Posters Fast Forward

1. "Massive Networks," Daniel Filonik @UNSW EPICentre, CSIRO Data61
2. "Hybrid Ray-Traced Ambient Occlusion," Morgan McGuire @NVIDIA
3. "A 2D to 3D Video Converter using Optical Flow Information and Least Squares Regression," Hui-Yun Lee @Chang Gung University
4. "Blender based Rendering-as-a-Service Platform for High Performance Computing Clusters," Milan Jaros @Technical University of Ostrava
5. "Energy Consumption Optimization of Rendering in Blender Cycles on x86 Architectures" Lubomir Riha @Technical University of Ostrava

Massive Networks

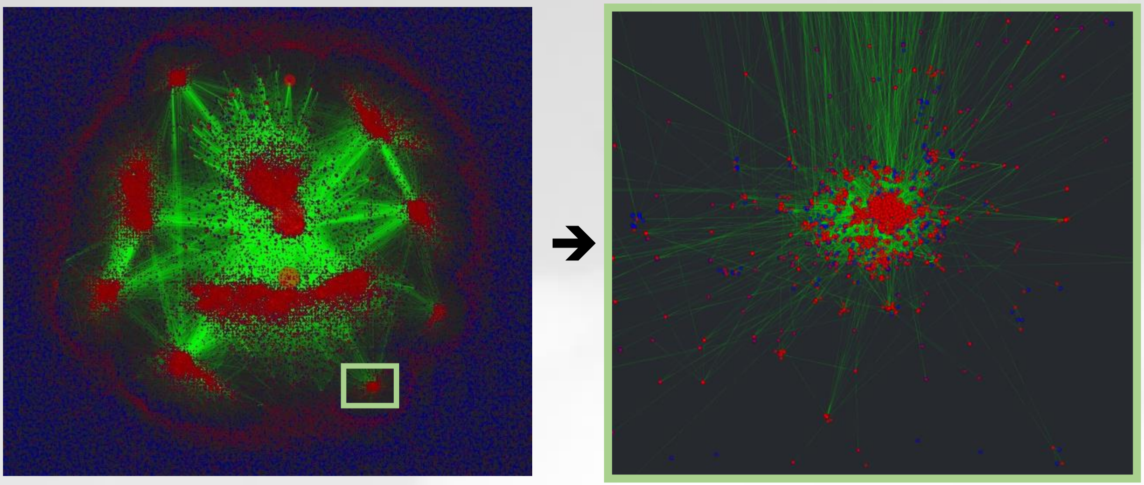
Daniel Filonik, Dominic Branchaud, Robert Lawther, Piotr Szul, Alex Collins, Tomasz Bednarz

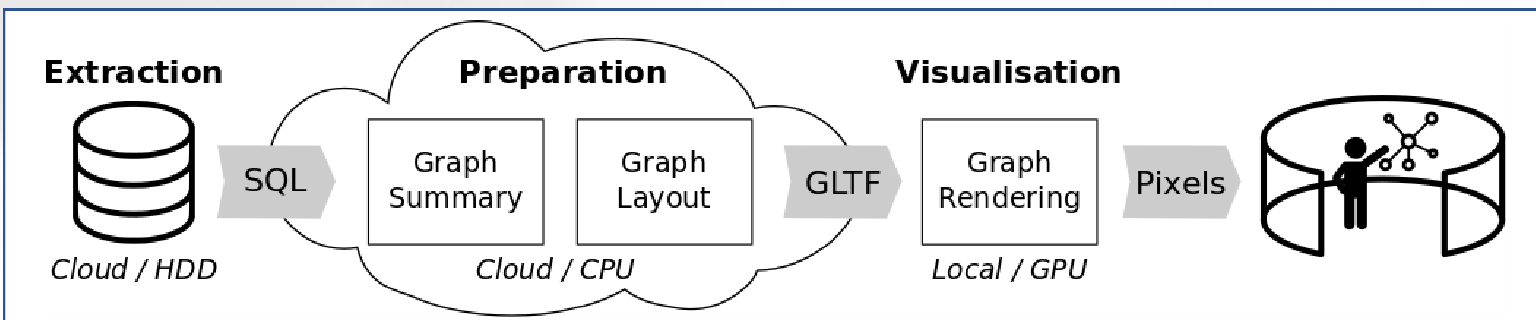
UNSW EPICentre, CSIRO Data61

hpg 2018

Massive Networks

Visualising Large-Scale Graphs in Immersive Environments

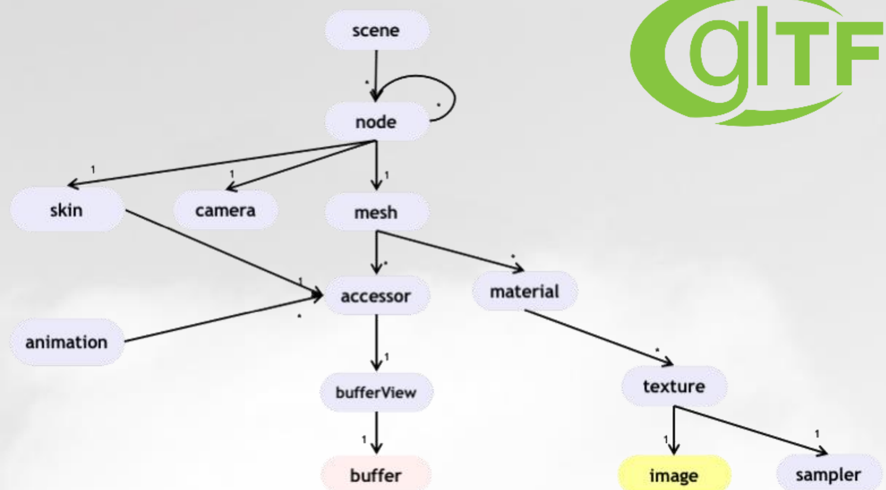


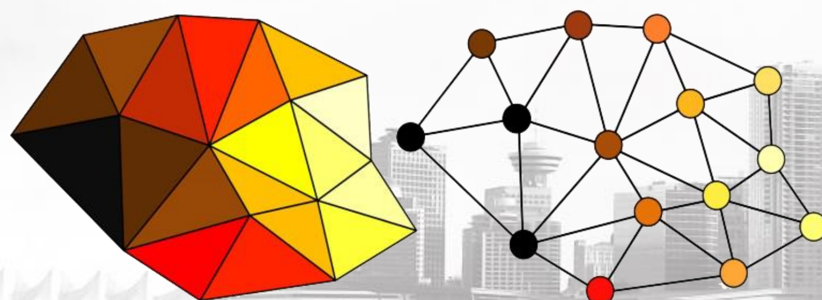


Massive Networks

Visualising Large-Scale Graphs in Immersive Environments

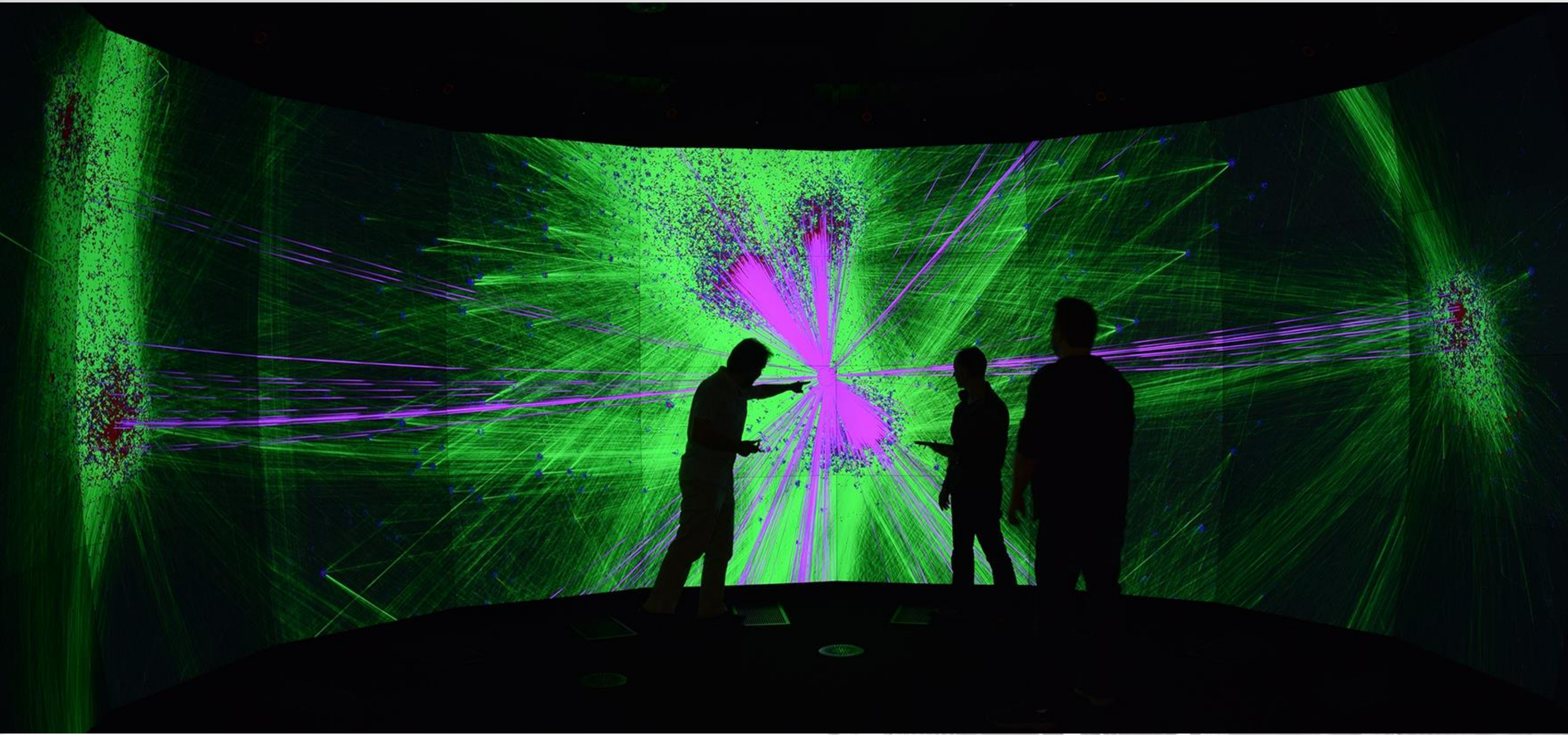
```
"meshes": [{
    "name": "Graph",
    "primitives": [{
        "attributes": {
            "POSITION": 0,
            "COLOR_0": 1,
            "_DEGREE": 2,
            "_WEIGHT": 3
        },
        "indices": 4,
        "mode": 0 // GL_POINTS
    },{
        "attributes": {
            "POSITION": 5,
            "COLOR_0": 6,
            "_WEIGHT": 7
        },
        "indices": 8,
        "mode": 1 // GL_LINES
    }]
}]
```





Massive Networks

Visualising Very Large-Scale Graphs in Immersive Environments



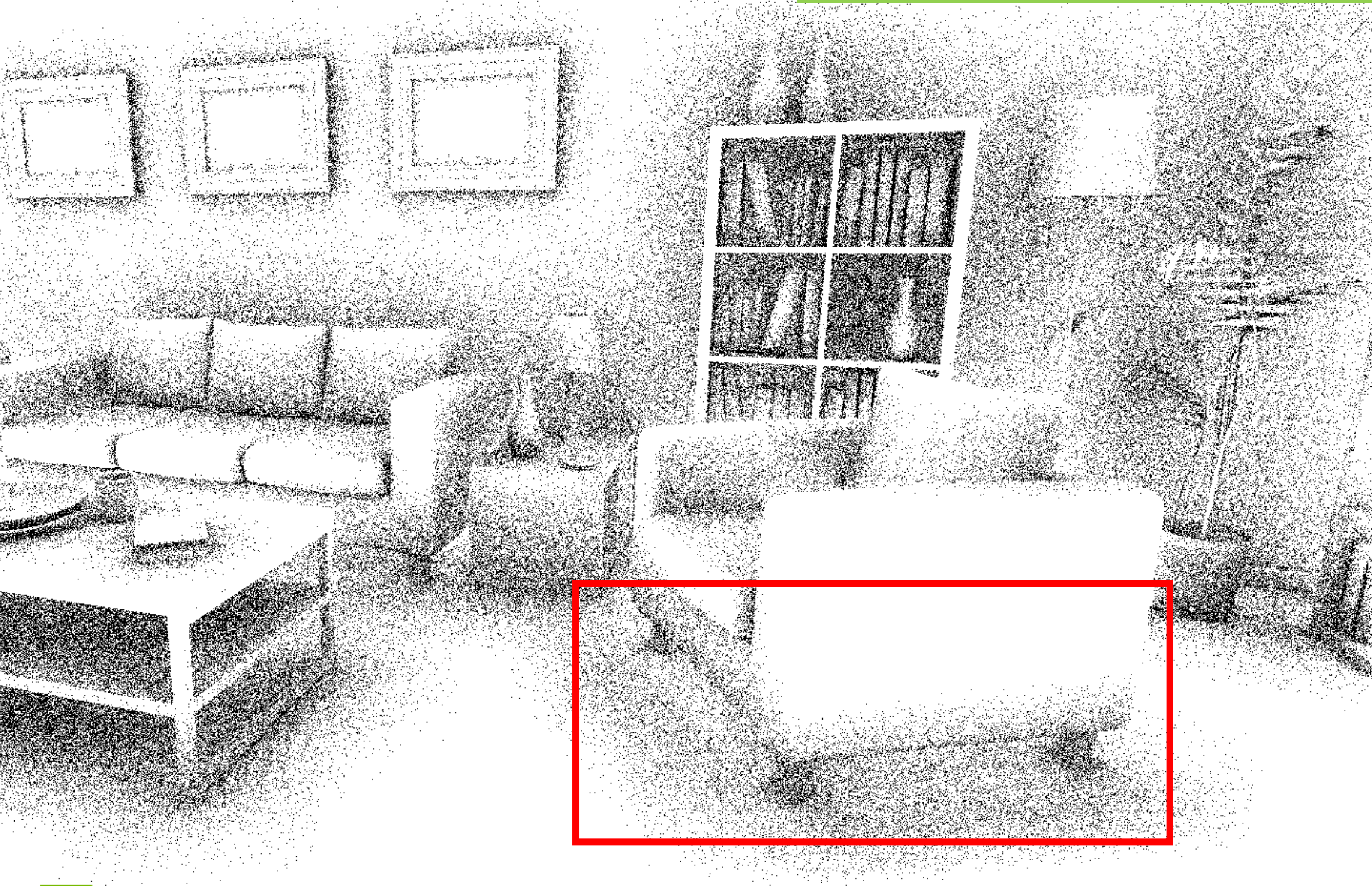
2018

Hybrid Ray-Traced Ambient Occlusion

Louis Bavoil Edward Liu Peter Shirley Morgan McGuire

NVIDIA

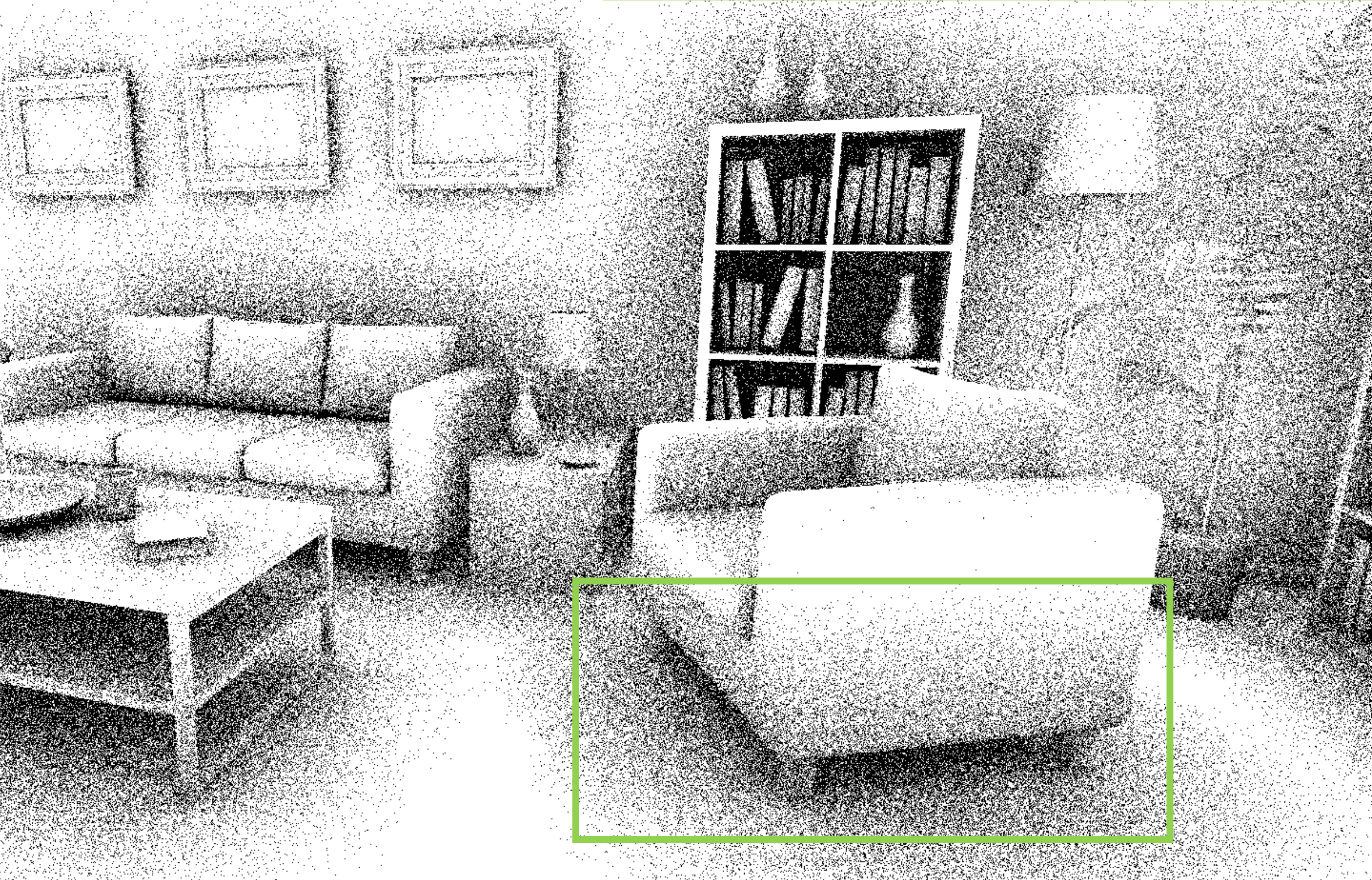
Biased: 2 Screen Rays



NVIDIA

Tracing: **1.5 ms** on TITAN V @ 1080p

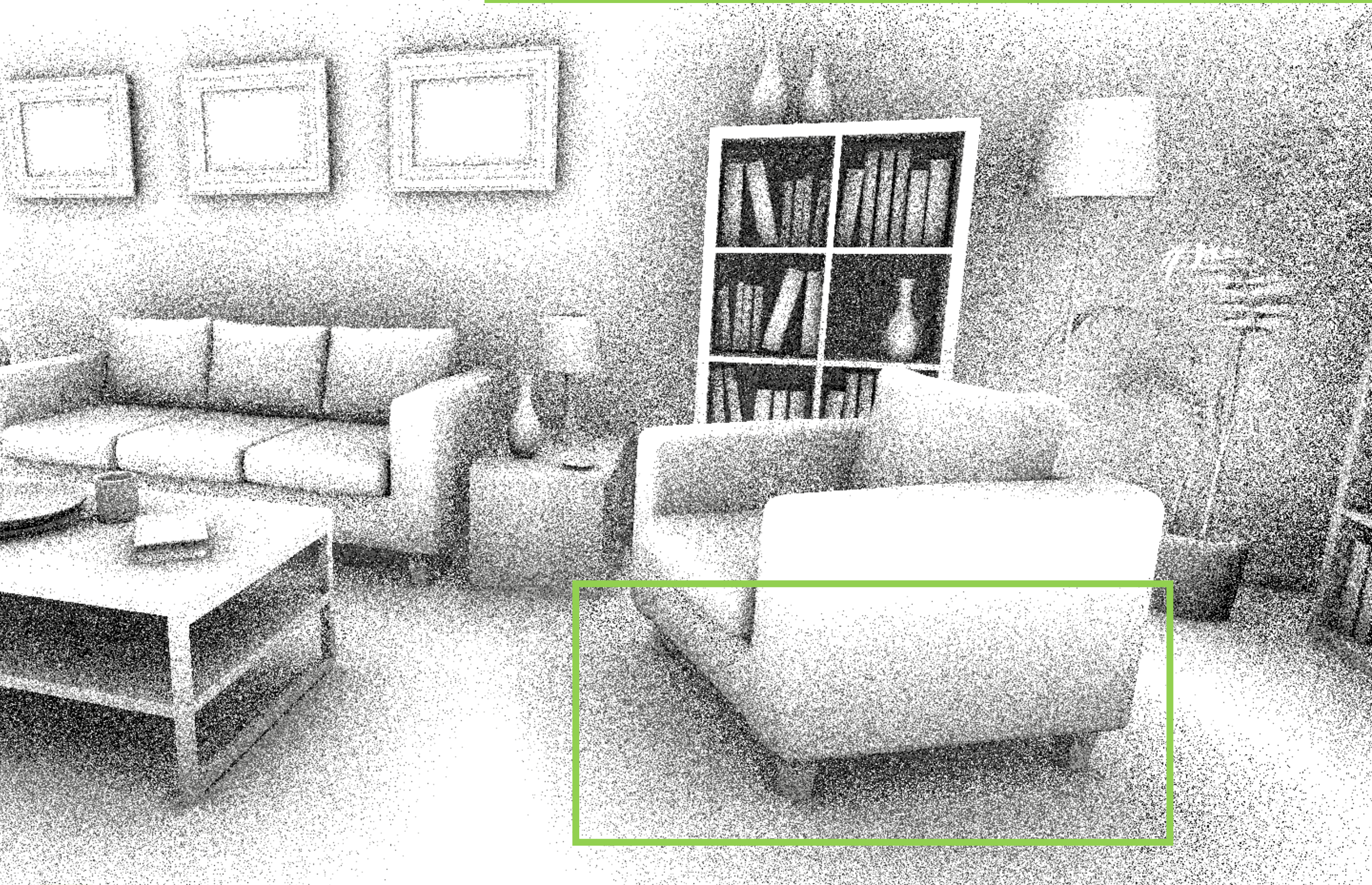
Brute force: 2 Geometric Rays



NVIDIA

Tracing: **6.4 ms** on TITAN V @ 1080p

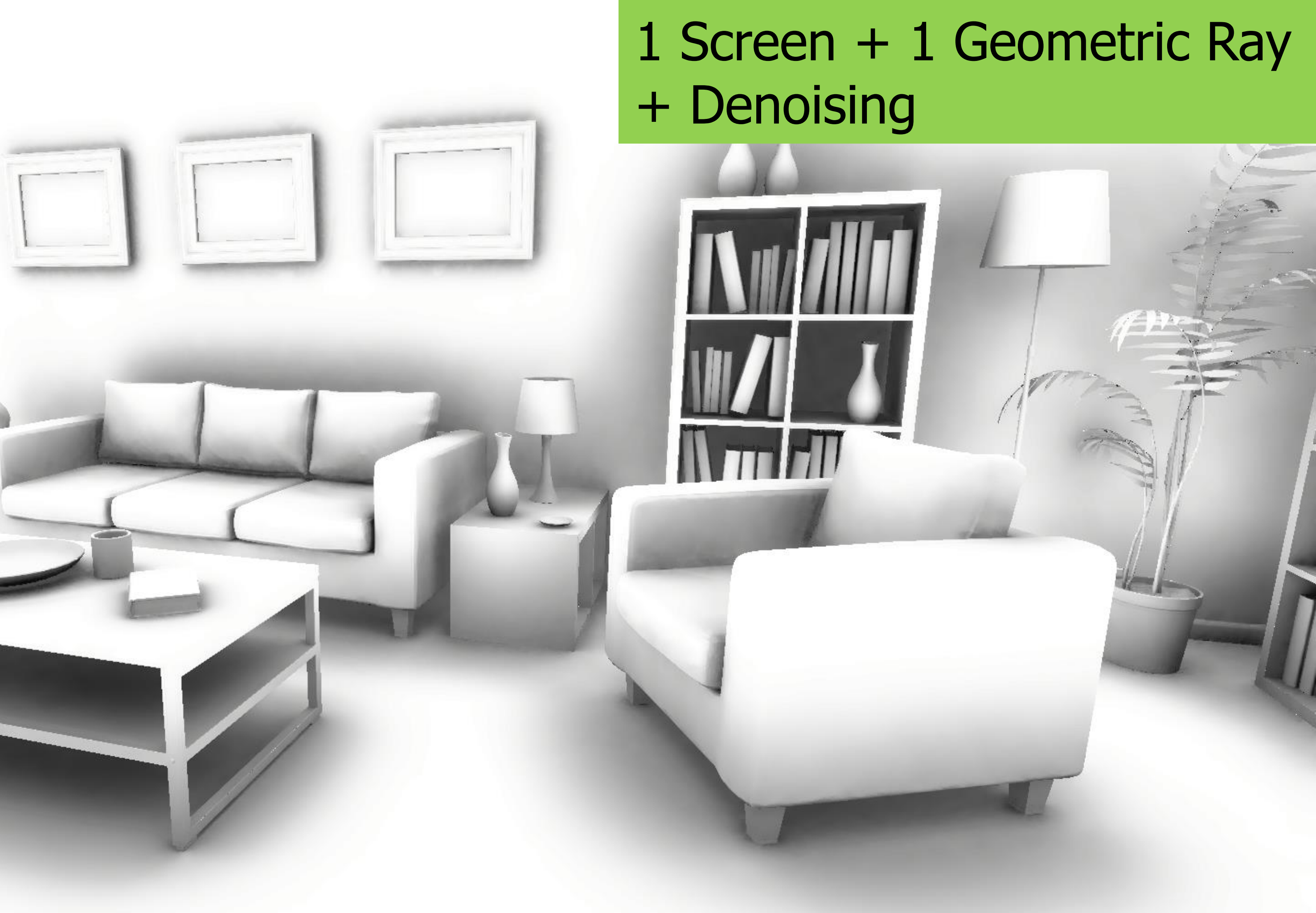
New: 1 Screen + 1 Geometric Ray



NVIDIA

Tracing: **5.8 ms** on TITAN V @ 1080p

1 Screen + 1 Geometric Ray + Denoising



NVIDIA

Tracing: **5.8 ms** on TITAN V @ 1080p

hpg 2018

A 2D to 3D Video Converter using Optical Flow Information and Least Squares Regression

Hui-Yun Lee
Chang Gung University

Introduction

- 3D visual technology and media contents
  - more attractive than traditional media
  - increasing attention from people
- Current methods to model 3D scenes
  - hardware solutions: dual lens, motion capture
  - software solutions: Maya and 3Ds max.
- For existing 2D media
  - can only develop a method to transferring 2D to 3D
  - also an economic method to produce 3D contents

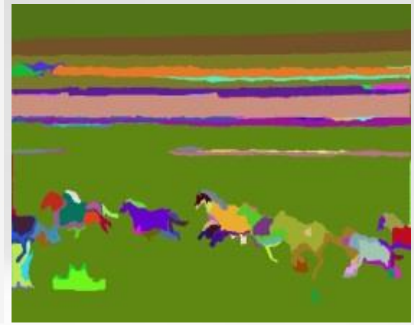
Our Method

- Read the video into frames.
- Calculate the optical flow values for each frame.
- Assign six grades, from 0 to 5 with uniform steps, as the depth values to the corresponding pixels.
- Use the mean-shift technique to partition a given image frame into superpixels, and assign the maximum depth grade thereof to everywhere of the same mean-shift region.
- Smooth the change of depth by building a high order polynomial surface according to the depth map.
- Render the reorganized 3D video.

Examples

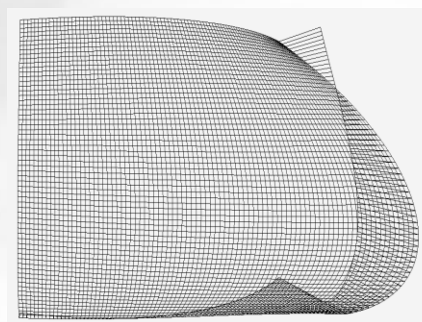
(a)

(b)



(c)

(d)



(e)

(f)

Examples

hpg 2018

Blender based Rendering-as-a-Service Platform for High Performance Computing Clusters

August 10-12, Vancouver, BC

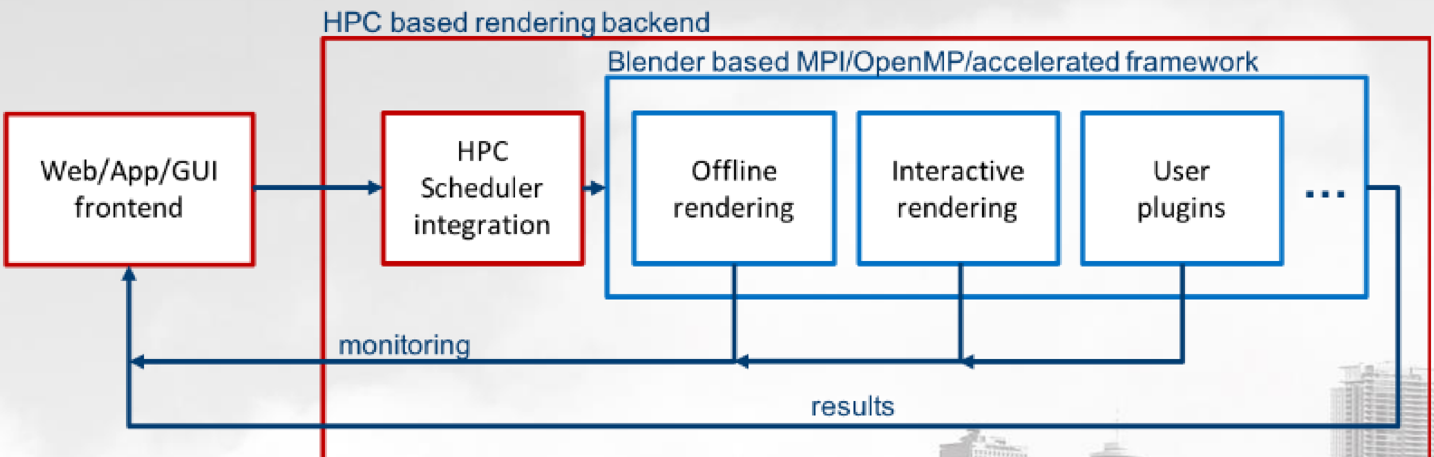
MŠMT

IT4Innovations
national01$#&0
supercomputing
center@#01%101

Authors: Milan Jaros, Petr Strakos, Lubomir Riha

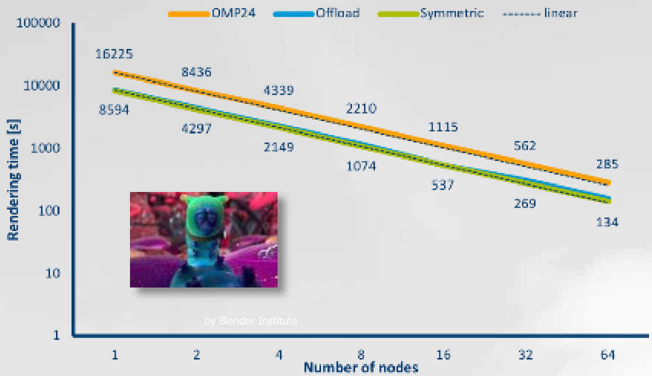
Our approach

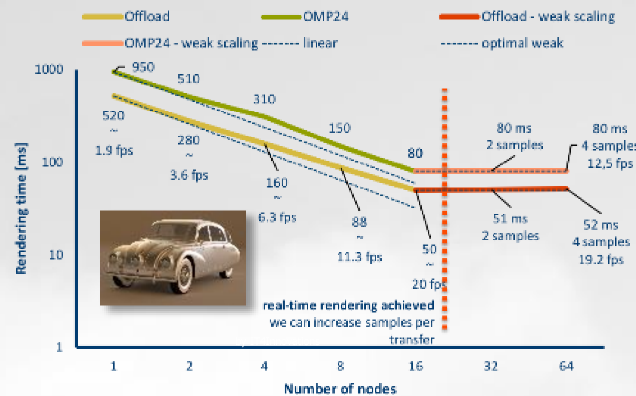
- Our platform is based on Blender renderers and upgrades them with HPC technologies.
- In this way we can offer not only standard offline but also interactive rendering mode which relies on fast HPC interconnecting networks.



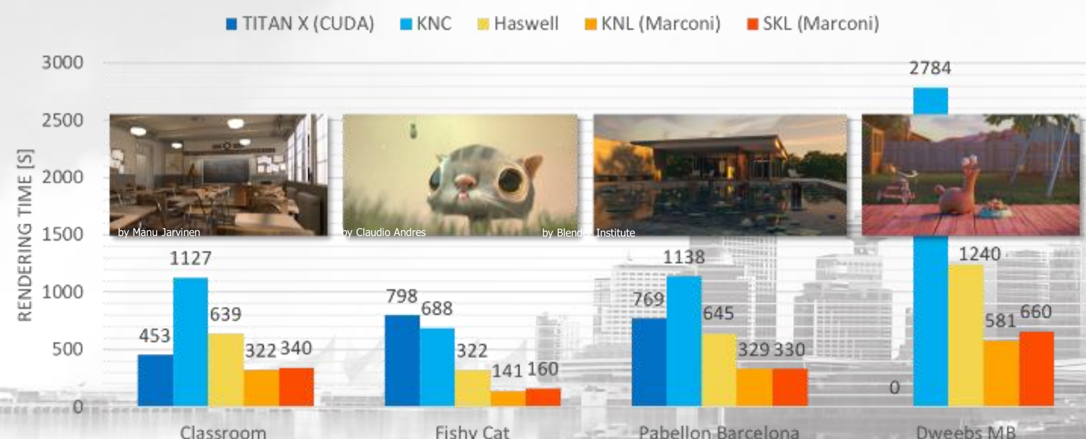
Rendering tests

- Scalability performance in offline rendering mode





- Rendering time comparison



hpg 2018

Energy Consumption Optimization of Rendering in Blender Cycles on x86 Architectures

August 10-12, Vancouver, BC

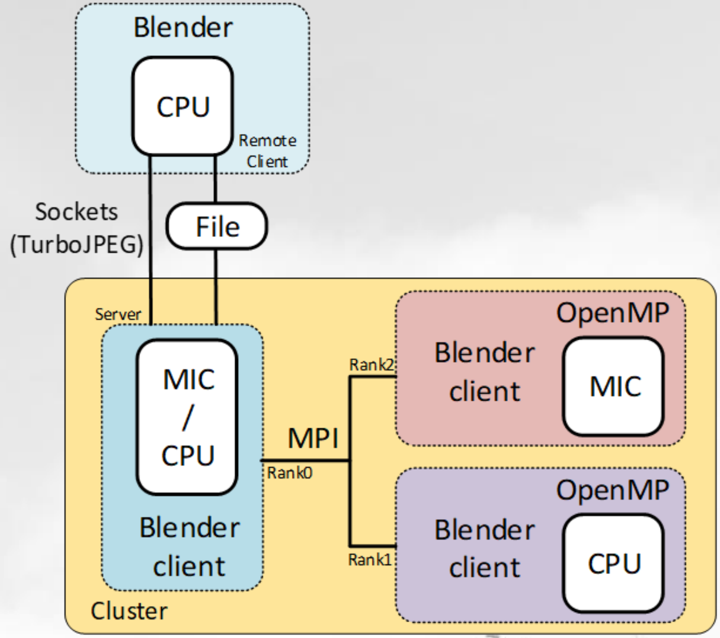
IT4Innovations national supercomputing center

READEX Runtime Exploitation of Application Dynamism for Energy-efficient eXascale computing

Authors: Milan Jaros, Ondrej Vysocky, Petr Strakos, Lubomir Riha

Our approach

- Extend Blender's renderer to support HPC resources and allow optimization of the energy consumption.



- The energy measurement of the whole node is defined by the equation:

$$E = energy_{cpu} + baseline * time$$

Architectures comparison

- Comparing between architectures **up to 18% of energy can be saved while increasing the rendering time just by 3%** (The Fishy Cat scene).

Platform	Default settings	Default HW configuration	Optimal settings	Optimal HW configuration	Energy and time savings
Classroom scene					
HSW AC	**19318 J; 65 s**	3 GHz (U); 2.8 GHz (C)	18286 J; 79 s	1.6GHz (U); 2.4 GHz (C)	E+5%; T-22%
HSW DLC	18699 J; 65 s	3 GHz (U); 2.8 GHz (C)	**17001 J; 79 s**	1.6GHz (U); 2.4 GHz (C)	E+12%; T-22%
KNL AC	16681 J; 66 s	1.5 GHz (C)	**16681 J; 66 s**	1.4 GHz (C)	E+14%; T-2%
Dweebs scene					
HSW AC	**19072 J; 64 s**	3 GHz (U); 2.8 GHz	18249 J; 78 s	1.8 GHz (U); 2.4 GHz (C)	E+4%; T-22%
HSW DLC	18541 J; 64 s	3 GHz (U); 2.8 GHz	**17093 J; 78 s**	1.8 GHz (U); 2.4 GHz (C)	E+10%; T-22%
KNL AC	15978 J; 62 s	1.5 GHz (C)	**15743 J; 66 s**	1.3 GHz (C)	E+17%; T-3%
Fishy Cat scene					
HSW AC	**18794 J; 63 s**	3 GHz (U); 2.8 GHz (C)	17755 J; 73 s	1.8 GHz (U); 2.4 GHz (C)	E+6%; T-16%
HSW DLC	18211 J; 63 s	3 GHz (U); 2.8 GHz (C)	**16672 J; 73 s**	1.8 GHz (U); 2.4 GHz (C)	E+11%; T-16%
KNL AC	15607 J; 61 s	1.5 GHz (C)	**15431 J; 65 s**	1.3 GHz (C)	E+18%; T-3%
Pabellon B. scene					
HSW AC	**17833 J; 60 s**	3 GHz (U); 2.8 GHz (C)	17220 J; 73 s	1.8 GHz (U); 2.4 GHz (C)	E+3%; T-22%
HSW DLC	17068 J; 60 s	3 GHz (U); 2.8 GHz (C)	**15732 J; 73 s**	1.8 GHz (U); 2.4 GHz (C)	E+12%; T-22%
KNL AC	16096 J; 63 s	1.5 GHz (C)	**15872 J; 67 s**	1.3 GHz (C)	E+11%; T-12%