SIGGRAPH ASIA 2016 MACAO

Sponsored by acm

Z^2 Traversal Order for VR Stereo Rendering on Tile-based Mobile GPUs

Jae-Ho Nah, Yeongkyu Lim,
Sunho Ki, and Chulho Shin

LG Electronics

SA2016.SIGGRAPH.ORG

CONFERENCE: 5 - 8 DECEMBER 2016 • EXHIBITION: 6 - 8 DECEMBER 2016 • THE VENETIAN MACAO, MACAO

INTRODUCTION

VIRTUAL REALITY

- Explosion in VR market
  - Thanks to technical advances of HMDs and GPUs
  - A wide range of applications

- High computational costs for VR rendering
  - High screen resolution (e.g., FHD to UHD)
  - High frame/refresh rates (e.g., 60-120Hz)
  - Stereo rendering for the left and right eyes

- Thus, efficient VR rendering techniques are required for realistic VR experiences!

MAIN CONTRIBUTIONS OF OUR WORK

- Focus on efficient **GPU H/W architectures** for mobile VR applications

- Novel cache-efficient tile traversal orders for VR rendering
  - Interleaved version of the traditional Z-order curve [Morton 1966]
  - Two variants:

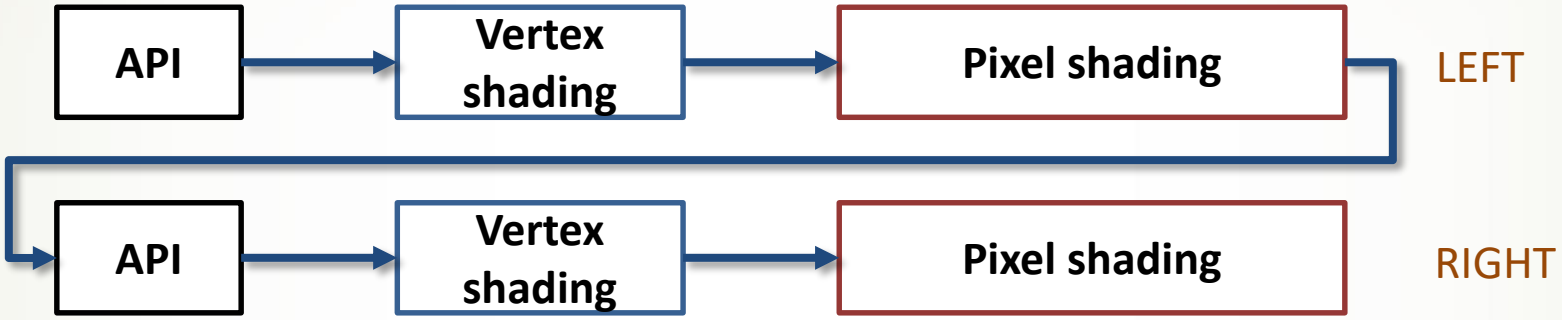
    Z^2 LRTA (left-right tile assignment) & Z^2 STA (simultaneous tile access)

- Implementation of a simulation environment
  - Mesa OpenGL renderer + Oculus VR library + in-house H/W simulator

BACKGROUND AND RELATED WORK

VR STEREO RENDERING

- Brute-force approach

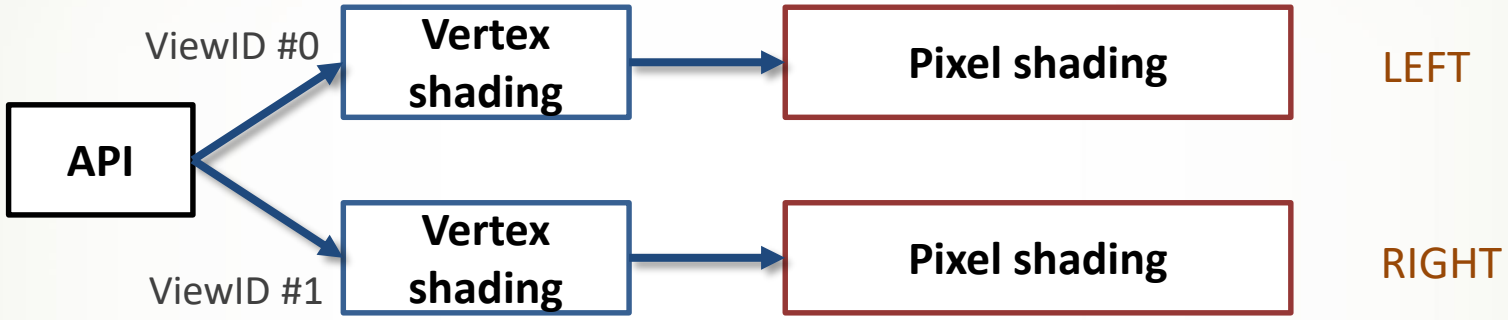


  - 2X draw calls are required

* This classification refers to Reed and Sancho [2015]

VR STEREO RENDERING

- Shader multiview [Reed and Sancho, GDC 2015]

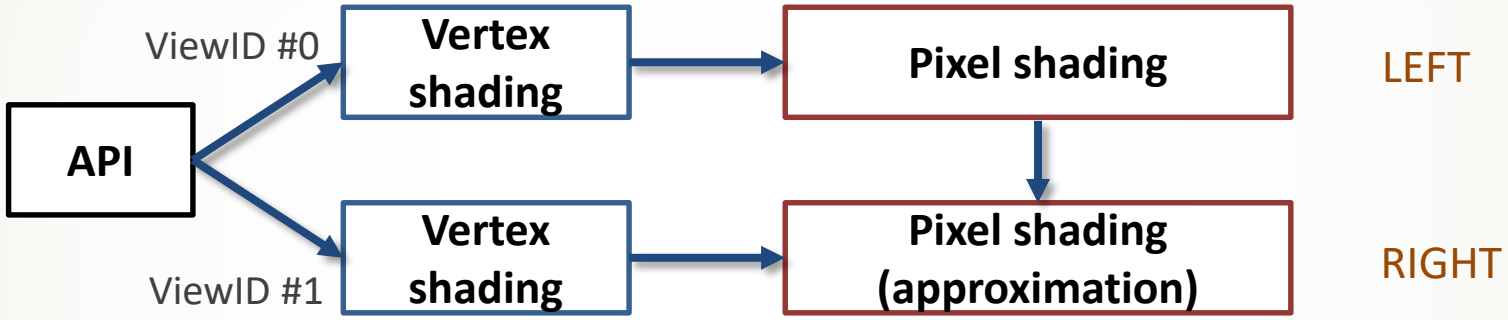


  - API overhead reduction
  - OpenGL extensions (e.g., GL_OVR_multiview2)

* This classification refers to Reed and Sancho [2015]

VR STEREO RENDERING

- Shading reuse [Hasselgren and Akenine-Möller, EGSR 2006]

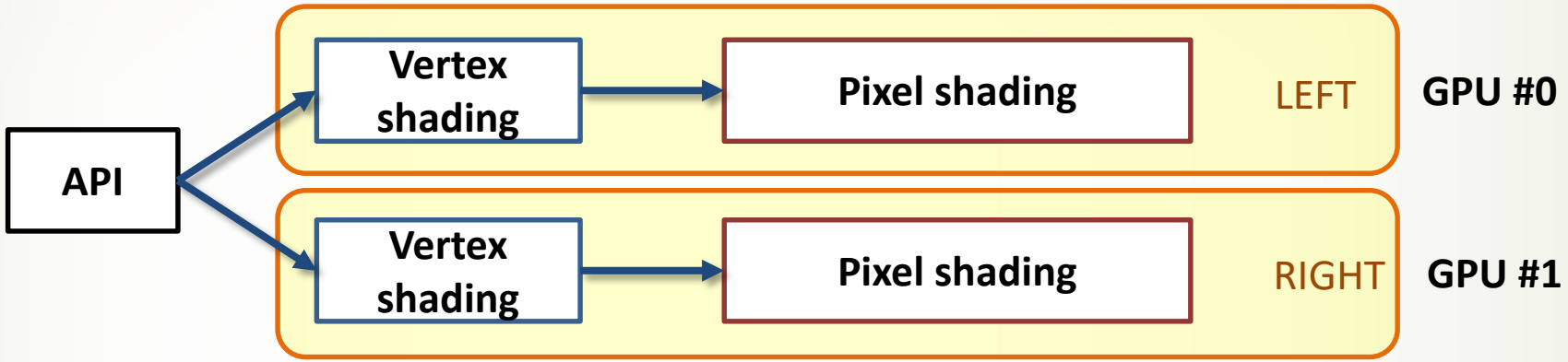


- Instead of exact PS evaluation, approximate PS evaluation is performed on the texture space for the right view (or multiple views)
- Aggressive & efficient for reducing PS costs
- Possibility of image quality degradation:
  problematic on view-dependent shading (e.g., specular highlights)

* This classification refers to Reed and Sancho [2015]

VR STEREO RENDERING

- VR SLI/CrossFire

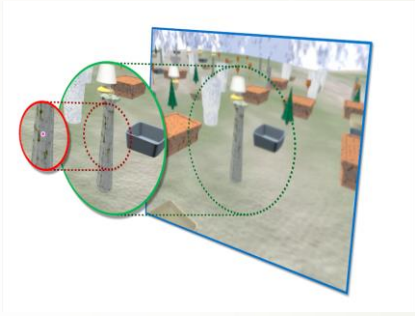


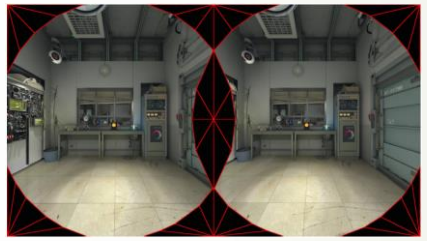
- NVIDIA VRWorks (2016) & AMD LiquidVR (2015)
- Left and right screens are distributed into two or more GPU cards connected by the SLI or CrossFire interface

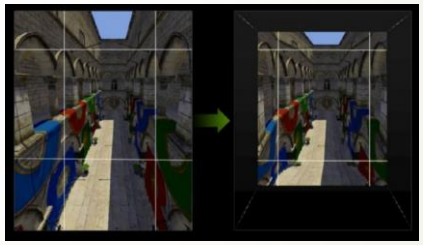
* This classification refers to Reed and Sancho [2015]

TECHNIQUES FOR REDUCING SHADED FRAGMENTS OF EACH VIEW

- Foveated rendering [Guenter et al., TVCG 2012]
  - High res. image in the fovea
  - Low res. image in the periphery
- Stencil mesh [Vlachos, GDC 2015]
  - Cull area hidden by warping in advance
- Multi-resolution shading [NVIDIA 2016]
  - Low res. image in the edges of the screen distorted by warping and lens distortion







TILE-BASED GPU ARCHITECTURES AND TILE TRAVERSAL ORDERS

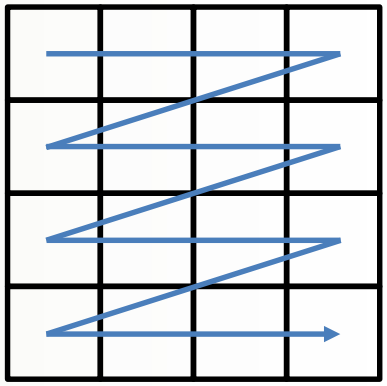
- Tile-based GPU architectures

Stone Age Begins © Netmarble Games Corp. 2016

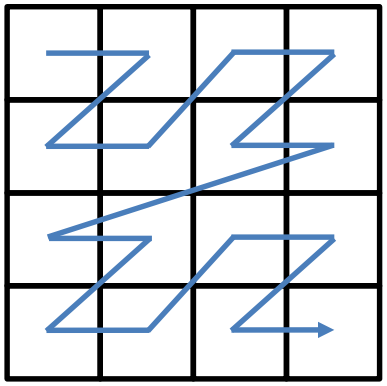
- The choice of most mobile GPUs: Adreno, Mali, and PowerVR
- Aiming at minimizing external memory accesses by using a local tile buffer
- # of tiles > # of GPU cores → various traversal orders can exist

TILE-BASED GPU ARCHITECTURES AND TILE TRAVERSAL ORDERS

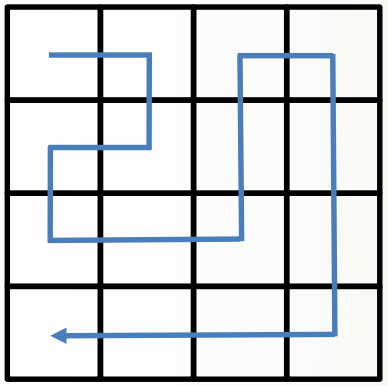
- Tile traversal orders
  - Cache-friendly traversal order can increase GPU performance
  - Representative examples



Scanline order



Z-order curve
[Morton 1966]

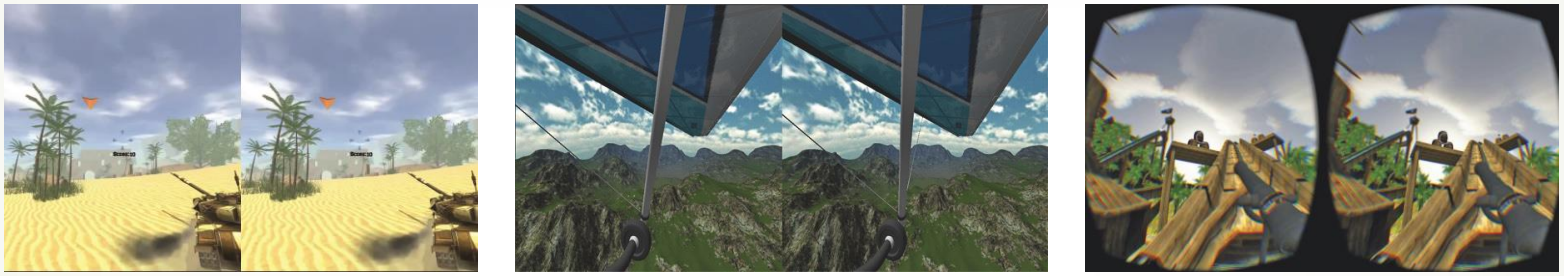


Zig-zag pattern
[Ellis et al. 2015]

Z^2 TILE TRAVERSAL ORDER

OUR OBSERVATION

- The left and right views are usually similar each other; binocular disparity is not very high

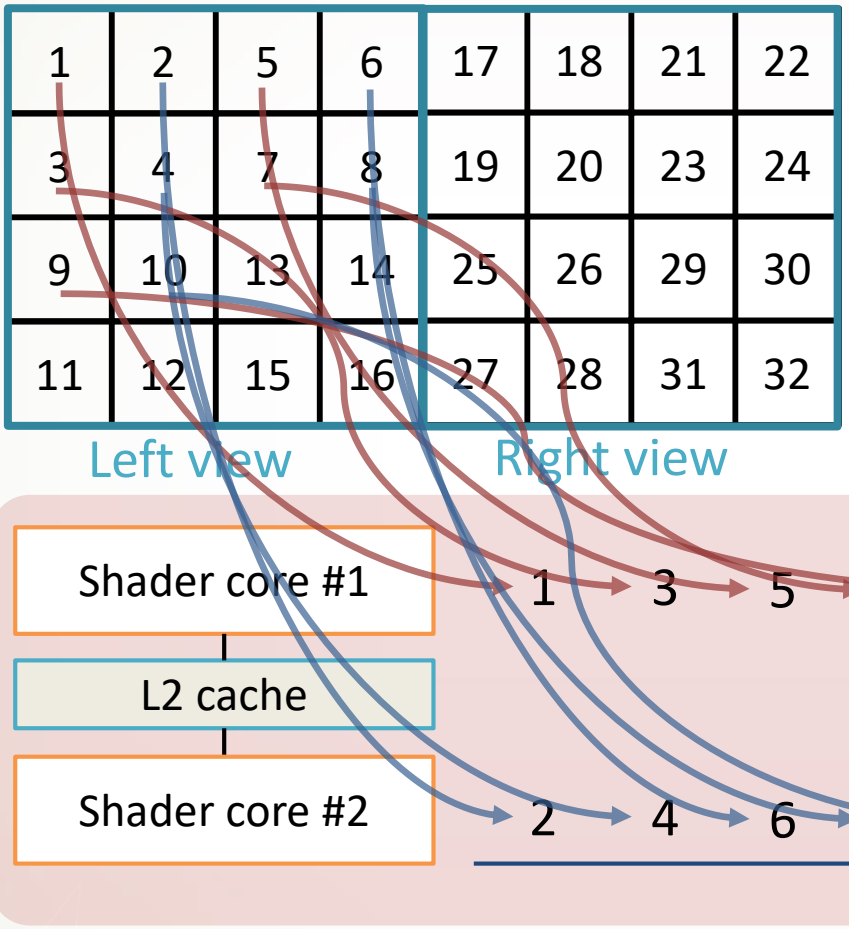


Three Cardboard games: VR Tank Training, Hang Gliding, and Swivel Gun! Log Ride

- Interleaved tile traversal between the left and right views can increase texture cache efficiency
  - This idea was inspired by Hasselgren and Akenine-Möller [2006]

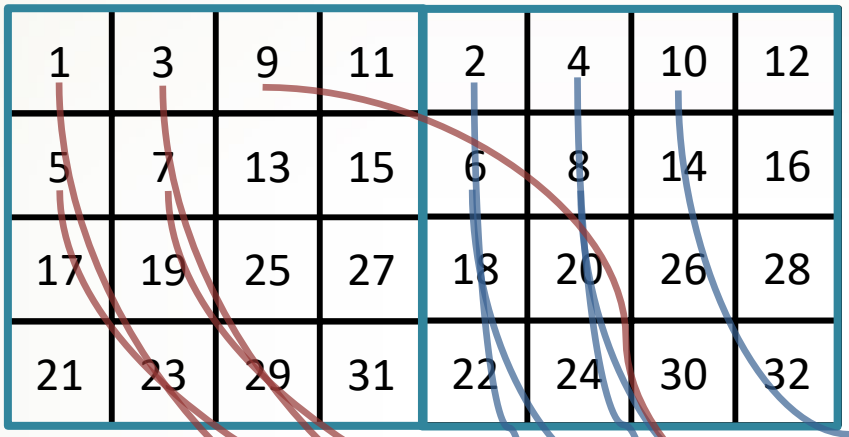
- Our main idea: Z-order curve + left-right interleaving

Z-ORDER CURVE

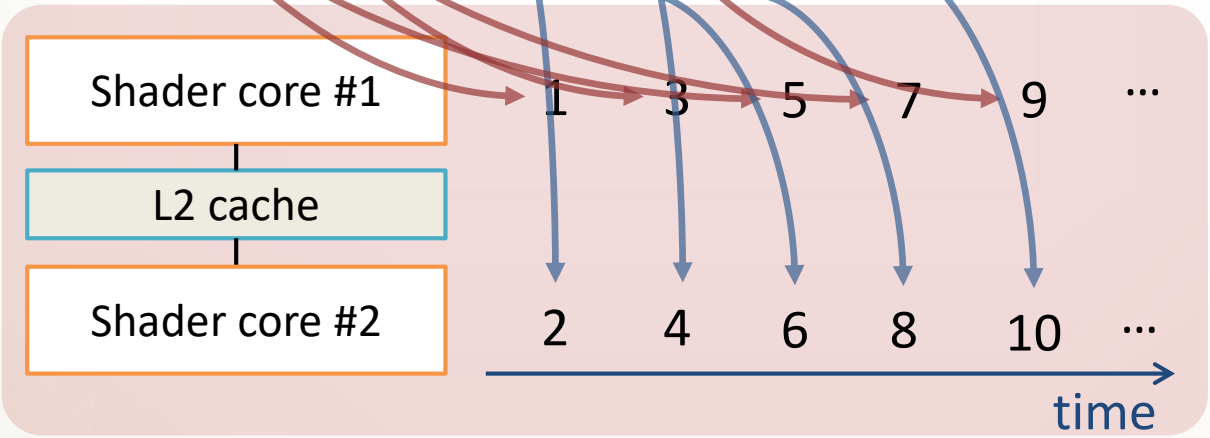


- Not effectively exploit locality between the left and right views

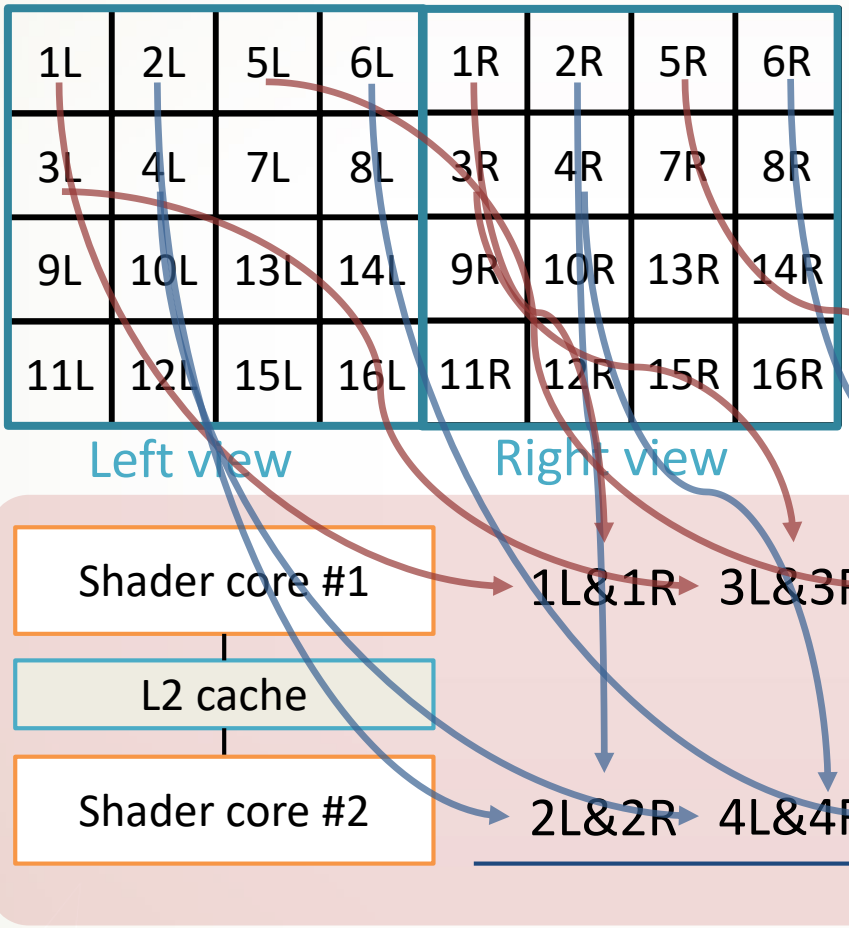
Z^2 LRTA (LEFT-RIGHT TILE ASSIGNMENT)



- Render the tiles in the left and right screens by turns
- Exploit locality between the left and right views
- Simple implementation



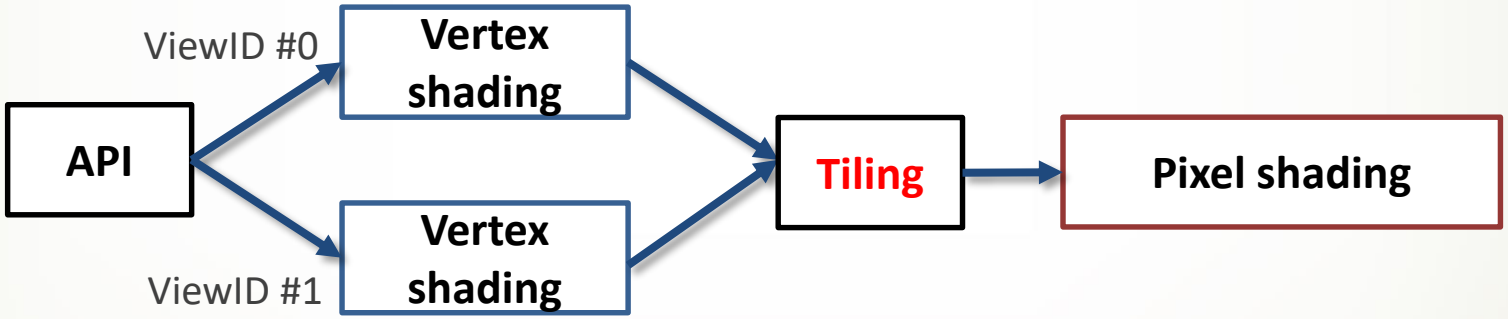
Z^2 STA (SIMULTANEOUS TILE ACCESS)



- Fetch two tiles in the left and right screens simultaneously
- Triangles in the left and right triangle lists are rendered by turns
- Best for two screen tiles consisting the same or similar triangle lists
- Require double-sized tile memory

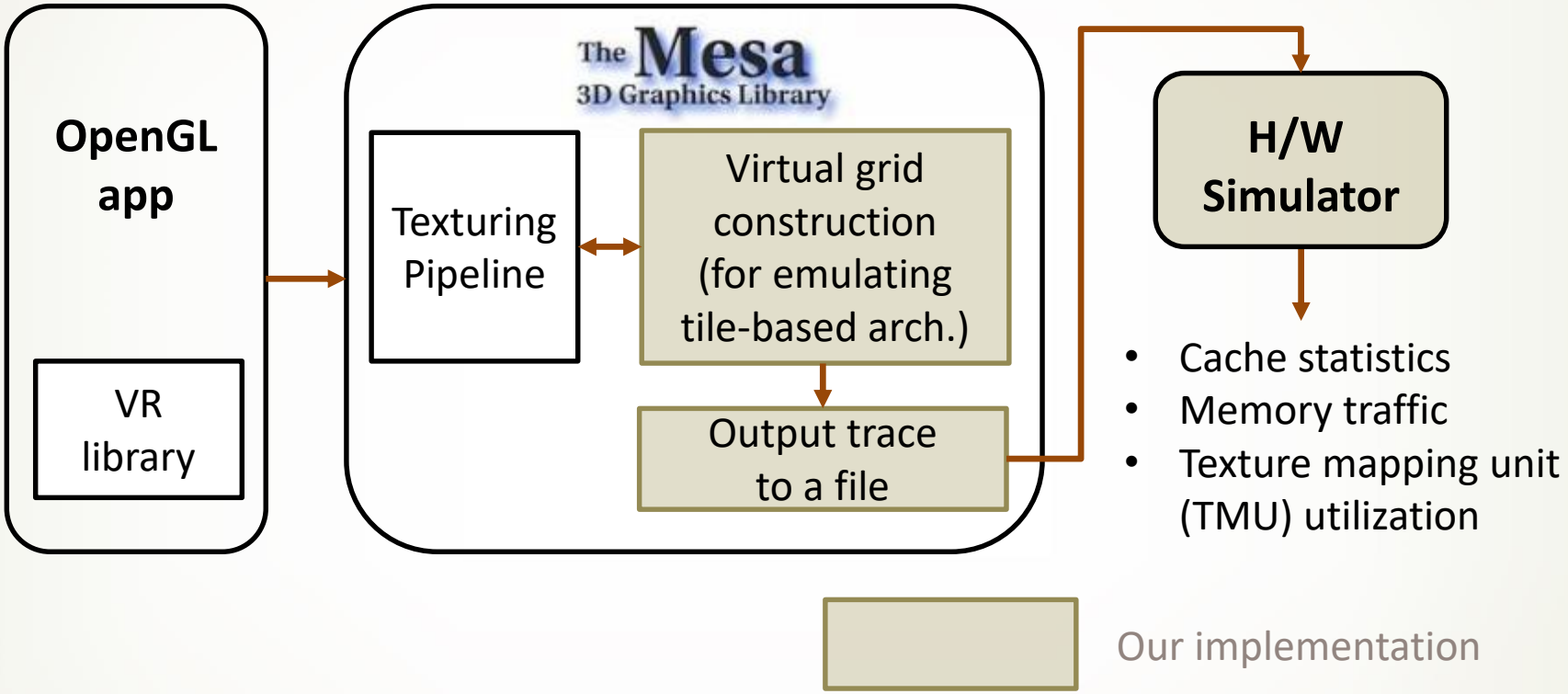
PREREQUISITE FOR Z^2 ORDER

- Use of **shader multiview** extensions is required
  - Brute-force case: no clue how to obtain the geometry lists of the left and right views simultaneously
  - Our traversal order can be enabled **only if** the tiling stage can sort all geometry of the left and right views together in a single render pass



EXPERIMENTS AND RESULTS

SIMULATION ENVIRONMENT



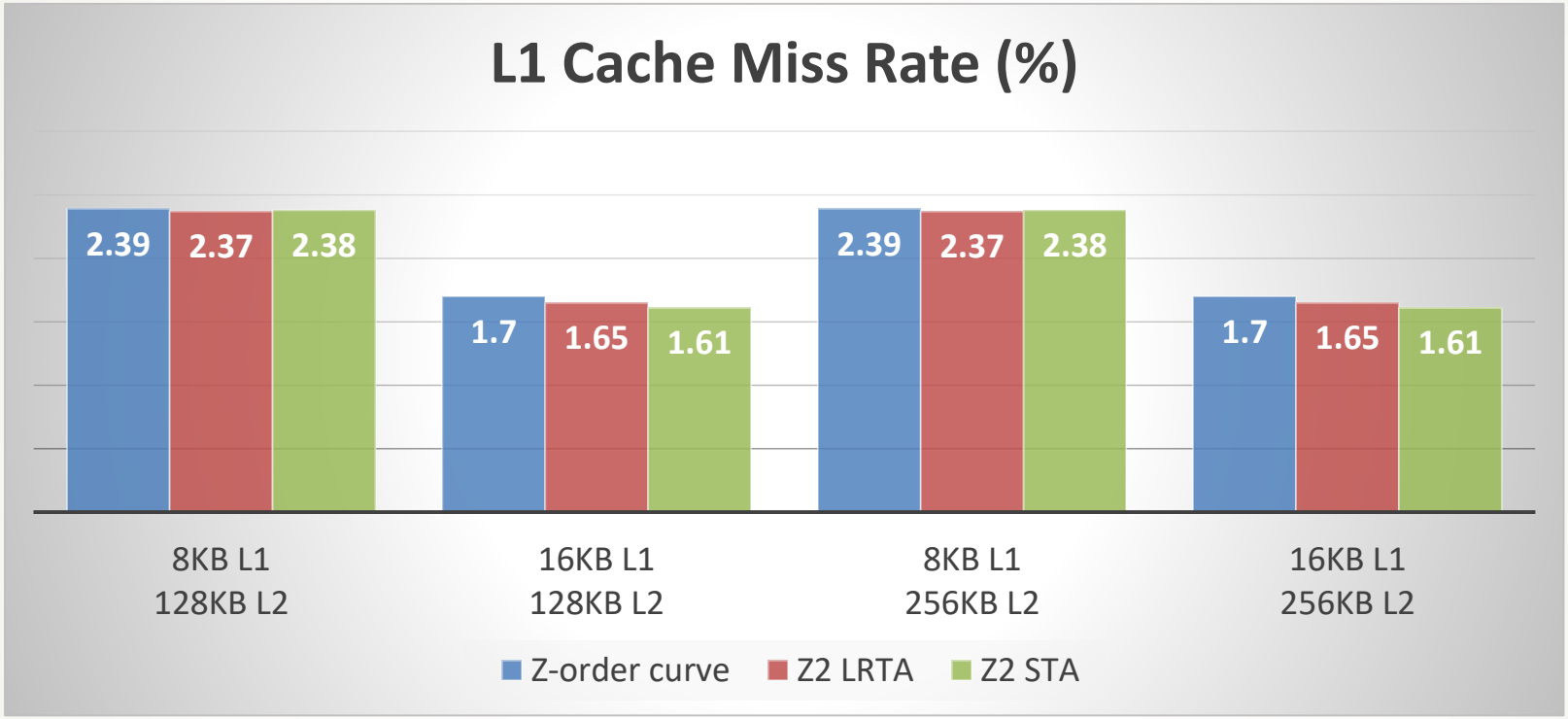
EXPERIMENTAL SETUP

- H/W assumptions in our experiment
  - MP2 GPU configuration:
    two L1 caches (8-16KB) and an L2 cache (128-256KB)
  - L1 and L2 miss penalties: 20 and 200 cycles, respectively
  - The texture pipeline is stalled when a cache miss occurs
  - 16×16 tile size

- Test scene
  - GFXBench T-Rex rendered
    with Oculus VR library
  - Resolution: 960×1080×2
  - DXT1 compressed textures

RESULTS

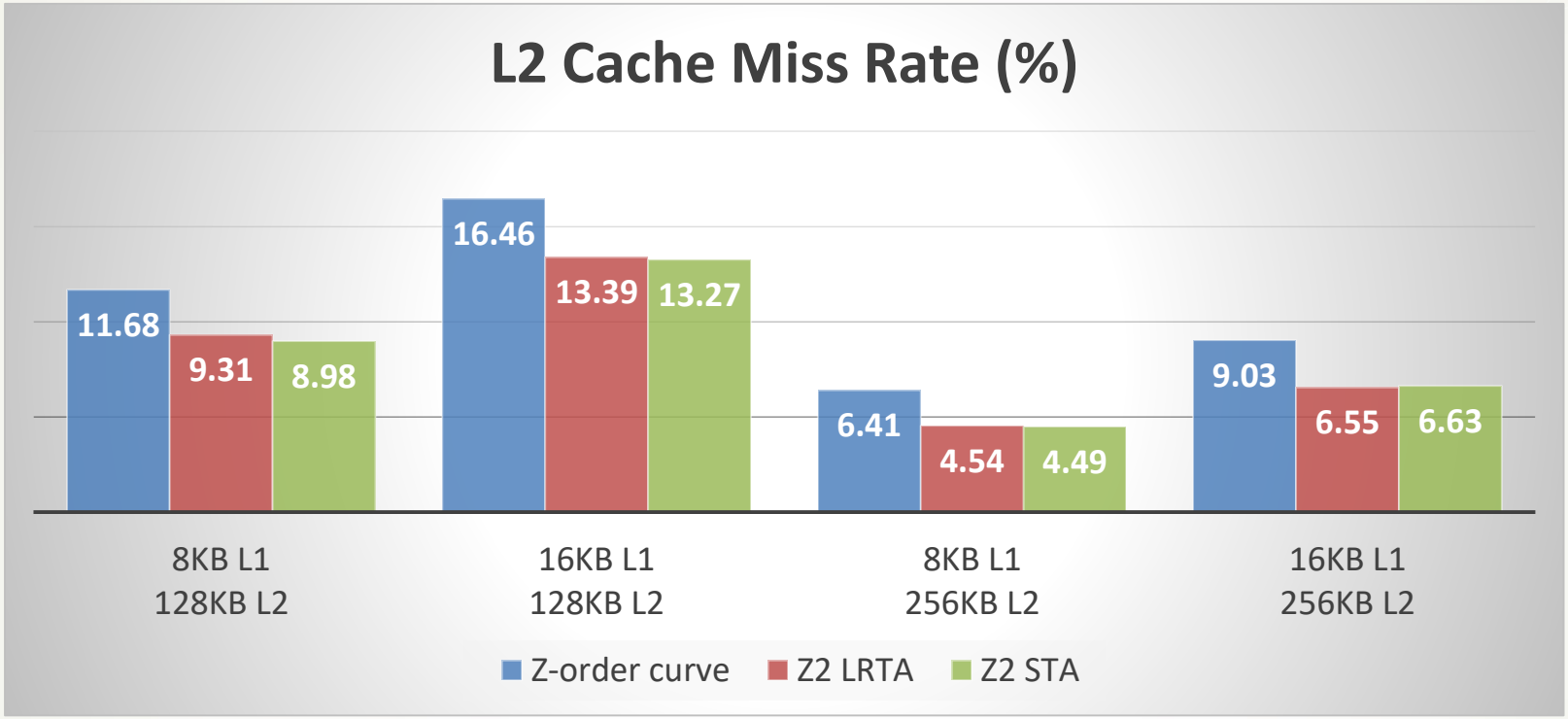
Lower is better



- Both Z^2 LRTA and Z^2 STA show slightly lower L1 cache miss rates than Z-order curve

RESULTS

Lower is better

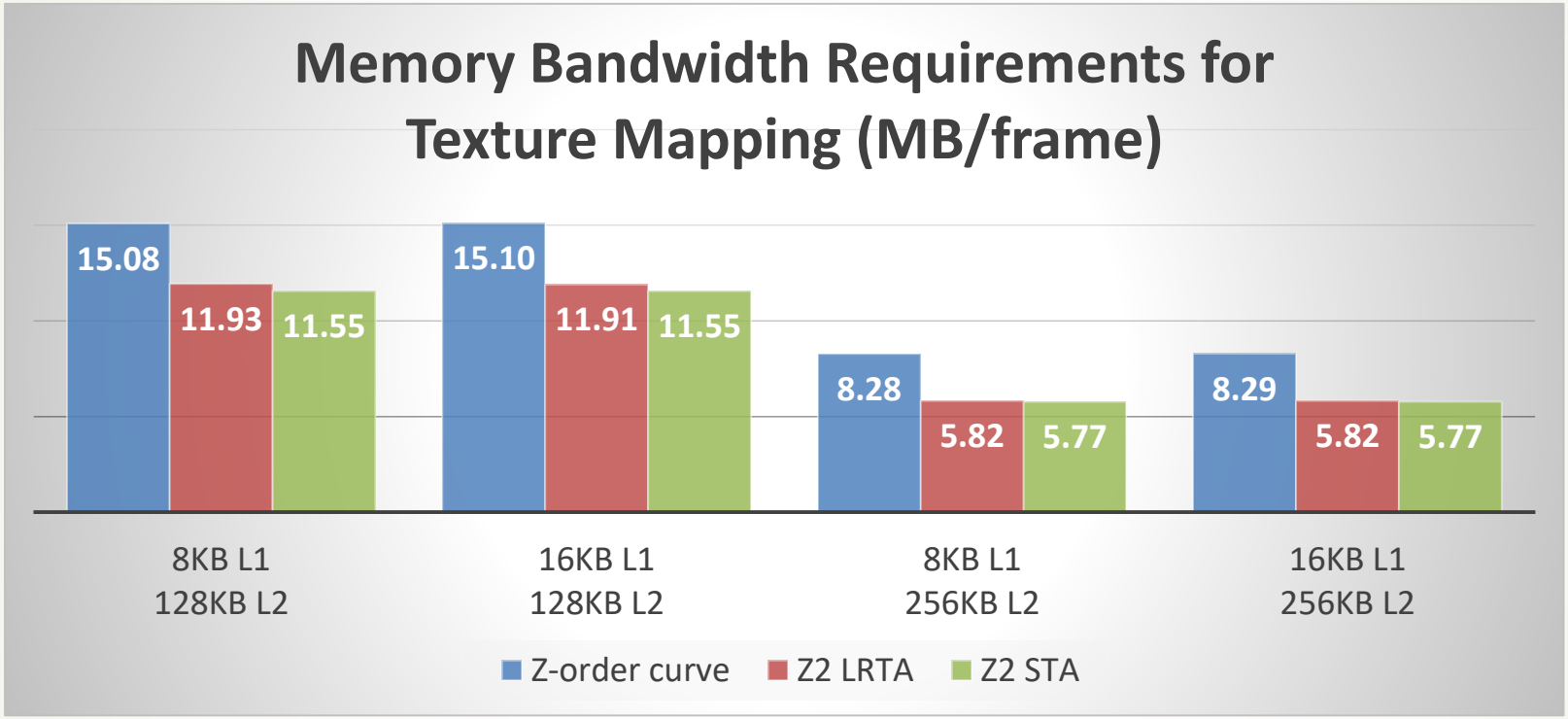


- Z^2 LRTA considerably reduces L2 cache miss rates
- Z^2 STA reduces the miss rates further in most cases

SA2016.SIGGRAPH.ORG

RESULTS

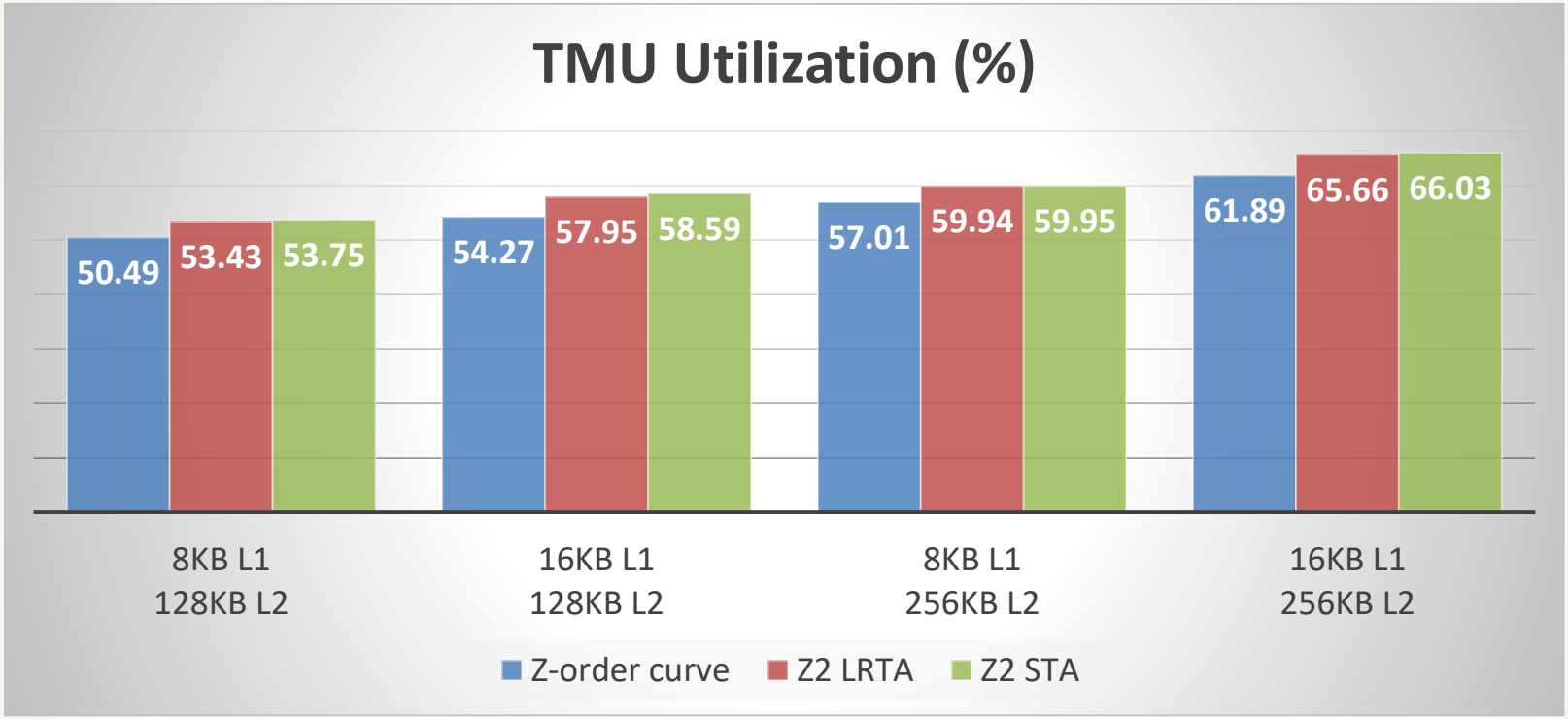
Lower is better



- Z^2 LRTA and Z^2 STA reduce memory traffic by 21-30%
- Z^2 STA shows slightly better results than Z^2 LRTA

RESULTS

Higher is better



- Z^2 LRTA and Z^2 STA can achieve 5-8% performance gain over Z-order curve in texture-heavy scenes

WRAP UP

CONCLUSIONS AND FUTURE WORK

- Two variations of the Z-order curve for tile-based GPUs
  - Specially designed for VR stereo rendering
  - Z^2 LRTA – simpler implementation and no area overhead
  - Z^2 STA – higher efficiency
  - Orthogonal to other VR acceleration techniques

- Future work
  - Experiments in more scenes
  - Investigation on the effects of disparity manipulation techniques (e.g., gloss depiction [Templin et al., TOG 2012])
  - Extension to ray tracers using Z-order curve (e.g., RayCore [Nah et al., TOG 2014])

ACKNOWLEDGEMENTS

Anonymous reviewers

Byeongjun Choi

Jinhong Park

Kishonti Ltd. (GFXBench)

SIGGRAPH ASIA 2016 MACAO

CONTACT INFORMATION

Jae-Ho Nah

nahjaeho@gmail.com

Yeongkyu Lim

postrain70@gmail.com