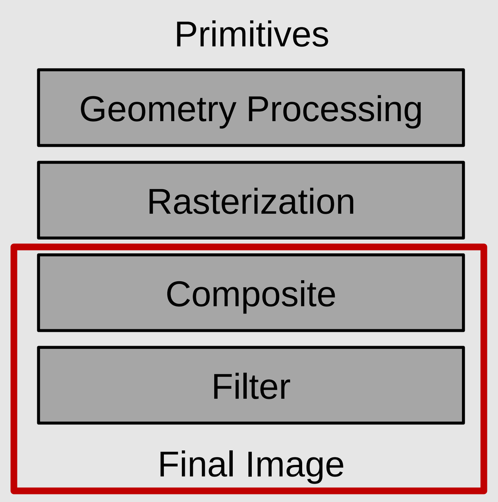
Fragment-Parallel Composite and Filter

Anjul Patney, Stanley Tzeng, John D. Owens
University of California, Davis

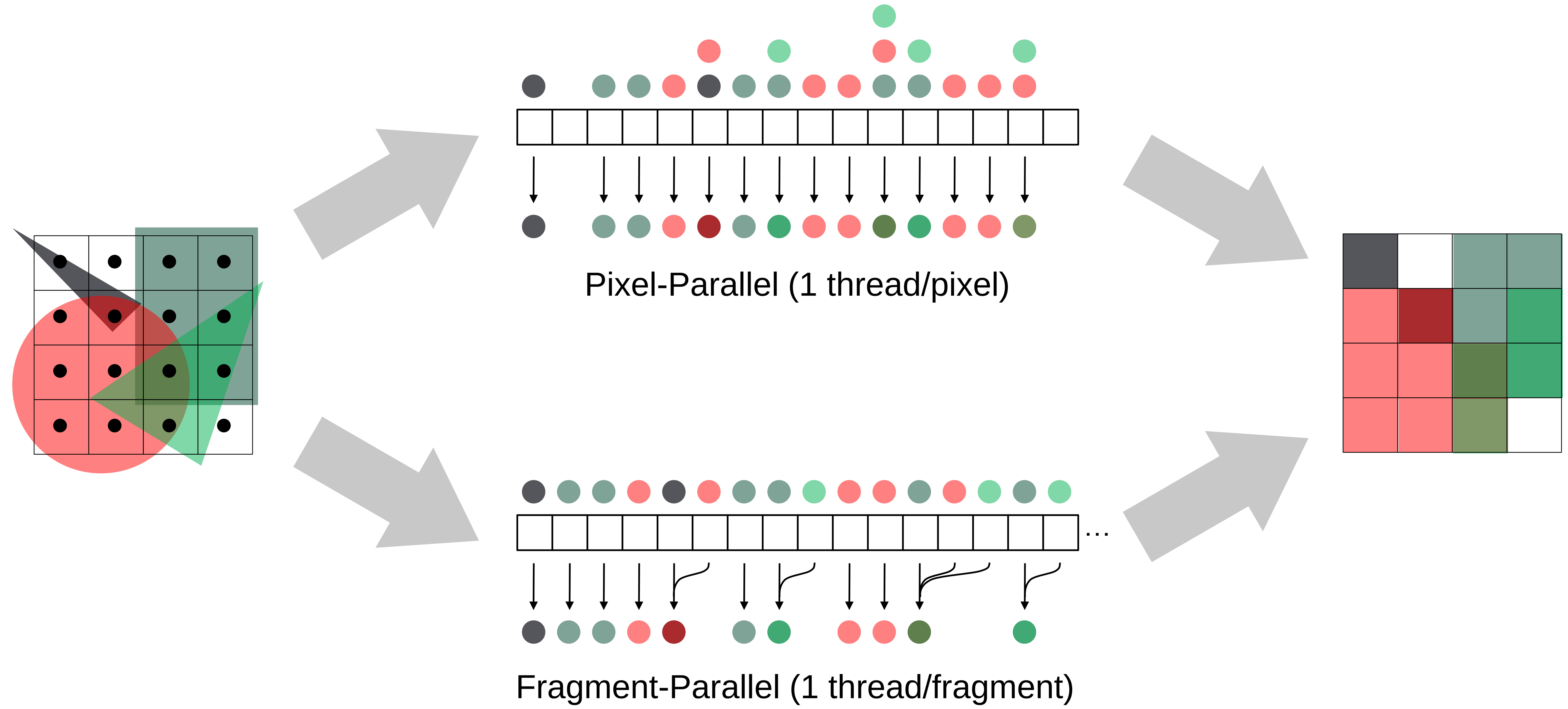
Introduction

We study the parallelization of composite/filter on a modern GPU architecture, and propose a scheme that parallelizes the operation not just across pixel or subpixel locations, but over all available fragments. We demonstrate the applicability of this technique for several applications, and compare the obtained performance of the proposed fragment-parallel composite (FPC) against a conventional pixel-parallel composite (PPC) scheme, i.e. one that parallelizes only across pixel/subpixel locations. Using a synthetic benchmark, we also study the variation in the relative performance of FPC and PPC across the continuum from scenes with a few depth layers to those with a large number of depth layers. Our implementation allows an arbitrary number of semi-transparent layers for each subpixel location. Our contributions are as follows:



A feed-forward pipeline showing the domain of this work

- We identify a novel strategy to parallelize composite and filter across individual depth fragments, and describe an implementation on a modern GPU. We demonstrate that this technique is applicable to several areas with four examples: a particle system, a volume renderer, a polygon rasterizer and an interactive Reyes pipeline.
- We compare the performance of the proposed fragment-parallel strategy against a pixel-parallel version, and study the behavior of both with varying depth complexity. We show that for the same number of fragments, a pixel-parallel technique loses performance with high as well as variable depth complexity, while the performance of a fragment-parallel version is much more consistent.

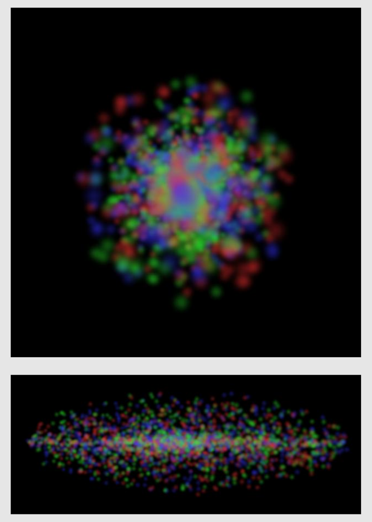


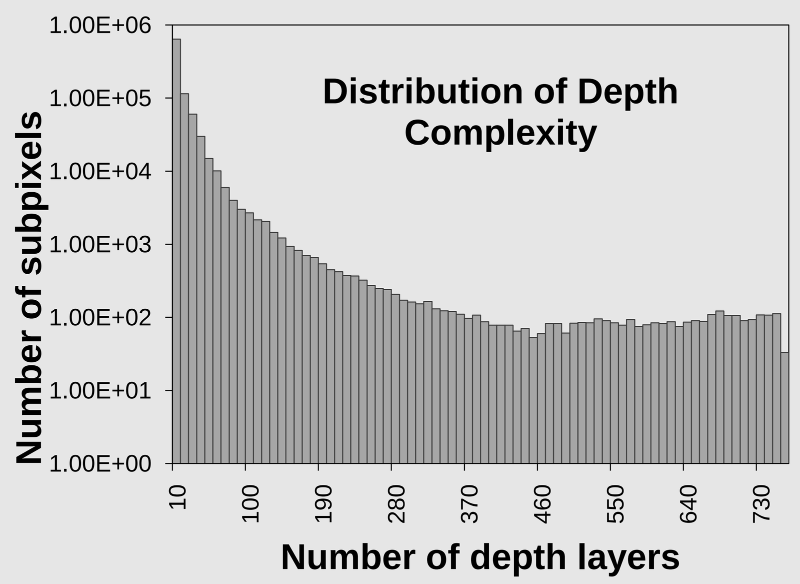
Motivation

Most graphics applications exhibit low depth-complexity
- Number of pixels/subpixels offers sufficient parallelism
- A pixel-parallel approach is preferred

We are interested in graphics applications
- With high depth complexity
- With high variation in depth complexity
- Where a depth-sequential approach is inefficient

Further:
- Future GPUs will have higher degrees of parallelism
- High depth-complexity can limit tile sizes and reduce pixel-parallelism





Basic Idea

- Input: Unordered list of fragments
- Output: Pixel colors
- Revisit the composite equation:

$$C_s = \alpha_1 C_1 + (1-\alpha_1)\{\alpha_2 C_2 + (1-\alpha_2)(\ldots(\alpha_N + (1-\alpha_N)C_B)\ldots\}$$

$$C_s = 1.\alpha_1.C_1 + (1-\alpha_1).\alpha_2.C_2 + (1-\alpha_1)(1-\alpha_2).\alpha_3.C_3 + \ldots$$

$$+ \underbrace{(1-\alpha_1)(1-\alpha_2)\ldots(1-\alpha_{k-1})}_{\text{Global Contribution } G_k}.\underbrace{\alpha_i.C_k}_{\text{Local Contribution } L_k} + \ldots + (1-\alpha_1)\ldots(1-\alpha_N).C_B$$

$$C_s = G_1.L_1 + G_2.L_2 + G_3.L_3 \ldots G_N.L_N$$

$G_k = (1-\alpha_1).(1-\alpha_2)\ldots(1-\alpha_{k-1})$ is a product scan
$L_k = \alpha_k.C_k$ is trivially parallel

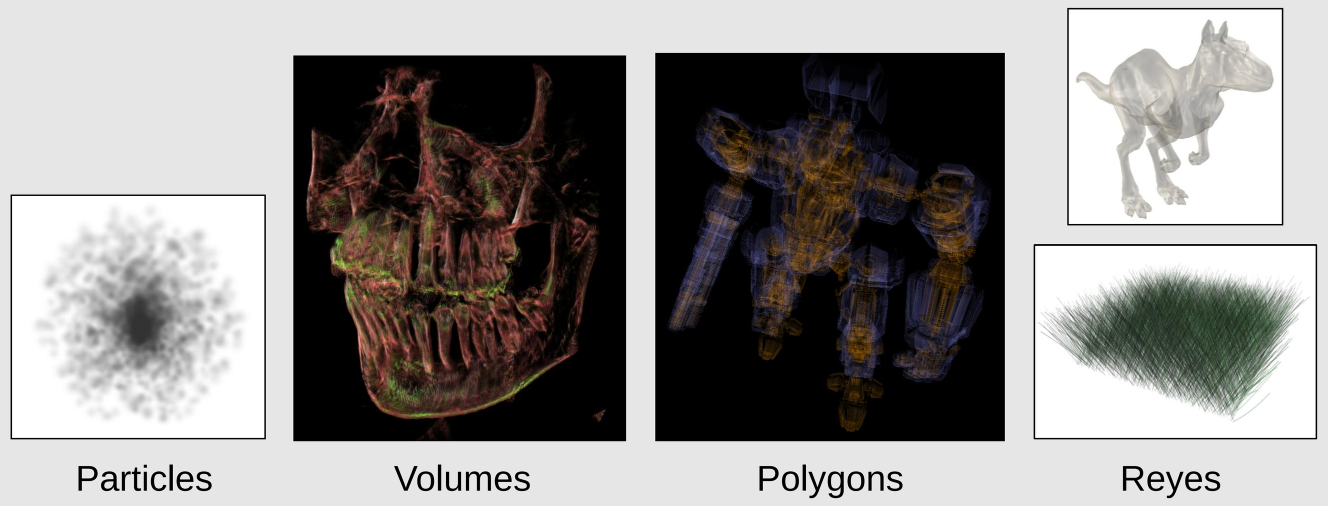
- Filter:

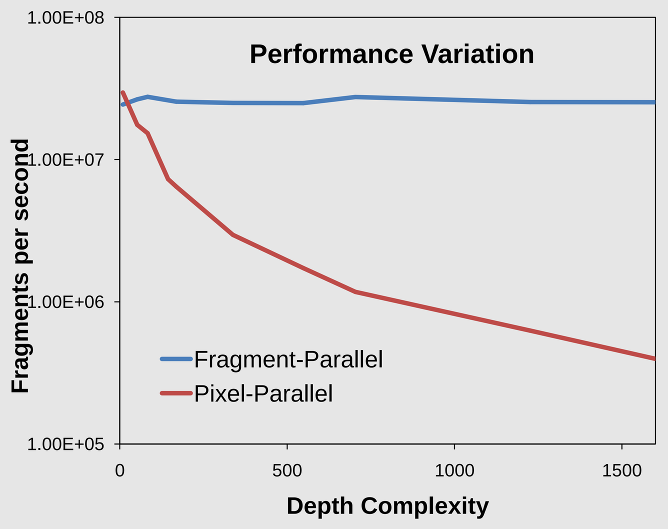
$$C_p = C_{s1}.\kappa_1 + C_{s2}.\kappa_2 + \ldots + C_{sM}.\kappa_M$$

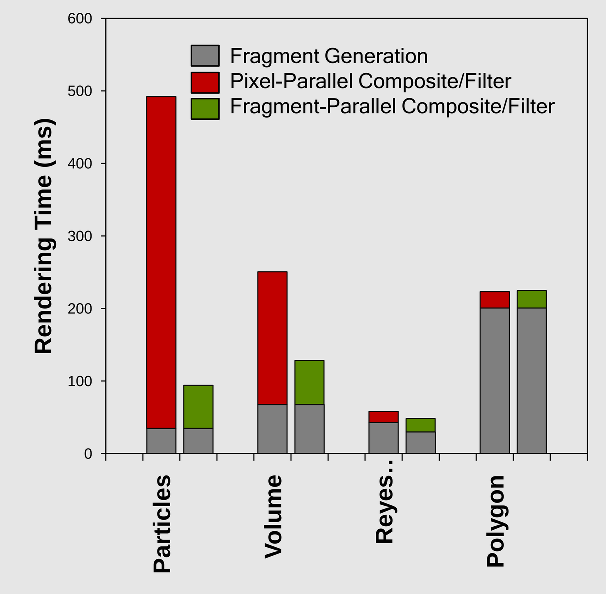
- Filter can be expressed as a parallel reduction

Final Algorithm
1. Two-key sort (Subpixel ID, depth)
2. Segmented Scan (obtain G_k)
3. Premultiplication with weights (L_k, κ_m)
4. Segmented Reduction

Results







Summary

Conclusions

- Parallel formulation of composite equation
  - Maps well to known parallel primitives
  - Can be integrated with filter operation
  - Consistent performance across varying workloads
- FPC is applicable to future programmable rendering pipelines
  - Exploits higher degree of parallelism at the level of fragments
  - Better related to the size of rendering workload

Limitations

- Increased memory traffic
  - Several passes through CUDPP primitives
- Unclear how to optimize for special cases
  - Threshold opacity
  - Threshold depth complexity

Future Work

- Direct3D 11 implementation
  - Will provide direct access to hardware rasterizer
  - Efficient parallel primitives not yet available
- A hybrid PPC-FPC technique

Anjul Patney, Stanley Tzeng, and John D. Owens, "Fragment-Parallel Composite and Filter,"
Computer Graphics Forum, 29(4), 1251-1258, June 2010 (EGSR 2010)