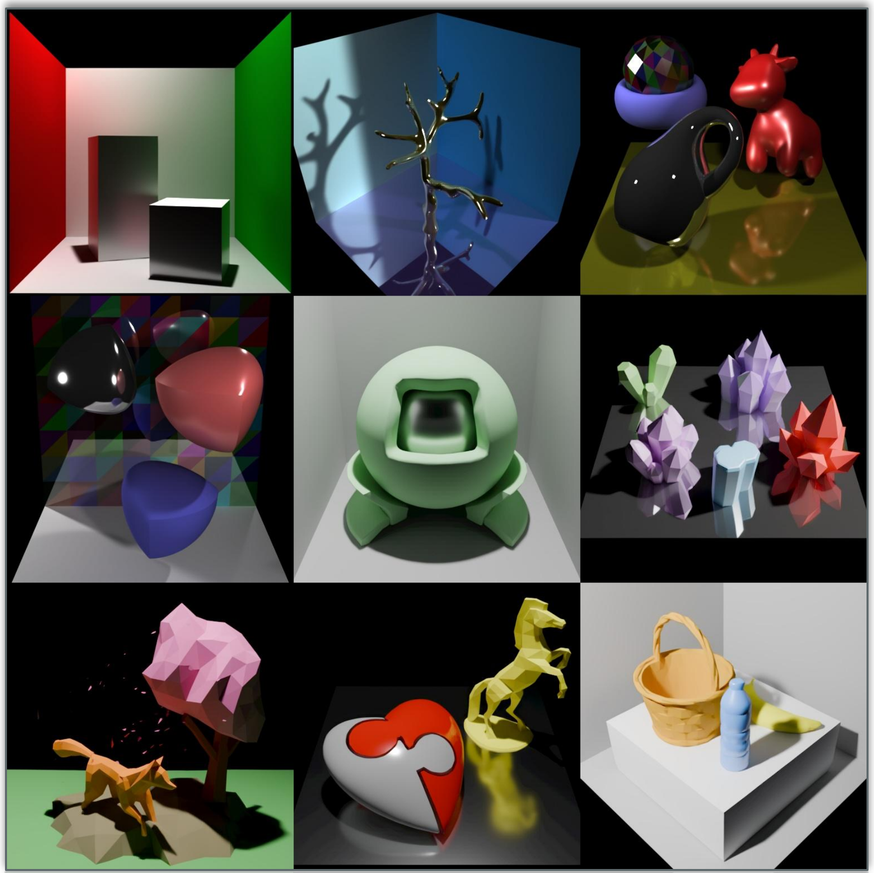
RenderFormer

Mid-term Project Presentation

10th Oct, 2025

Team 4
Kyaw Ye Thu, Janghyun



https://microsoft.github.io/renderformer/

CS580: Computer Graphics

Outline

- Related Work & their Problems
- Overview of RenderFormer
- Limitations of RenderFormer
- Our Plan to Overcome One Limitation

Related Work & Their Limitations

Solving the Rendering Equation

$$L_r(x \to \Theta) = \int_{\Psi} L(x \leftarrow \Psi) f_r(x, \Psi \to \Theta) \cos\theta_x dw_{\Psi}$$

Monte Carlo Estimation of Path Integral

- Pa
- M
- Path Guiding
- Multiple Importance Sampling
- ...

Recursive

Radiosity Techniques (Finite Elements Methods)

- Pre-computation
- Per-scene Training

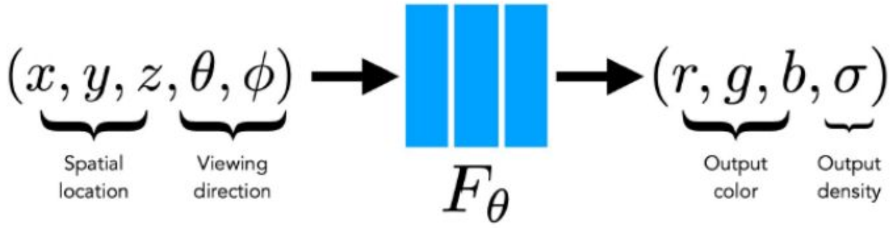
- Precomputed Radiance Transfer (PRT)
- ...

Slide Source: https://saeedhd96.github.io/neural-radiosity/

3D Inductive Bias (Representation Level)

NeRF

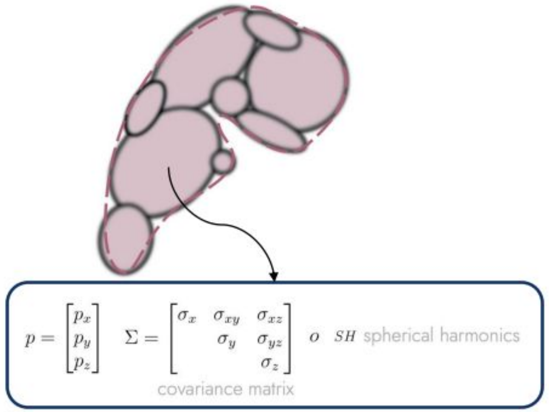
Continuous Volumetric Field: Assumes the scene can be represented as a continuous 5D function.



$(\underbrace{x, y, z}_{\text{Spatial location}}, \underbrace{\theta, \phi}_{\text{Viewing direction}}) \rightarrow F_\theta \rightarrow (\underbrace{r, g, b}_{\text{Output color}}, \underbrace{\sigma}_{\text{Output density}})$

3DGS

Gaussian Primitives: Scenes are composed of fuzzy blobs (Gaussians) that can be combined



$p = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$
$\Sigma = \begin{bmatrix} \sigma_x & \sigma_{xy} & \sigma_{xz} \\ & \sigma_y & \sigma_{yz} \\ & & \sigma_z \end{bmatrix}$
covariance matrix
o
SH spherical harmonics

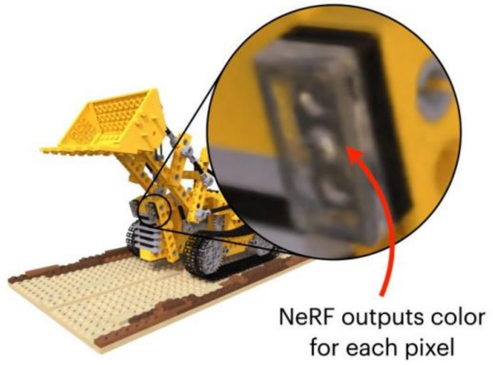
3D Inductive Bias

Why 3D Inductive Biases Are Good

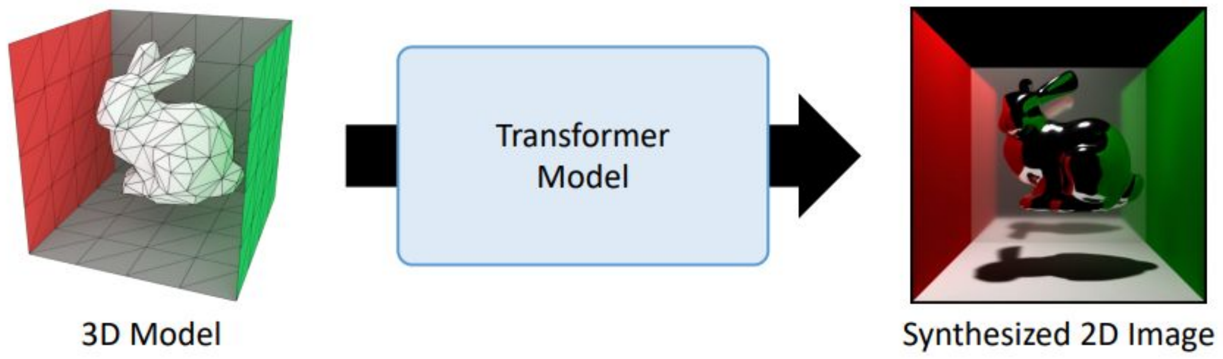
1. **Efficiency**: Reduce the learning space.
2. **Sample efficiency**: With strong priors, models can generalize from fewer examples

Why 3D Inductive Biases Are Bad

1. **Limited flexibility**: Real-world scenarios may not match the built-in assumptions.
2. **Poor generalization**: A model designed for one type of object/geometry might fail on other types of objects/geometries.



RenderFormer: A Fully Transformer Rendering Pipeline



✅ End-to-end with Raw 3D Input

✅ Minimal Prior Constraints

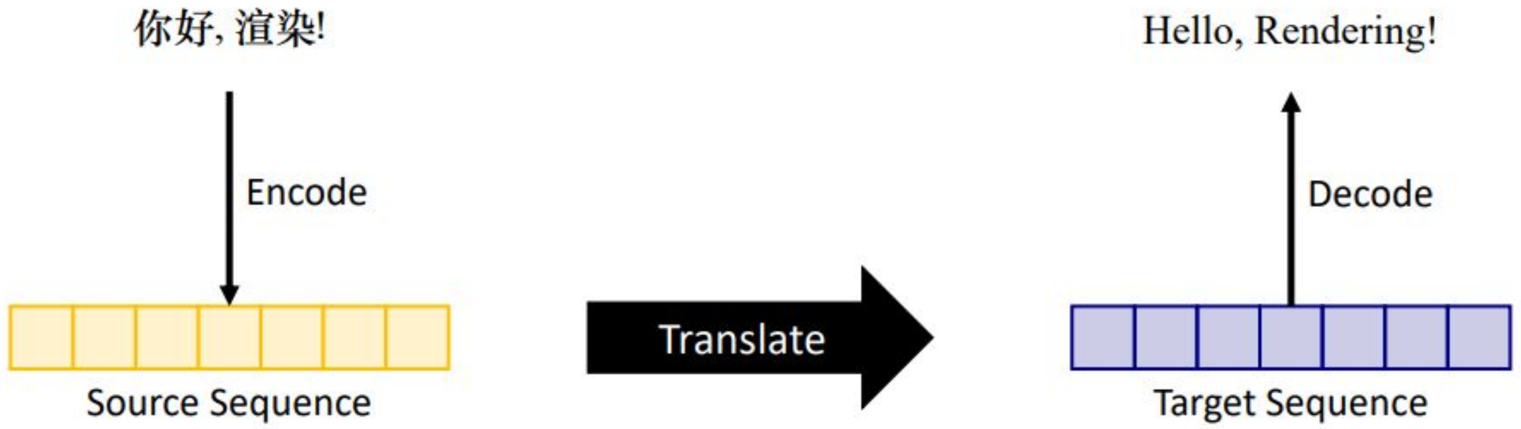
✅ No Per-scene Training

✅ Full Global Illumination Effects

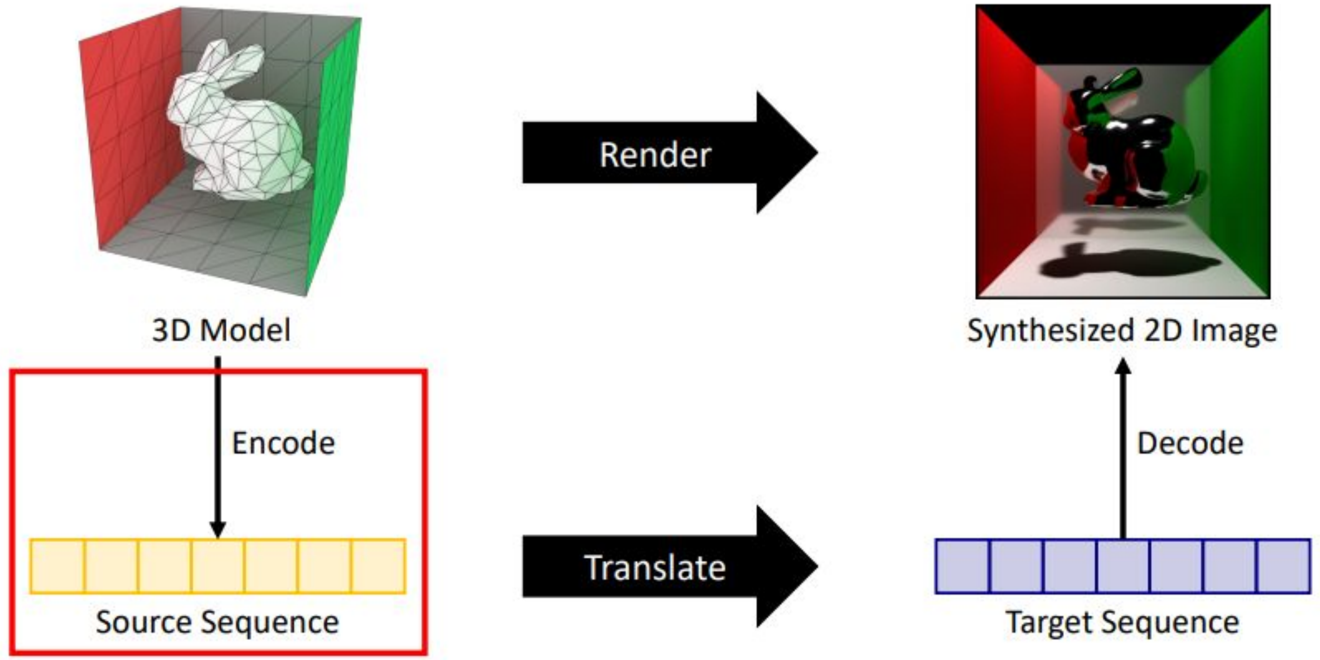
Slide Source: https://microsoft.github.io/renderformer/

Overview of RenderFormer

Sequence-to-Sequence Machine Translation

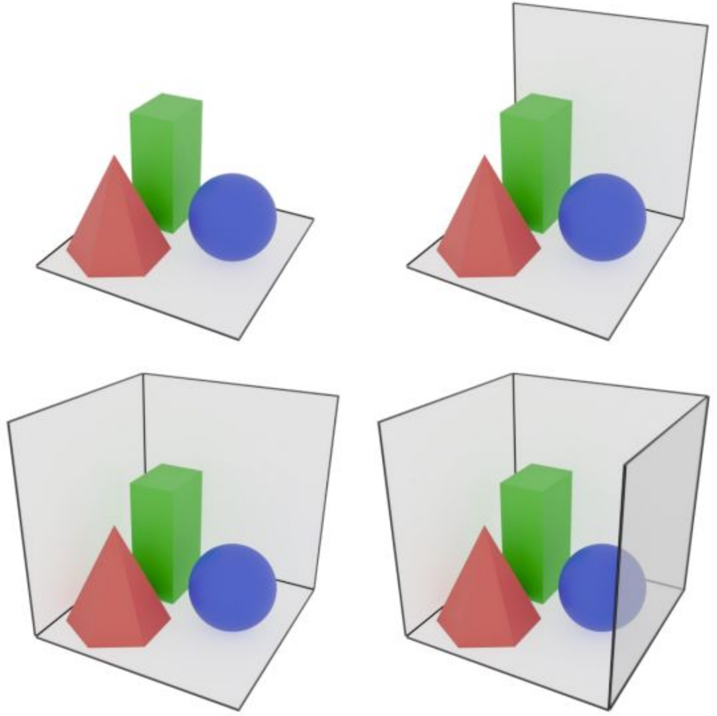


Idea: 3D Rendering = Translating 3D to 2D



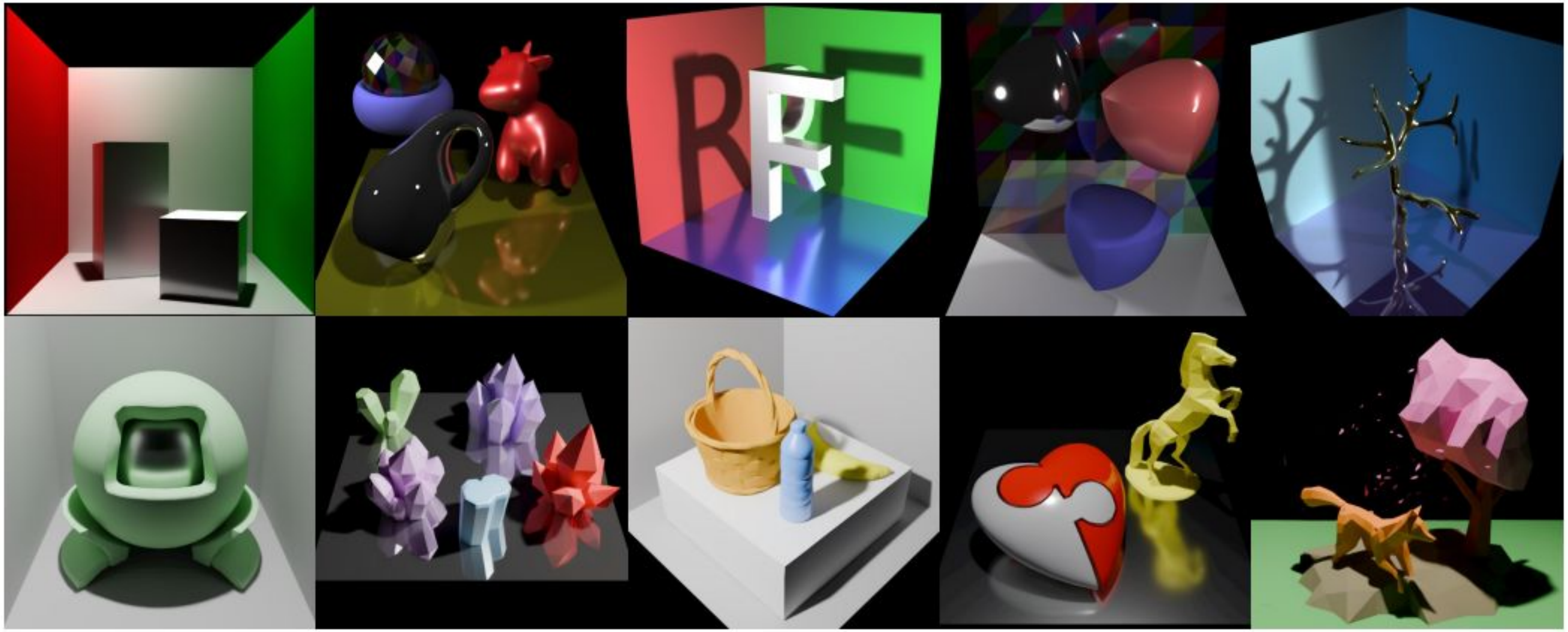
Slide Source: https://microsoft.github.io/renderformer/

Training Data – Template-based Scene Generation



4,096 Max #Triangles	**0.01-1.0** Roughness Range
8 Max #Lights	**2.1-2.7** Light Distance
30-60 Camera FOV	**1.5-2.0** Camera Distance

Visual Results – Static Scene Rendering

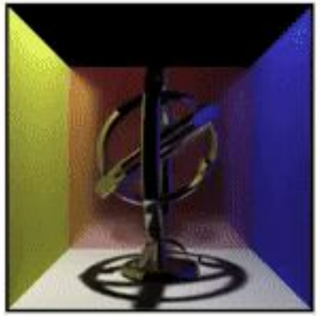


Slide Source: https://microsoft.github.io/renderformer/

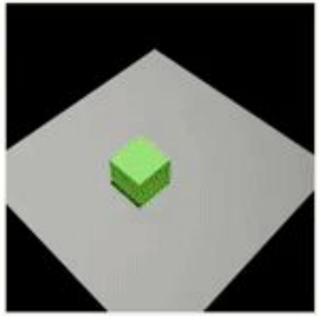
Visual Results – Dynamic Scene Rendering

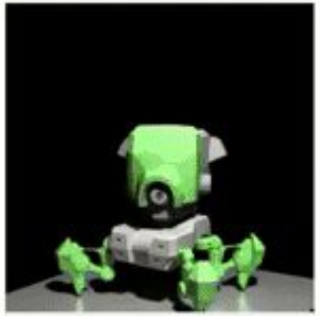
Cascade Cube Animation
Tycho Magnetic Anomaly

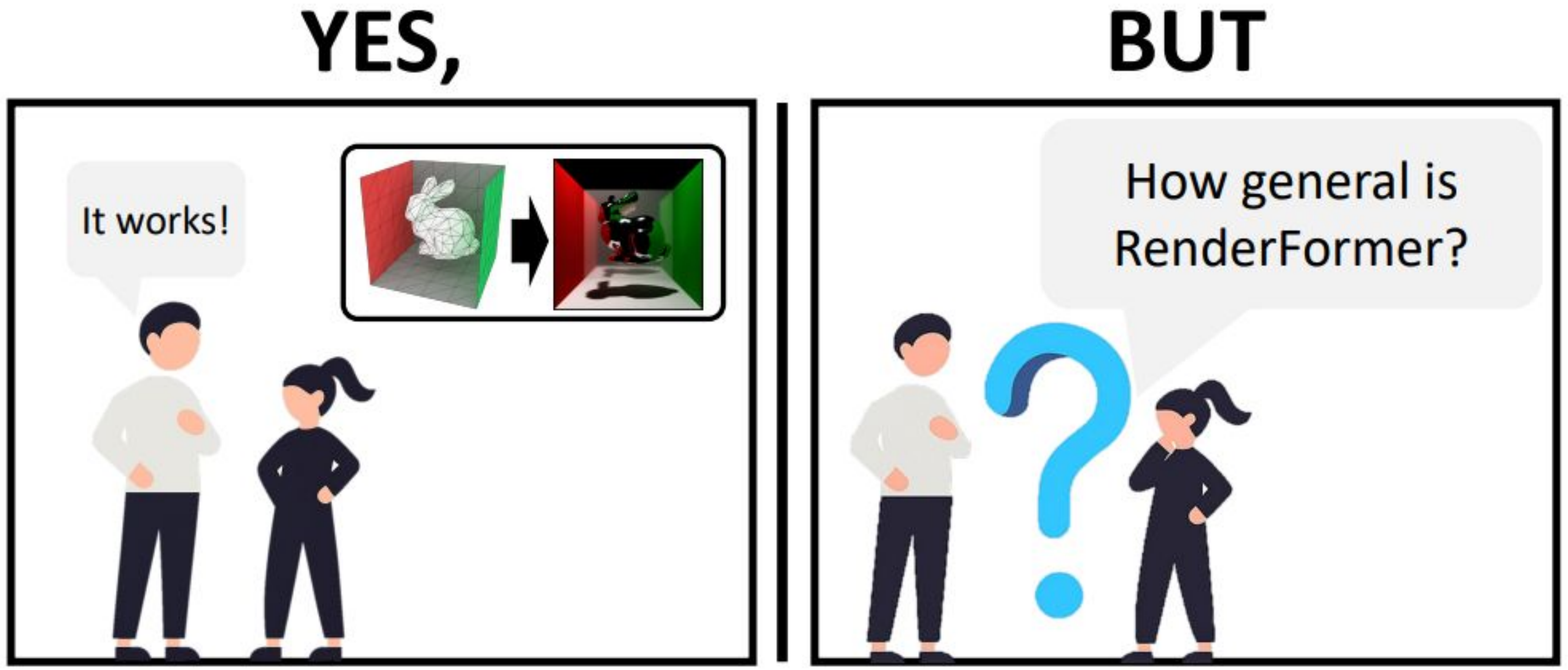
Animated Crab
Bohdan Lvov


Gyroscope Motion
reddification

Animated Character
mortaleiros

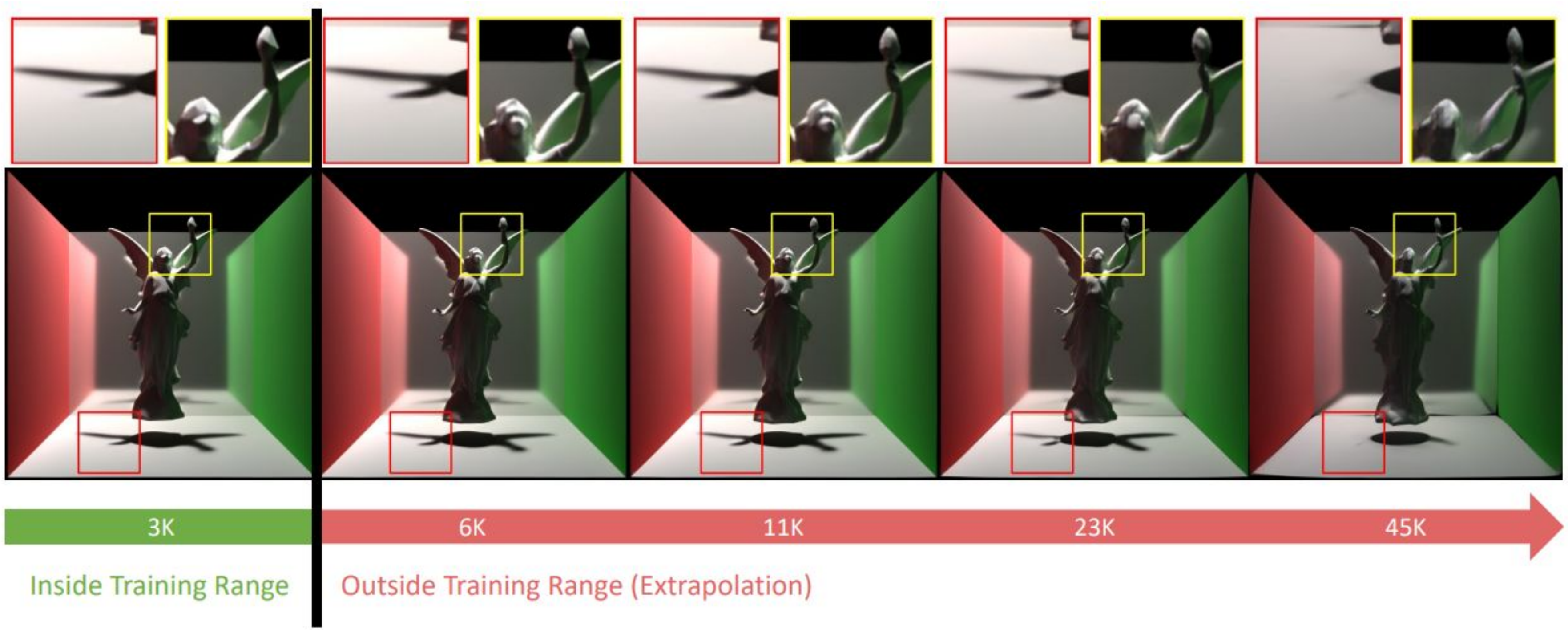

Marching Cubes Animation
Tycho Magnetic Anomaly


Robot Animation
Goshadouken


YES,
BUT
It works!
How general is RenderFormer?

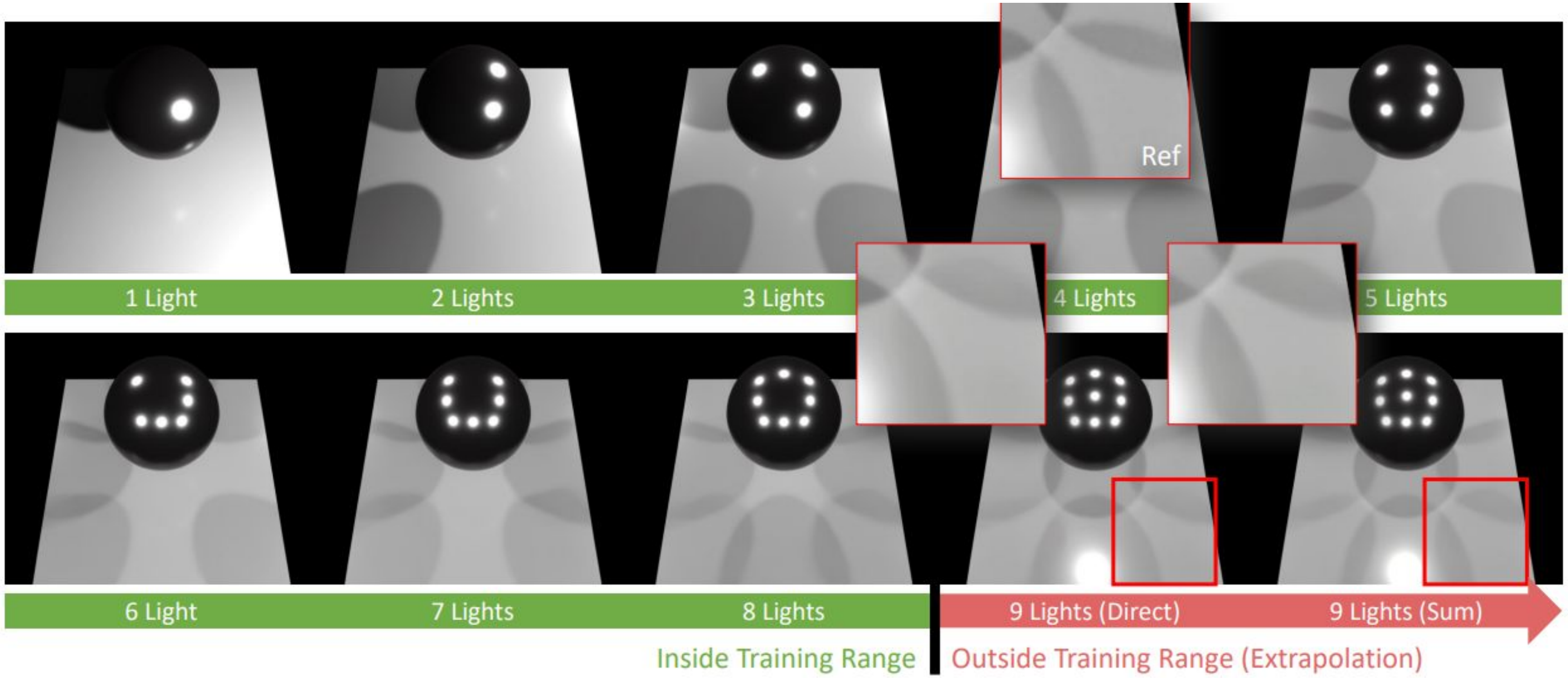
Limitations of RenderFormer

1. Generalization (# Triangles)

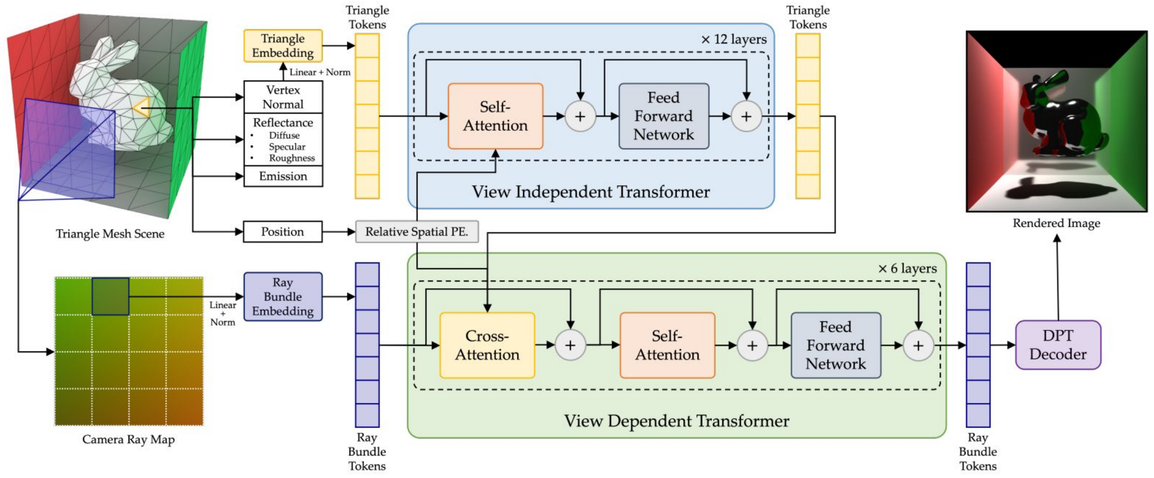


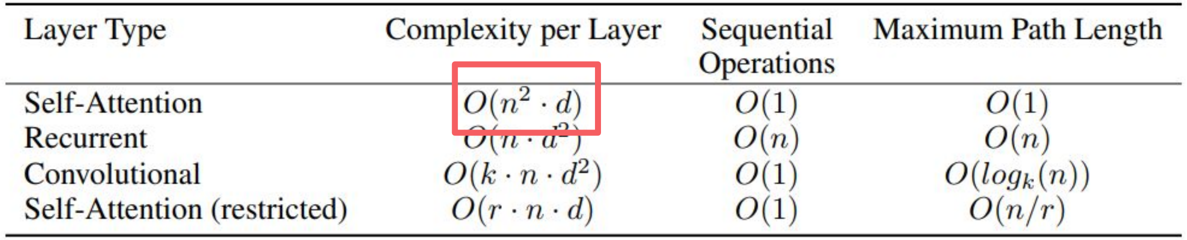
Slide Source: https://microsoft.github.io/renderformer/

1. Generalization (# light resources)



2. Time complexity of Attention





Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$\boxed{O(n^2 \cdot d)}$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

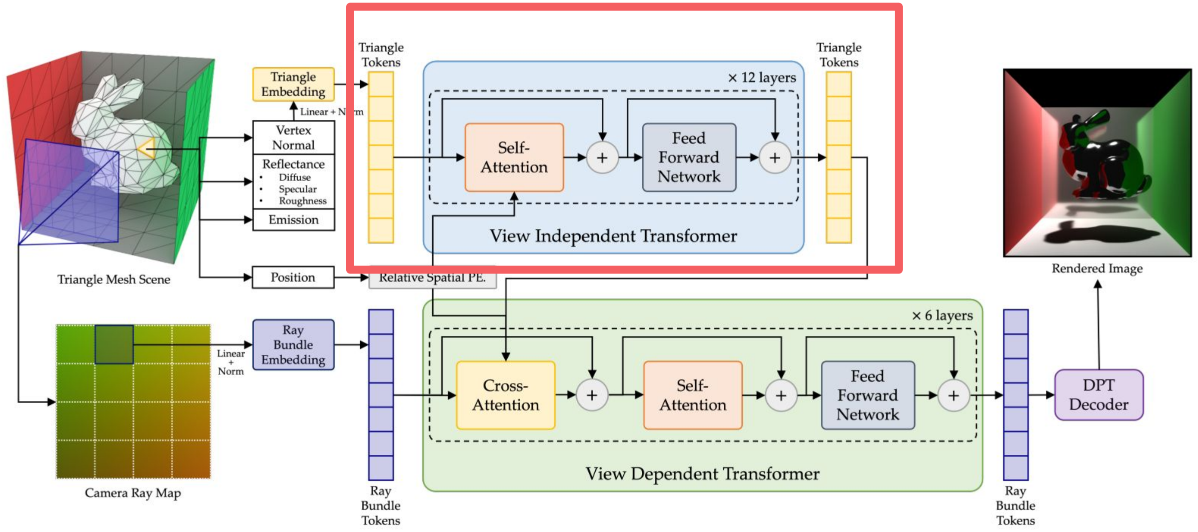
n = sequence length

d = dimension

k = kernel size of convolutions

A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, vol. 30

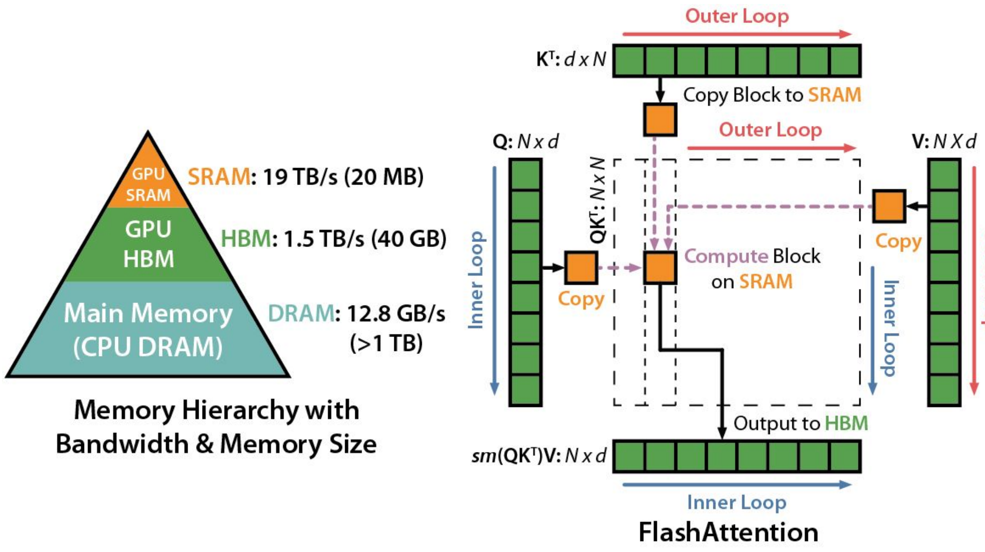
Time complexity of Renderformer



- The view-dependent stage's time complexity is roughly by O(#Bundle^2)
- The view-independent stage's time complexity is roughly by O(#tri^2)
- Because of this, limit the total number of triangles in scenes to 4,096

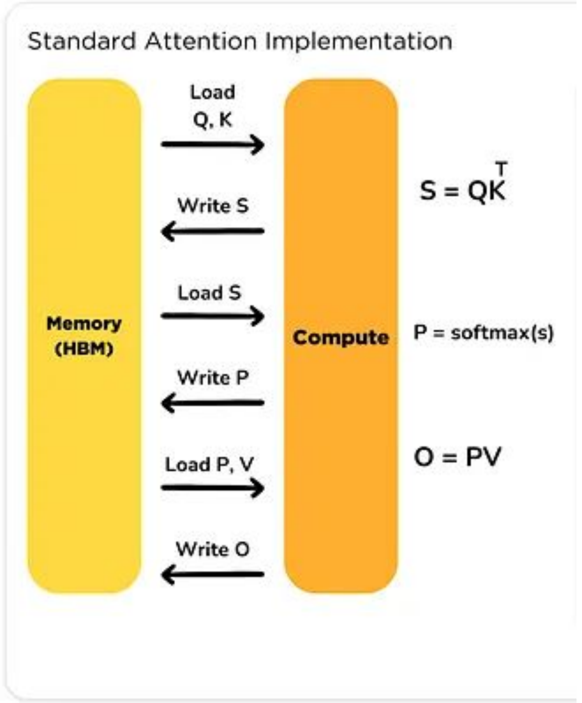
C. Zeng, Y. Dong, P. Peers, H. Wu, and X. Tong, "RenderFormer: Transformer-based Neural Rendering of Triangle Meshes with Global Illumination," in *Proceedings of the ACM SIGGRAPH 2025 Conference*

Actually, they used Flash Attention-2



- They used FA-2 which can help to reduce calculation time and memory usage.
- Flash attention is attention algorithm that tiles softmax-attention to on-chip SRAM.

How Flash Attention work?

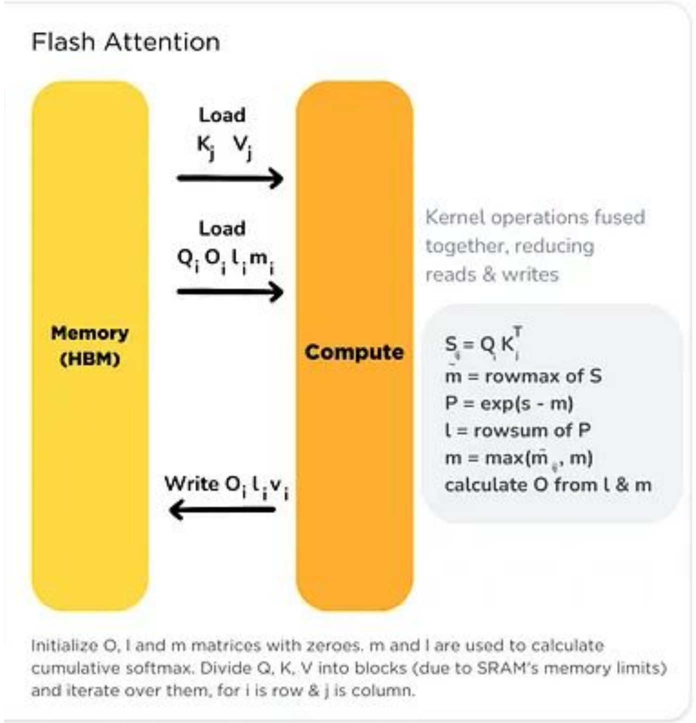


Algorithm 0 Standard Attention Implementation

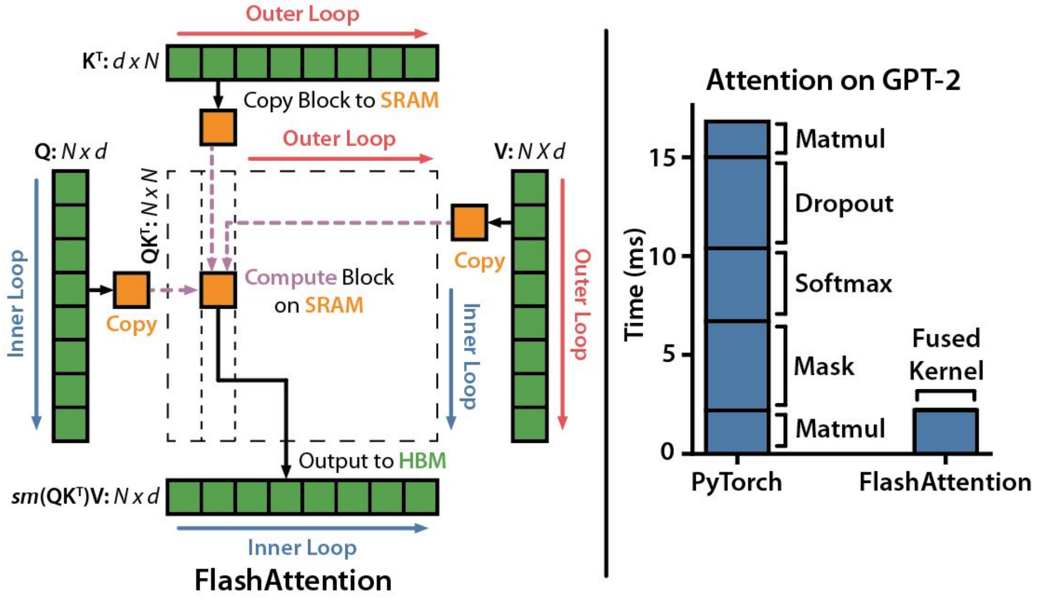
```
Require: Matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N \times d}$ in HBM.
  1: Load $\mathbf{Q}, \mathbf{K}$ by blocks from HBM, compute $\mathbf{S} = \mathbf{Q}\mathbf{K}^\top$, write $\mathbf{S}$ to HBM.
  2: Read $\mathbf{S}$ from HBM, compute $\mathbf{P} = \mathrm{softmax}(\mathbf{S})$, write $\mathbf{P}$ to HBM.
  3: Load $\mathbf{P}$ and $\mathbf{V}$ by blocks from HBM, compute $\mathbf{O} = \mathbf{P}\mathbf{V}$, write $\mathbf{O}$ to HBM.
  4: Return $\mathbf{O}$.
```

T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FLASHATTENTION: Fast and Memory-Efficient Exact Attention with IO-Awareness," in *Proceedings of the 36th International Conference on Neural Information Processing Systems* (NeurIPS), 2022

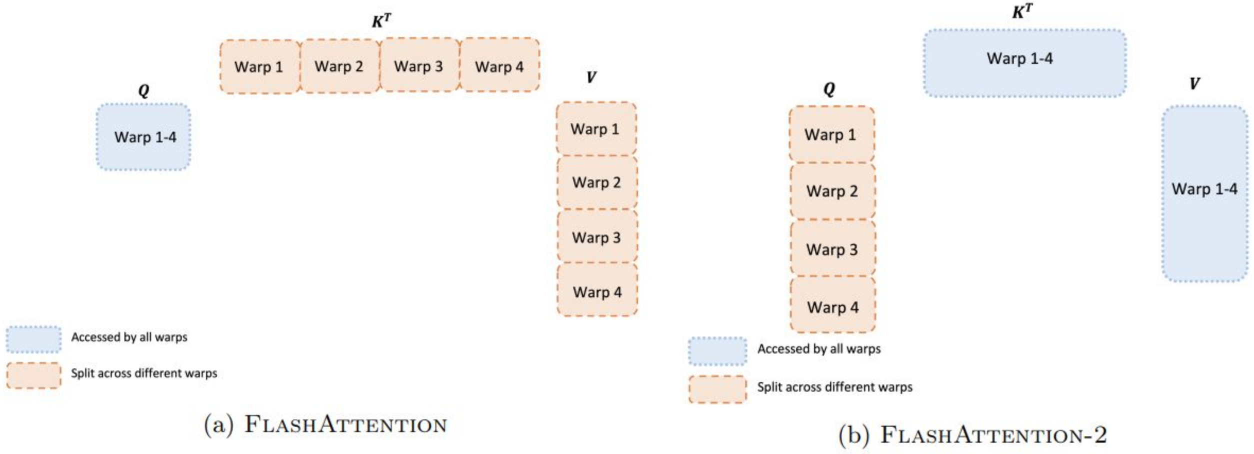
How Flash Attention work?



Initialize O, l and m matrices with zeroes. m and l are used to calculate cumulative softmax. Divide Q, K, V into blocks (due to SRAM's memory limits) and iterate over them, for i is row & j is column.

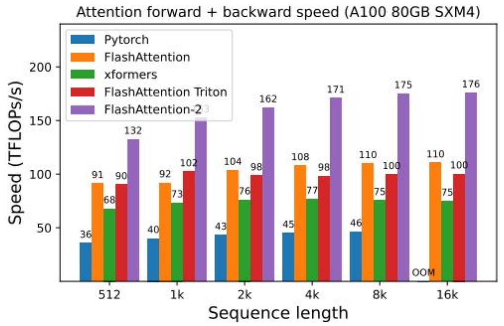


Improvement of Flash Attention-2

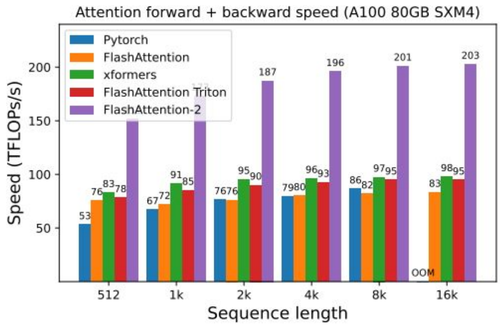


- Flash attention-2 make more fast speed by modifying the process to split Q and share K and V, eliminating the need for continuous synchronization of intermediate calculation results, instead of splitting K and V.

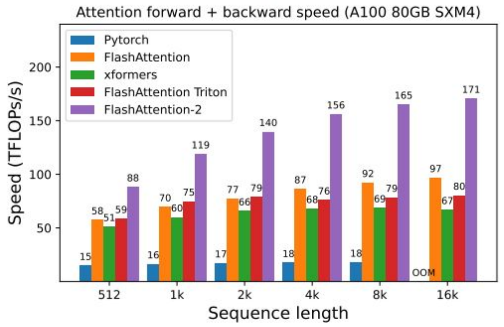
Result of Flash Attention-2



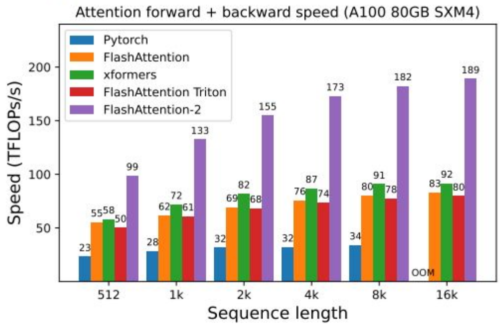
(a) Without causal mask, head dimension 64



(b) Without causal mask, head dimension 128



(c) With causal mask, head dimension 64

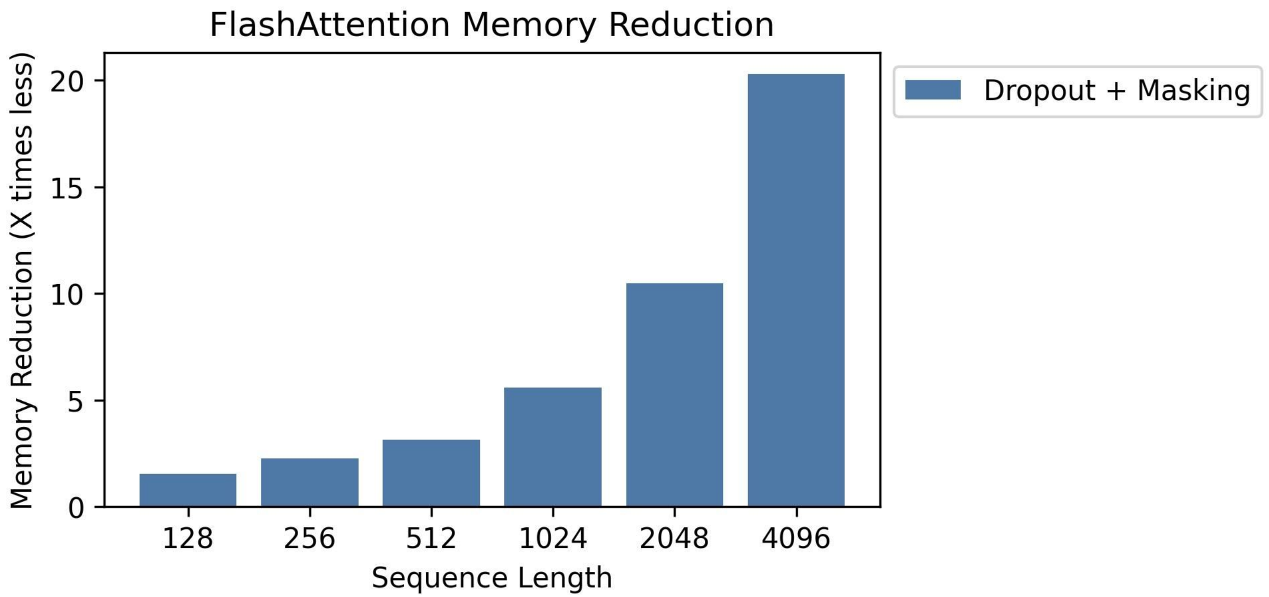


(d) With causal mask, head dimension 128

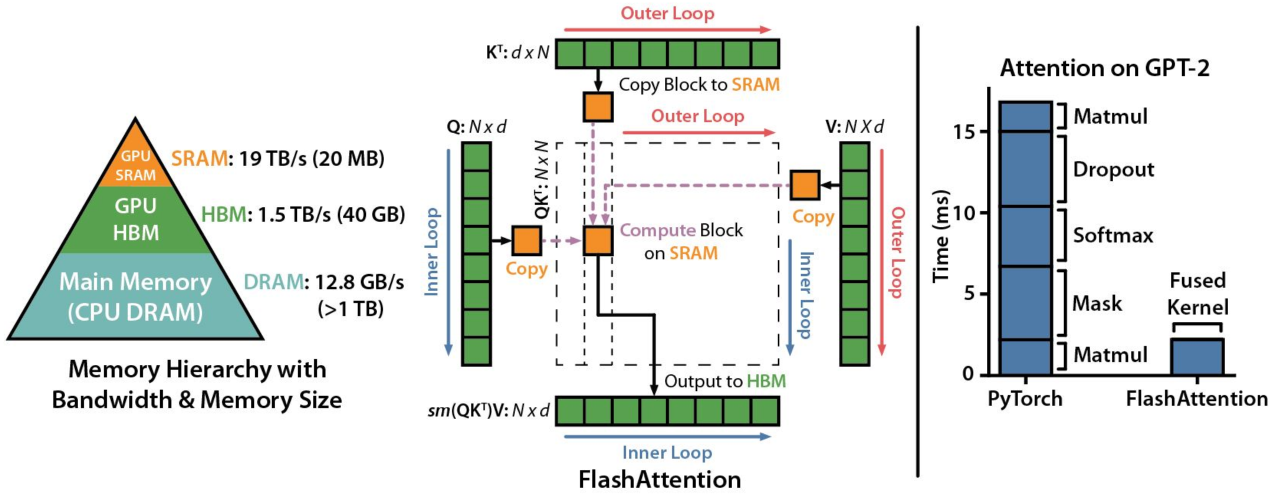
Figure 4: Attention forward + backward speed on A100 GPU

T. Dao, "FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning," in *Proc. Int. Conf. Learn. Representations* (ICLR), 2024

Result of Flash Attention-2



Limitation of Flash Attention-2



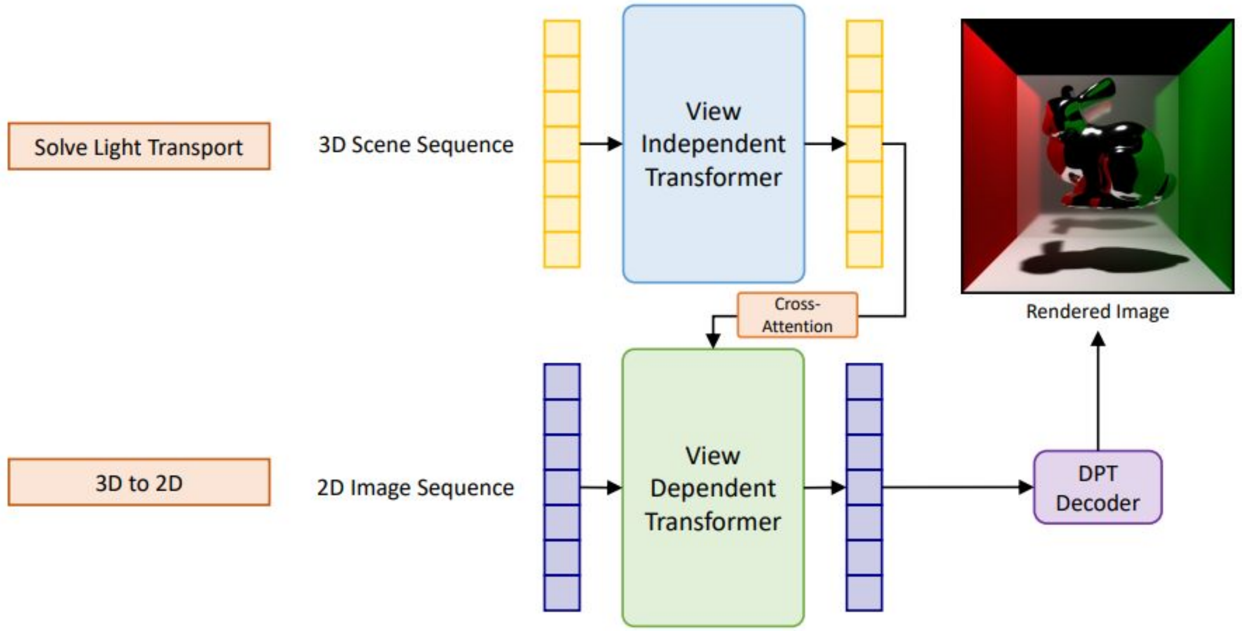
- Although they used FA-2 which can help to calculation time, its asymptotic time complexity is same.
- Thus, we need other approach for resolve this issue.

T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FLASHATTENTION: Fast and Memory-Efficient Exact Attention with IO-Awareness," in *Proceedings of the 36th International Conference on Neural Information Processing Systems* (NeurIPS), 2022

Our Plan

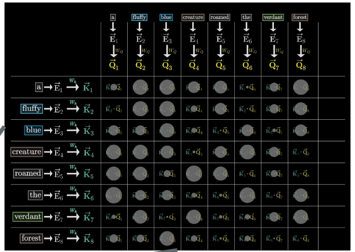
The Use of Original Transform Architecture

Both stages closely follow the original transformer architecture [Vaswani et al. 2017]



A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, vol. 30

How Attention can be Computed Faster



L x L

L x d   d x L   L x d

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{Q \cdot K^{\intercal}}{\sqrt{d_k}}\right) \cdot V$$

L - length of the sequence
d - # dimension of the sequence

How Attention can be Computed Faster

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{\overbrace{\overset{\text{L x d}}{Q} \cdot \overset{\text{d x L}}{K^{\intercal}}}^{\text{L x L}}}{\sqrt{d_k}}\right) \cdot \overset{\text{L x d}}{V}$$

- The view-independent stage scales roughly by O (#$tris$^2)
- view-dependent layers scales by O (#$bundles$^2 + #$bundles$ × #$tris$).

How Attention can be Computed Faster

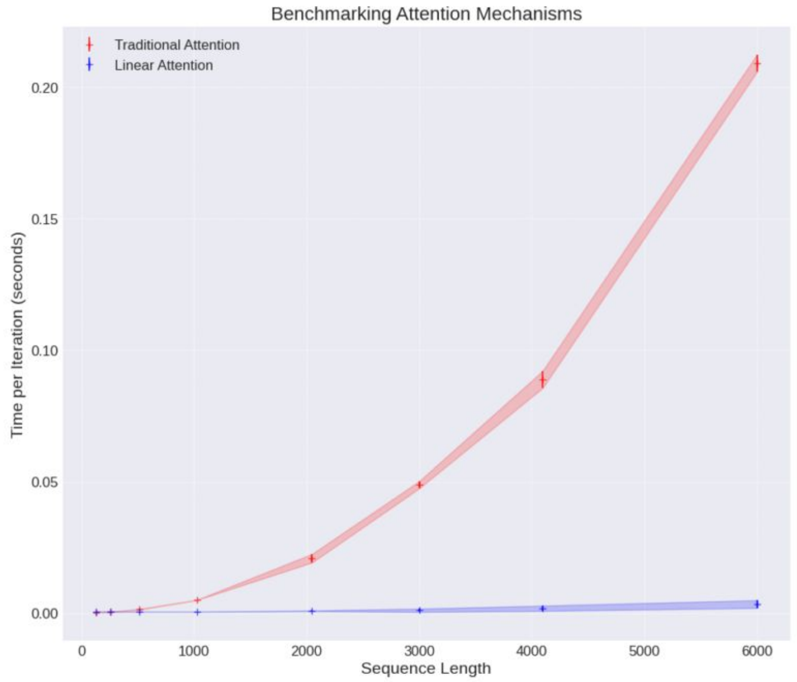
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{\overbrace{\overset{L \times d}{Q} \cdot \overset{d \times L}{K^\intercal}}^{L \times L}}{\sqrt{d_k}}\right) \cdot \overset{L \times d}{V}$$

$$\underbrace{\underset{d \times L}{} \qquad \underset{L \times d}{}}_{d \times d}$$

d = 768 (dimension of input sequence)
L = 4,096 (# triangles)

As long as d << L, significantly faster

Potential Speedup from Linear Attention



Time per Iteration (seconds)

Sequence Length

1D sequences or 2D images

?

3D

Figure Source: https://towardsdatascience.com/linear-attention-is-all-you-need-5fa9c845c1b5/

Our Plan

- Rewrite the **Inference** Part with Linear Attention or Performer
- Compare the Inference Speed and Rendering Result with the original
- Try **training** with Linear Attention or Performer
- Compare the Training Speed and Rendering Result with the original

Q&A

Quiz Time