Cache-Aware Sampling Strategies for Texture-Based Ray Casting on GPU

Junpeng Wang　Fei Yang　Yong Cao

VirginiaTech
Invent the Future

中国科学院
CHINESE ACADEMY OF SCIENCES

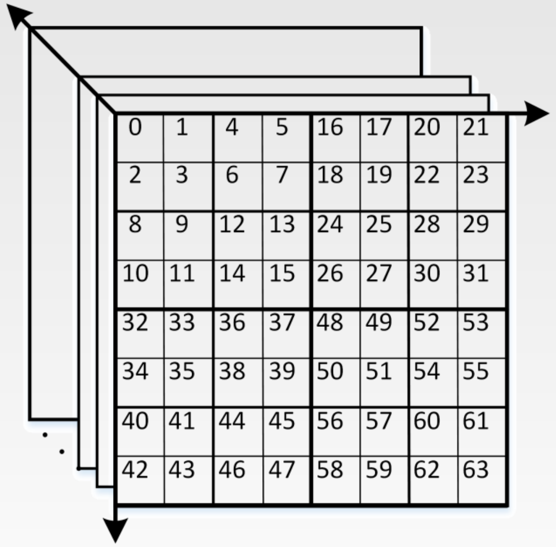
LDAV

VIS 2014
VAST · INFOVIS · SCIVIS
9-14 NOVEMBER 2014
PARIS, FRANCE

Overview

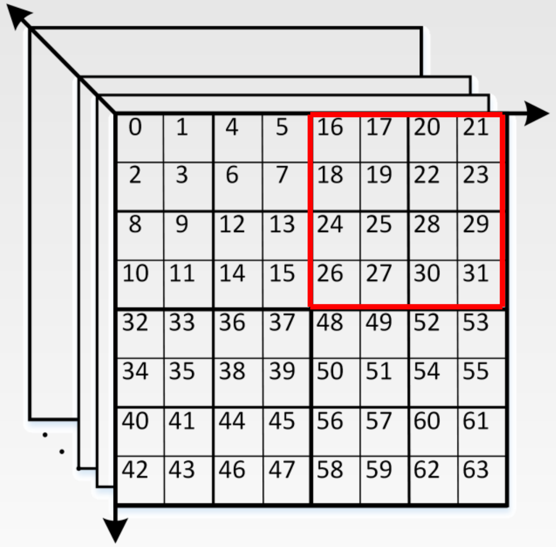
- Introduction/Motivation
- Related Work
- Contribution
- Result
- Application

Motivation



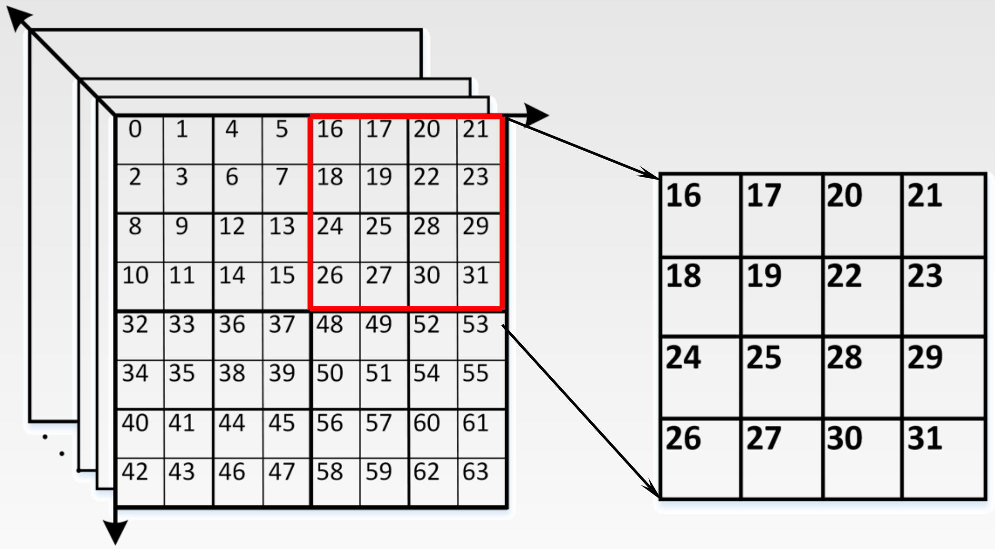
0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

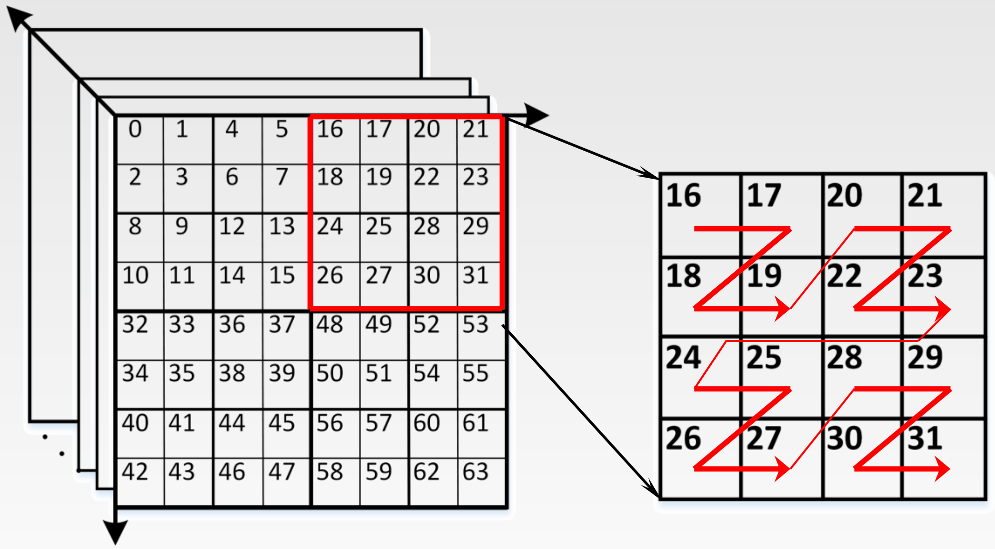
Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

16	**17**	**20**	**21**
18	**19**	**22**	**23**
24	**25**	**28**	**29**
26	**27**	**30**	**31**

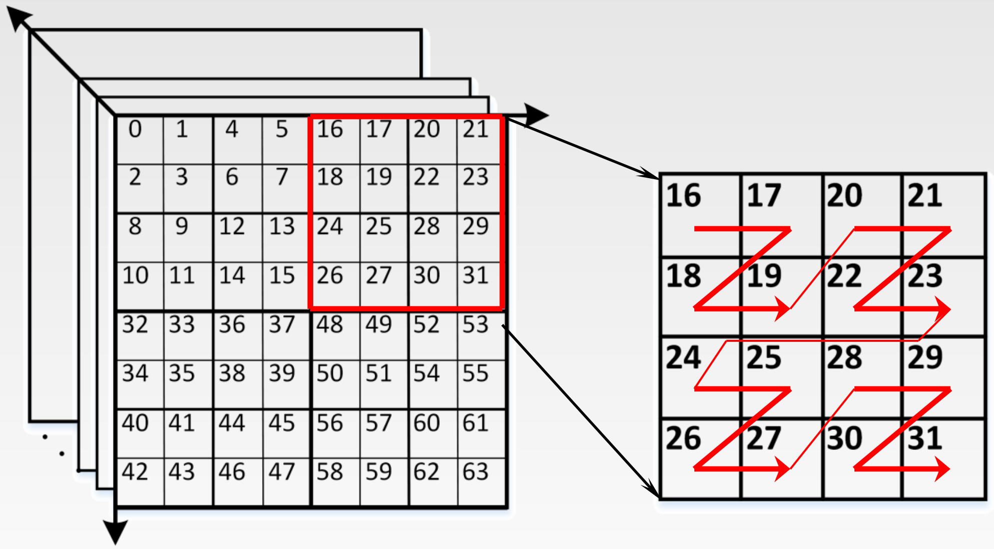
Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

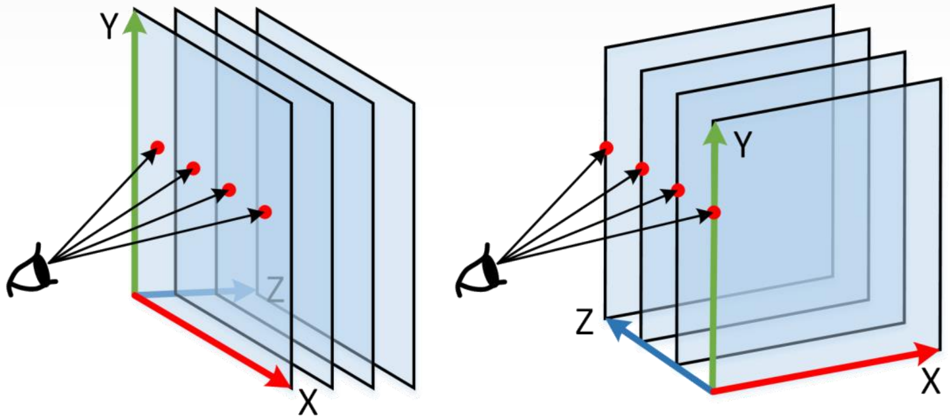
16	17	20	21
18	19	22	23
24	25	28	29
26	27	30	31

Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

16	17	20	21
18	19	22	23
24	25	28	29
26	27	30	31



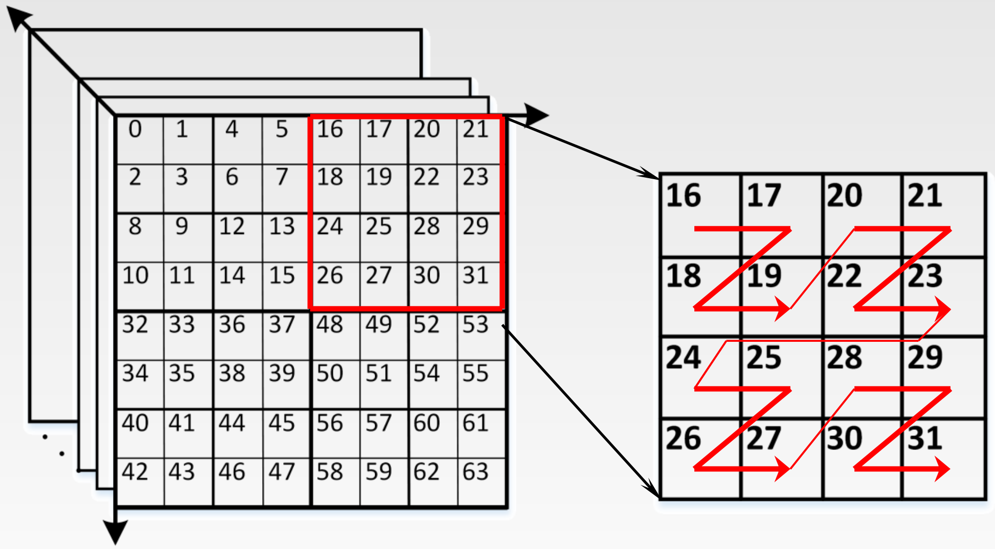
facing XY

facing ZY

Perpendicular

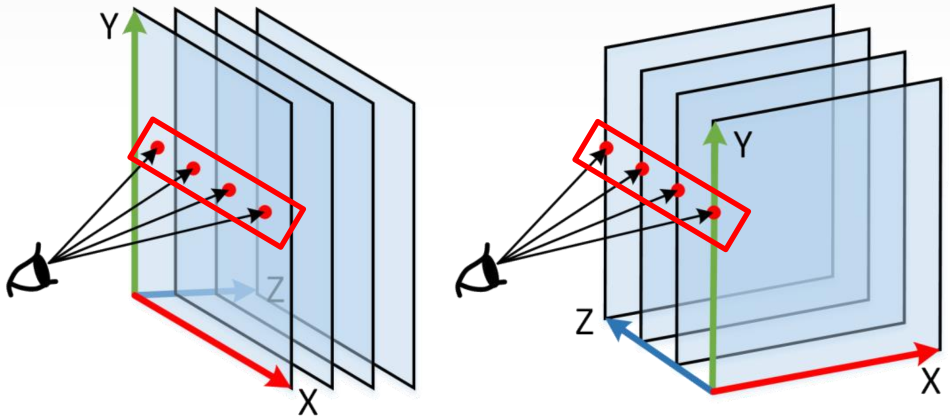
Parallel

Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

16	17	20	21
18	19	22	23
24	25	28	29
26	27	30	31



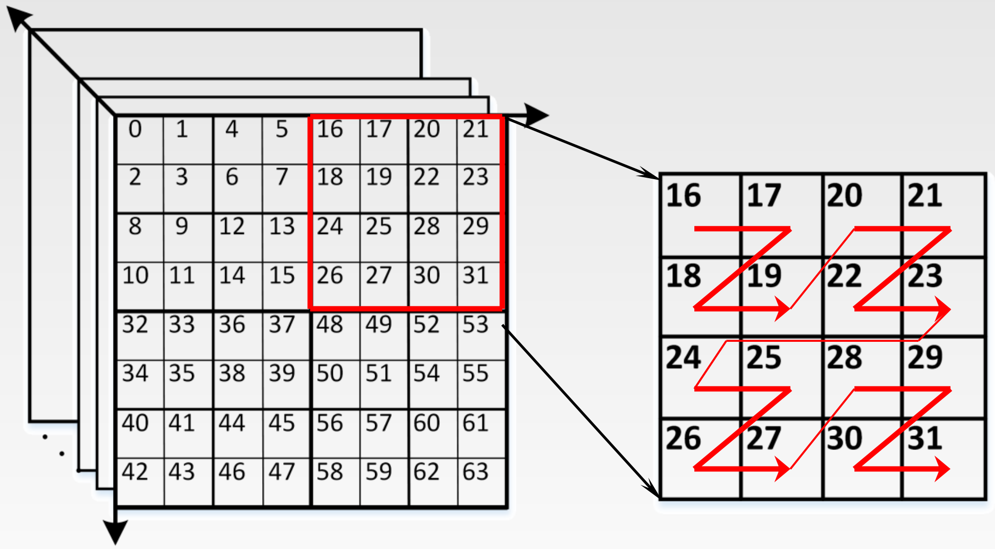
facing XY

facing ZY

Perpendicular

Parallel

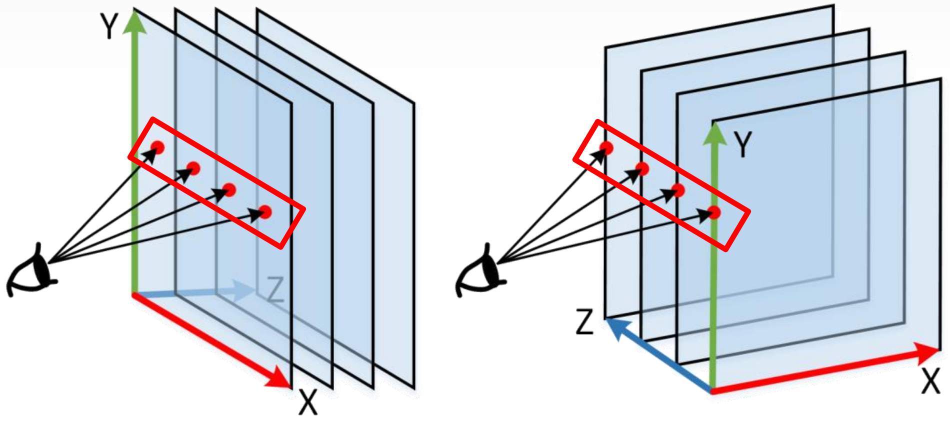
Motivation



0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

16	17	20	21
18	19	22	23
24	25	28	29
26	27	30	31

Viewing Direction



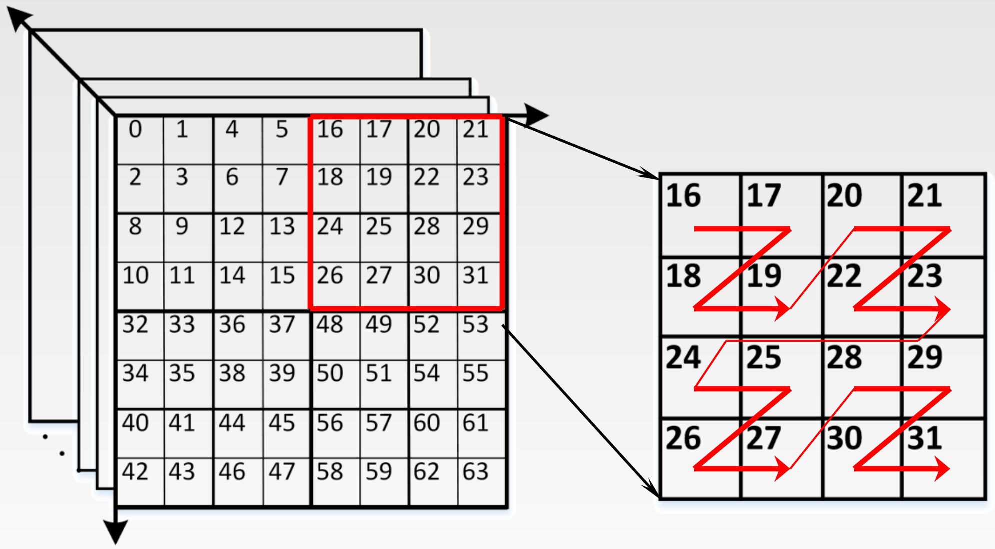
facing XY

Perpendicular

facing ZY

Parallel

Motivation

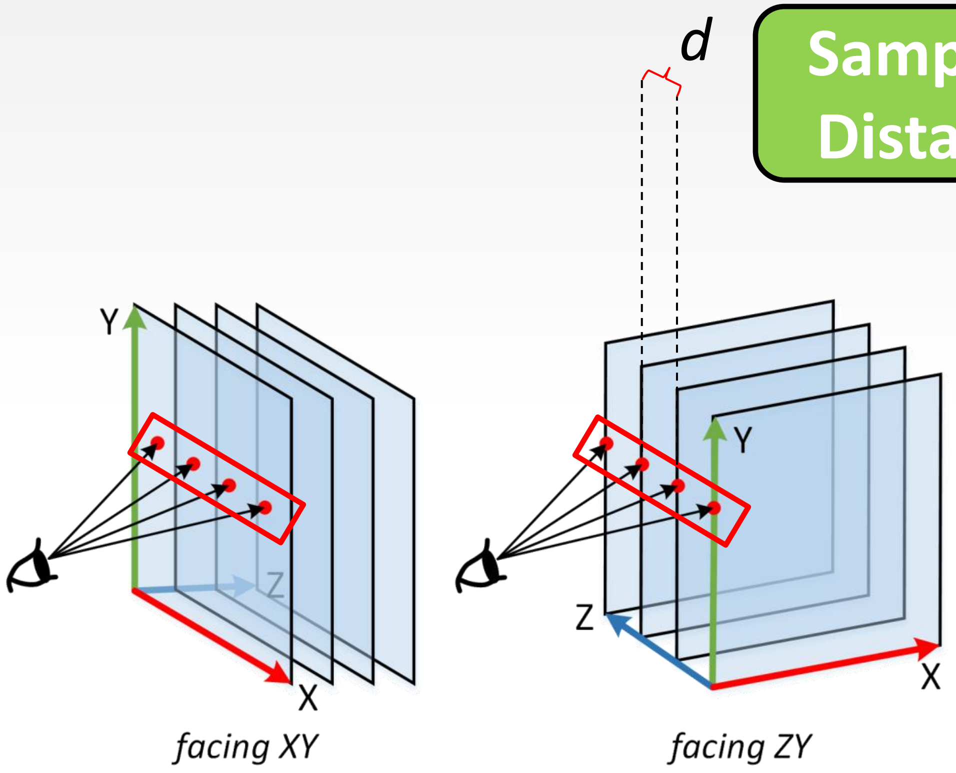


0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

16	17	20	21
18	19	22	23
24	25	28	29
26	27	30	31

Sampling Distance

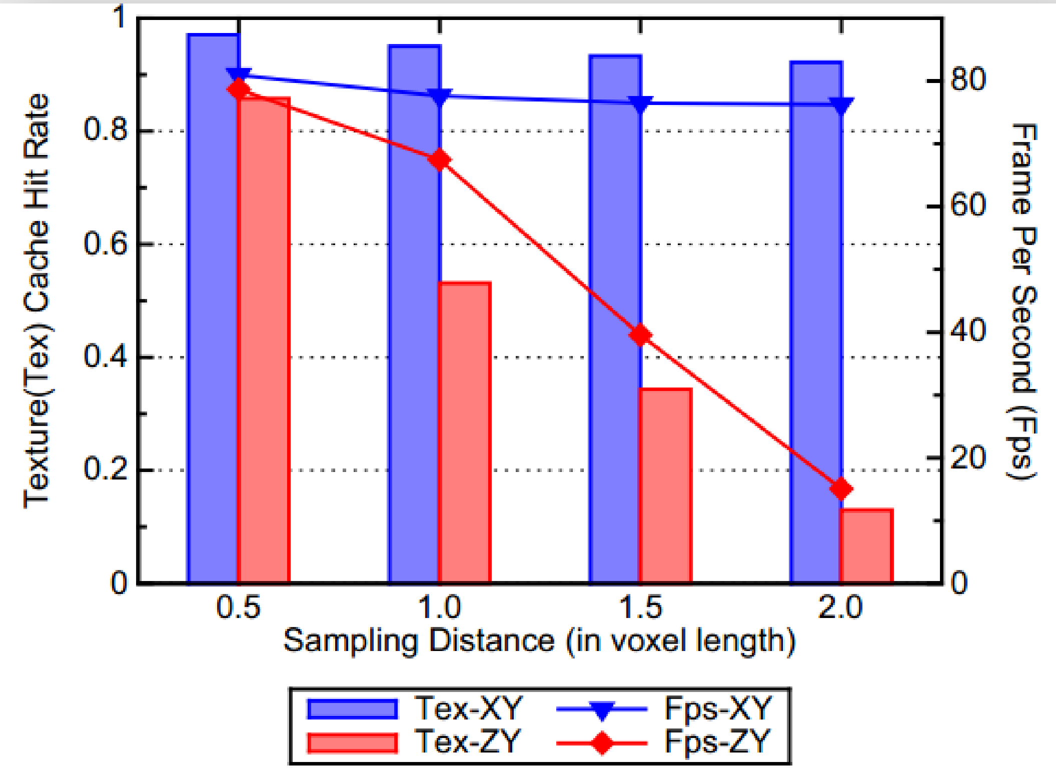
Viewing Direction



Perpendicular

Parallel

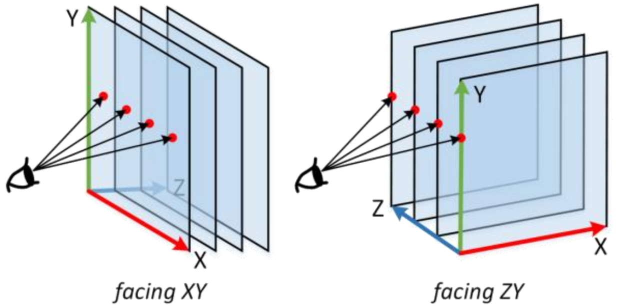
Motivation



GPU: GTX GeForce Titan

Volume size: 1024x1024x1024 x 8bit

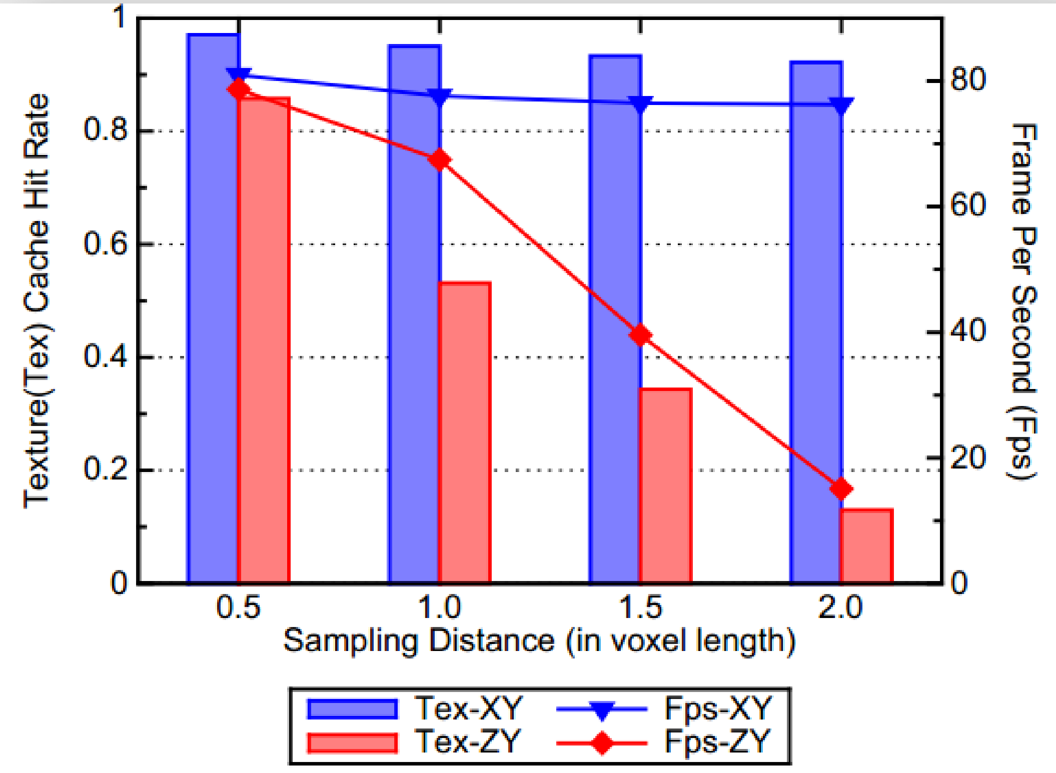
Rendered image size: 512x512



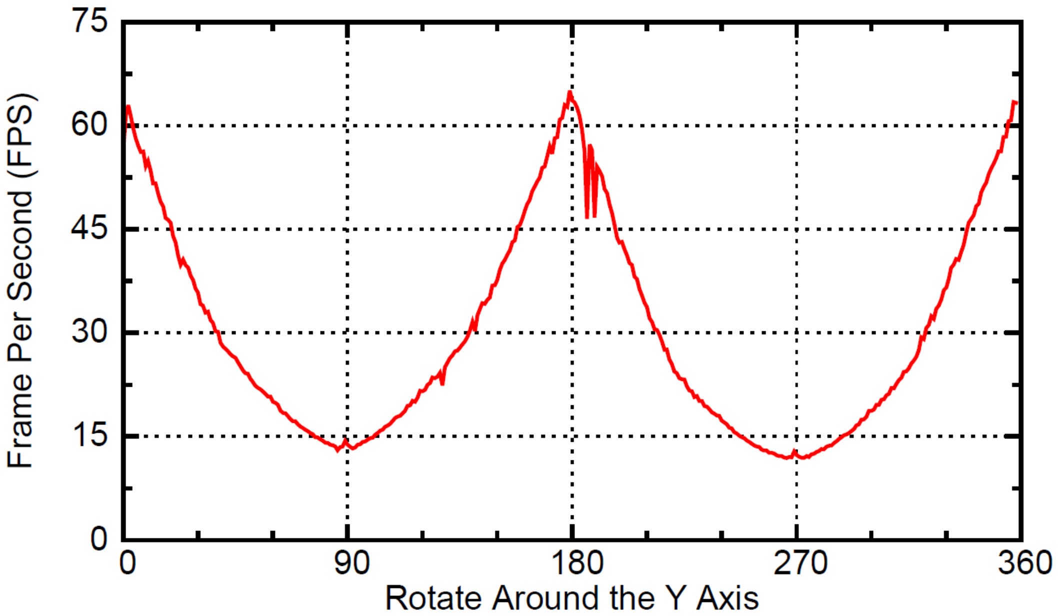
Perpendicular

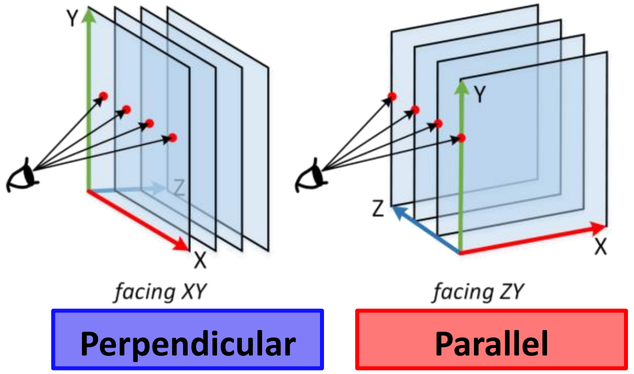
Parallel

Motivation



GPU: GTX GeForce Titan
Volume size: 1024x1024x1024 x 8bit
Rendered image size: 512x512





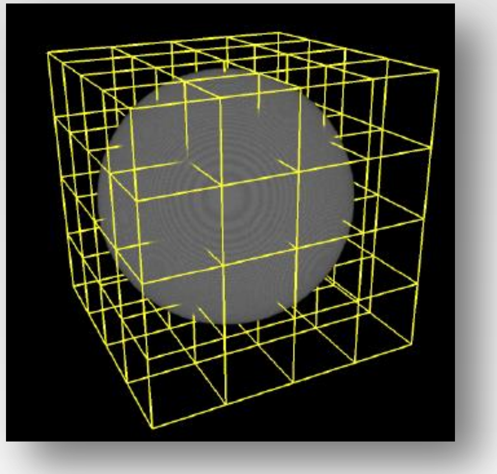
Related Work

Motivation
Related Work
Contribution
Result
Application

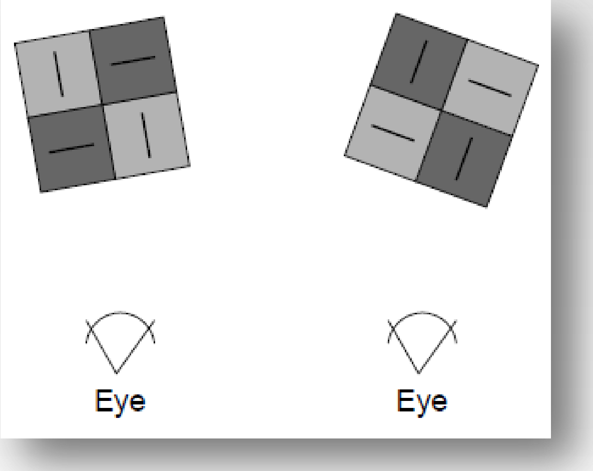
Motivation
Related Work
Contribution
Result
Application

Related Work

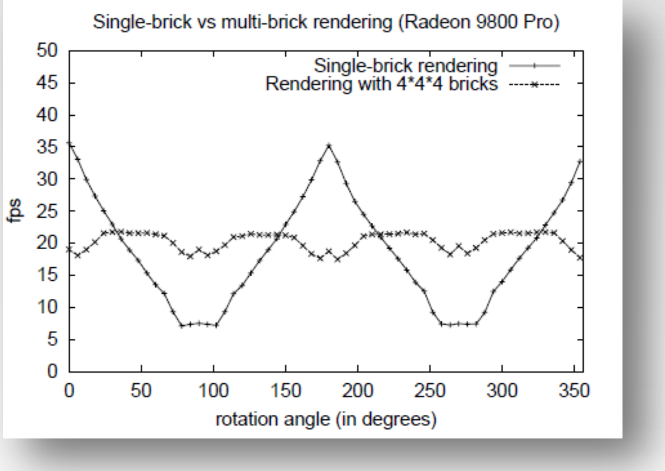
[Weiskopf04]



Partitioning a volume into small bricks



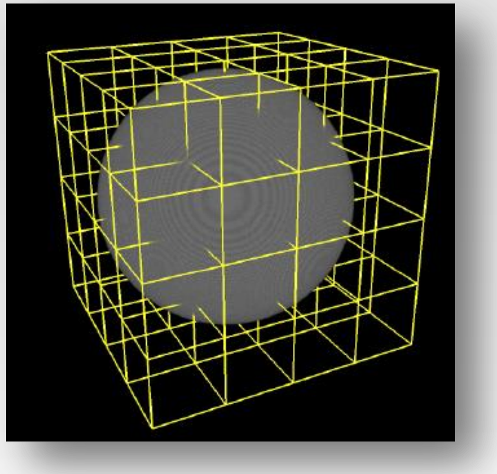
For any direction, 2 bricks are parallel and two bricks are perpendicular to the view



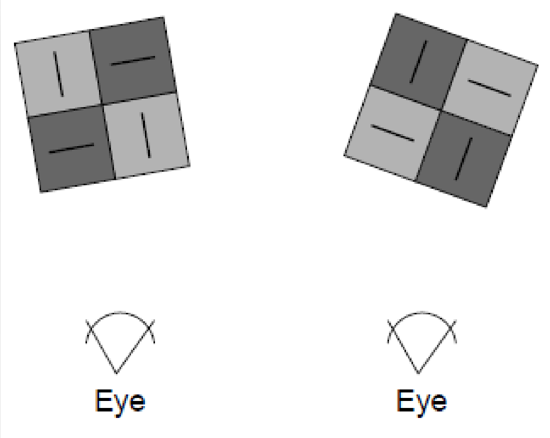
Achieve a roughly constant frame rate when rotating around the Y axis

Related Work

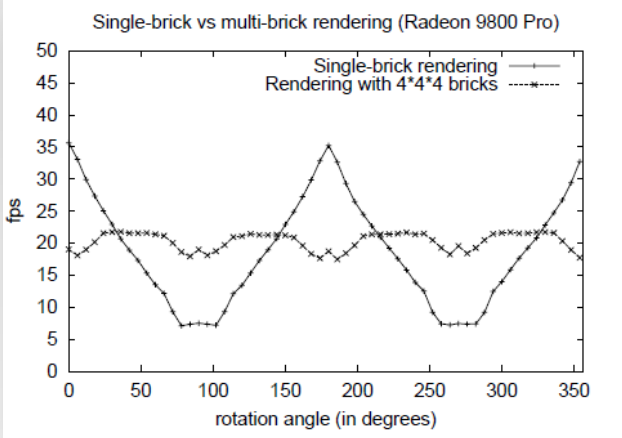
[Weiskopf04]



Partitioning a volume into small bricks

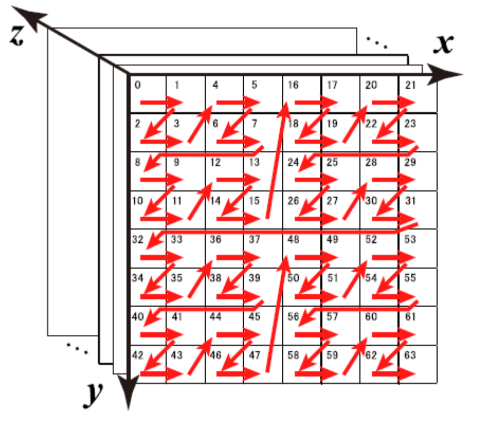


For any direction, 2 bricks are parallel and two bricks are perpendicular to the view



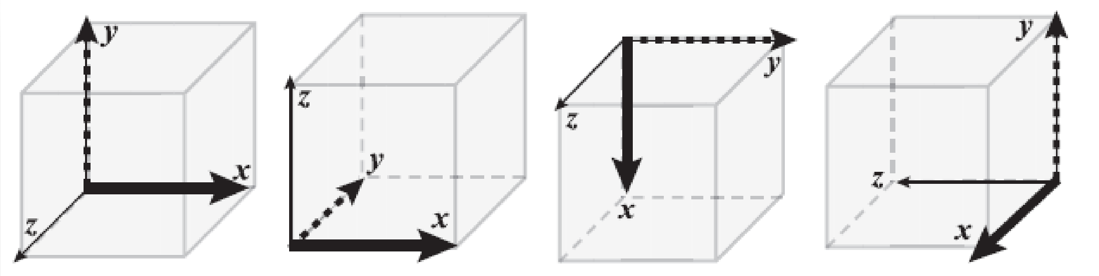
Achieve a roughly constant frame rate when rotating around the Y axis

[Sugimoto2012]
[Sugimoto2014]



Memory stride ratio of the 3D texture along X, Y and Z axis is **1:2:6**

GPU WARP



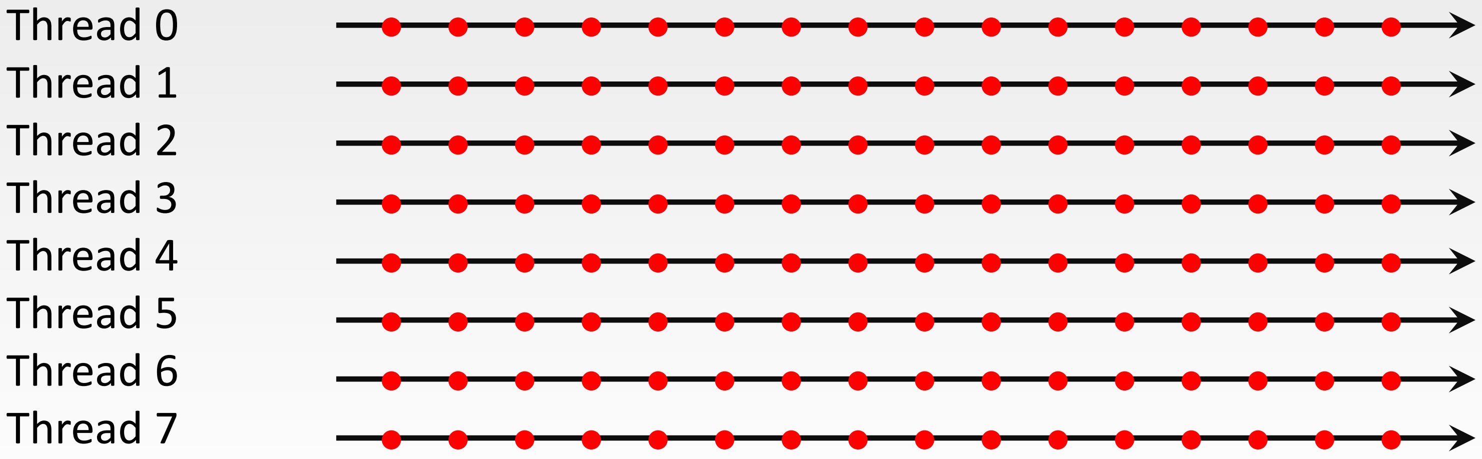
A warp of GPU threads should always take samples along the direction with smaller stride, so that higher cache locality could be achieved

Contribution

We are trying to improve the texture cache performance by minimizing the memory stride inside a WARP of GPU threads

Contribution

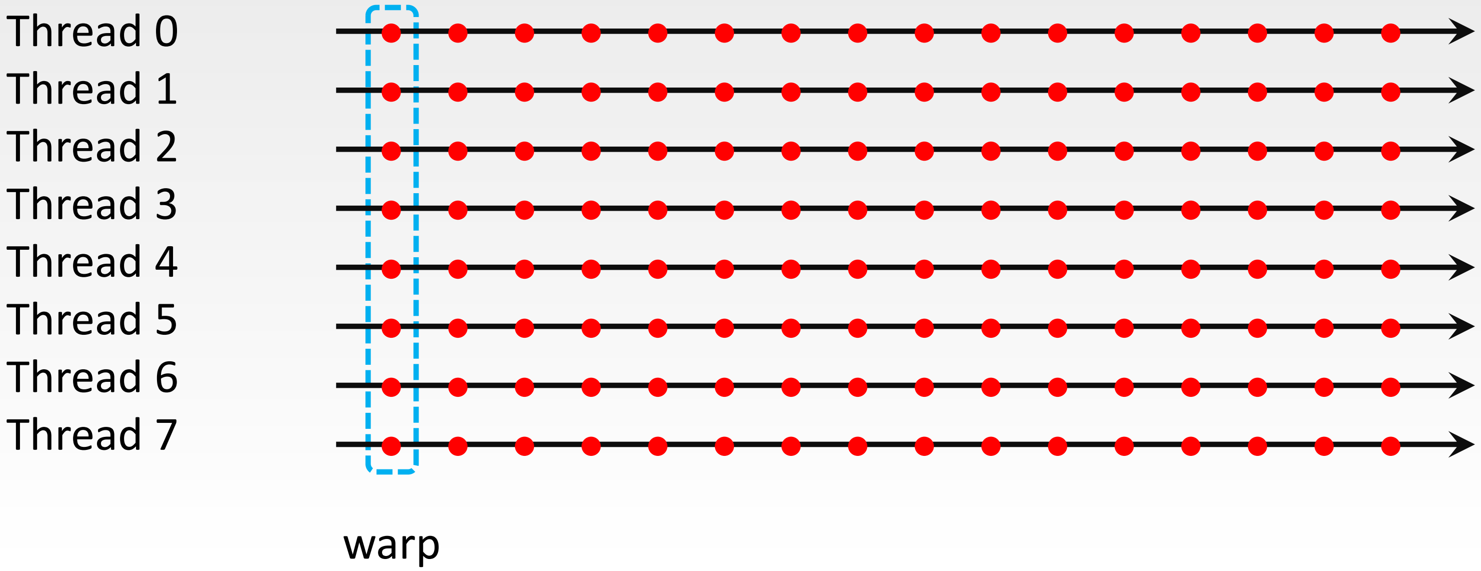
Map one thread to one ray (warp size = 8)



Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

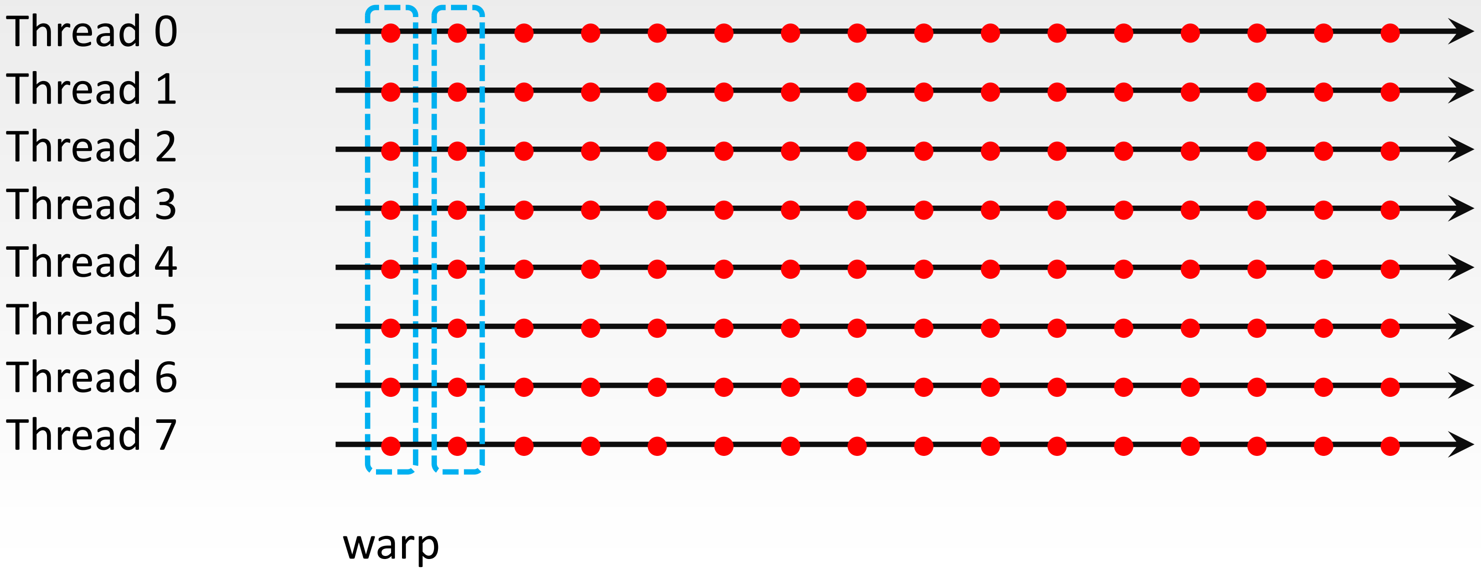
Contribution

Map one thread to one ray (warp size = 8)



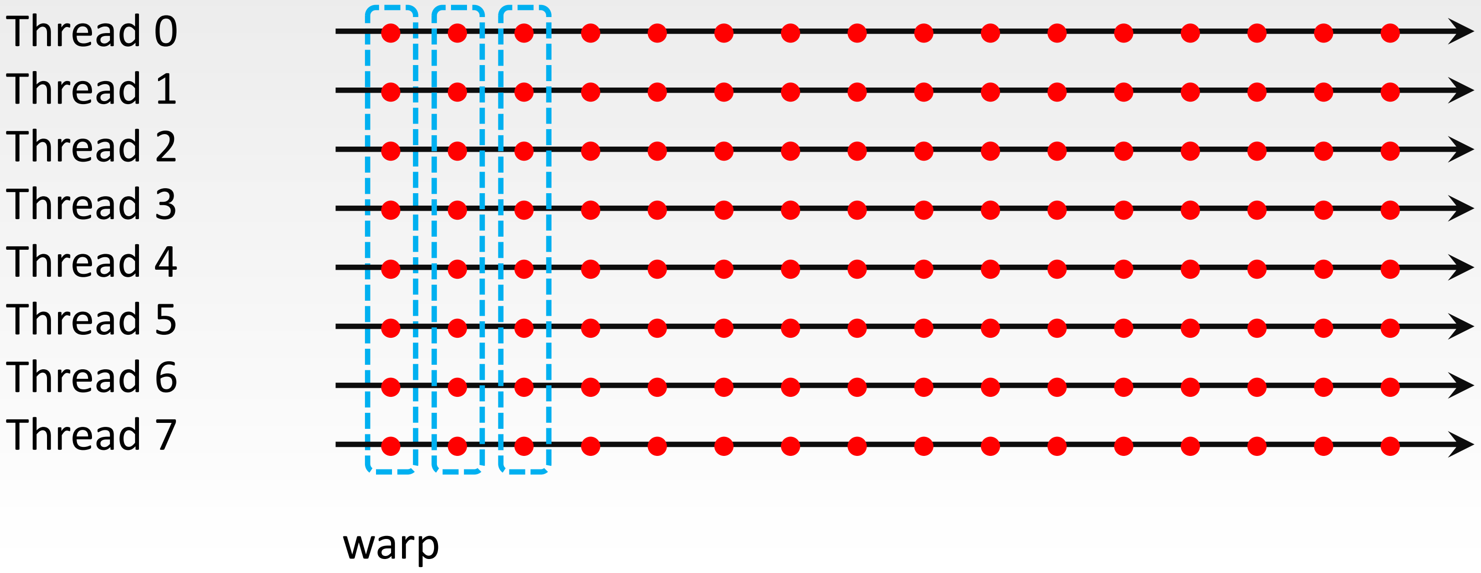
Contribution

Map one thread to one ray (warp size = 8)



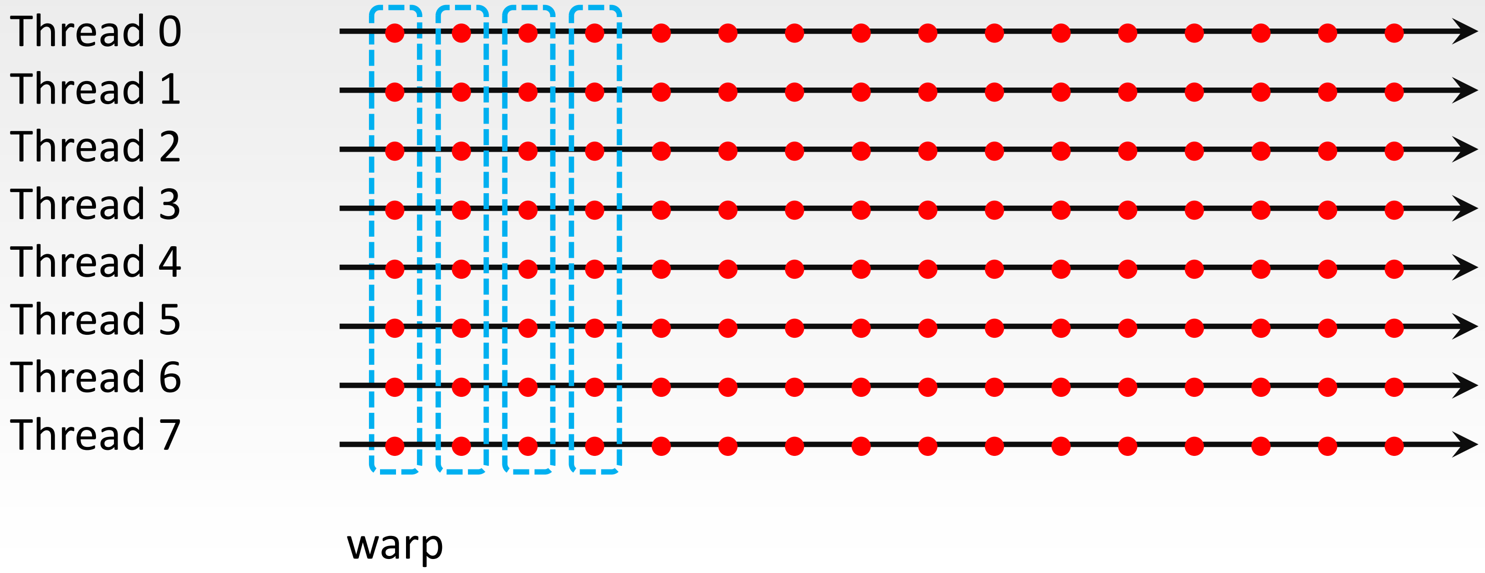
Contribution

Map one thread to one ray (warp size = 8)



Contribution

Map one thread to one ray (warp size = 8)

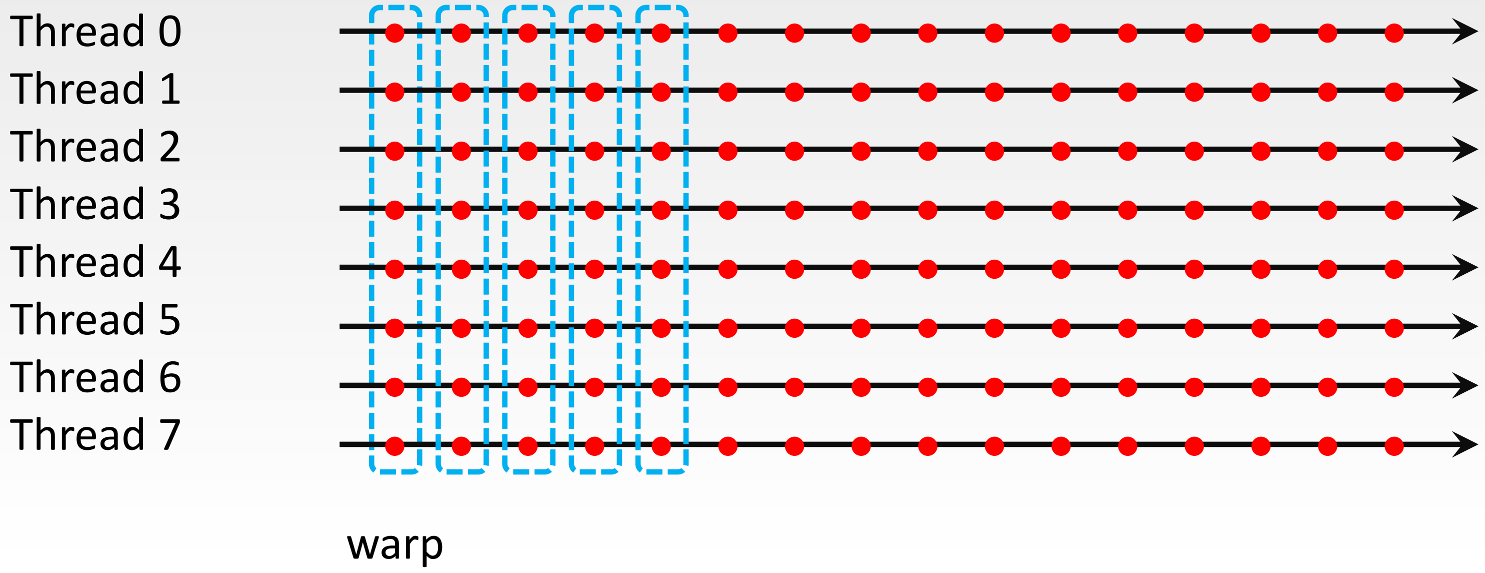


Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

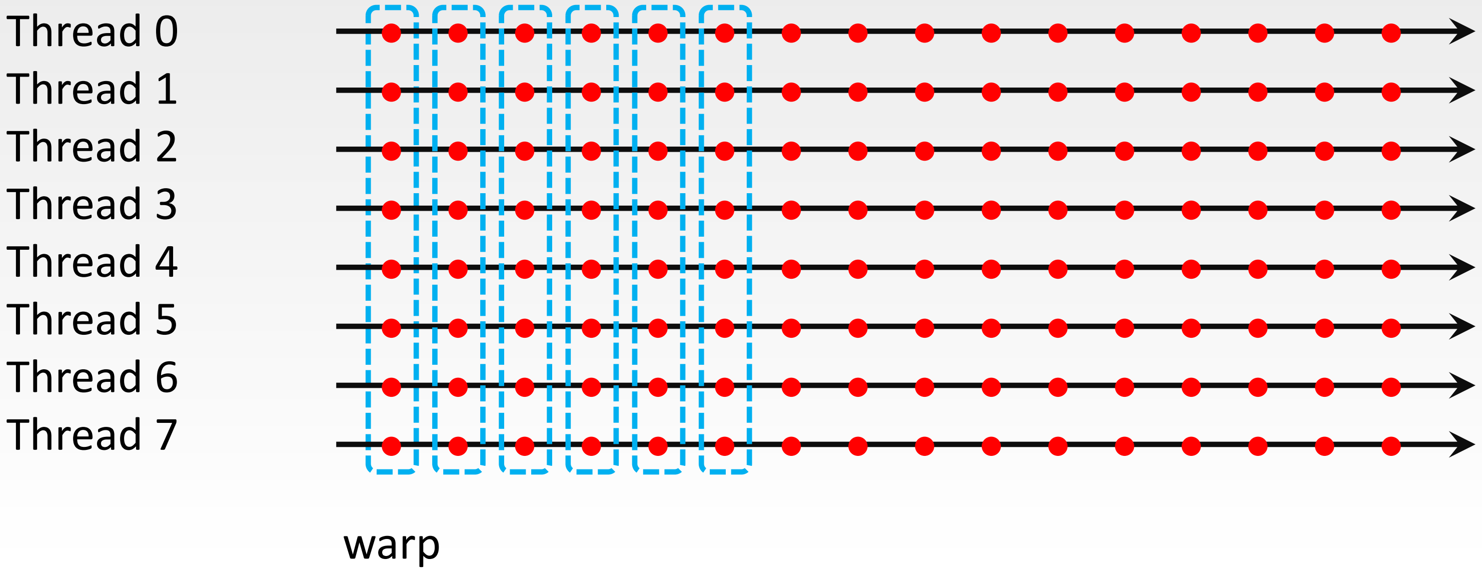
Contribution

Map one thread to one ray (warp size = 8)



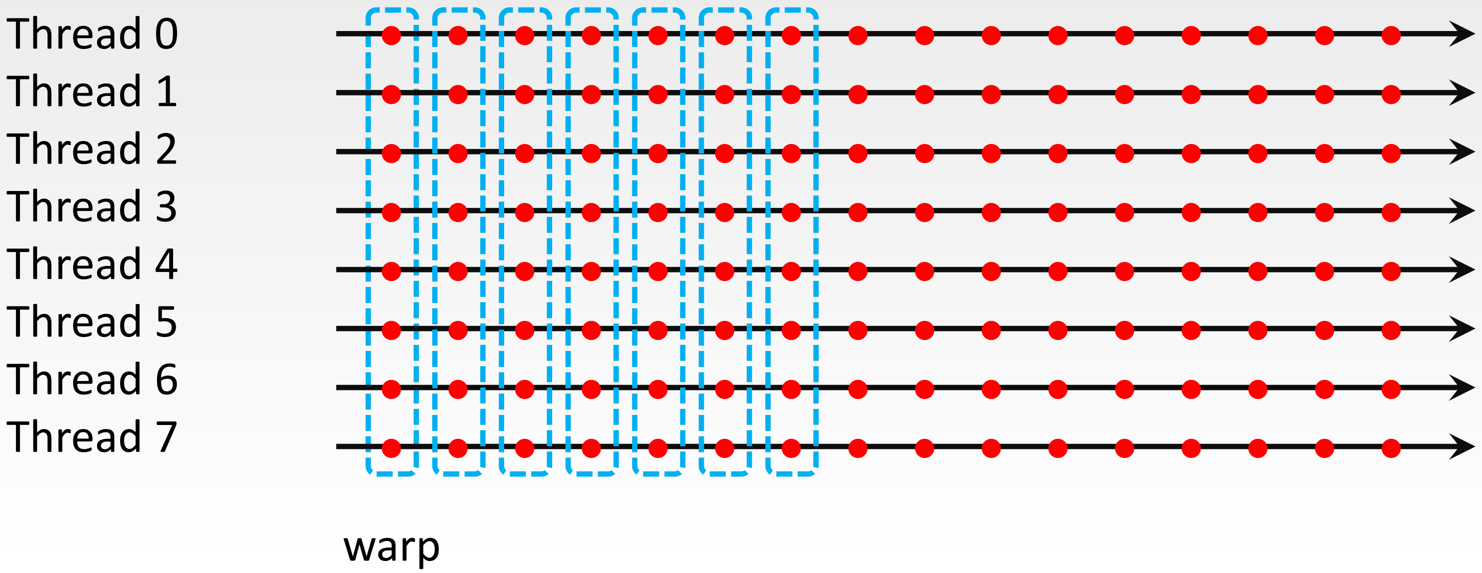
Contribution

Map one thread to one ray (warp size = 8)



Contribution

Map one thread to one ray (warp size = 8)

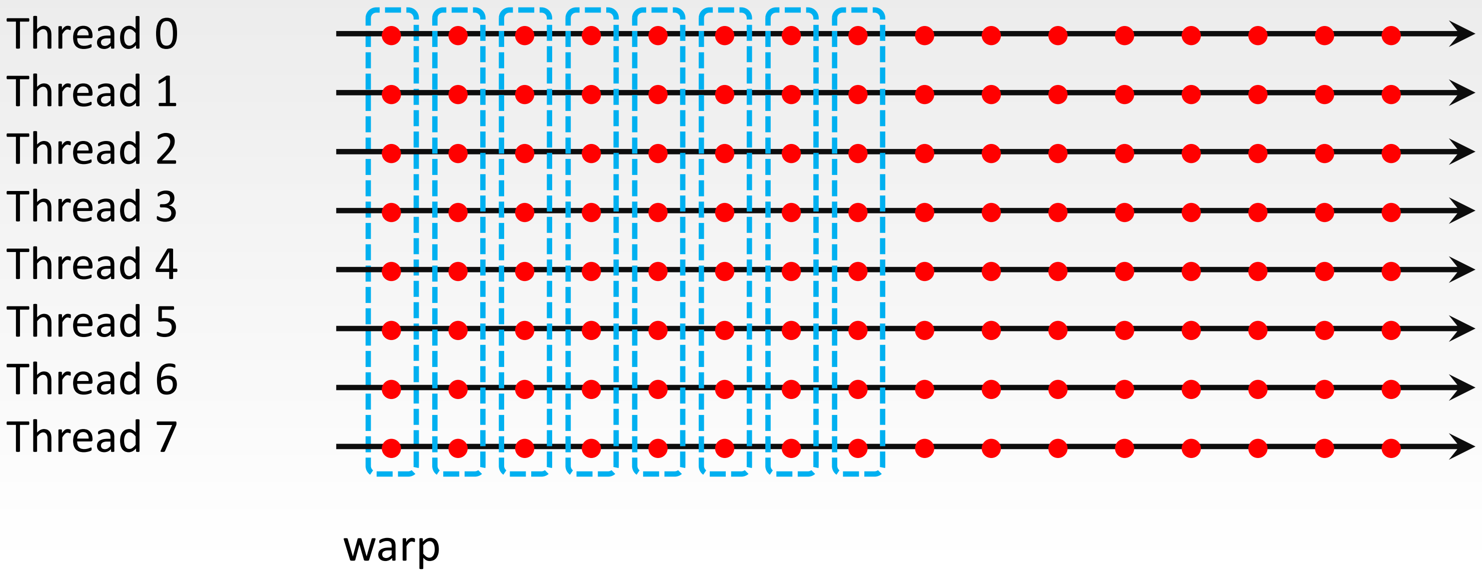


Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

Contribution

Map one thread to one ray (warp size = 8)

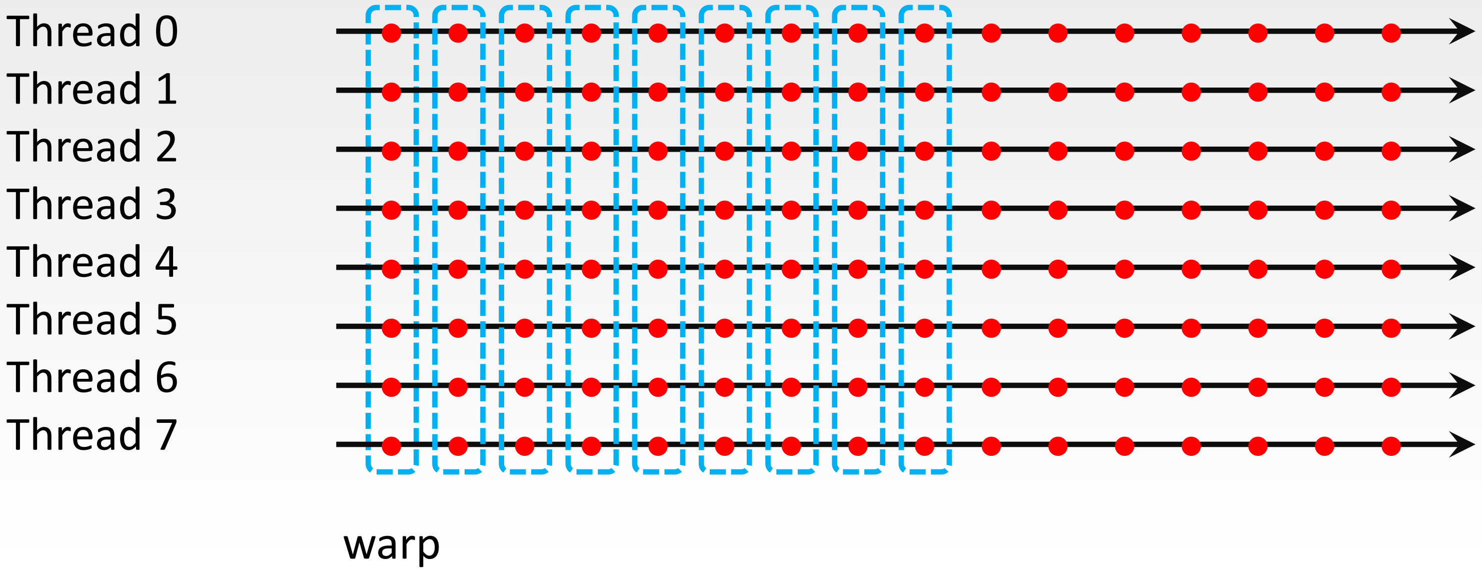


Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

Contribution

Map one thread to one ray (warp size = 8)



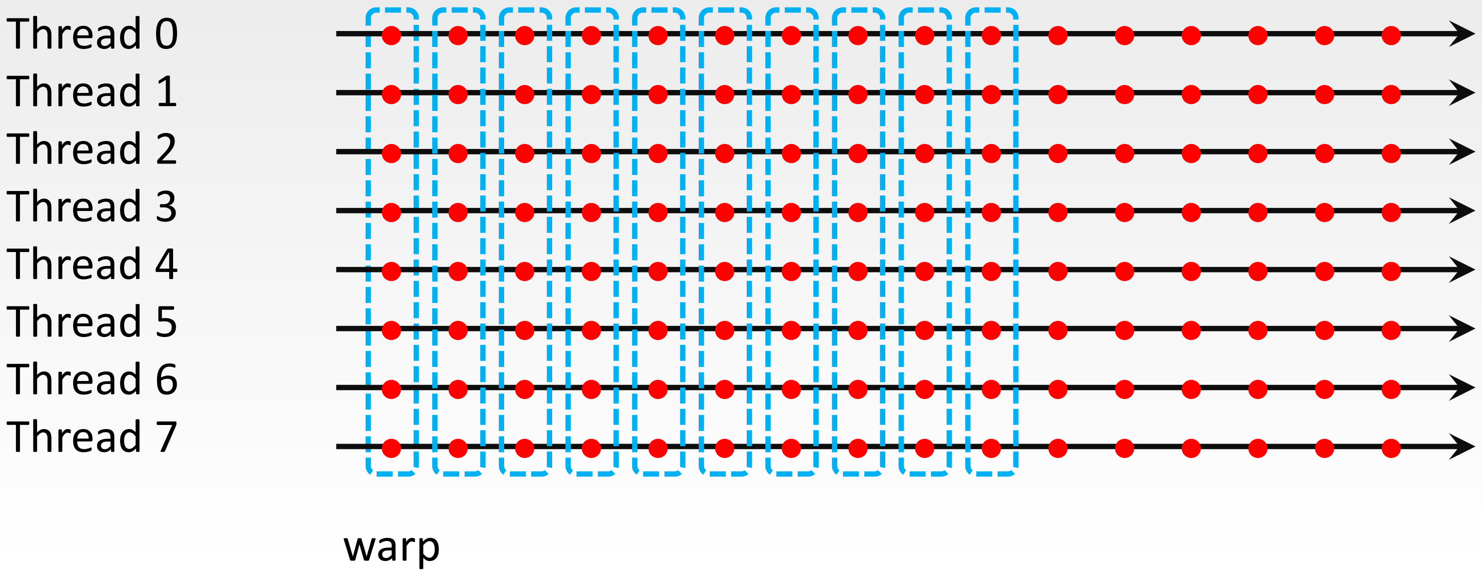
Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

Motivation
Related Work
Contribution
Result
Application

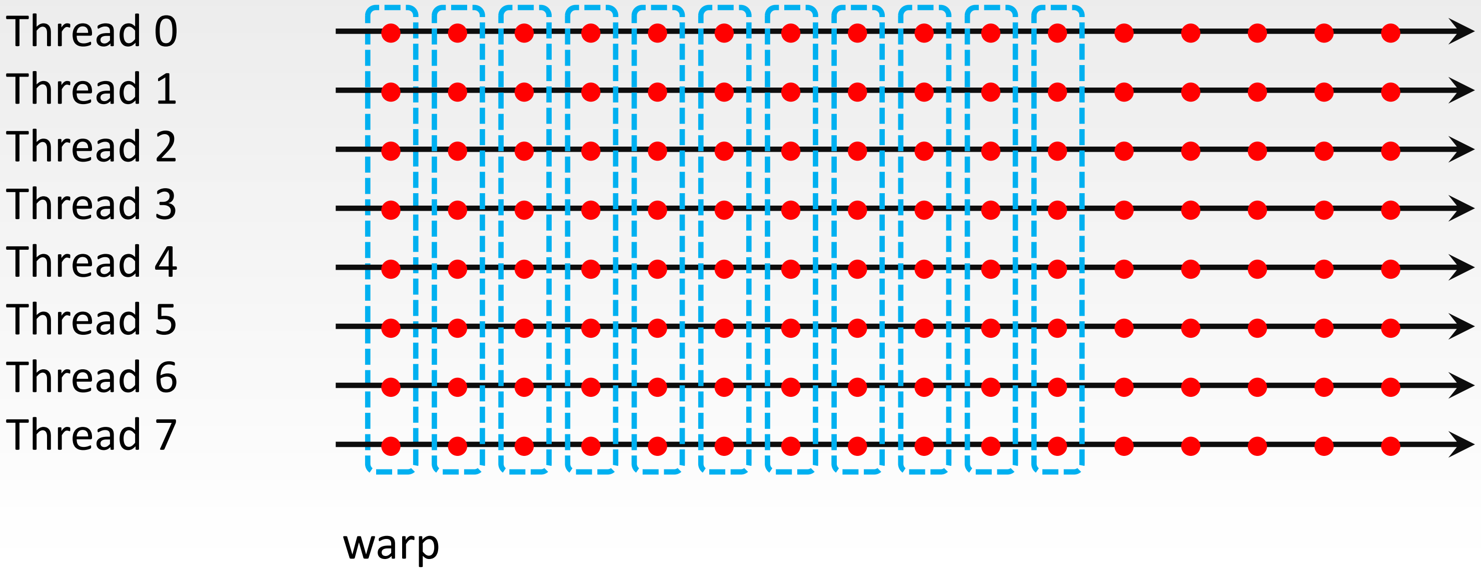
Contribution

Map one thread to one ray (warp size = 8)



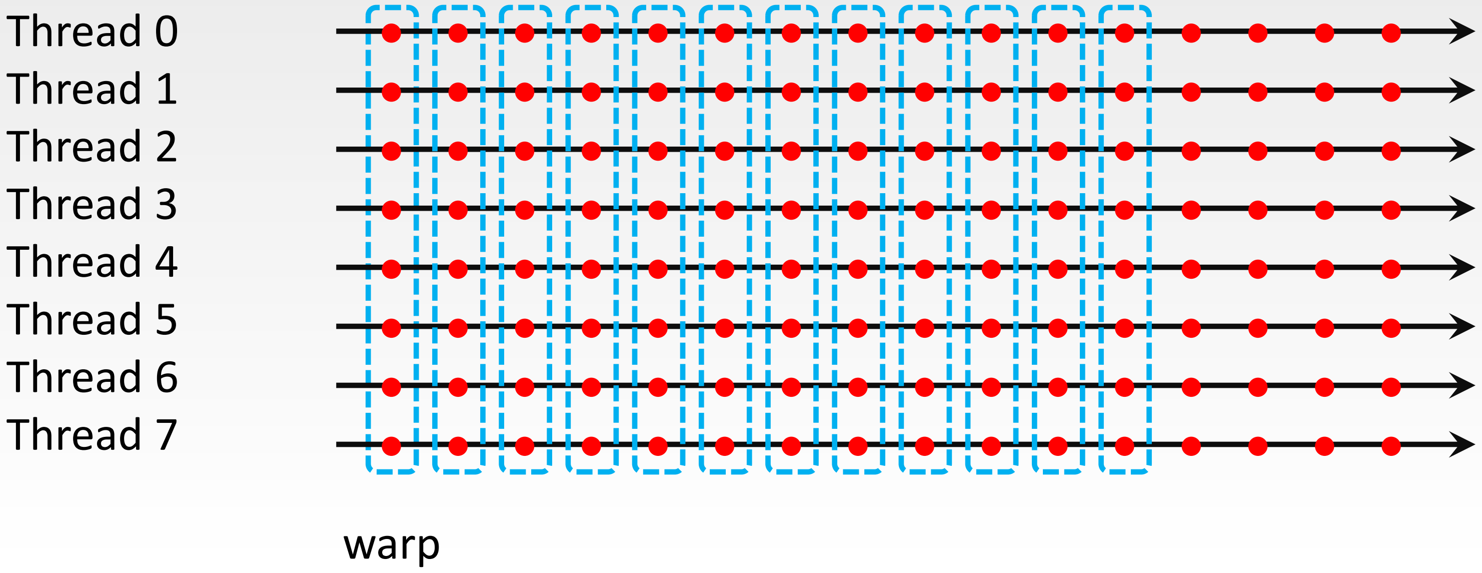
Contribution

Map one thread to one ray (warp size = 8)



Contribution

Map one thread to one ray (warp size = 8)

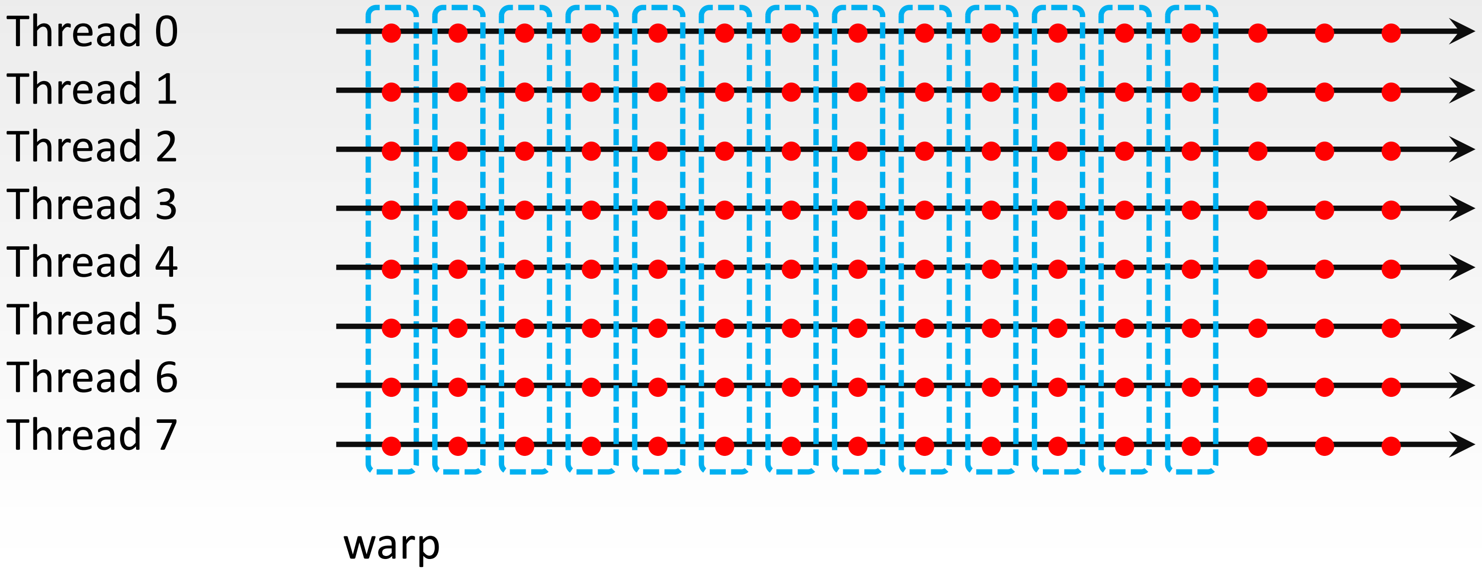


Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

Contribution

Map one thread to one ray (warp size = 8)

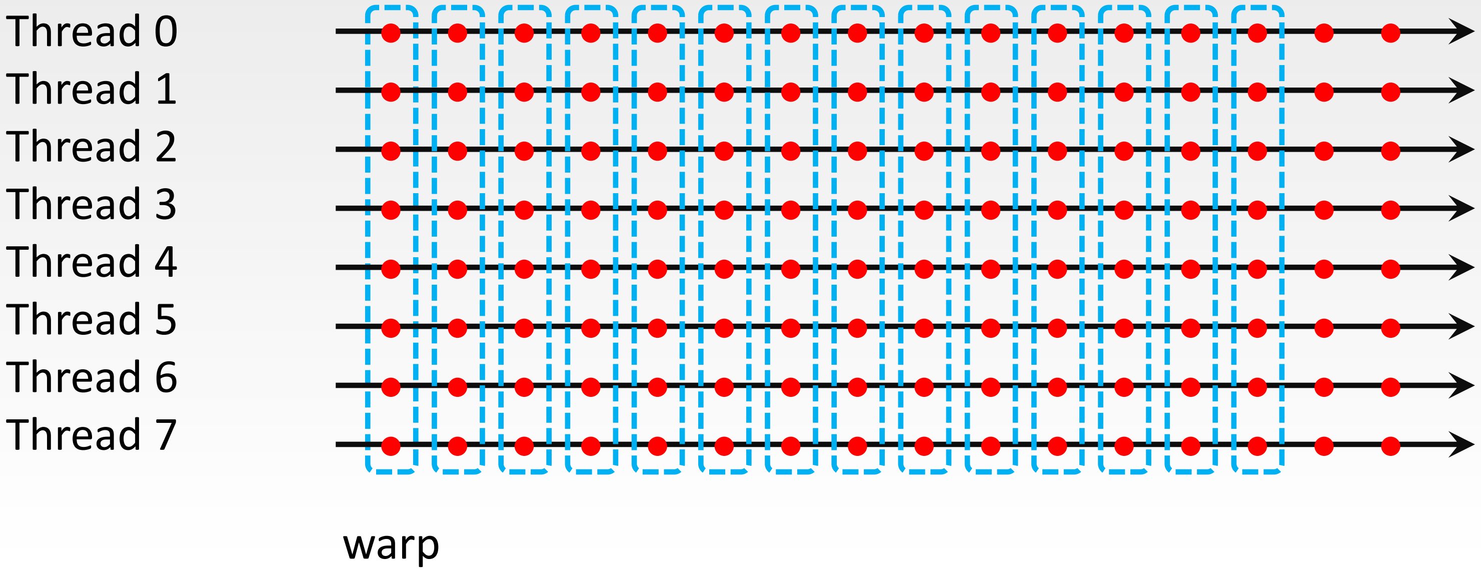


Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

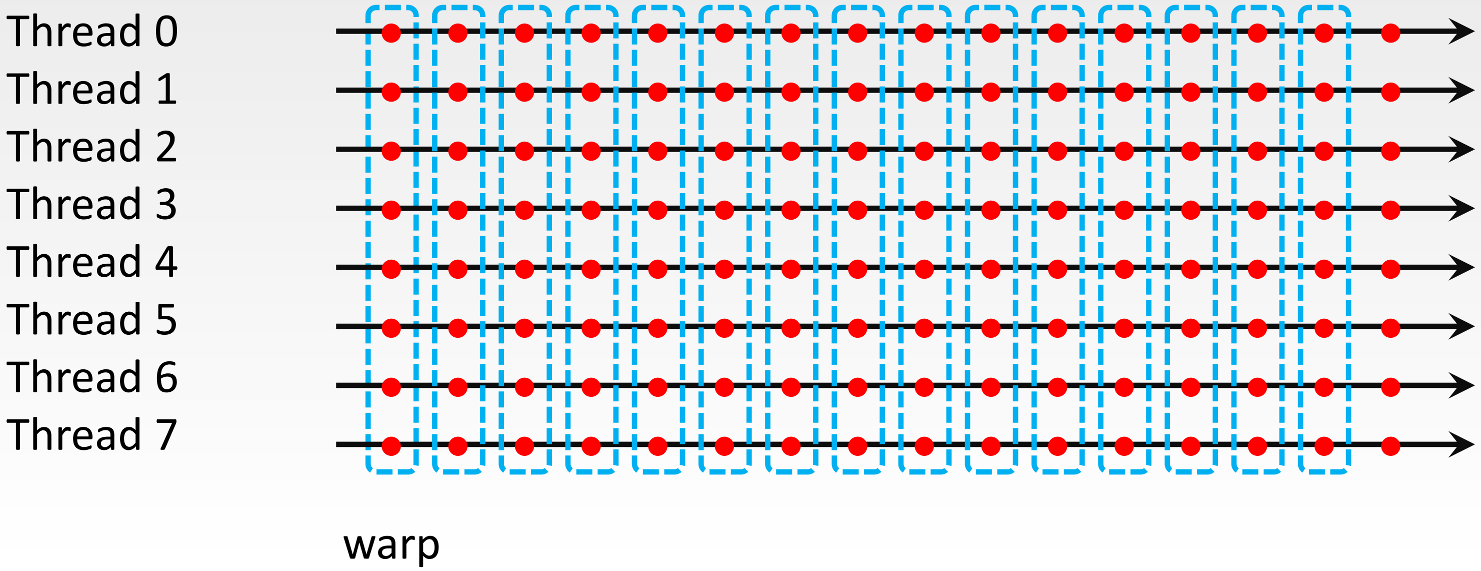
Contribution

Map one thread to one ray (warp size = 8)



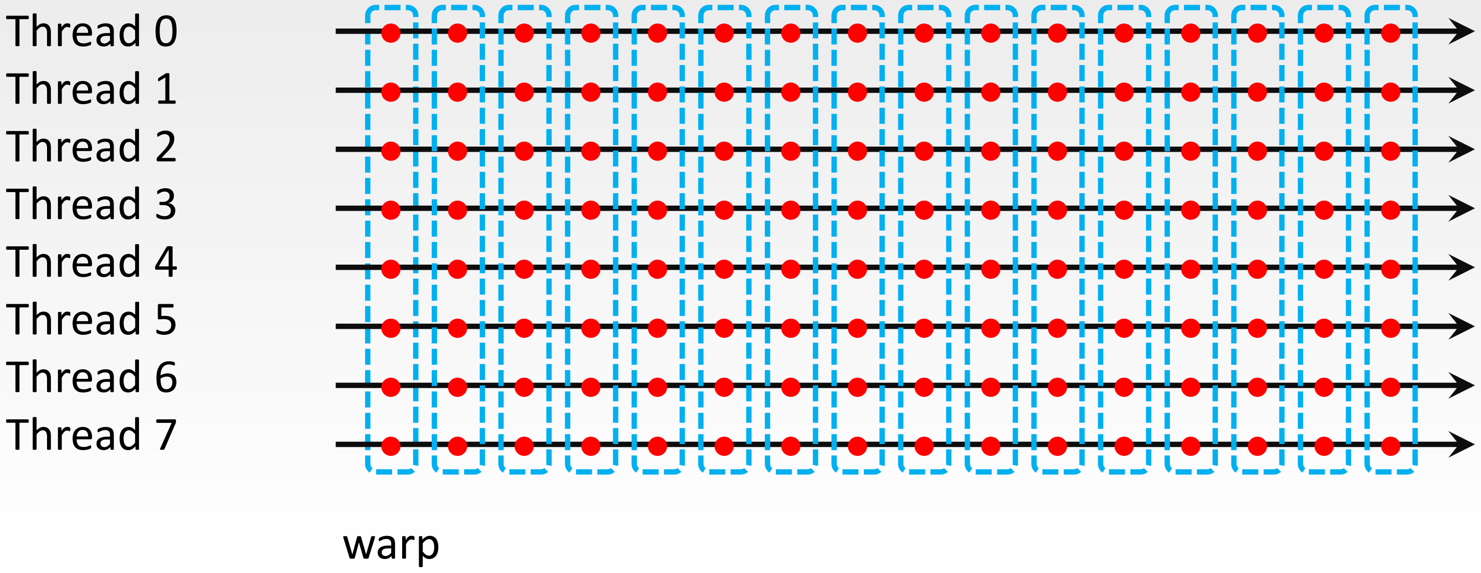
Contribution

Map one thread to one ray (warp size = 8)



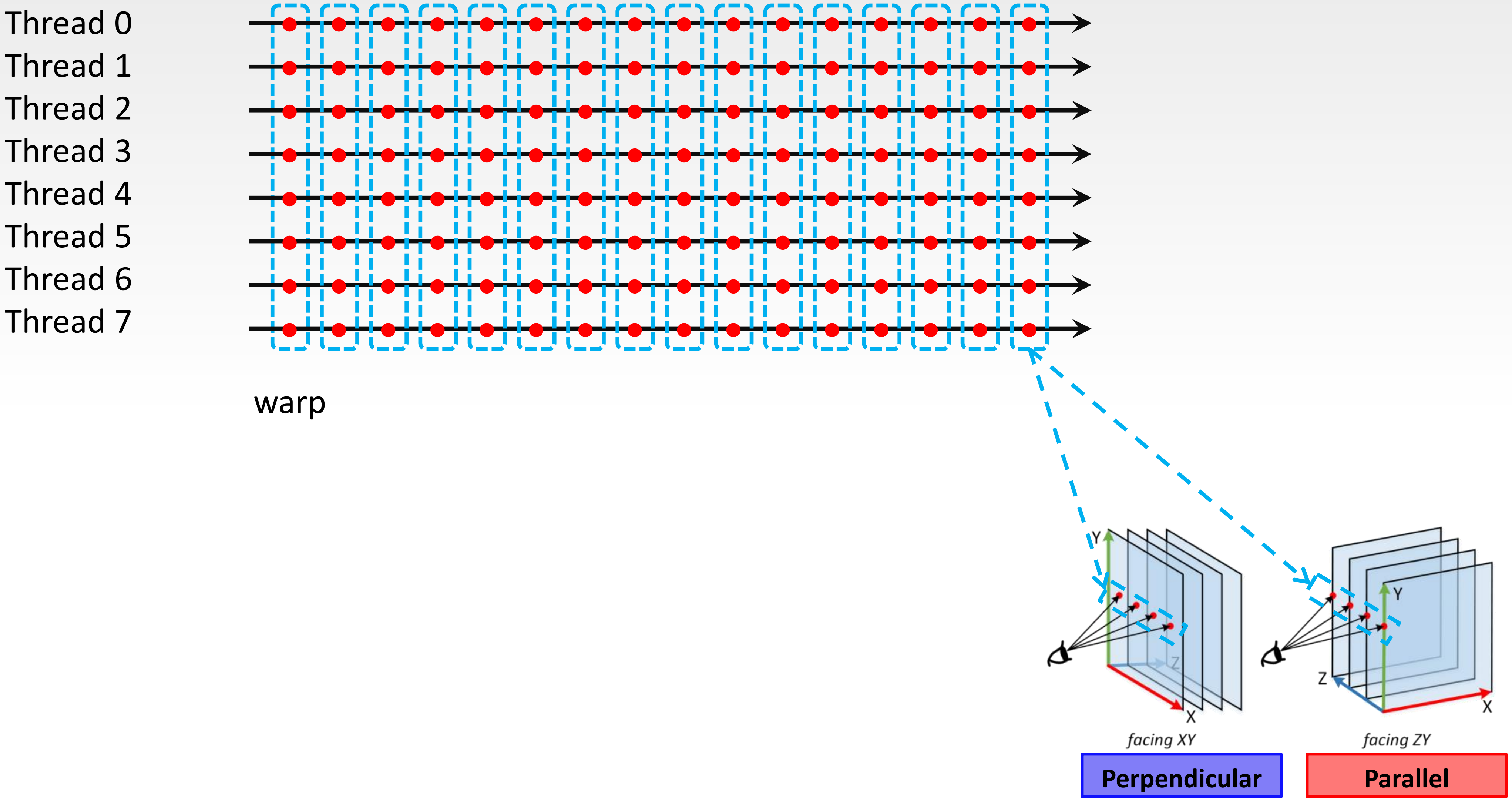
Contribution

Map one thread to one ray (warp size = 8)



Contribution

Map one thread to one ray (warp size = 8)



Thread 0
Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7

warp

facing XY

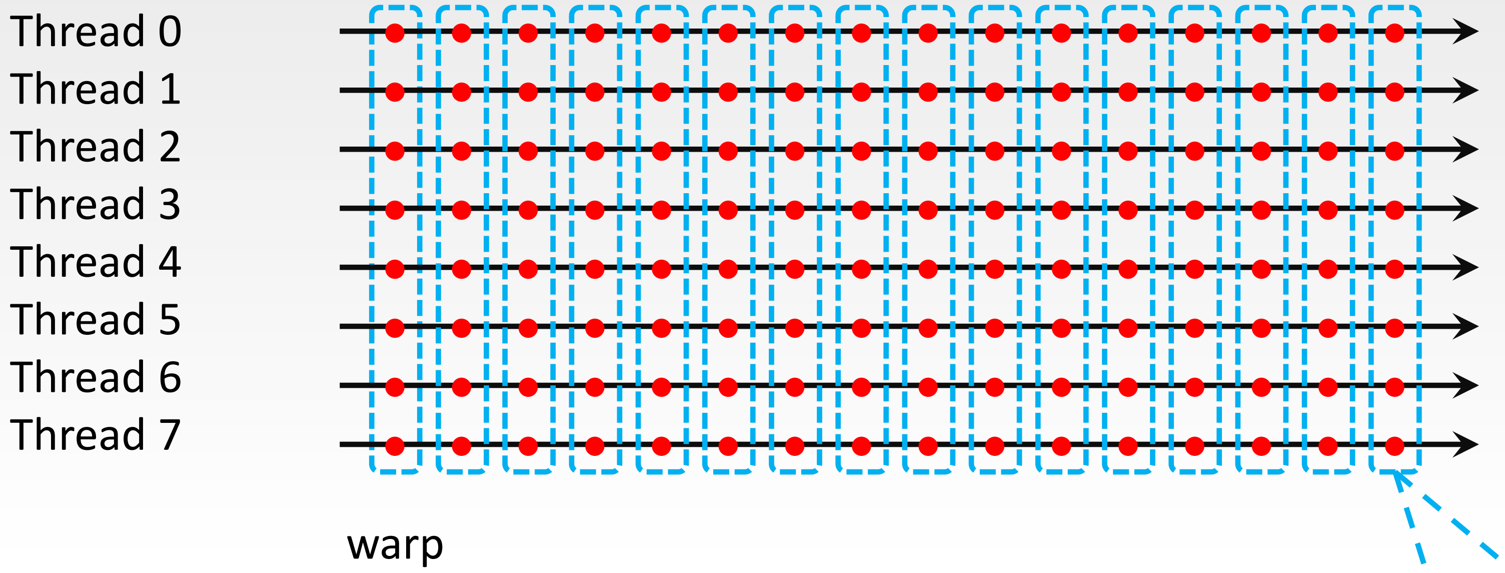
facing ZY

Perpendicular

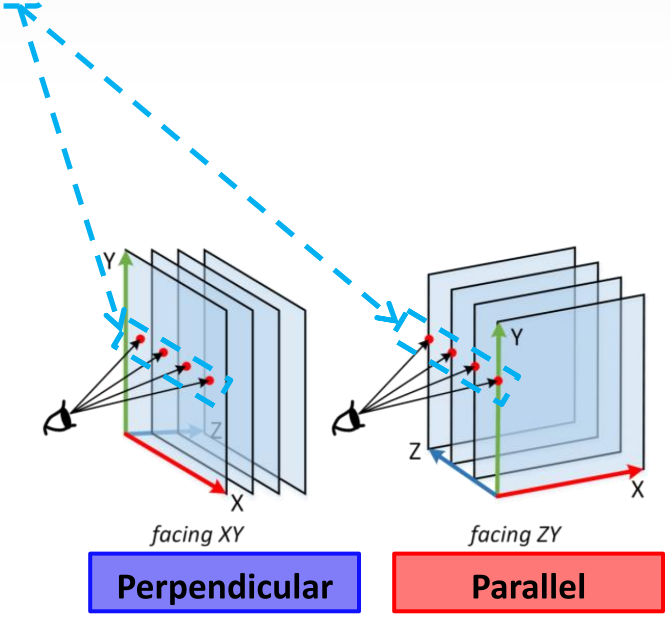
Parallel

Contribution

Map one thread to one ray (warp size = 8)

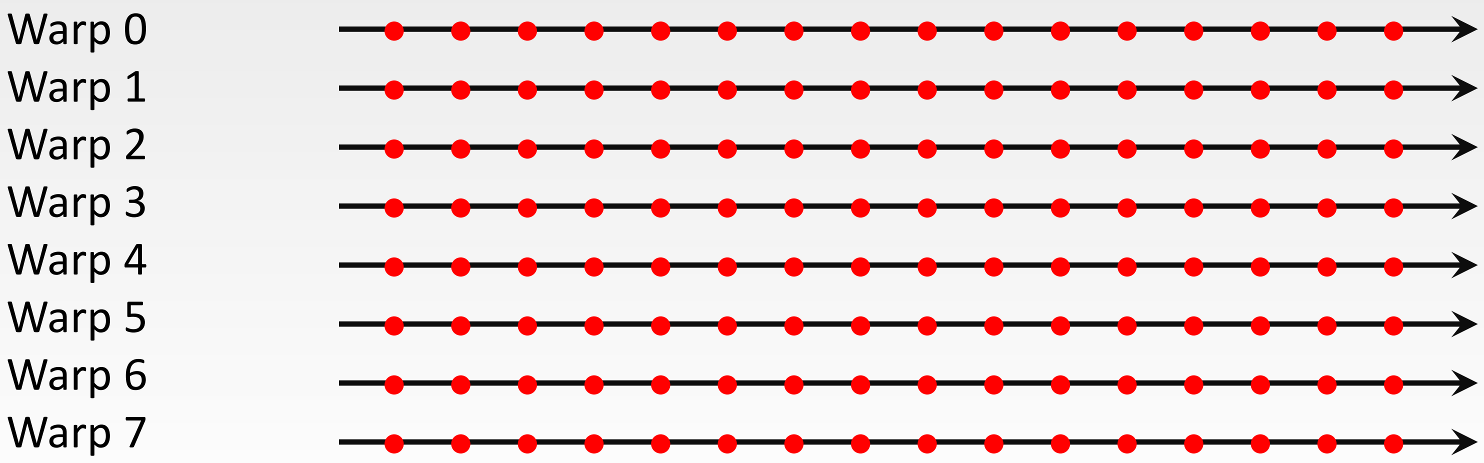


The Standard



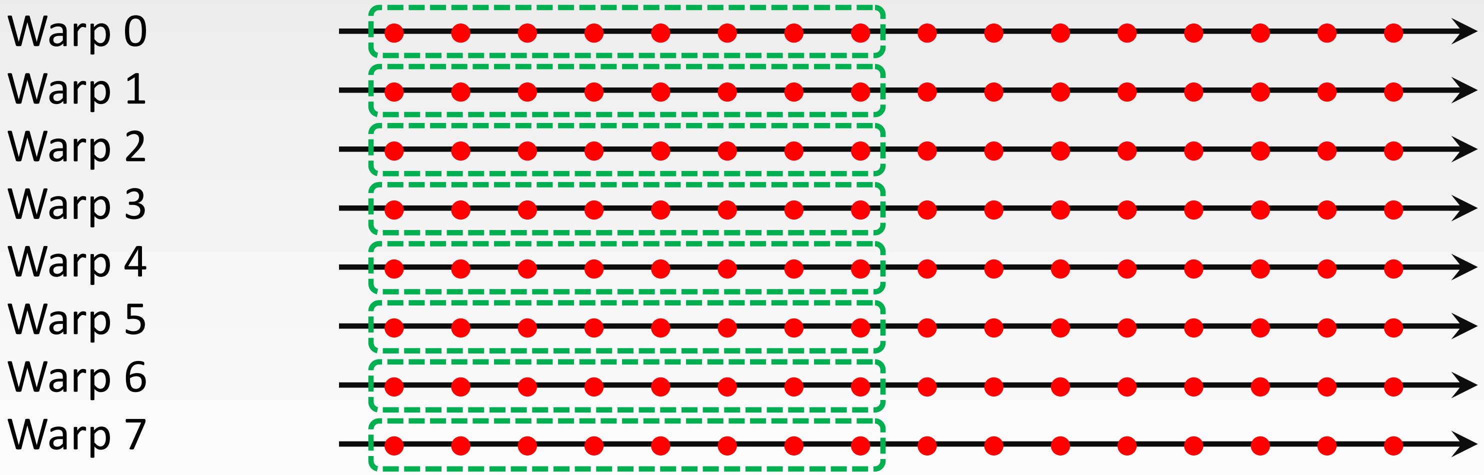
Contribution

Map one warp of threads to one ray



Contribution

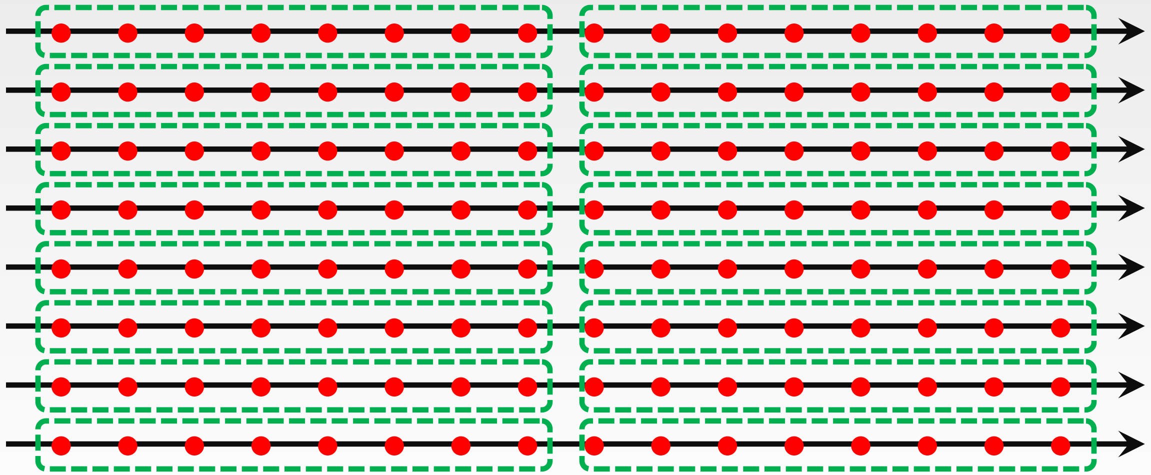
Map one warp of threads to one ray



Warp 0
Warp 1
Warp 2
Warp 3
Warp 4
Warp 5
Warp 6
Warp 7

Contribution

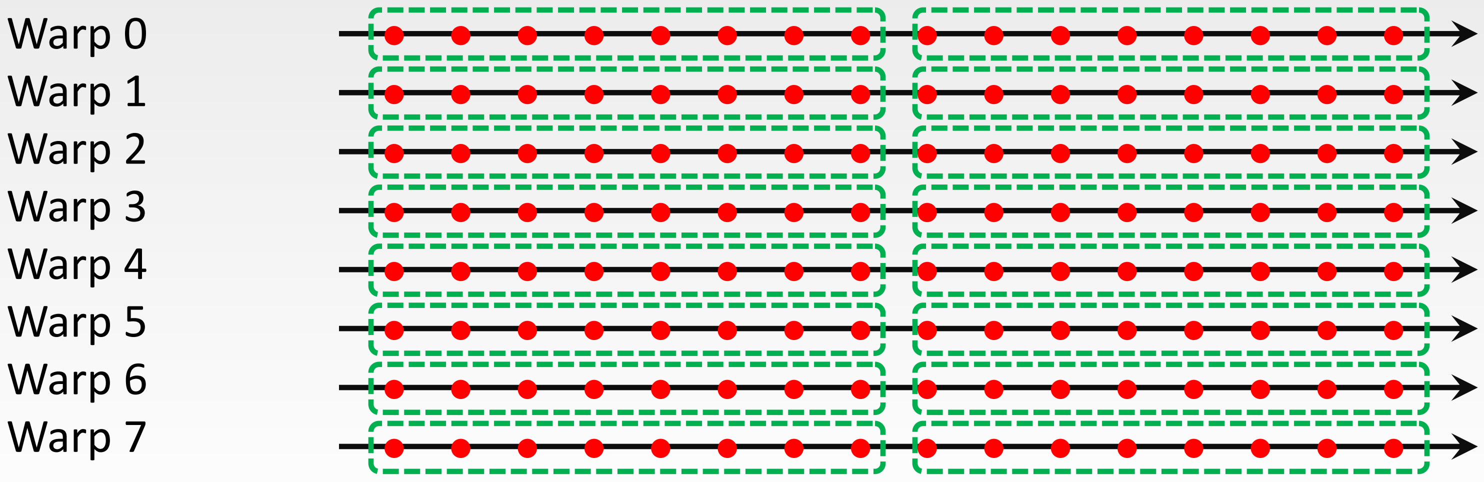
Map one warp of threads to one ray



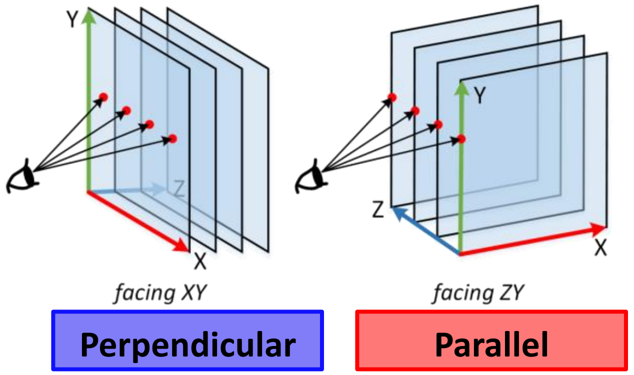
Warp 0
Warp 1
Warp 2
Warp 3
Warp 4
Warp 5
Warp 6
Warp 7

Contribution

Map one warp of threads to one ray



Warp 0
Warp 1
Warp 2
Warp 3
Warp 4
Warp 5
Warp 6
Warp 7



facing XY

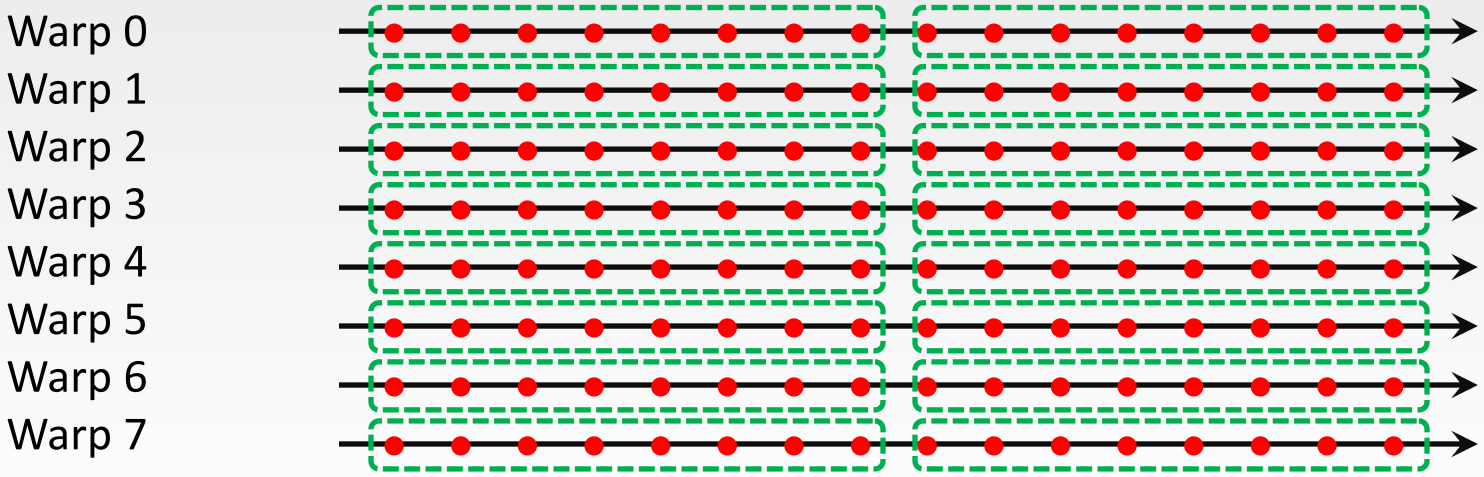
facing ZY

Perpendicular

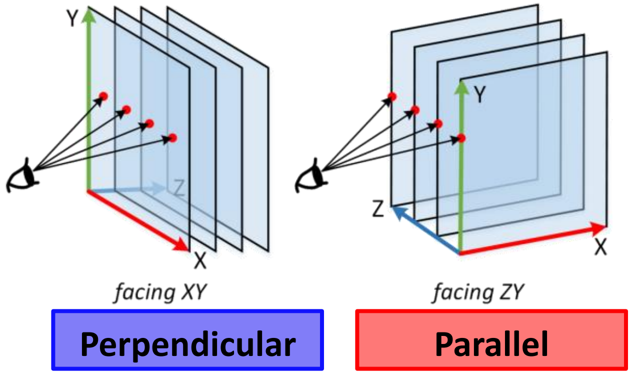
Parallel

Contribution

Map one warp of threads to one ray



Warp Marching

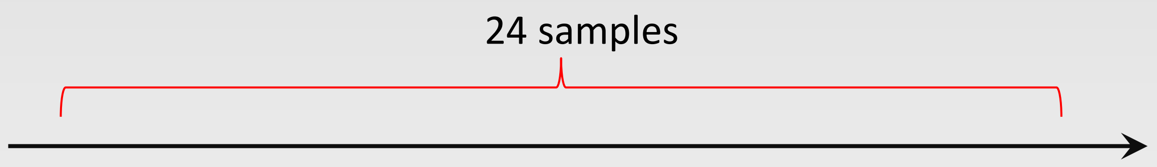


Motivation
Related Work
Contribution
Result
Application

Single Buffer Warp Marching

● *sample*
🟦 *buffer*
🟩 *intermediate result*

Single Buffer Warp Marching



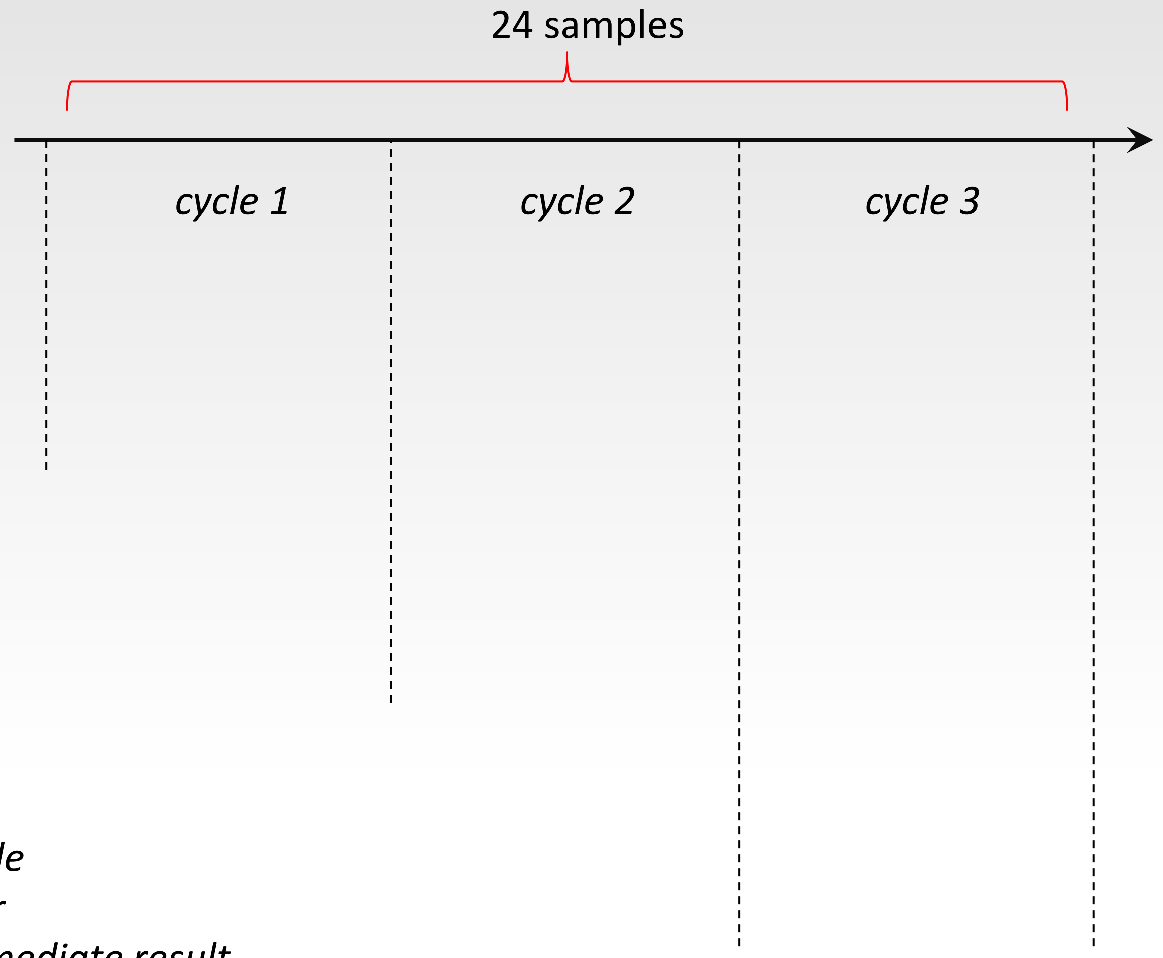
● *sample*

🟦 *buffer*

🟩 *intermediate result*

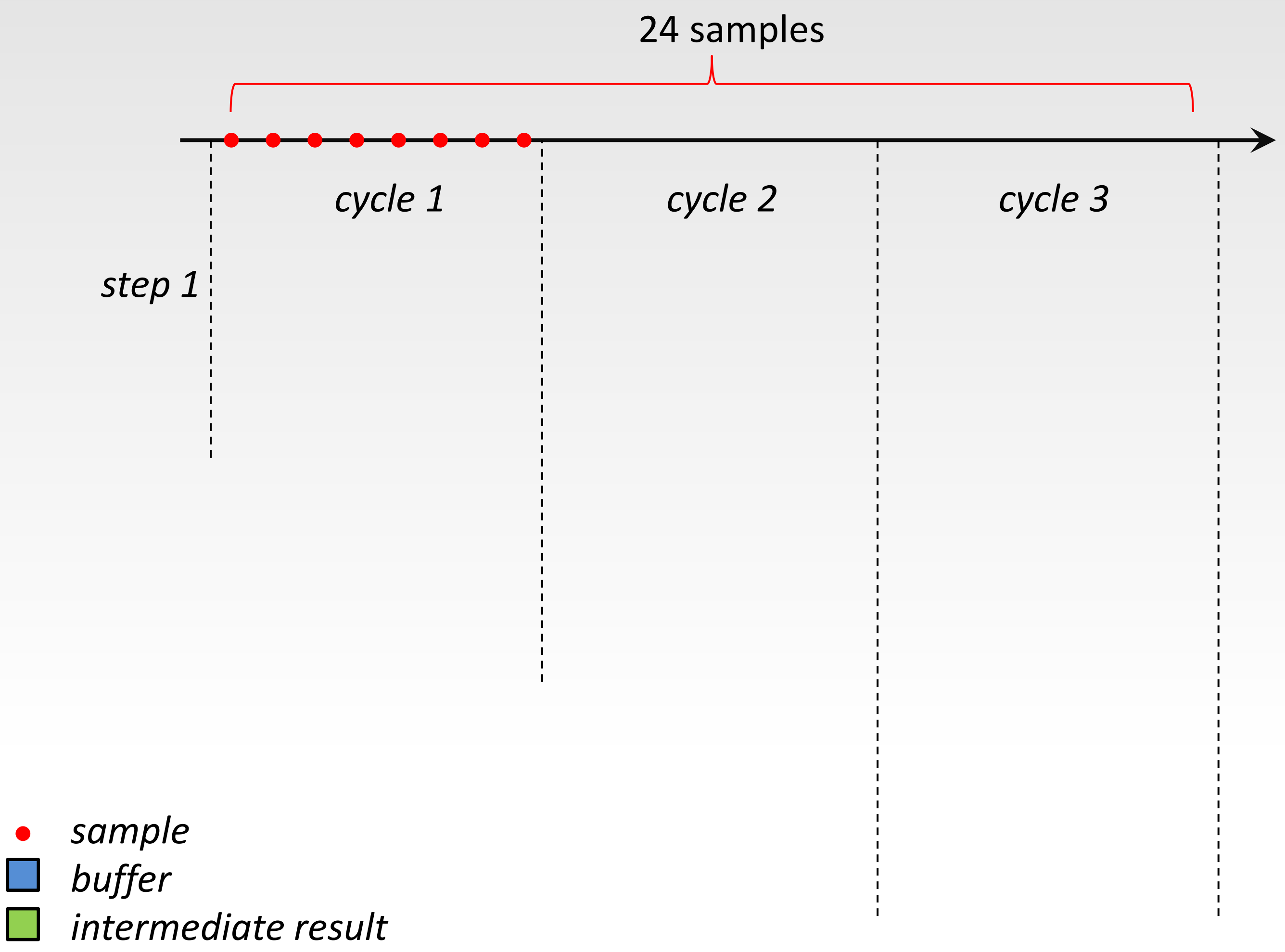
Motivation
Related Work
Contribution
Result
Application

Single Buffer Warp Marching

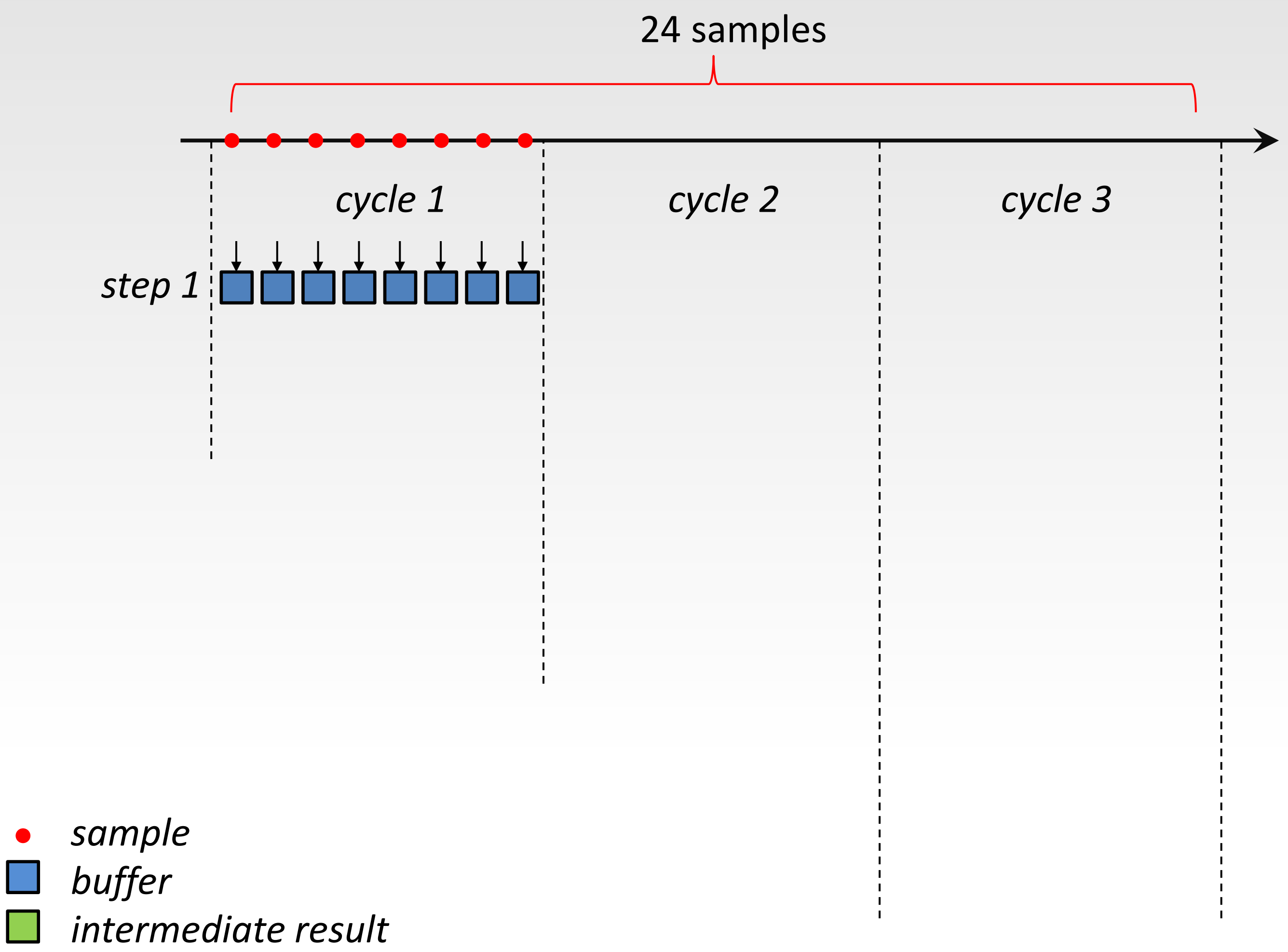


● *sample*
🟦 *buffer*
🟩 *intermediate result*

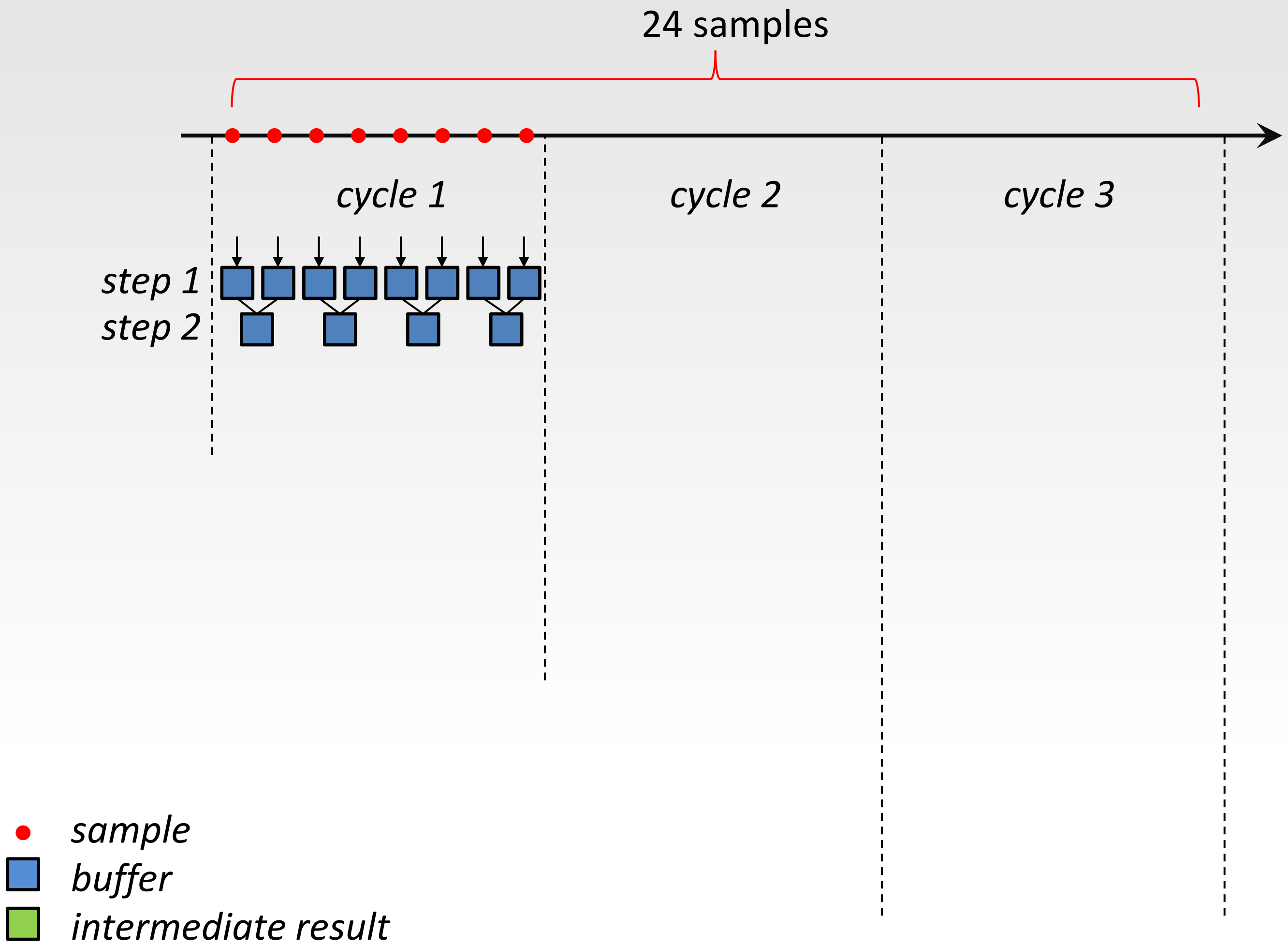
Single Buffer Warp Marching



Single Buffer Warp Marching

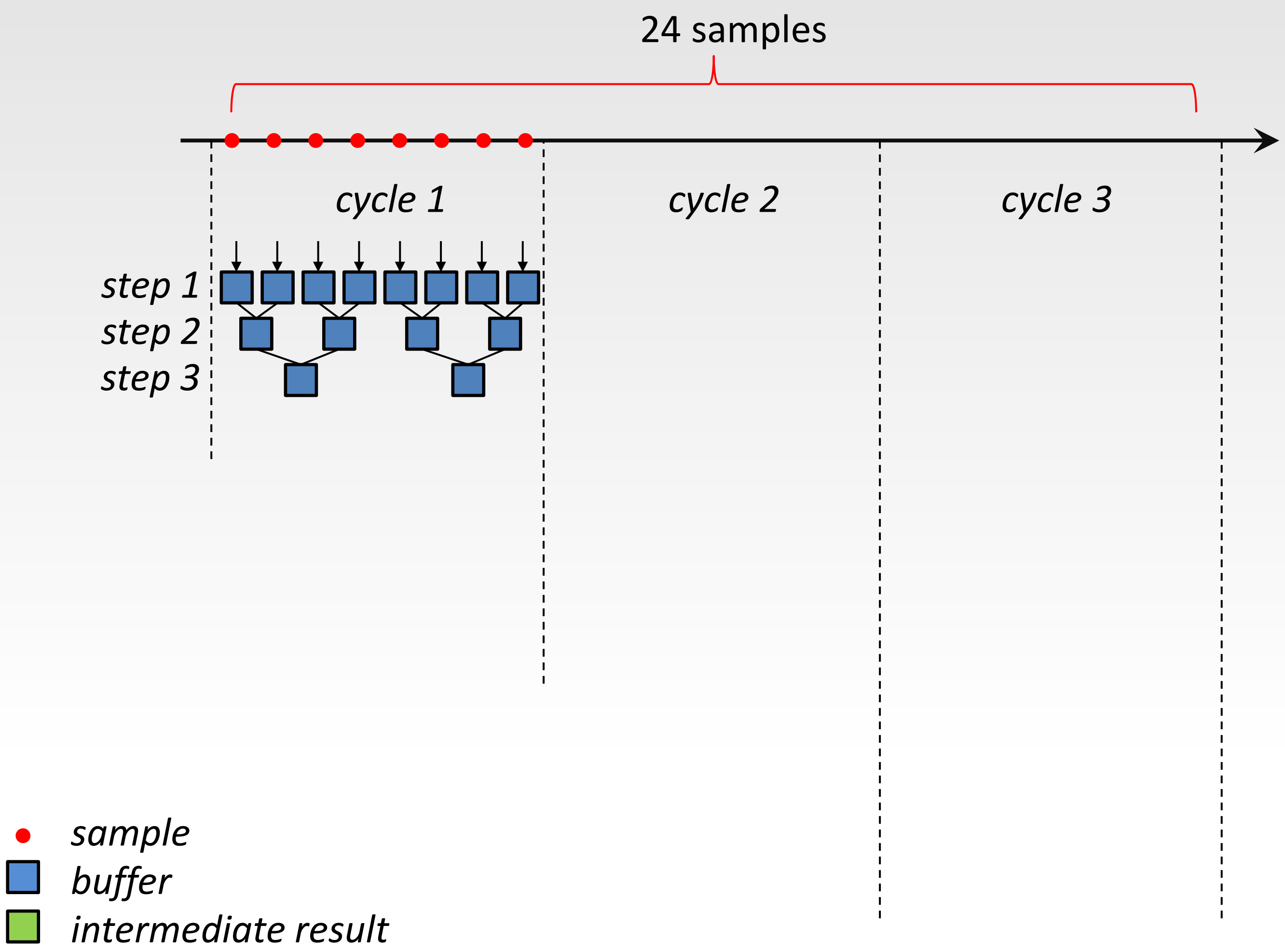


Single Buffer Warp Marching

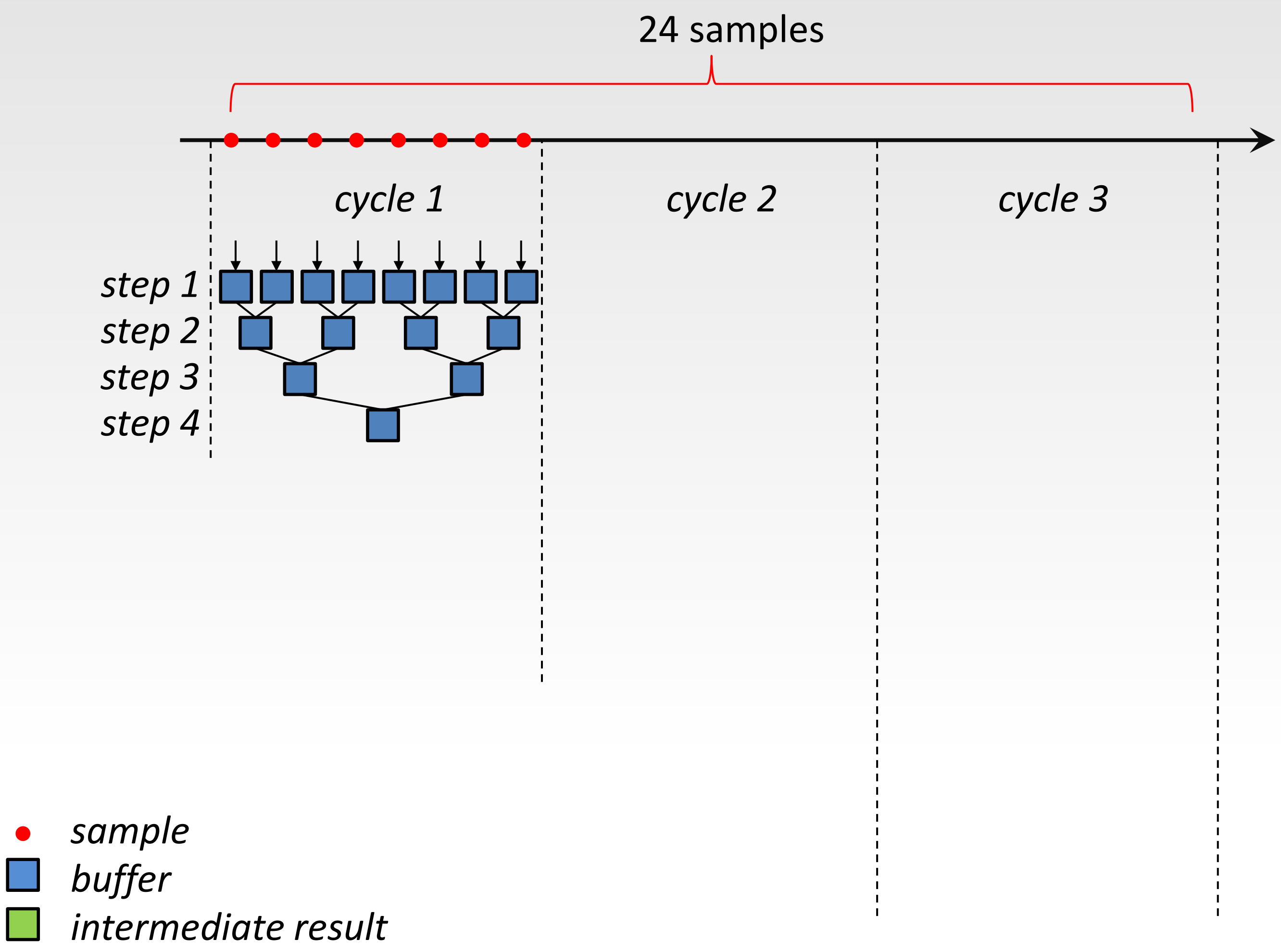


Motivation
Related Work
Contribution
Result
Application

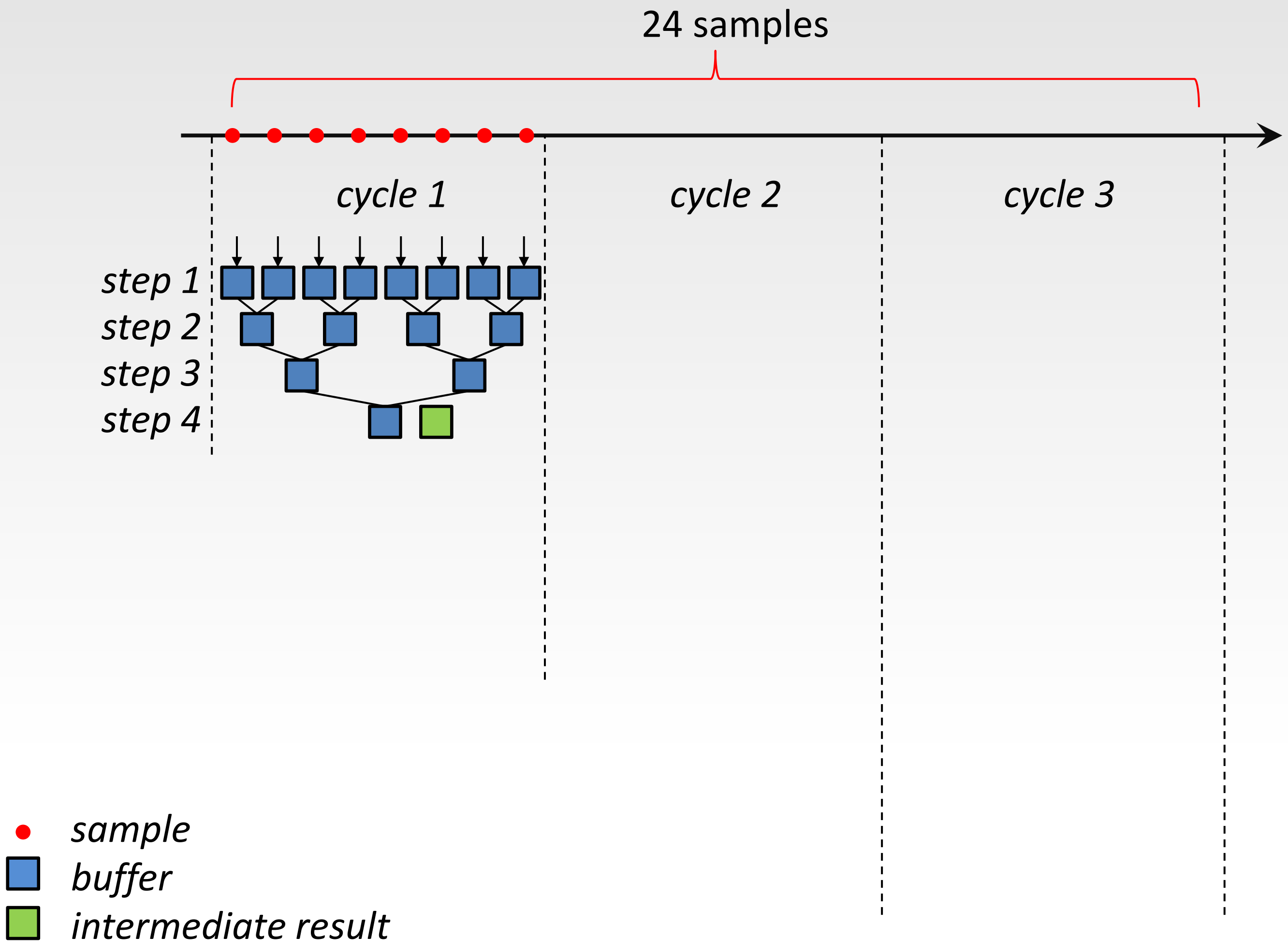
Single Buffer Warp Marching



Single Buffer Warp Marching



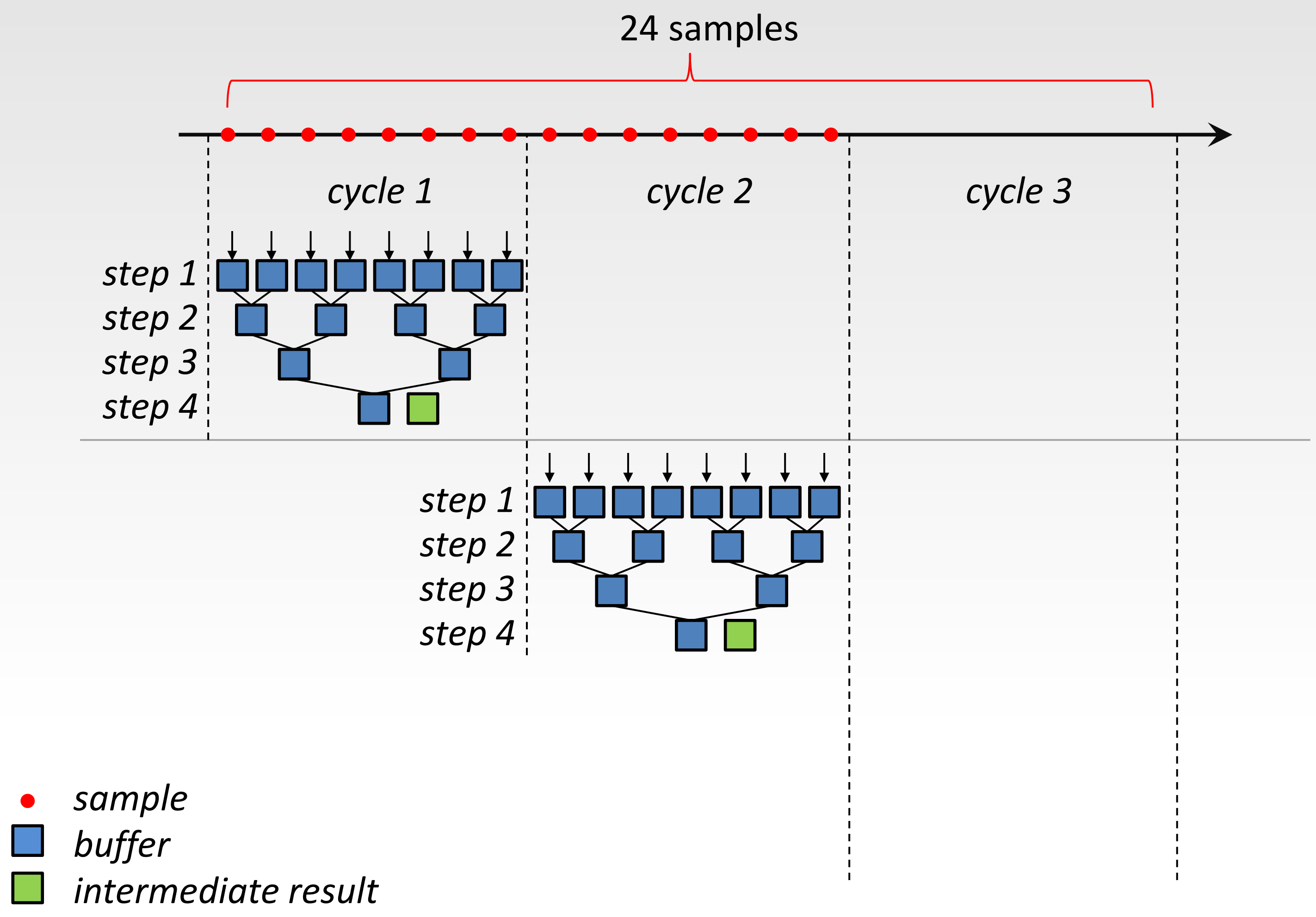
Single Buffer Warp Marching



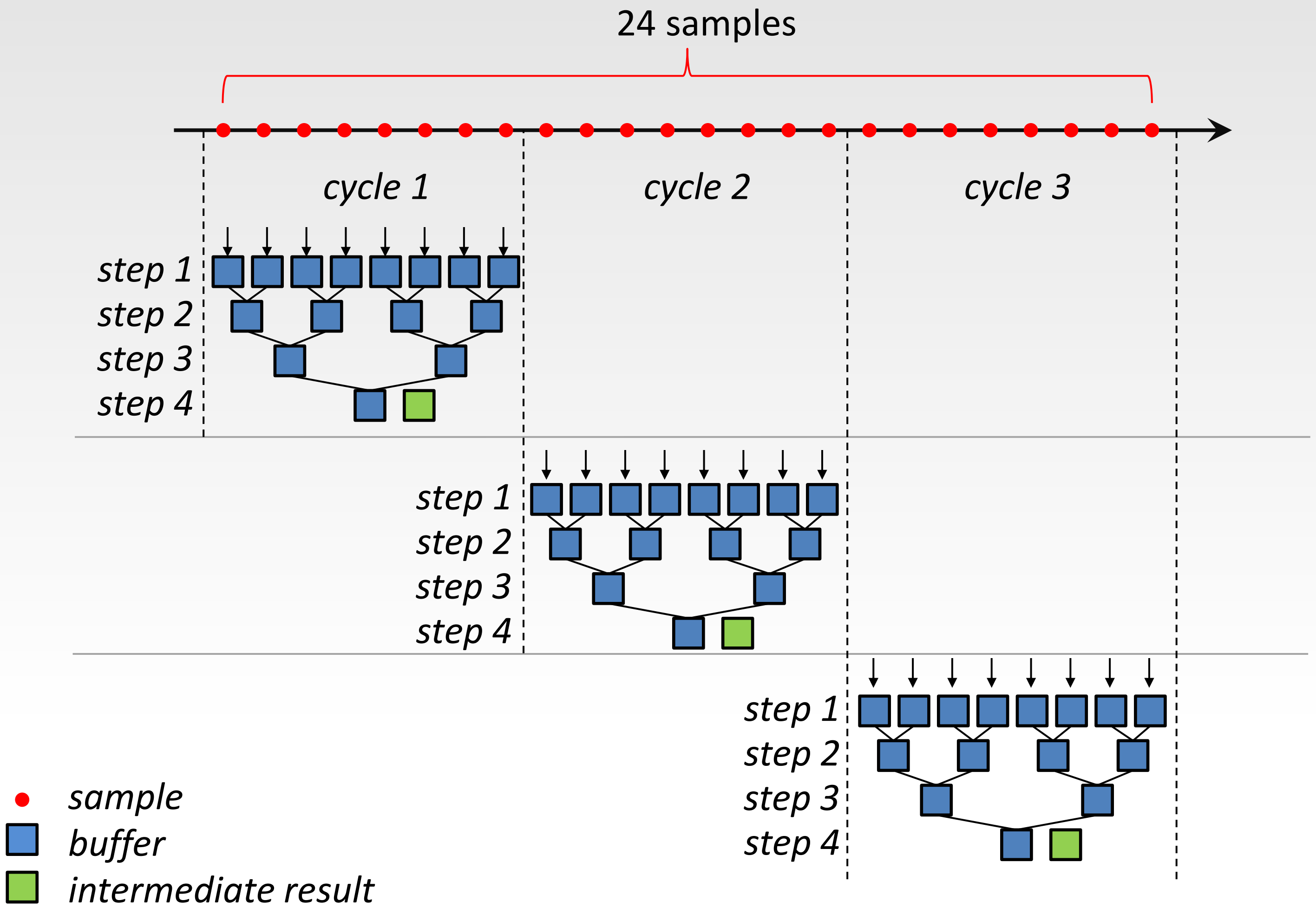
- ● *sample*
- 🟦 *buffer*
- 🟩 *intermediate result*

Motivation
Related Work
Contribution
Result
Application

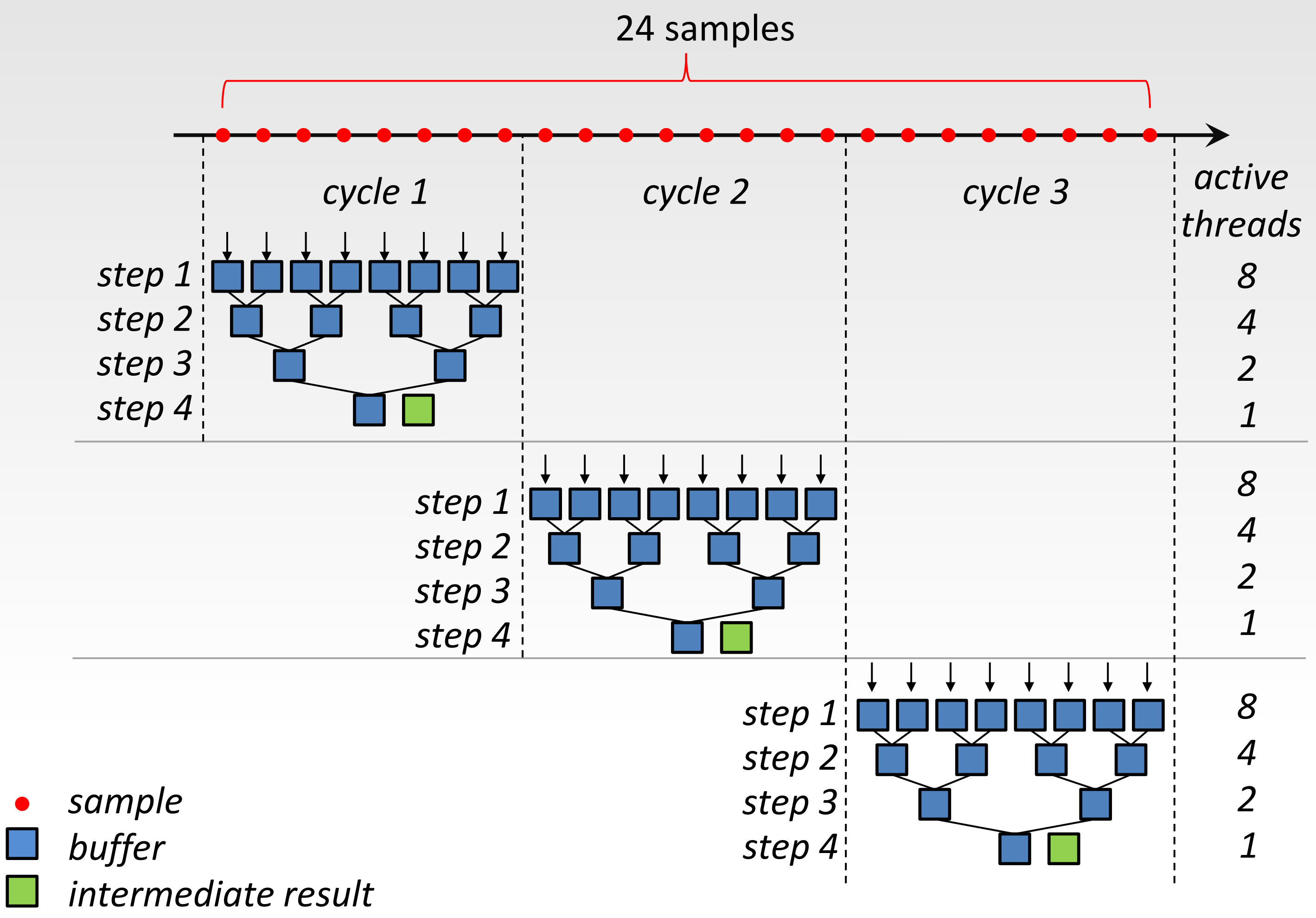
Single Buffer Warp Marching



Single Buffer Warp Marching



Single Buffer Warp Marching



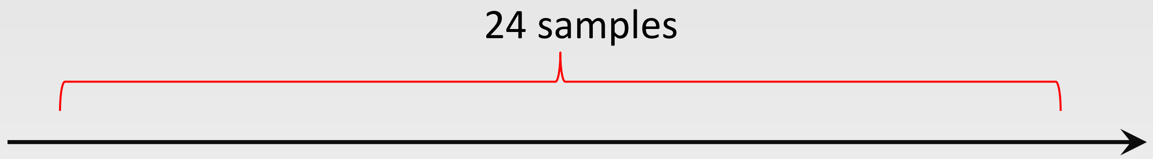
Double Buffer Warp Marching

● *sample*

🟦 *buffer 1*

🟩 *buffer 2*

Double Buffer Warp Marching

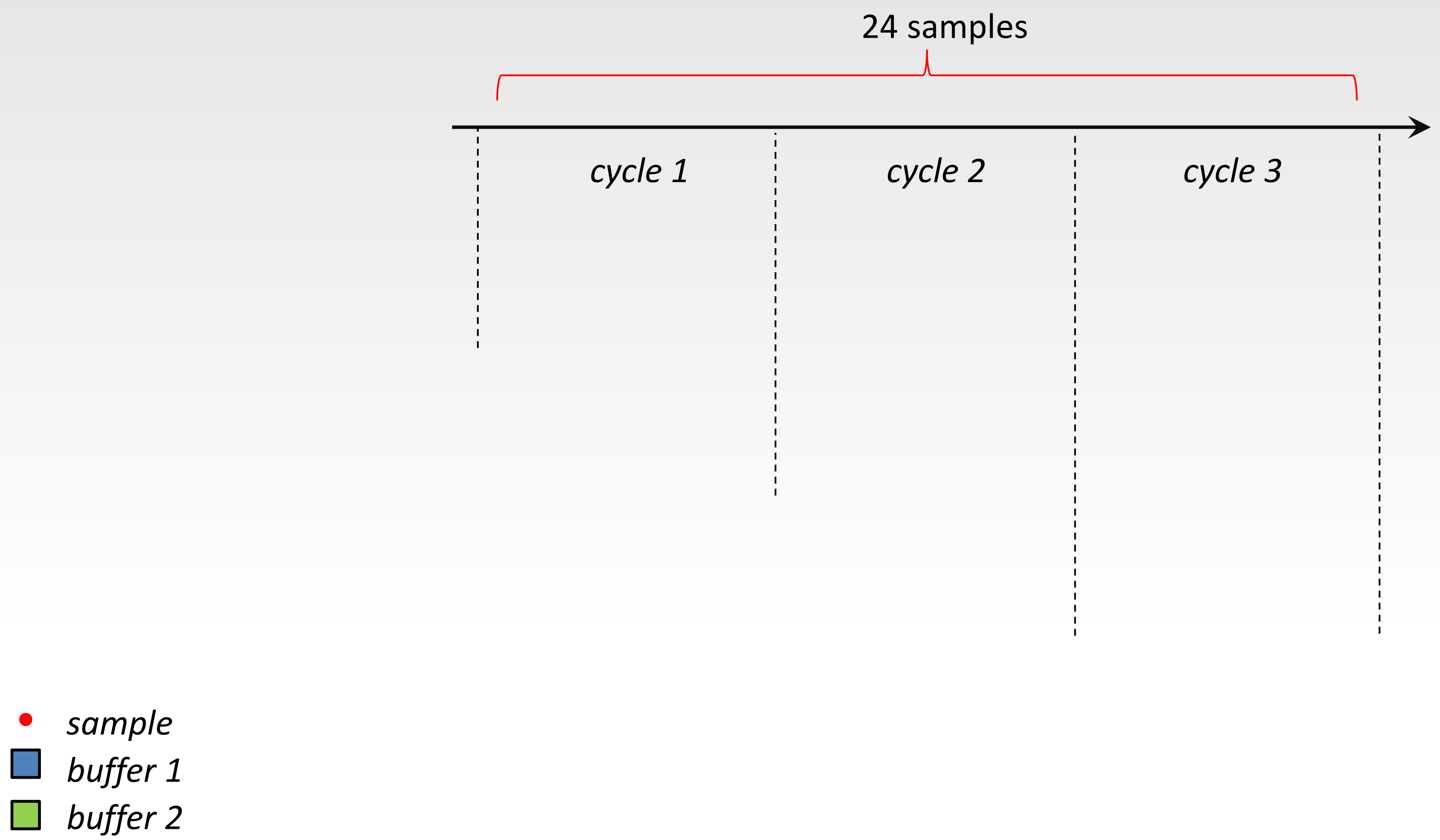


● *sample*

🟦 *buffer 1*

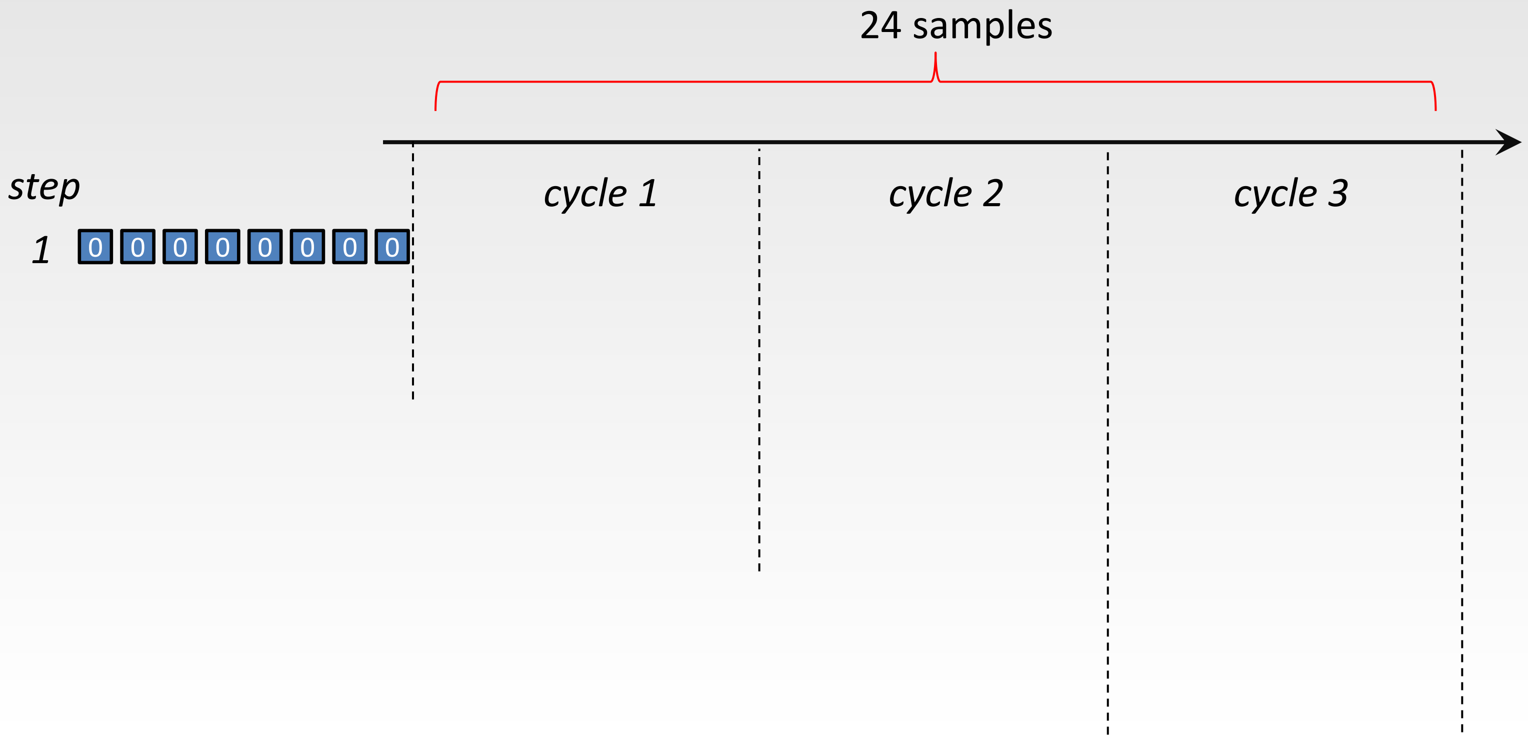
🟩 *buffer 2*

Double Buffer Warp Marching



Motivation
Related Work
Contribution
Result
Application

Double Buffer Warp Marching

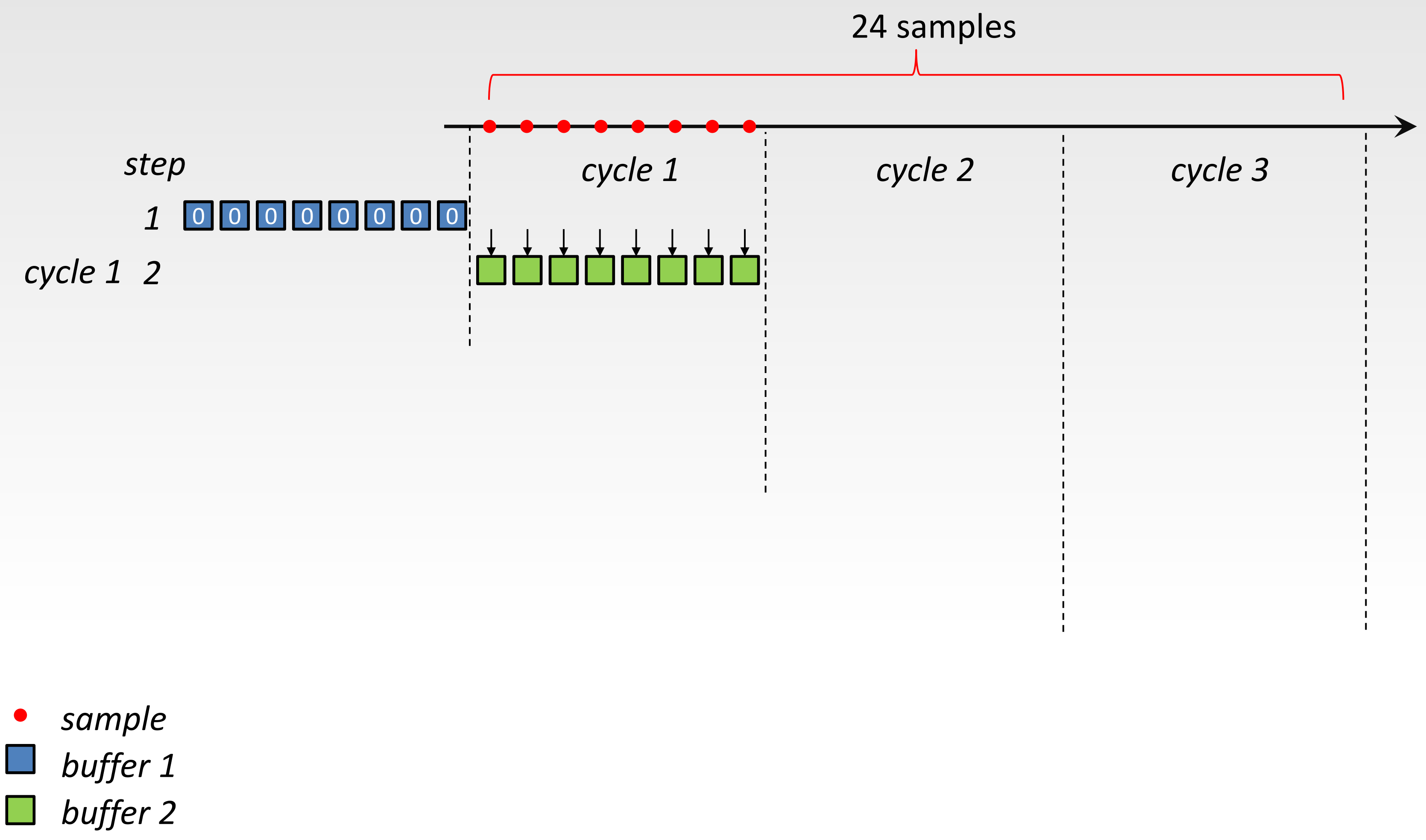


● *sample*

🟦 *buffer 1*

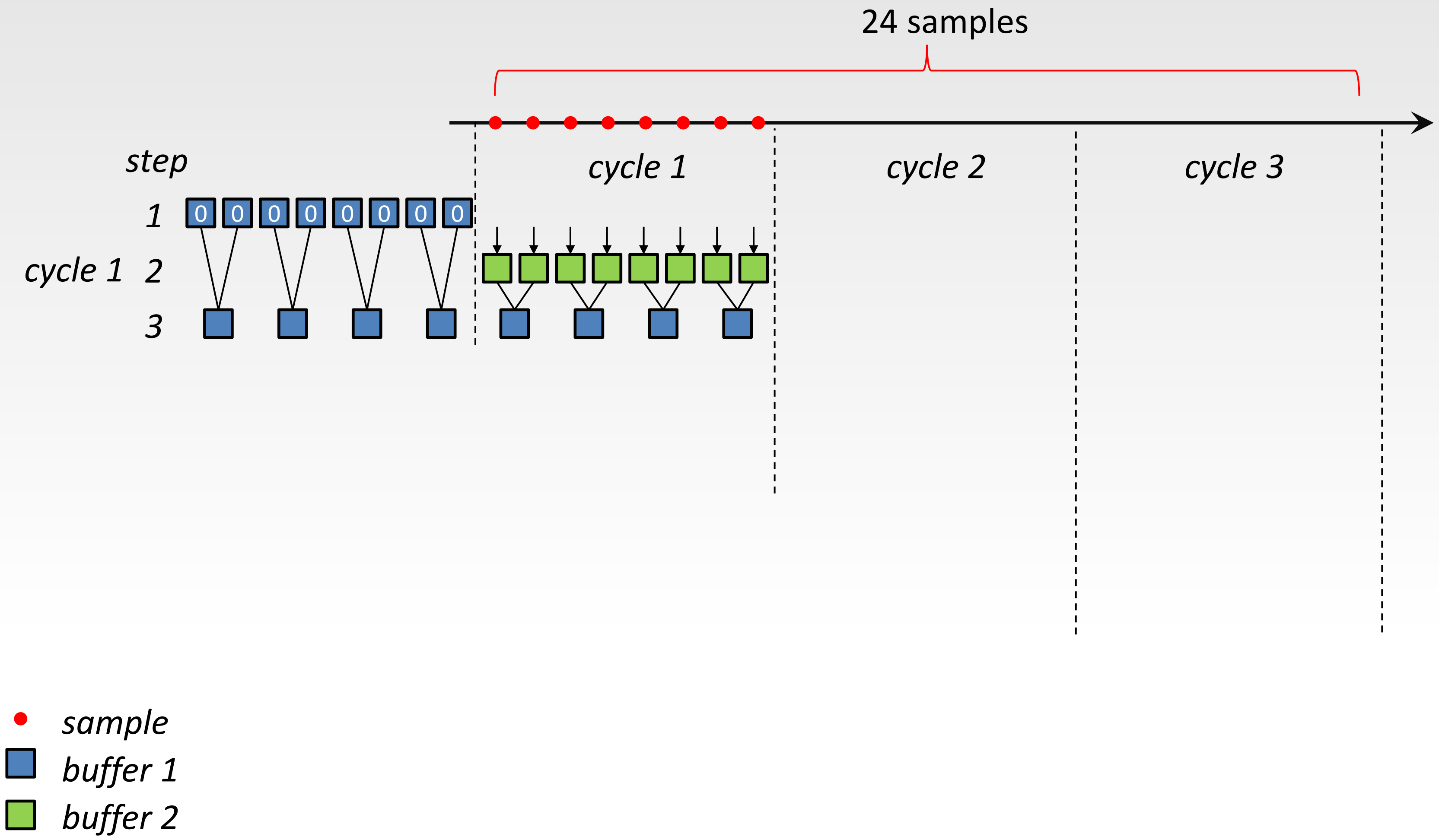
🟩 *buffer 2*

Double Buffer Warp Marching

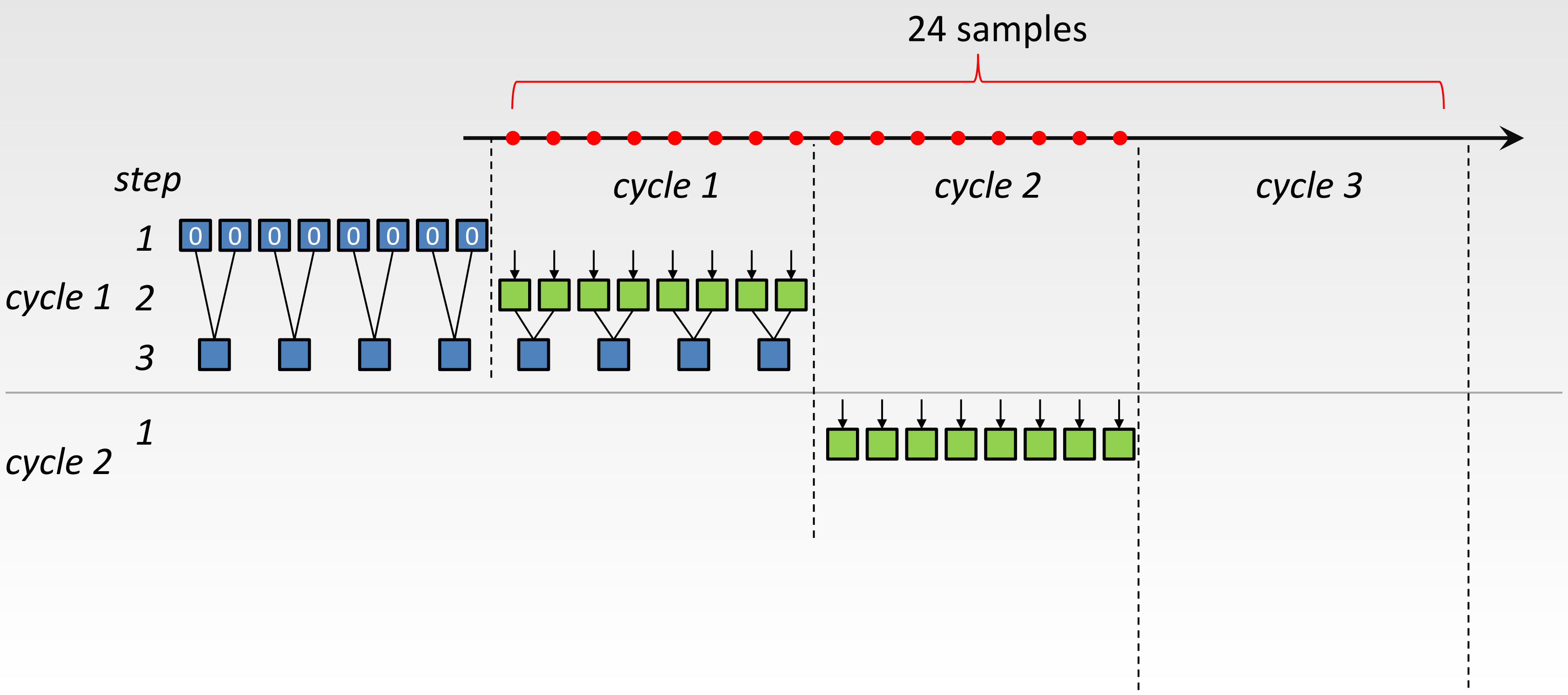


Motivation
Related Work
Contribution
Result
Application

Double Buffer Warp Marching



Double Buffer Warp Marching



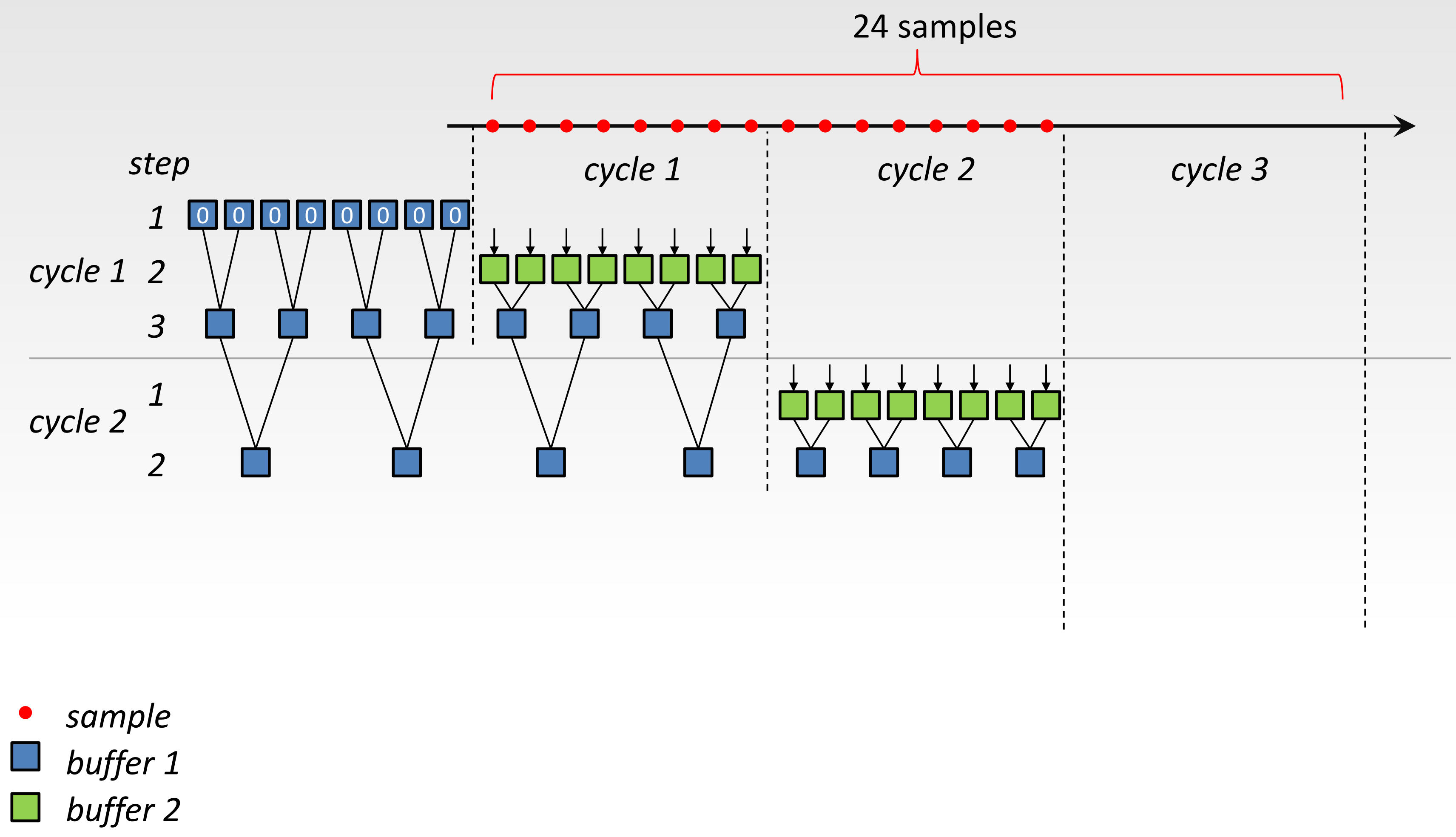
● *sample*

🟦 *buffer 1*

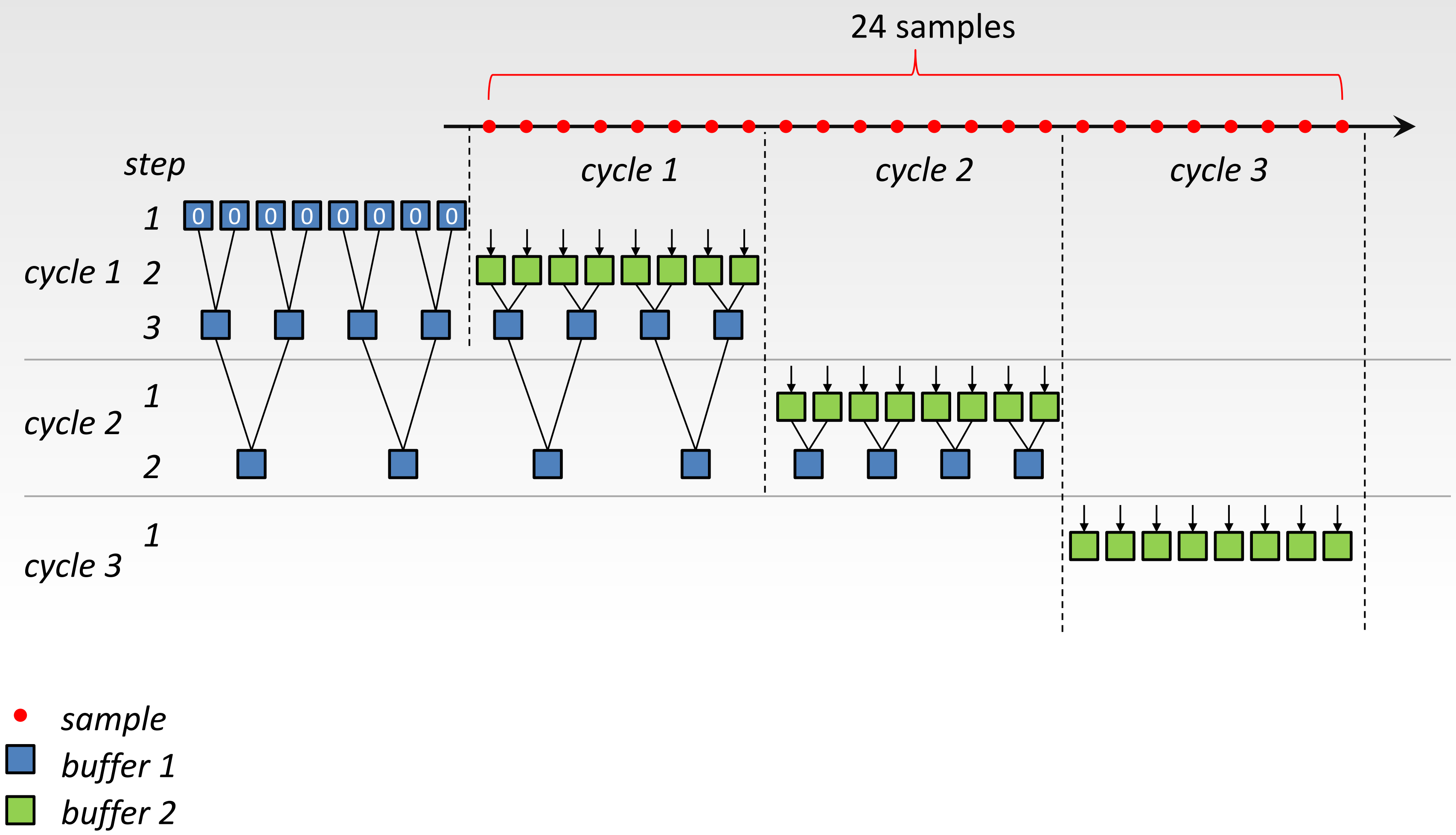
🟩 *buffer 2*

Motivation
Related Work
Contribution
Result
Application

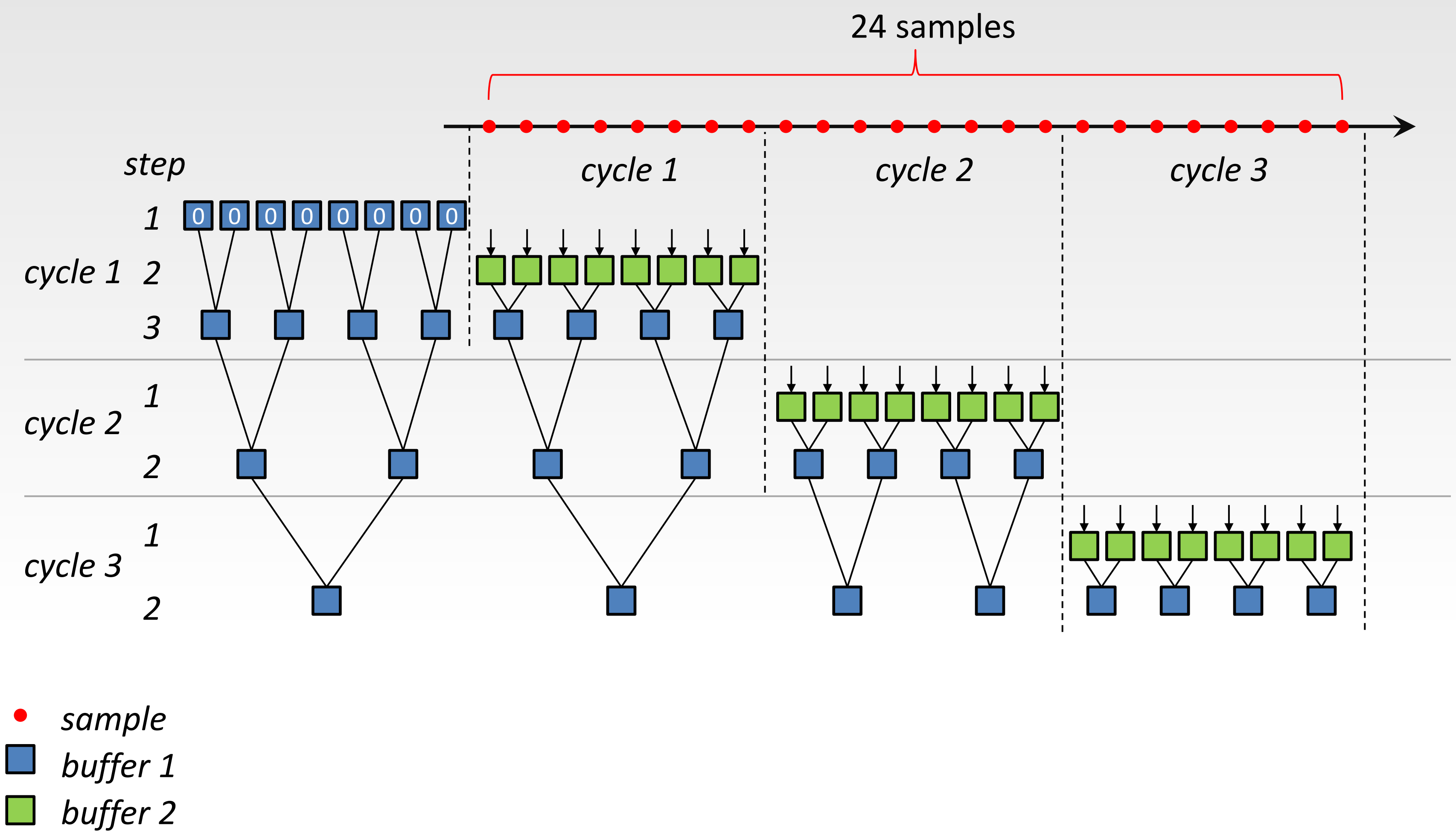
Double Buffer Warp Marching



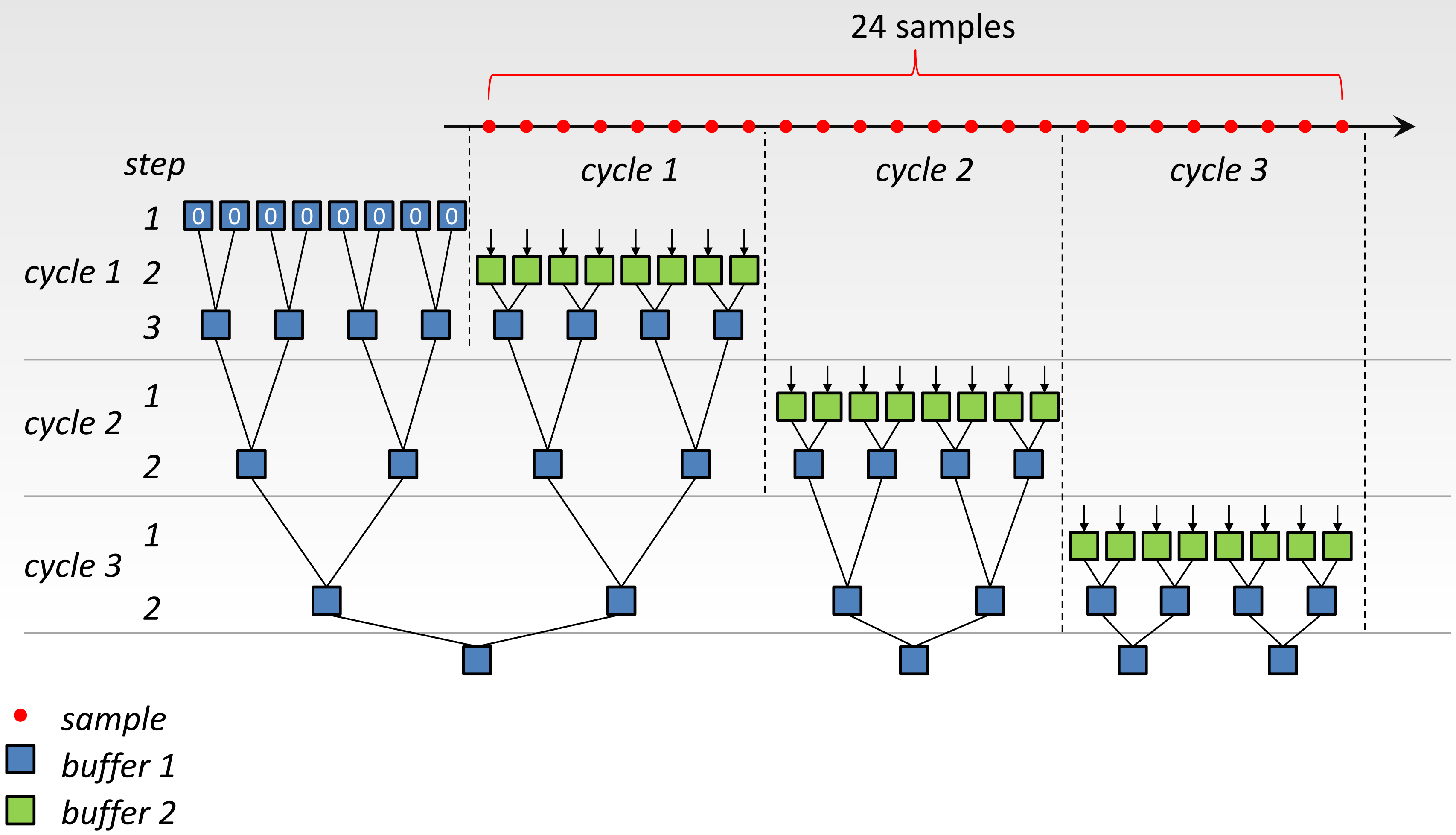
Double Buffer Warp Marching



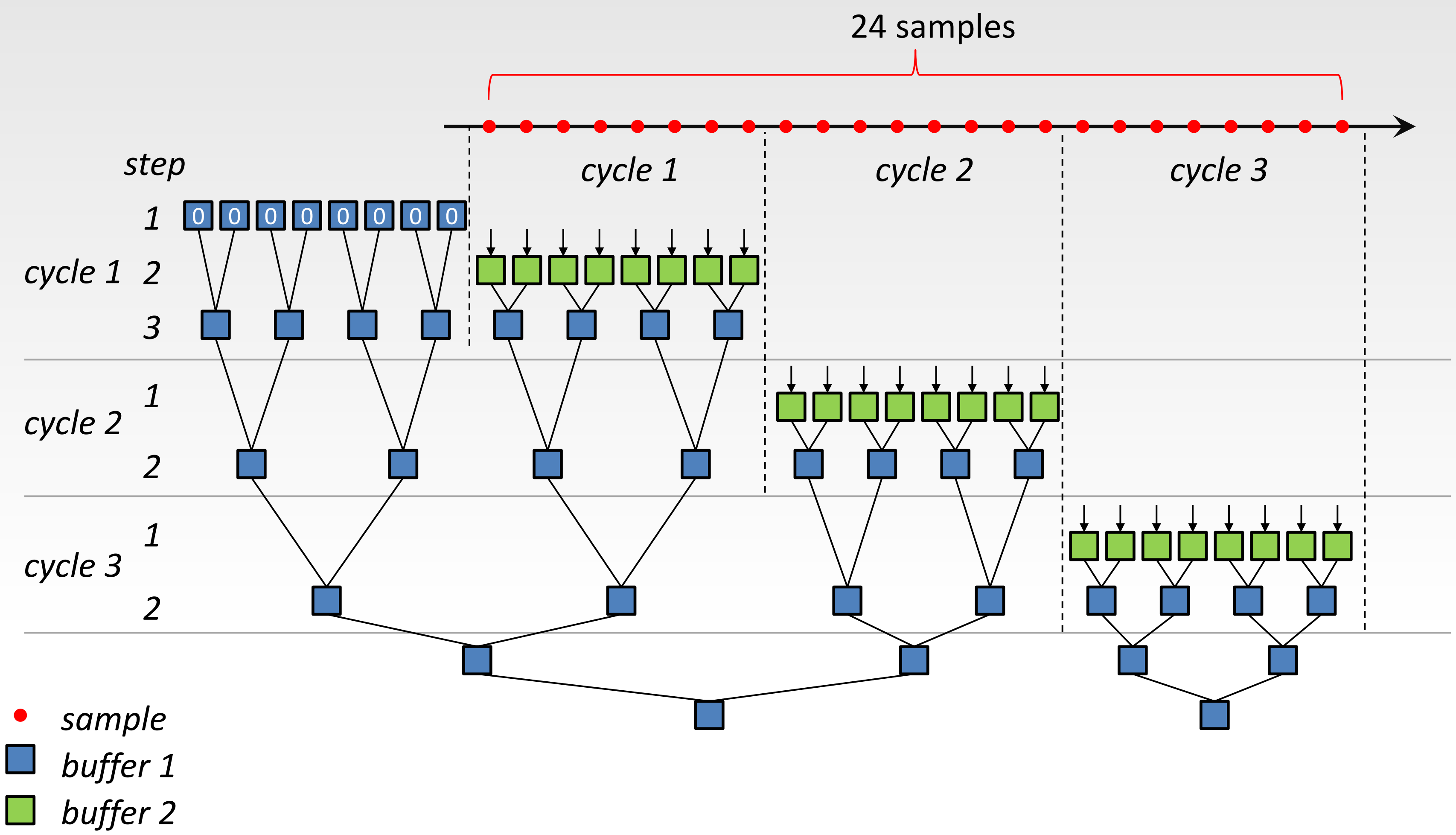
Double Buffer Warp Marching



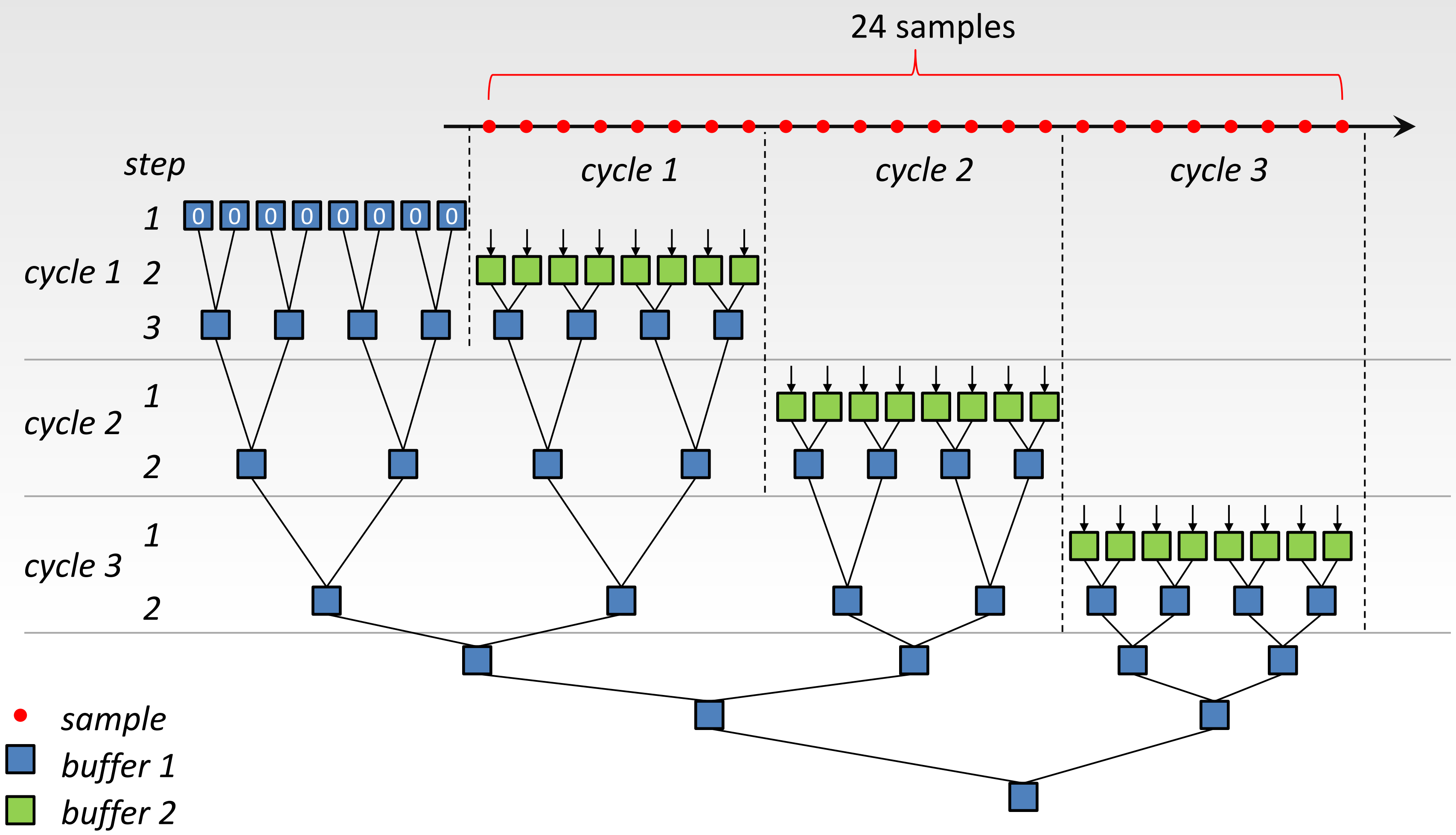
Double Buffer Warp Marching



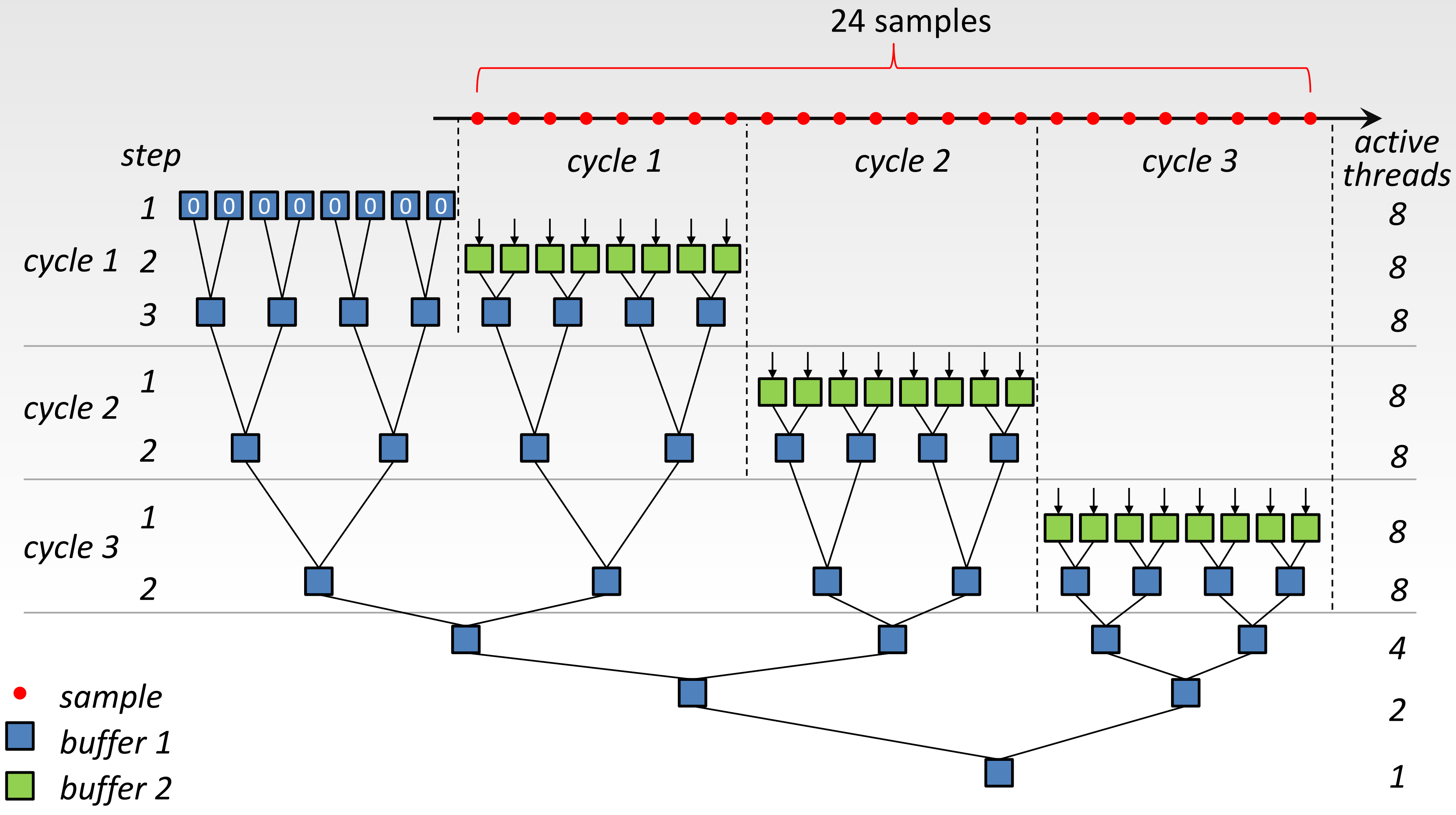
Double Buffer Warp Marching



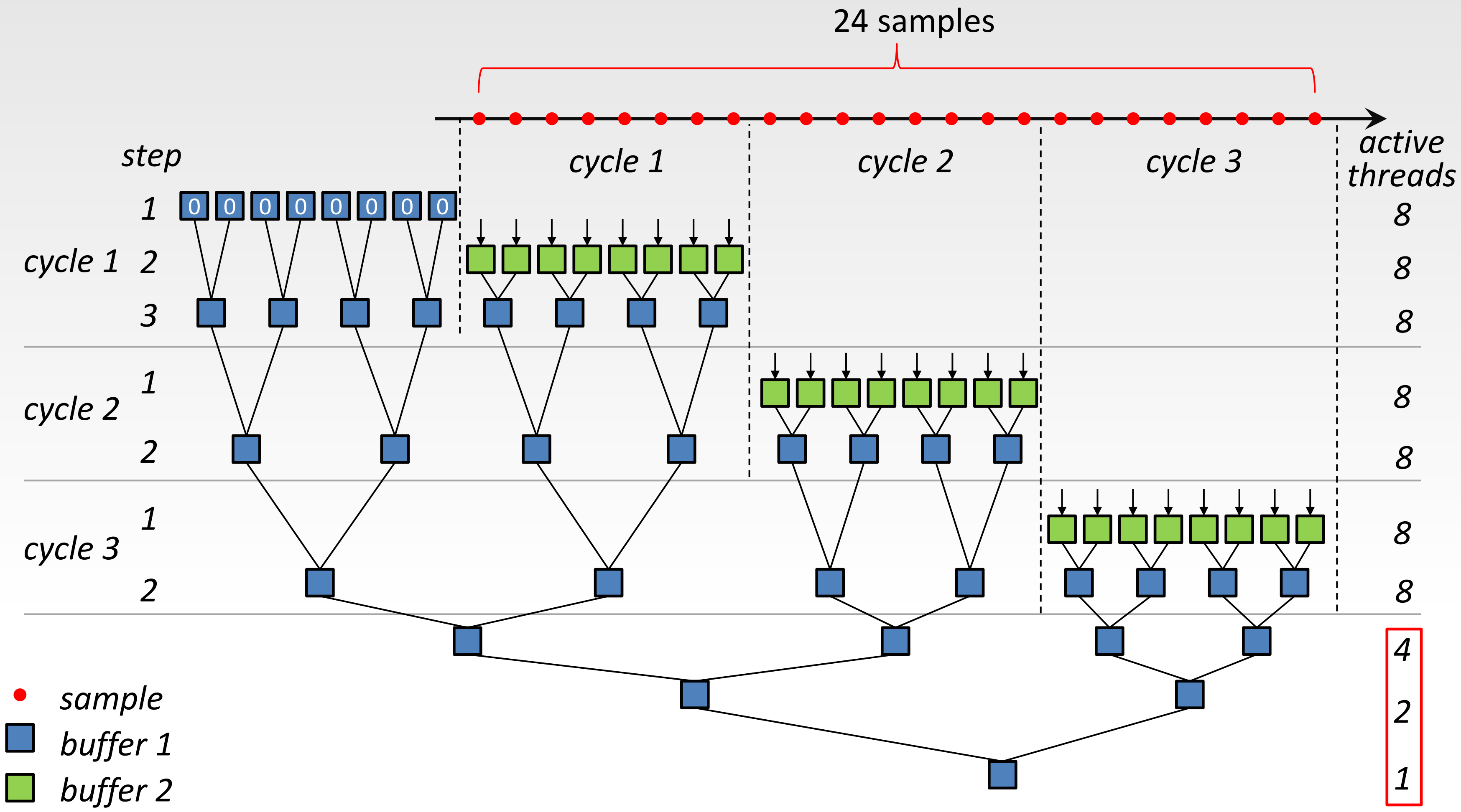
Double Buffer Warp Marching



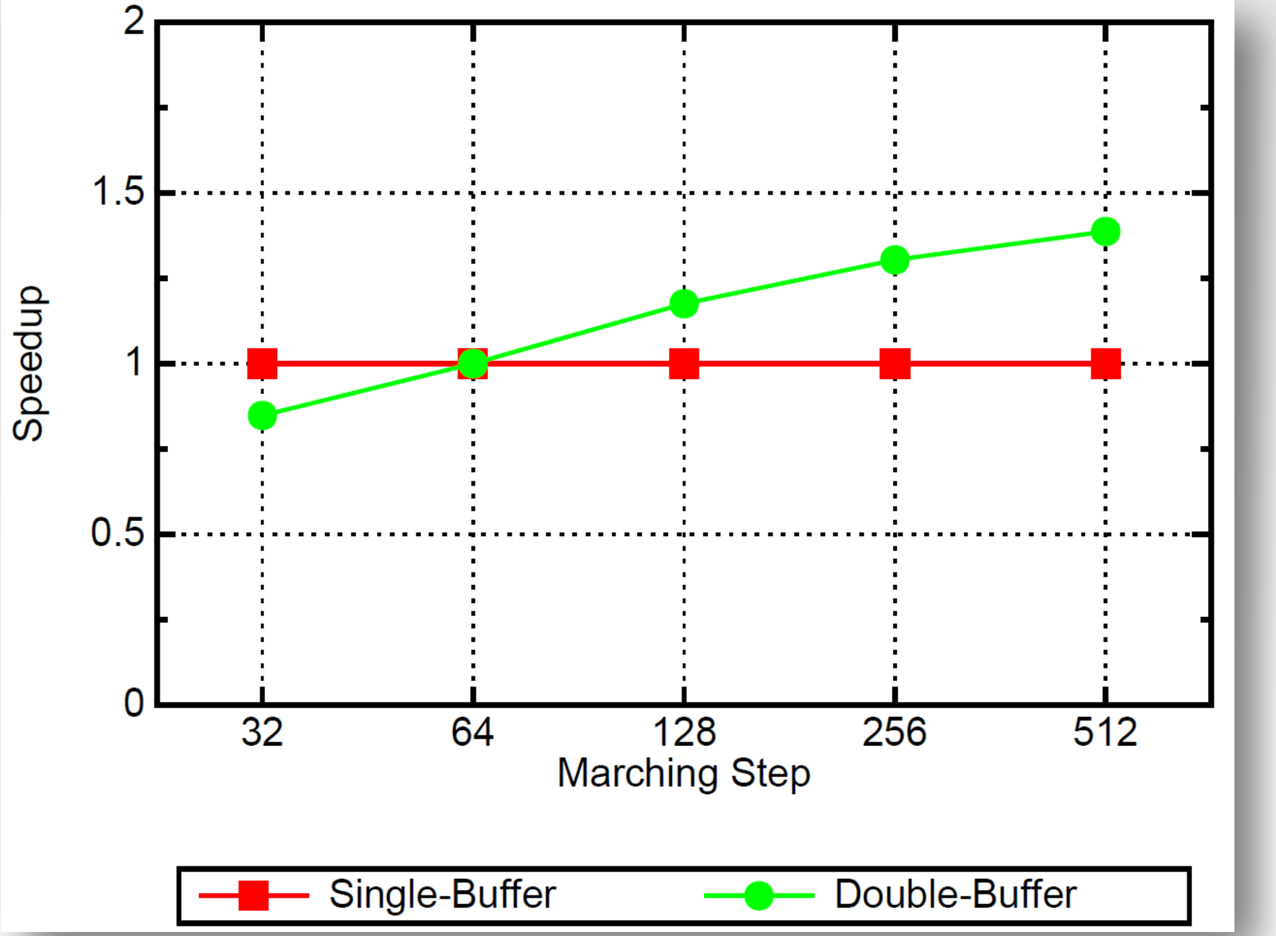
Double Buffer Warp Marching



Double Buffer Warp Marching

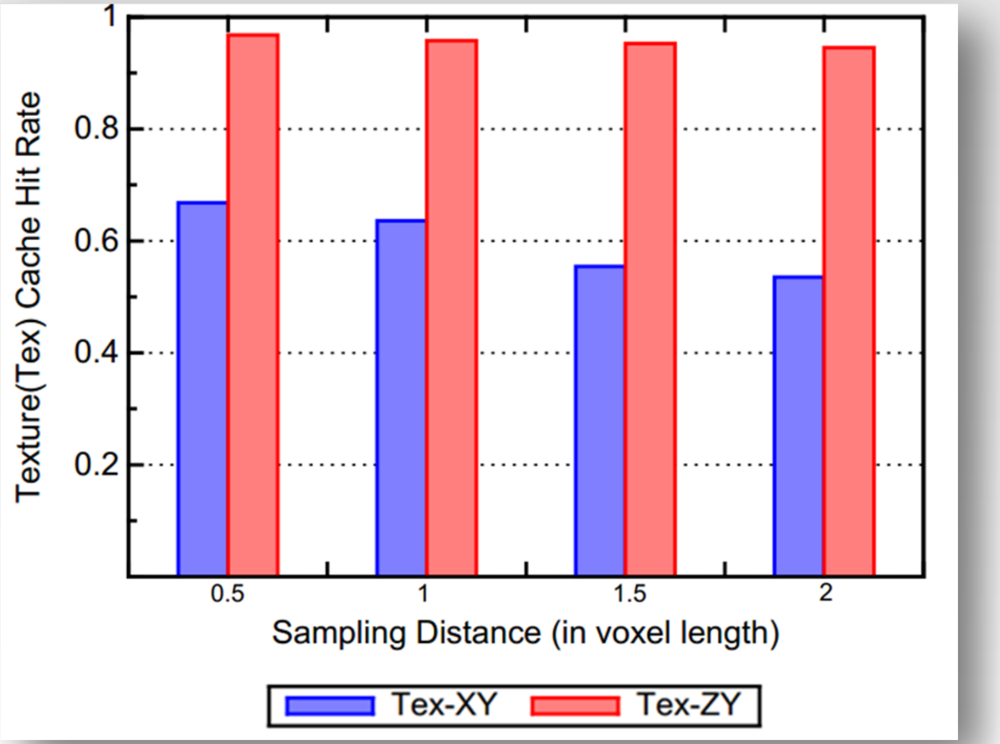


Optimization Result

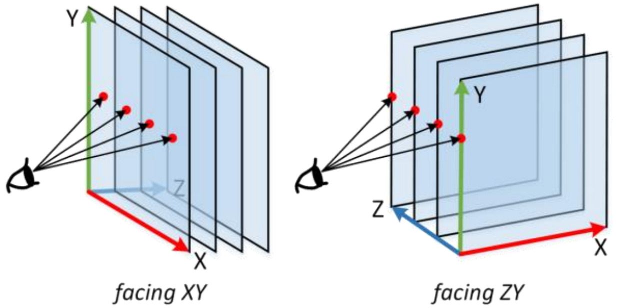


Warp size = 32

Texture Cache Performance



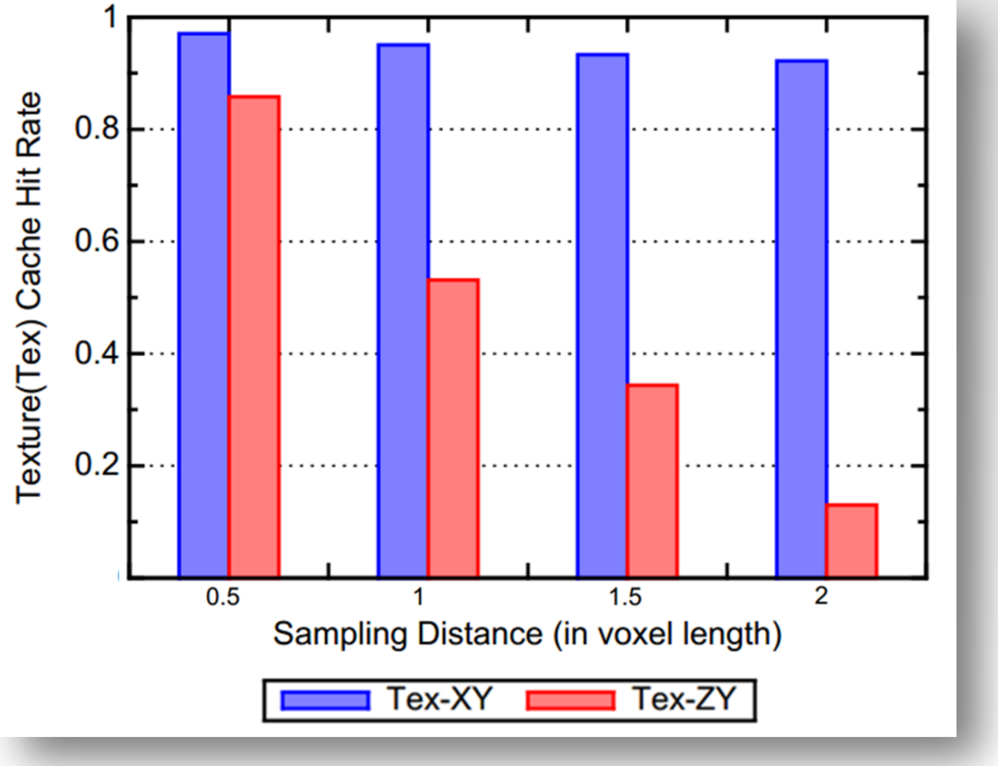
Warp Marching



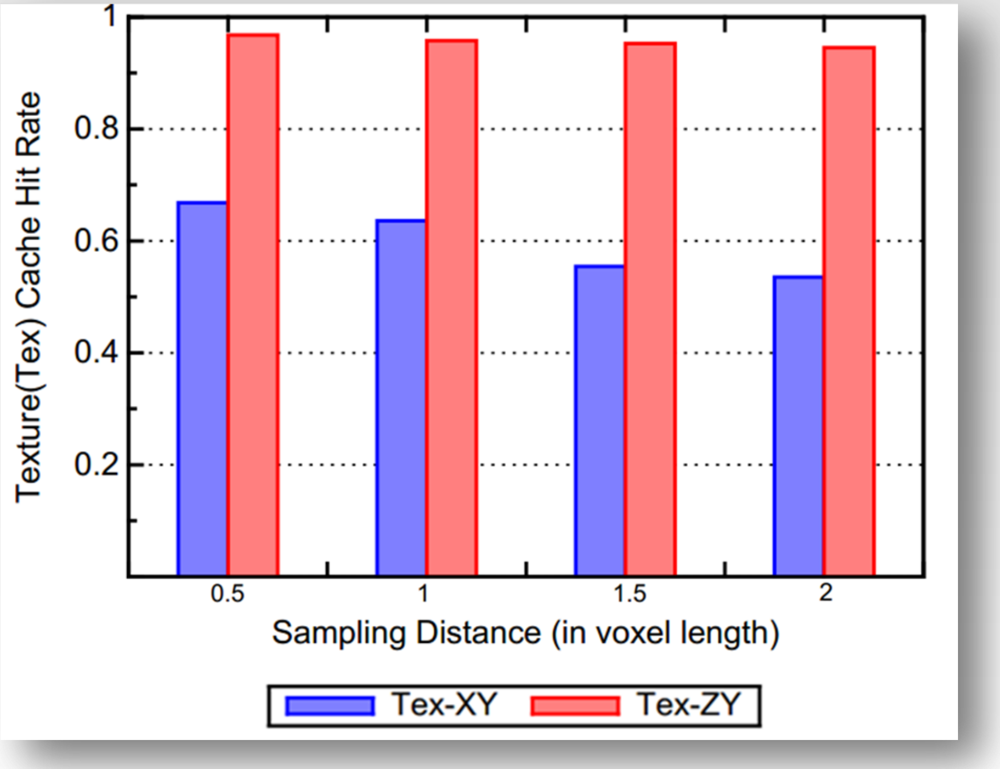
Perpendicular

Parallel

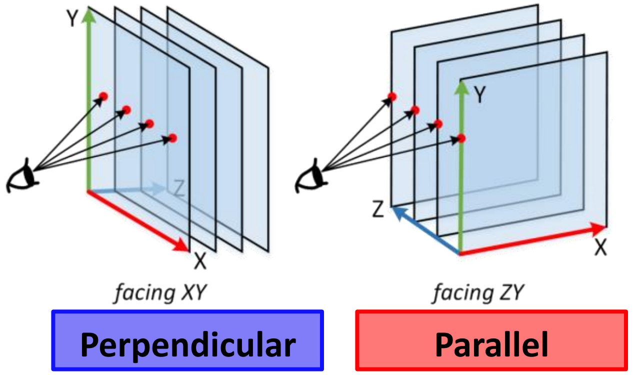
Texture Cache Performance



The Standard
(The Traditional)



Warp Marching



Perpendicular

Parallel

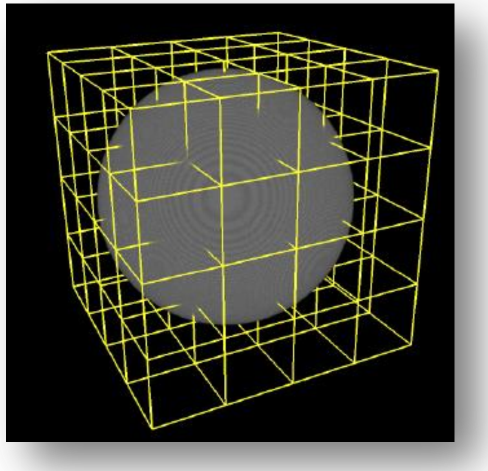
View Independent?

- Hybrid?
  - Perpendicular, the standard
  - Parallel, warp marching
- How about viewing directions in between?

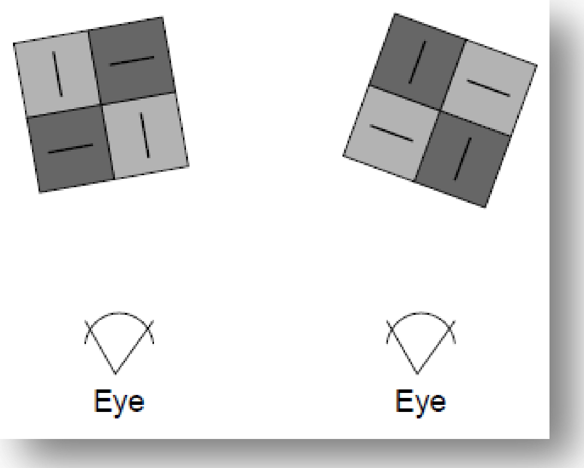
View Independent?

- Hybrid?
  - Perpendicular, the standard
  - Parallel, warp marching
- How about viewing directions in between?

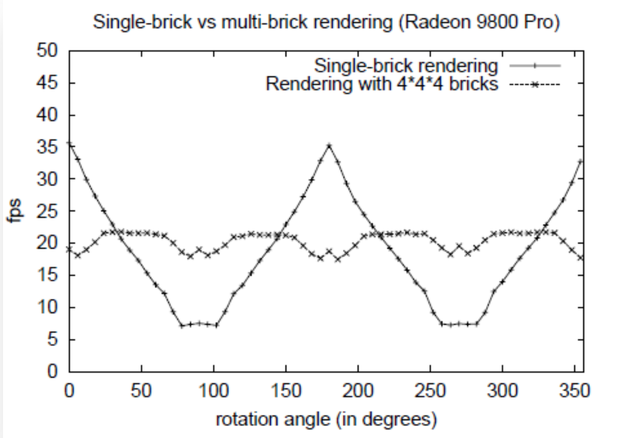
[Weiskopf04]



Partitioning a volume into small bricks



For any direction, 2 bricks are parallel and two bricks are perpendicular to the view



Achieve a roughly constant frame rate when rotating around the Y axis

Warp Shape

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8	32 x 1		
16 x 16	16 x 2		
8 x 32	8 x 4		
…	…	…	…

Warp Shape

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1		
16 x 16 x 1	16 x 2		
8 x 32 x 1	8 x 4		
…	…	…	…

Warp Shape

Motivation
Related Work
Contribution
Result
Application

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1 x 1		
16 x 16 x 1	16 x 2 x 1		
8 x 32 x 1	8 x 4 x 1		
…	…	…	…

Motivation
Related Work
Contribution
Result
Application

Warp Shape

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1 x 1	1x8x32	1 x 1 x 32
16 x 16 x 1	16 x 2 x 1	2x4x32	1 x 1 x 32
8 x 32 x 1	8 x 4 x 1	4x2x32	1 x 1 x 32
…	…	…	…

Warp Shape

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1 x 1	1x8x32	1 x 1 x 32
16 x 16 x 1	16 x 2 x 1	2x4x32	1 x 1 x 32
8 x 32 x 1	8 x 4 x 1	4x2x32	1 x 1 x 32
…	…	…	…

Warp Shape

Thread block size 256, warp size 32

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1 x 1	1x8x32	1 x 1 x 32
16 x 16 x 1	16 x 2 x 1	2x4x32	1 x 1 x 32
8 x 32 x 1	8 x 4 x 1	4x2x32	1 x 1 x 32
…	…	…	…

Combined Approach	
Block Shape	Warp Shape
2x16x8	2x2x8
4x16x4	4x2x4
…	…

Warp Shape

Thread block size 256, warp size 32

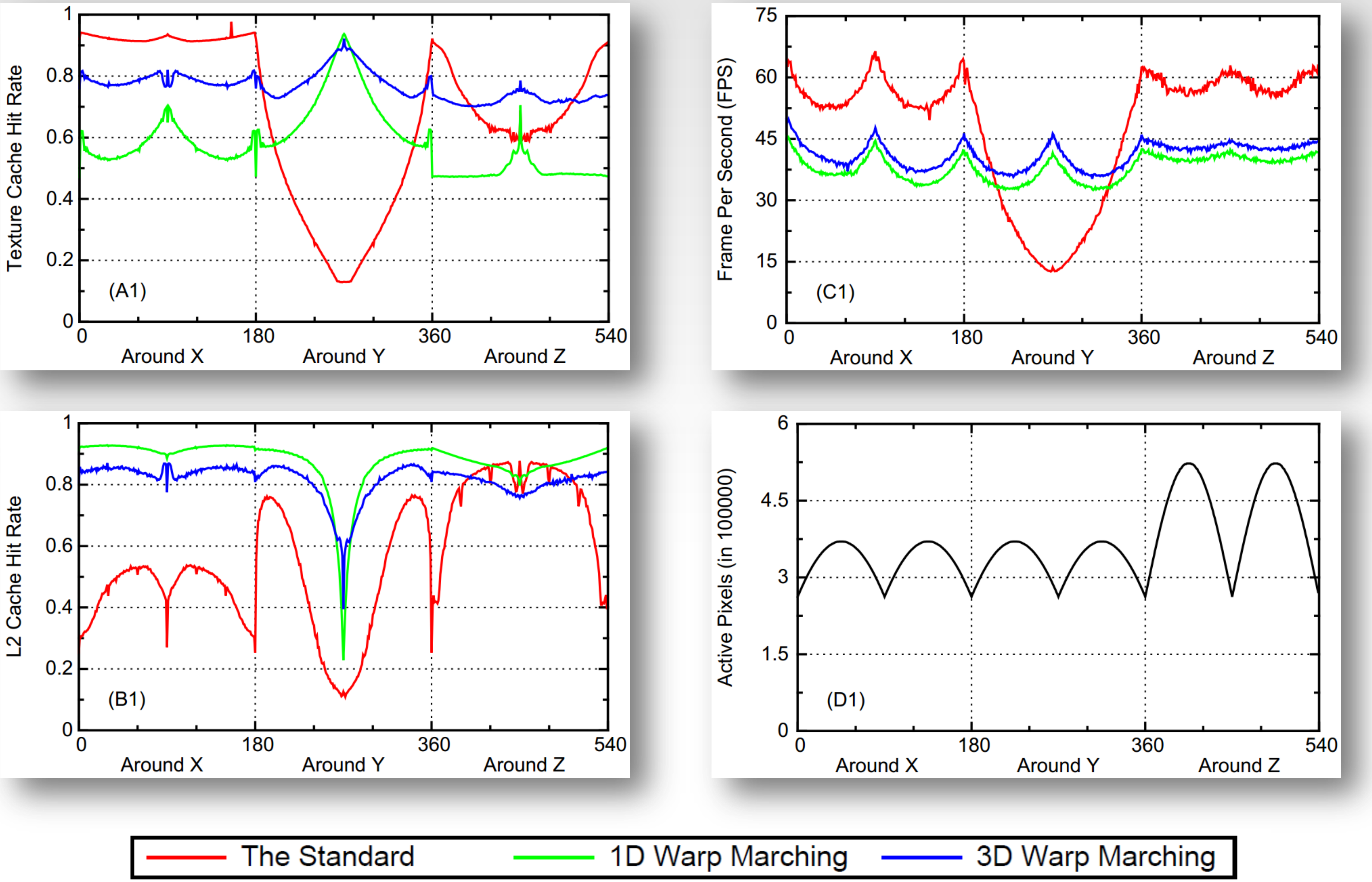
1D Warp Marching

The Standard Sampling		Warp Marching	
Block Shape	Warp Shape	Block Shape	Warp Shape
32 x 8 x 1	32 x 1 x 1	1x8x32	1 x 1 x 32
16 x 16 x 1	16 x 2 x 1	2x4x32	1 x 1 x 32
8 x 32 x 1	8 x 4 x 1	4x2x32	1 x 1 x 32
…	…	…	…

Combined Approach	
Block Shape	Warp Shape
2x16x8	2x2x8
4x16x4	4x2x4
…	…

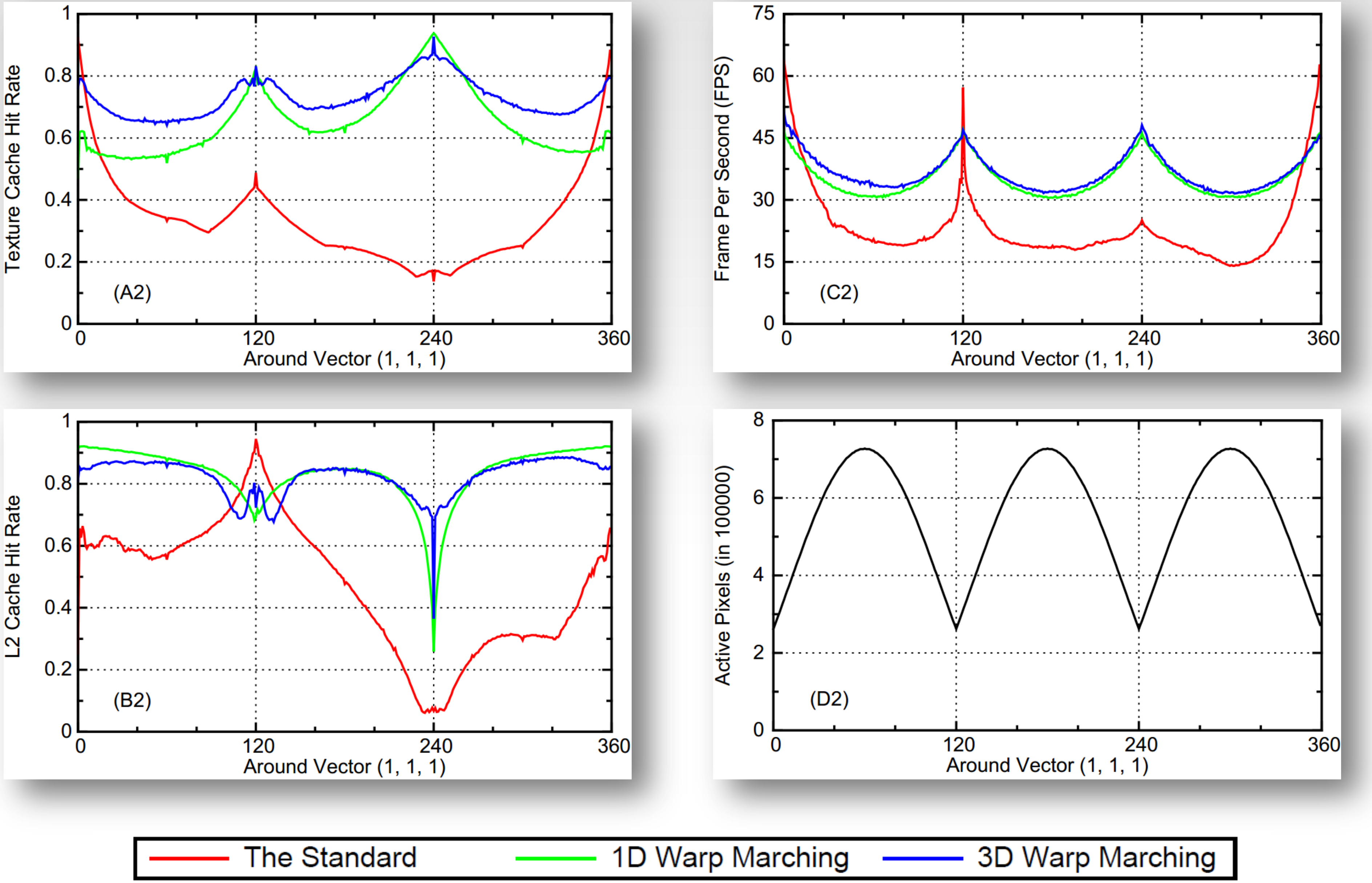
3D Warp Marching

3D Warp Marching



Rotate the volume around X, Y and Z axis 180 degree respectively

3D Warp Marching

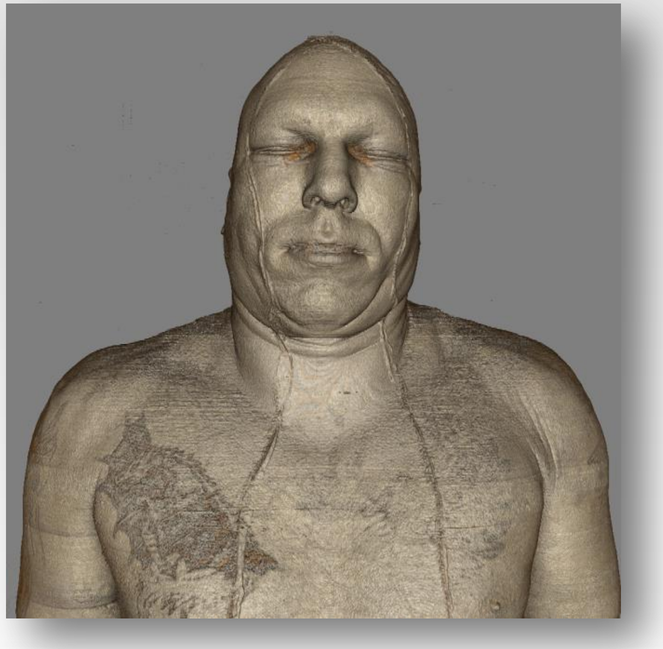


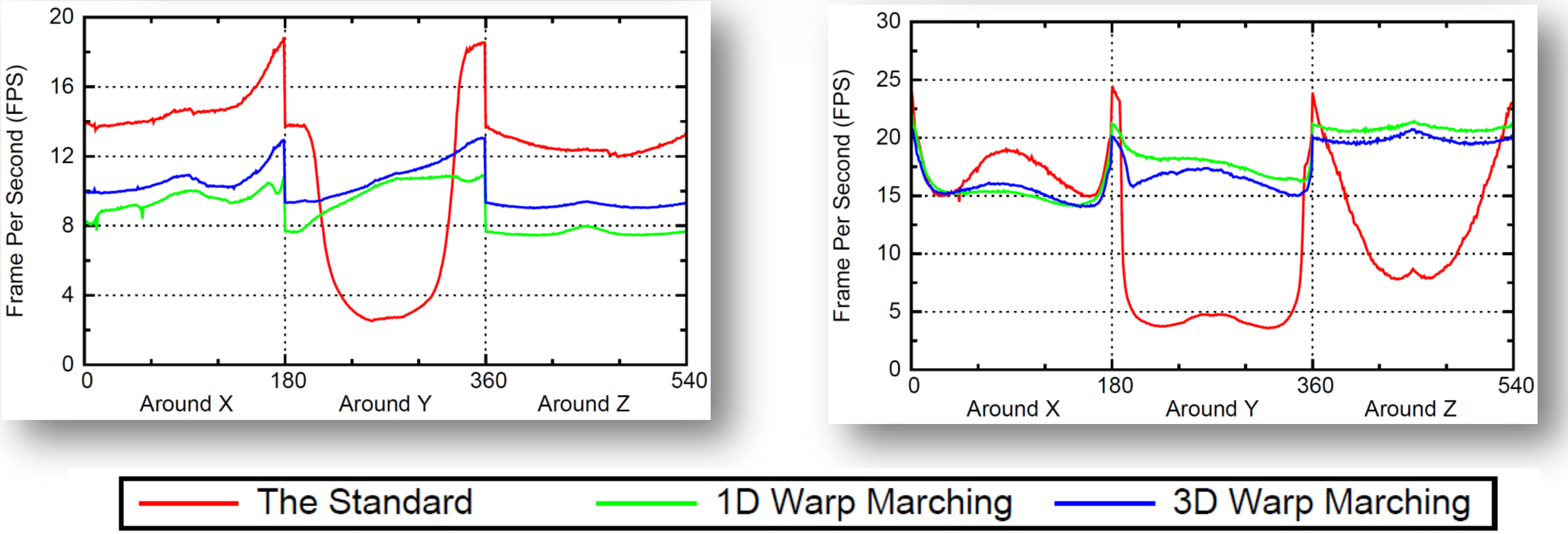
Rotate the volume around vector (1,1,1) 360 degree

Application

vhf_head (8-bit, 2.8GB)

vhm_body (8-bit, 3.8GB)





[Lum2004]: High-quality lighting and efficient pre-integration for volume rendering.

Conclusion & Future Work

- Conclusion
  - We design a cache-aware sampling strategy, i.e. warp marching, for the ray casting algorithm.
  - The 3D warp marching maintains a roughly constant texture cache hit rate regardless of volume orientation.

- Future Work
  - L2 cache performance
  - Other types of GPUs, varying warp sizes
  - New applications

Thank you

Questions?