Improved Algorithms for Volume Rendering and Mesh Processing

André Maximo

Computer Graphics Lab

Federal University of Rio de Janeiro

Advisor: Ricardo Farias

(CNPq Sandwich) Advisor: Amitabh Varshney

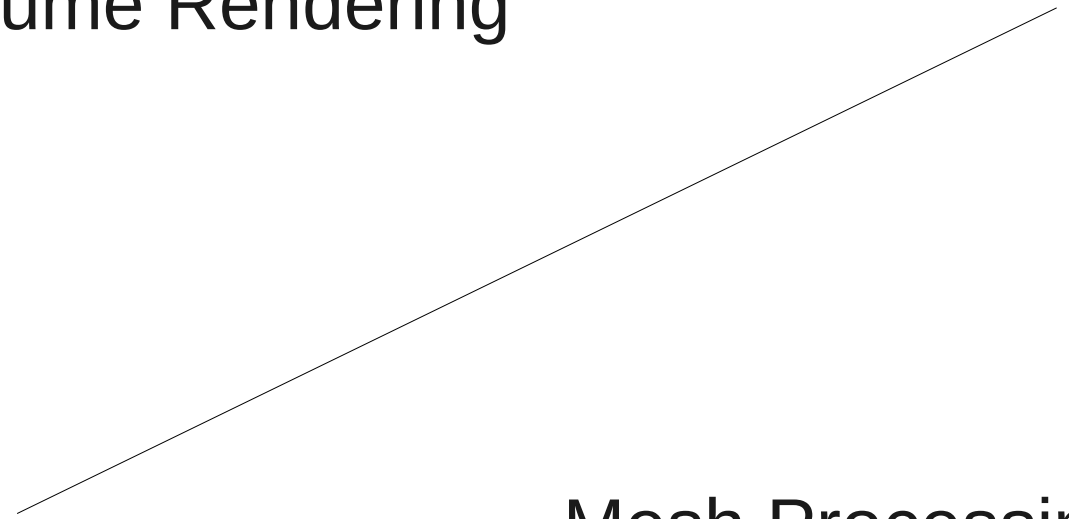
July 23, 2010
Rio de Janeiro, Brazil

introduction

introduction

focus

Volume Rendering



Mesh Processing

introduction

volume rendering

Goal

- Better insight of the volume data

Source

- MRI/CT acquisition
- CFD numerical simulations

Applications

- Geo-sciences
- Fluid Simulations
- Medical Images

introduction

volume rendering

Goal

- Better insight of the volume data

Source

- MRI/CT acquisition
- CFD numerical simulations

Applications

- Geo-sciences
- Fluid Simulations
- Medical Images

Motivation

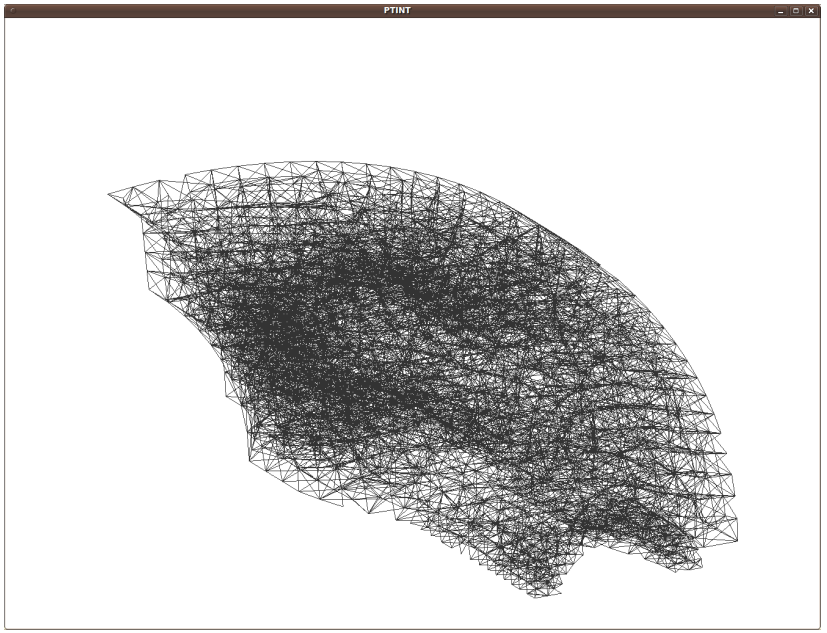
- Regular and Irregular Data
- Ray-Casting and Cell-Projection Algorithms
- Volume and Iso-surface Rendering
- GPU Programming

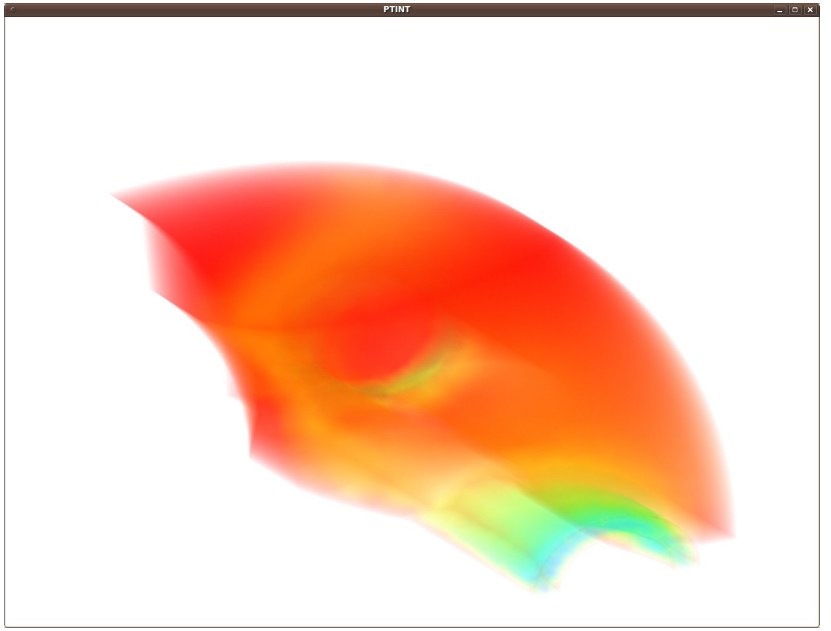
illustration

regular data rendering

illustration

irregular data rendering





introduction

mesh processing

Goal

- Manipulation of meshed surfaces

Source

- Generated by artists
- Scanned from real-world objects

Applications

- CAD
- CG Arts

introduction

mesh processing

Goal

- Manipulation of meshed surfaces

Source

- Generated by artists
- Scanned from real-world objects

Applications

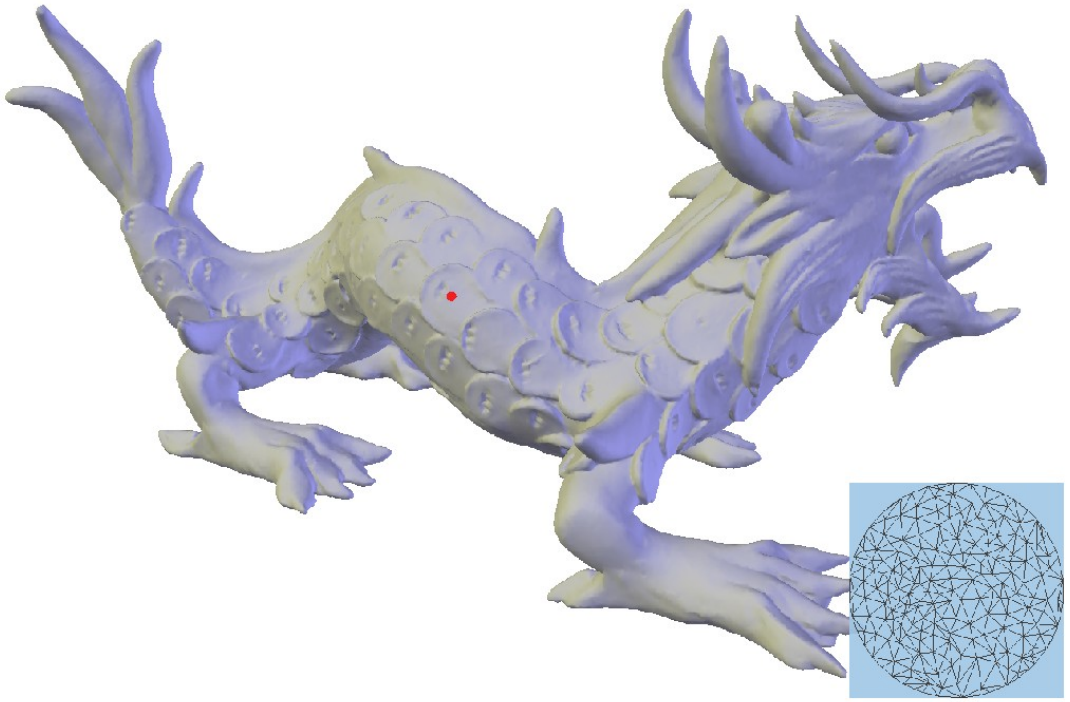
- CAD
- CG Arts

Motivation

- Similarity Descriptor
- Propagate Mesh Processing

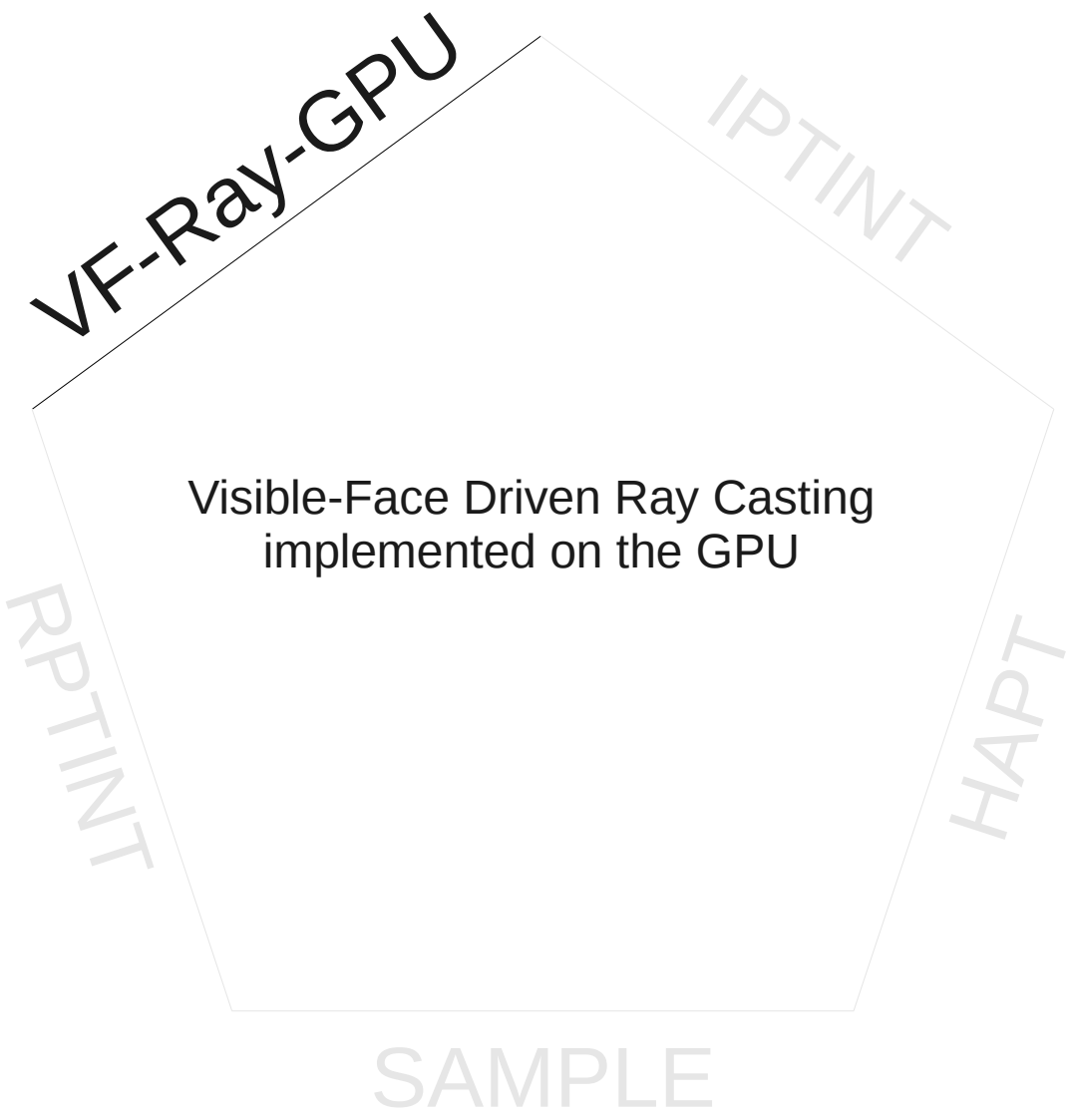
illustration

mesh parameterization



volume rendering

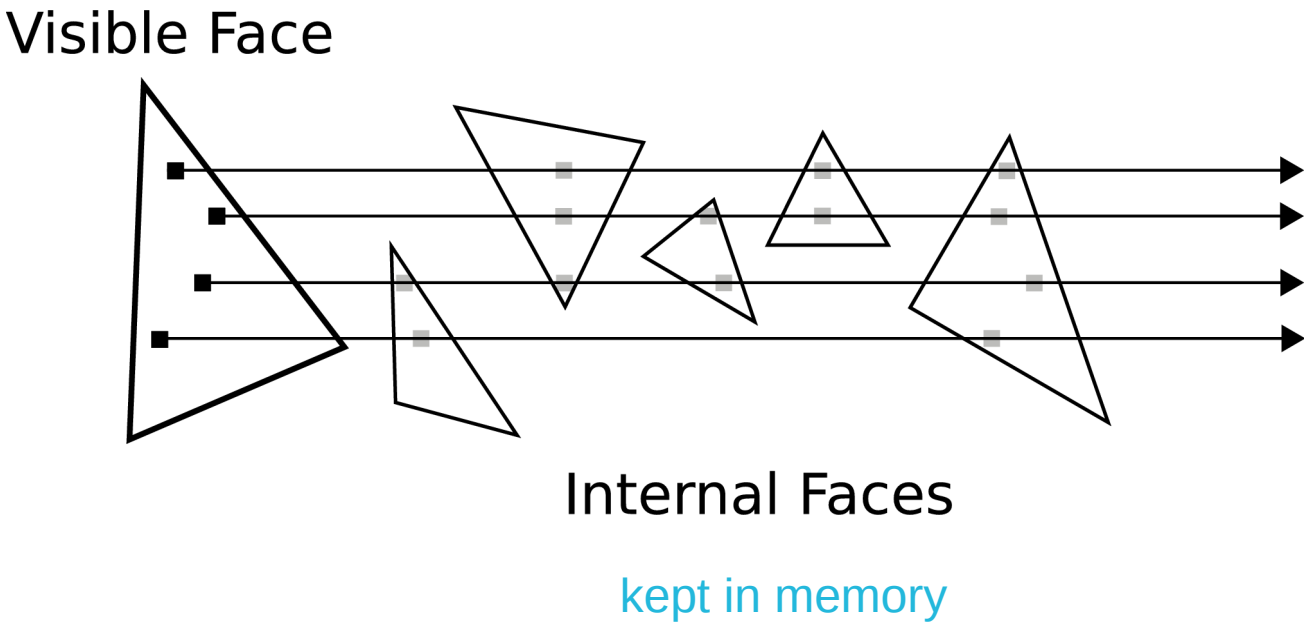
algorithm I



VF-Ray

Visible-Face Driven Ray Casting *

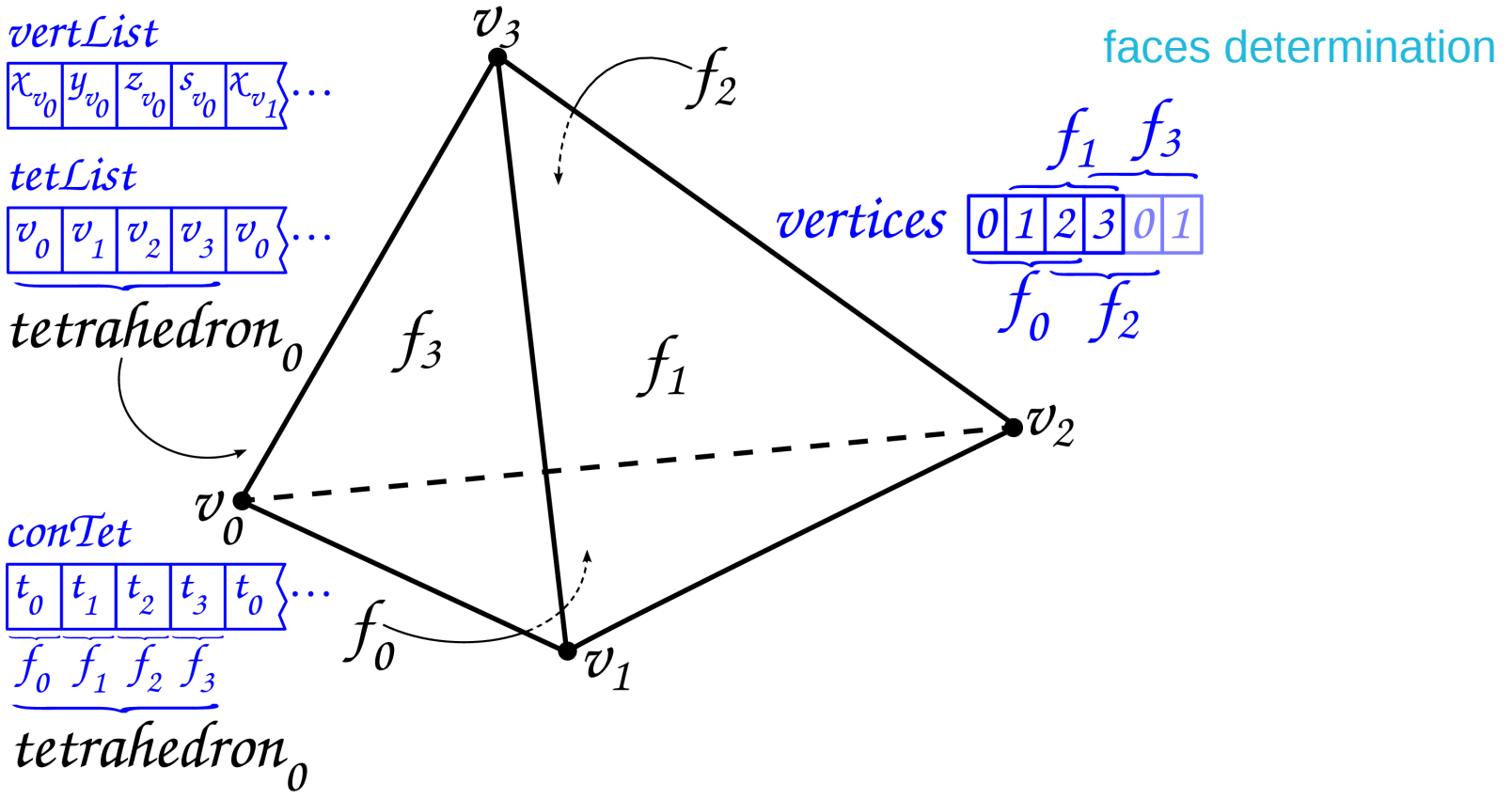
basic idea



* RIBEIRO, S., MAXIMO, A., BENTES, C., OLIVEIRA, A., FARIAS, R. **"Memory-Aware and Efficient Ray-Casting Algorithm"**, in Proceedings of *SIBGRAPI (IEEE Computer Society)*, pp. 147-154, Belo Horizonte, Minas Gerais, Brazil, 2007.

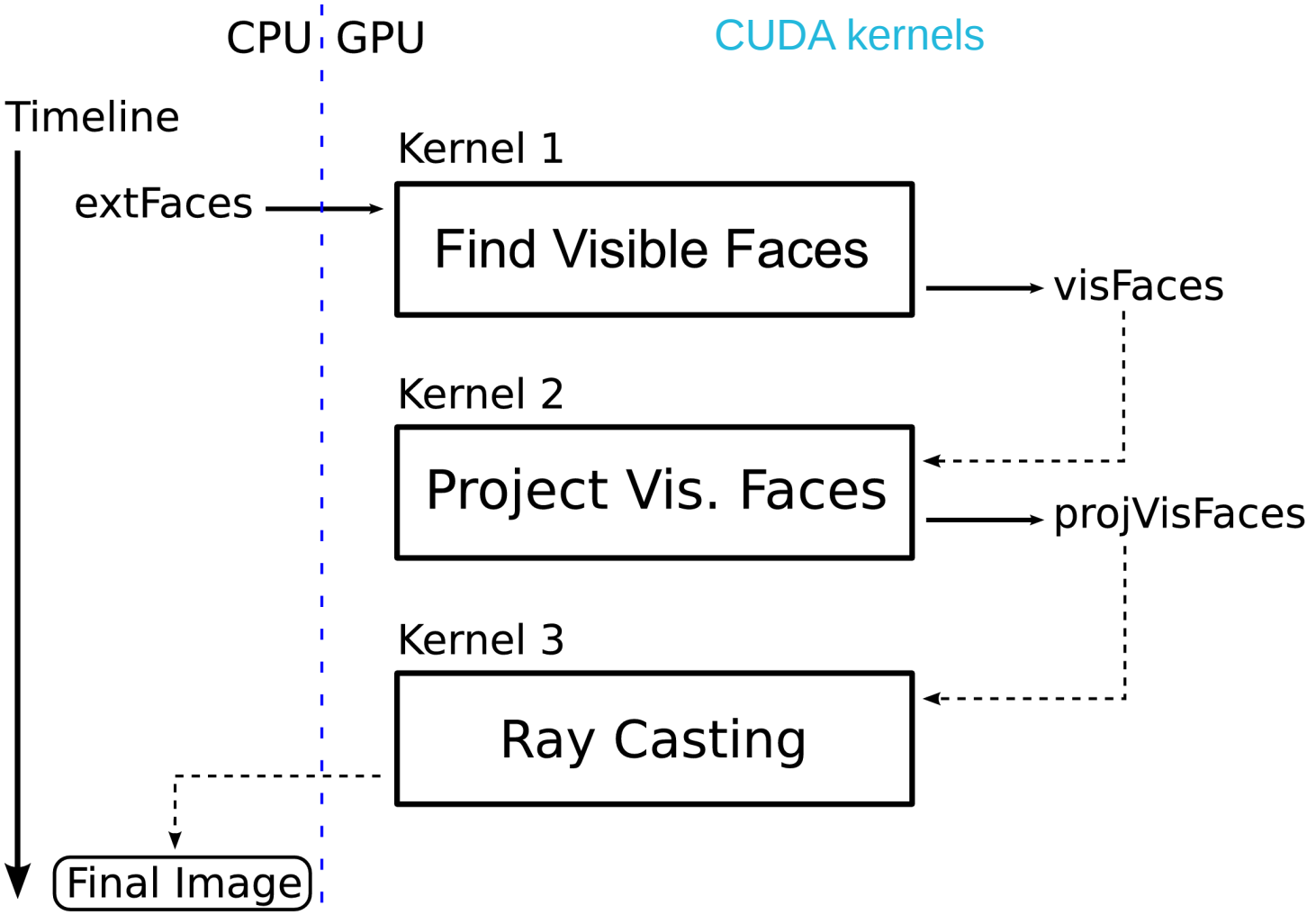
VF-Ray-GPU

basic data structures



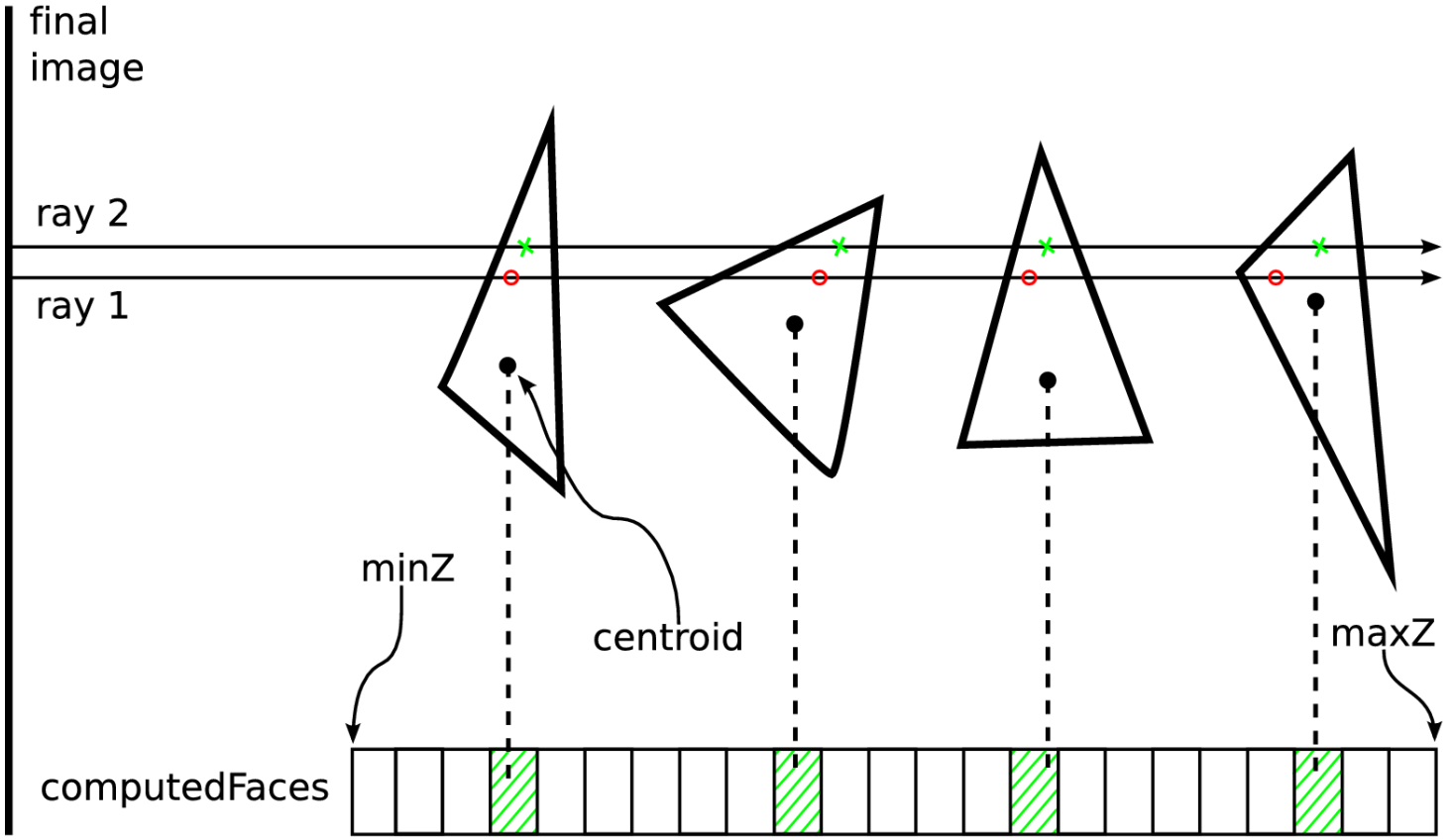
VF-Ray-GPU

overview



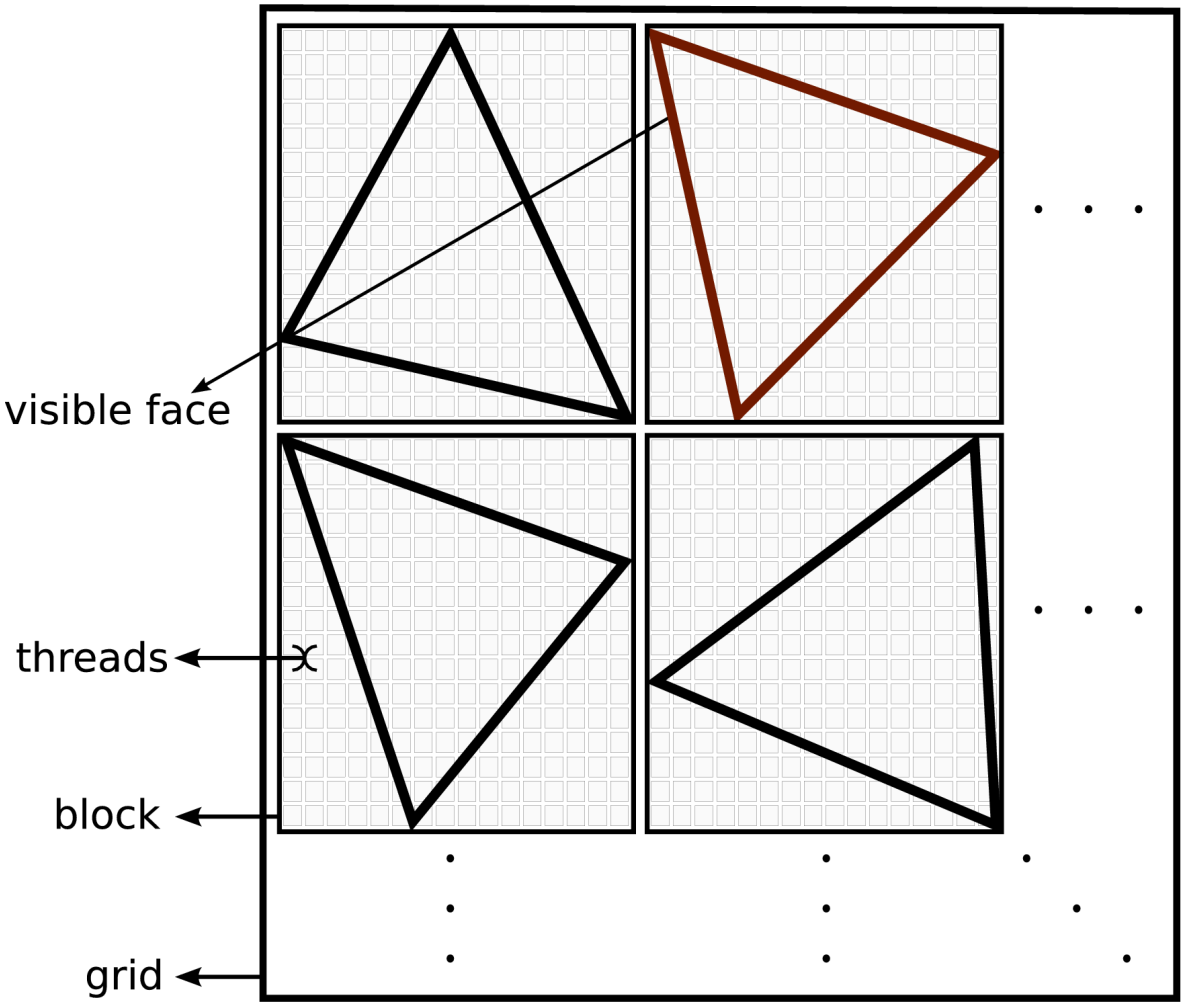
VF-Ray-GPU

hashed buffer



VF-Ray-GPU

threads scheme



VF-Ray-GPU

results

Memory aspects

Algorithm	*Bytes/Tet*	*Bytes/Pixel*	*Pre-Int*
VICP	456	–	16
HARC	160	96	16
HARC (INT)	96	96	1
VF-Ray-GPU	38	–	–

less than half
memory consumption

MB

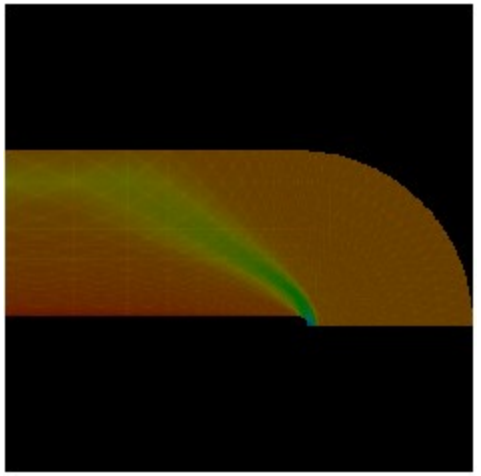
VF-Ray-GPU

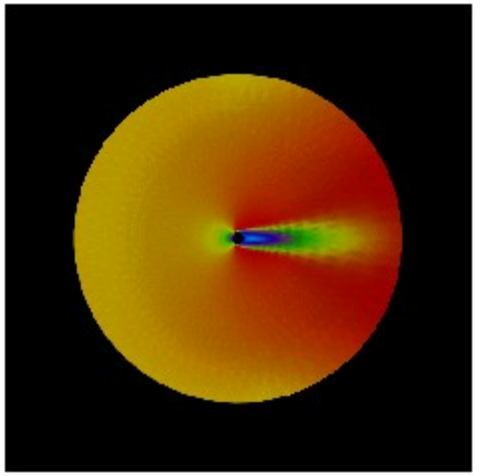
results

Memory/Timing comparisons

Algorithm	blunt		post	
	Memory (KB)	*Time (ms)*	*Memory (KB)*	*Time (ms)*
VICP	118,524	190	249,928	546
HARC	72,267	18	123,245	33
HARC (INT)	22,636	32	50,248	51
VF-Ray-GPU	7,029	186	19,494	370

~10x less memory





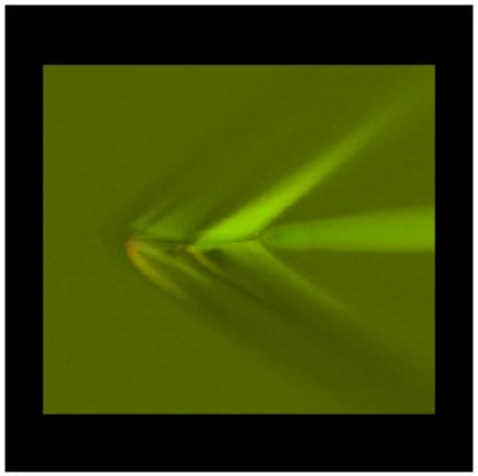
VF-Ray-GPU

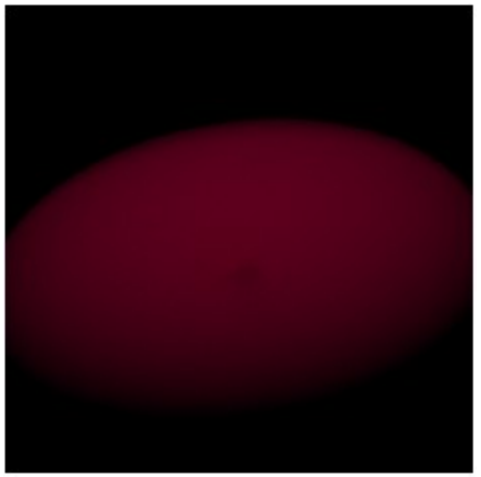
results

3/3

VF-Ray x VF-Ray-GPU

Dataset	Memory (MB)		Time (ms)	
	CPU	GPU	CPU	GPU
fighter+	876	426	58326	10035
f16	499	239	51235	8815





half memory consumption

~7x faster

VF-Ray-GPU

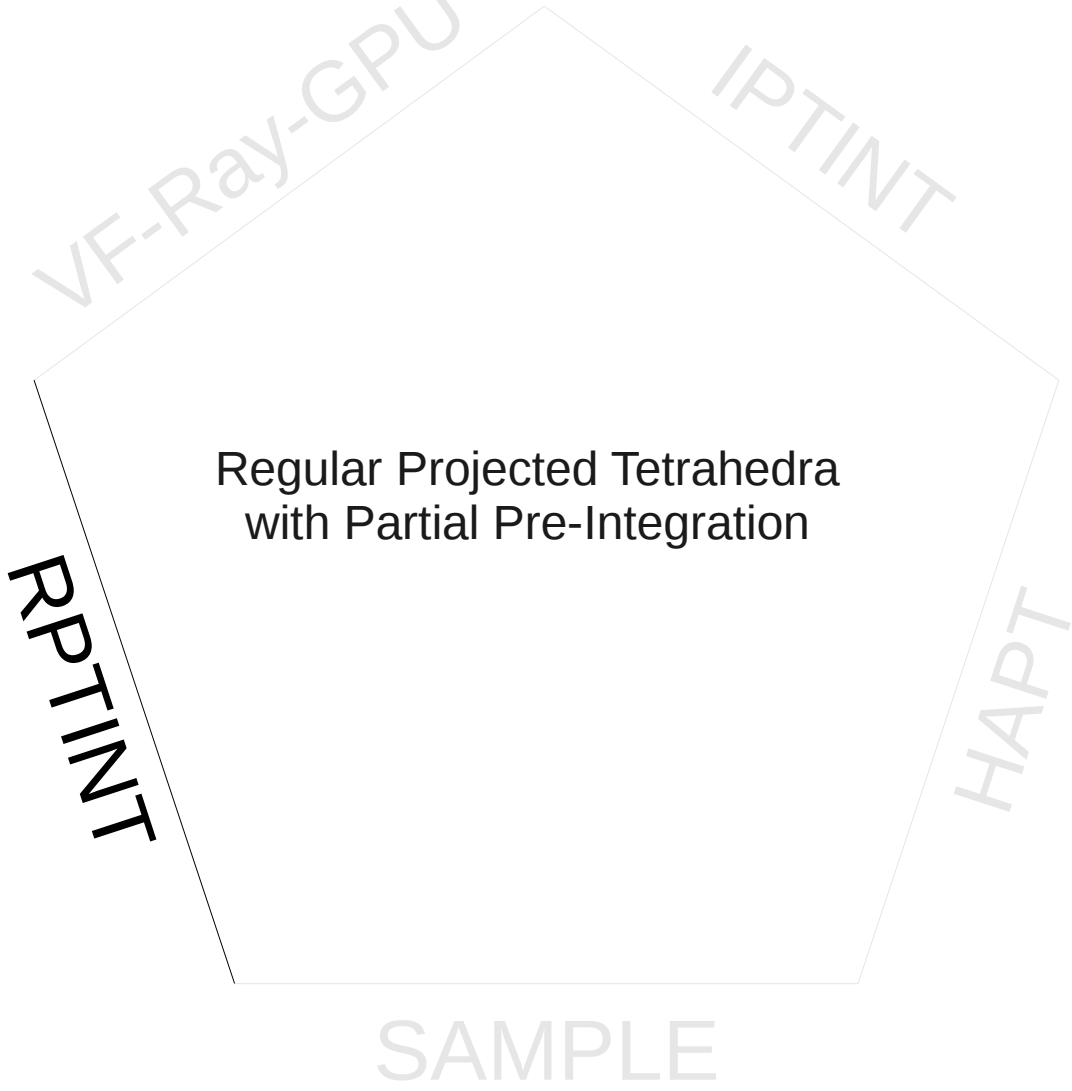
Visible-Face Driven Ray Casting implemented on the GPU *

code & publication

http://code.google.com/p/vfray

* MAXIMO, A., RIBEIRO, S., BENTES, C., OLIVEIRA, A., FARIAS, R. **“Memory-Efficient GPU-Based Ray Casting for Unstructured Volume Rendering”**, *in Proceedings of VG-PBG (Eurographics Association)*, pp. 155-162, Los Angeles, California, USA, 2008.

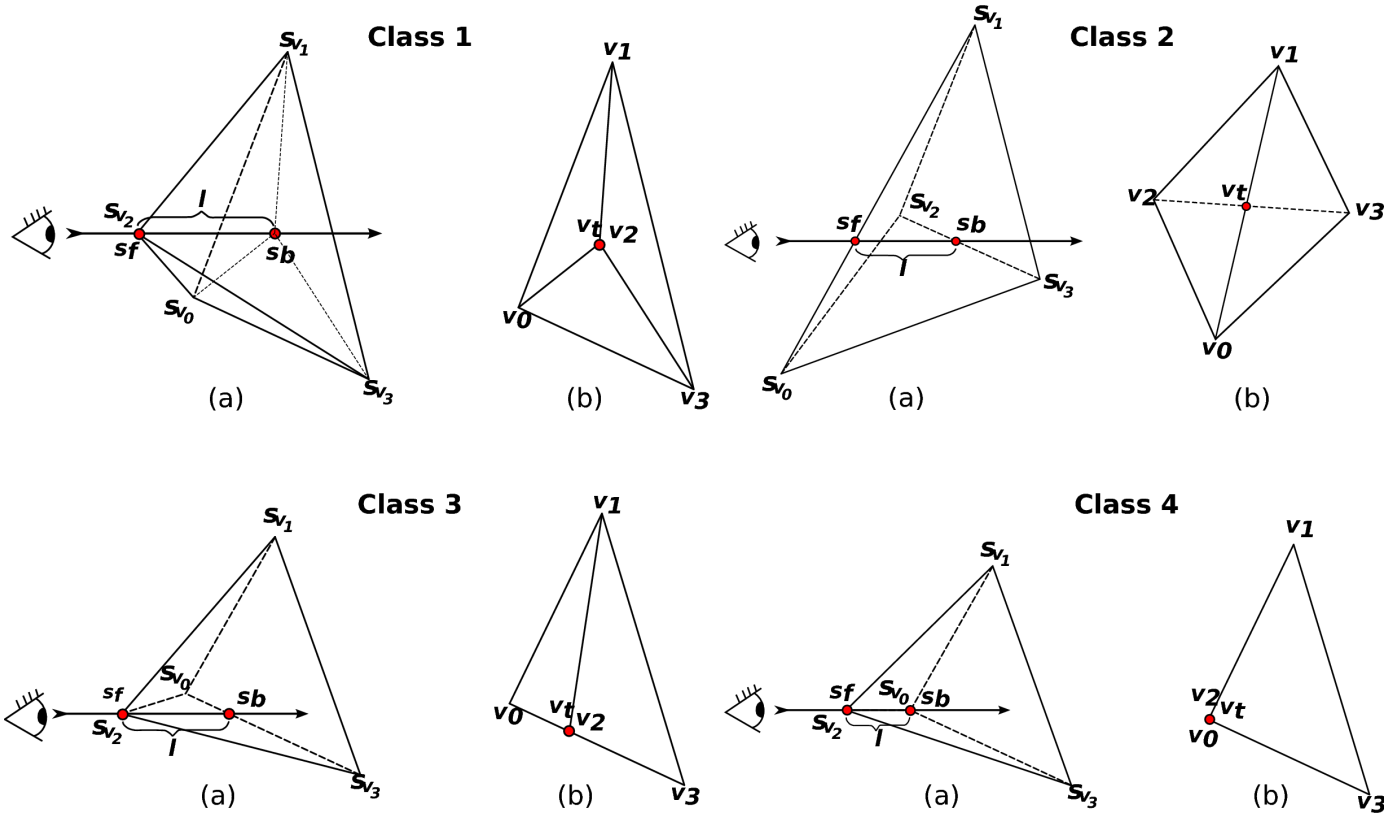
algorithm II



PT

Projected Tetrahedra *

first algorithm

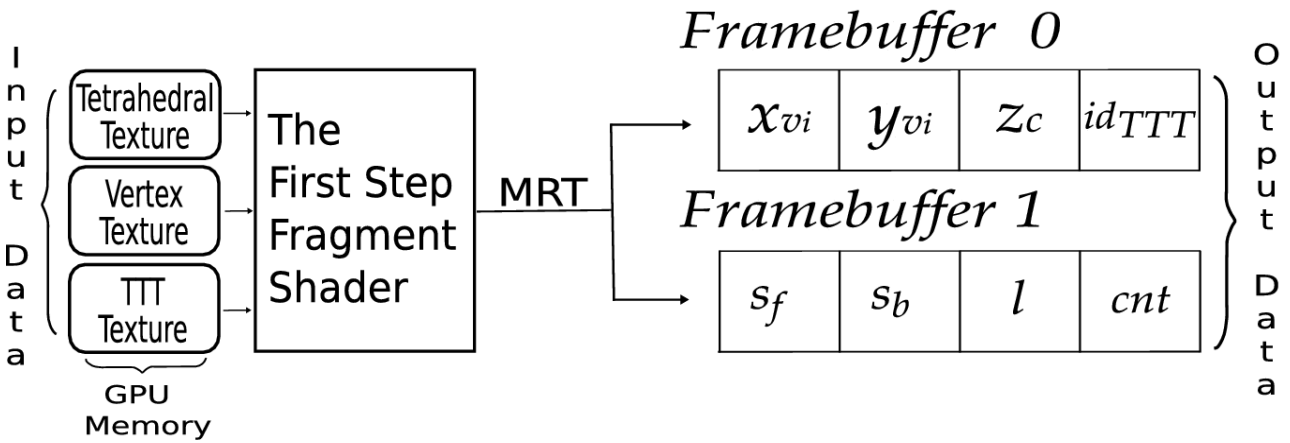


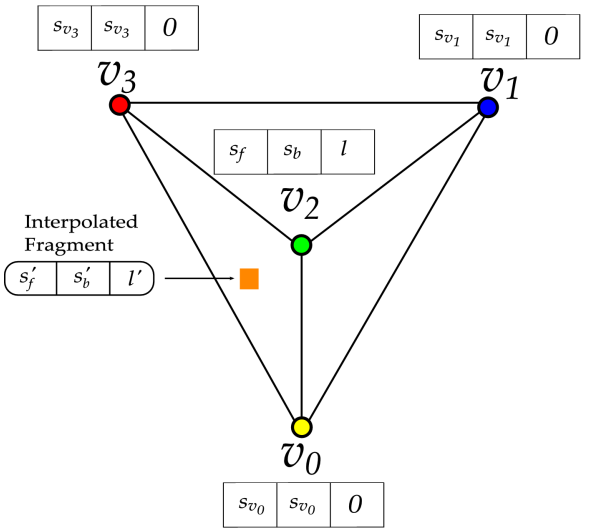
* SHIRLEY, P., TUCHMAN, A. **"A Polygonal Approximation to Direct Scalar Volume Rendering"**, *in Proceedings of SIGGRAPH*, v. 24, n. 5, pp. 63-70, 1990.

PTINT

Projected Tetrahedra with Partial Pre-Integration *

a two-step approach



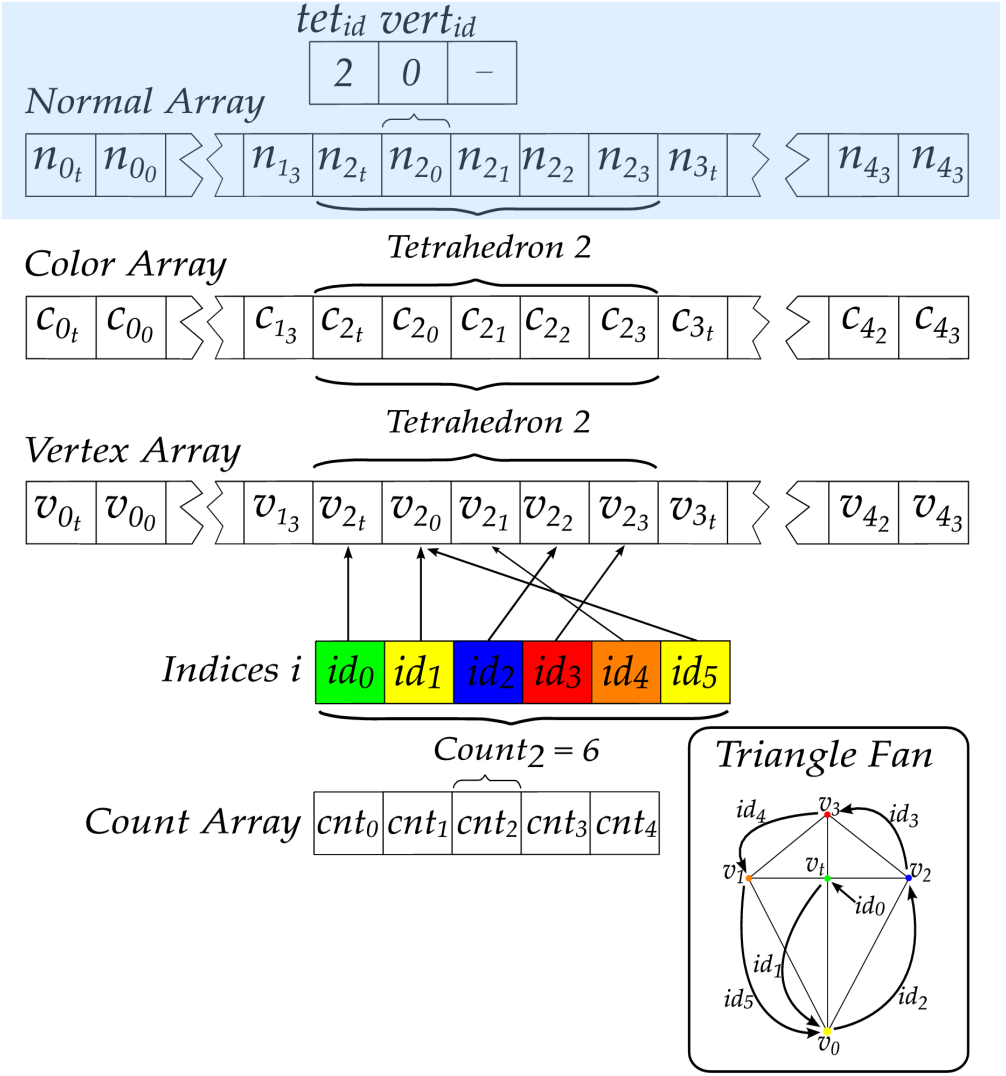


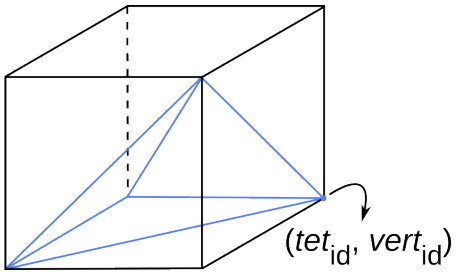
* MARROQUIM, R., MAXIMO, A., FARIAS, R., ESPERANÇA, C. **"GPU-Based Cell Projection for Interactive Volume Rendering"**, in Proceedings of *SIBGRAPI (IEEE Computer Society)*, pp. 147-154, Manaus, Amazonas, 2006. *Best Paper.*

RPTINT

data structures

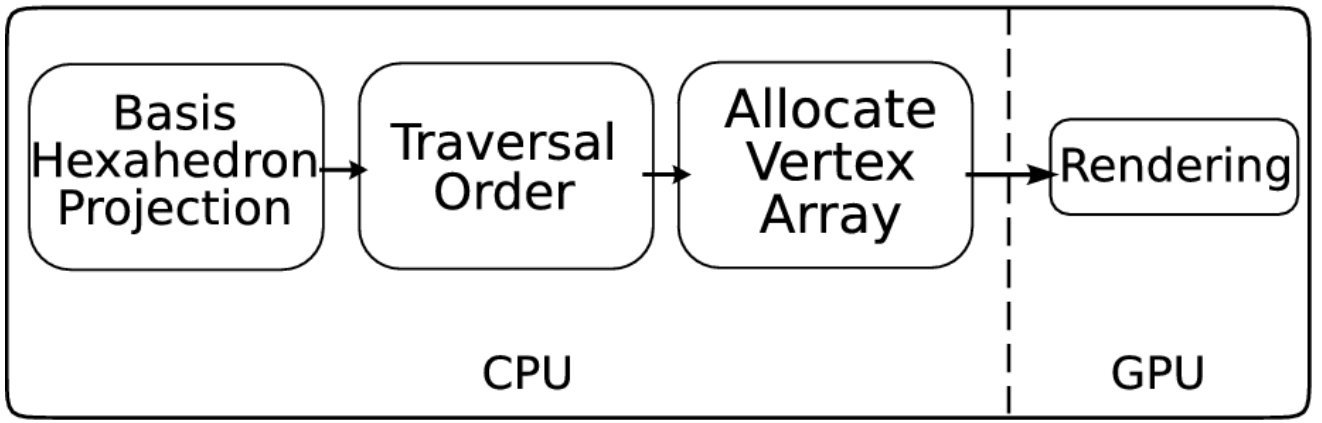
+





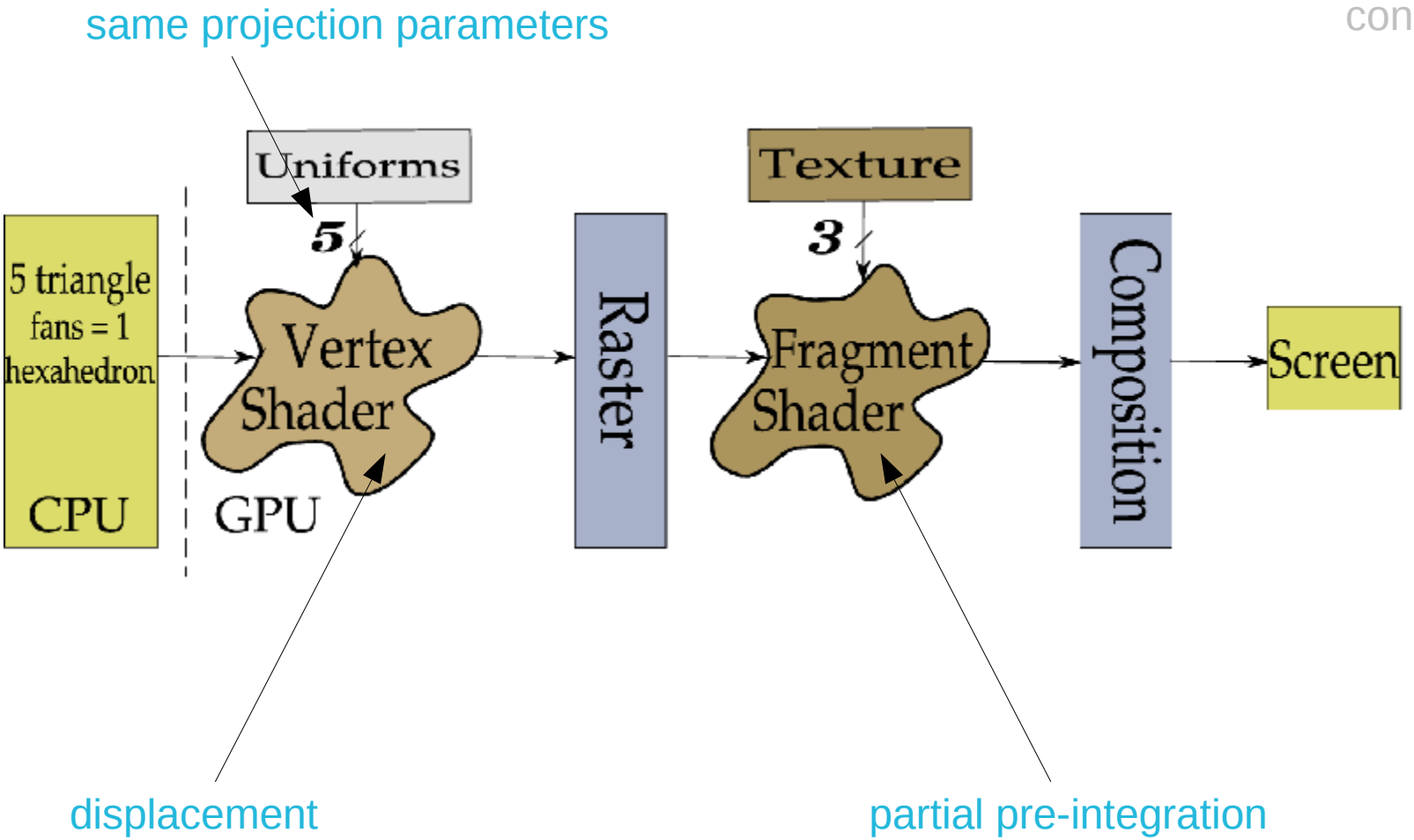
RPTINT

overview



RPTINT

4th step in GPU :: rendering pipeline

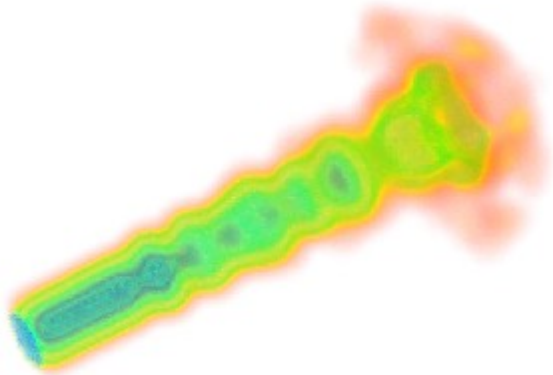


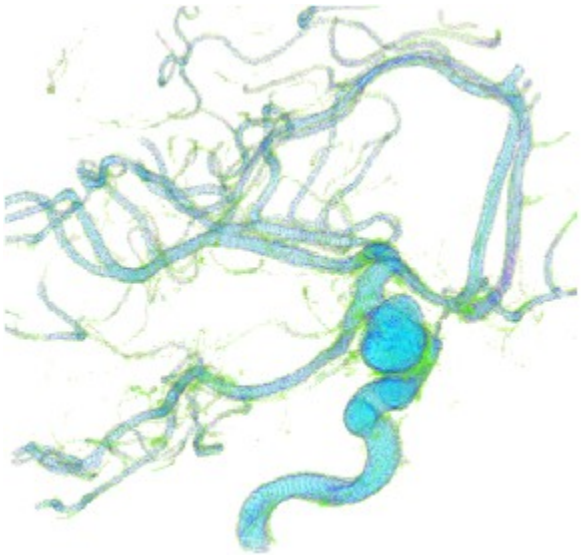
RPTINT

results :: performance measures

256x256x256

Dataset	# Verts	# Tet	*fps*	*M Tet/s*
fuel	262 K	1.2 M	70.78	88.5 (5.99)
tooth–	1 M	5 M	1.22	13.1 (6.30)
tooth	10 M	52 M	0.24	12.7 (6.61)
foot	16 M	83 M	0.81	67.7 (6.23)
skull	16 M	83 M	0.61	51.7 (6.25)
aneurysm	16 M	83 M	2.42	201 (5.35)



RPTINT

results :: setup x render

Dataset	Setup	Render	% Total
fuel	0.005 s	0.009 s	64.28 %
tooth	1.377 s	6.484 s	82.47 %
foot	4.556 s	9.074 s	66.57 %
skull	4.606 s	8.593 s	65.10 %
aneurysm	0.199 s	0.210 s	51.34 %

1st – 3rd steps (setup)
x
4th step (rendering)

theoretical limit

RPTINT

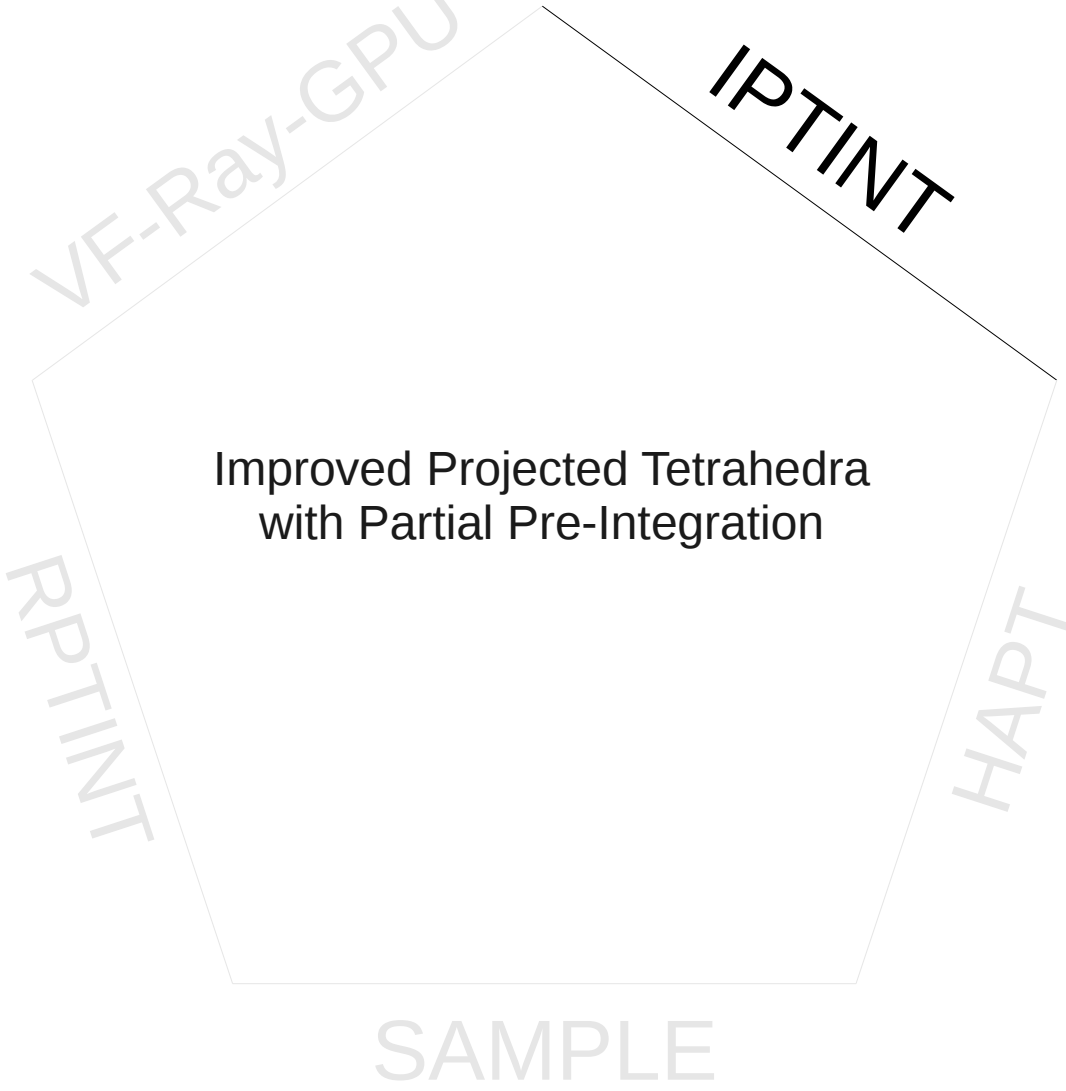
Regular Projected Tetrahedra
with Partial Pre-Integration *

code & publication

http://code.google.com/p/rptint

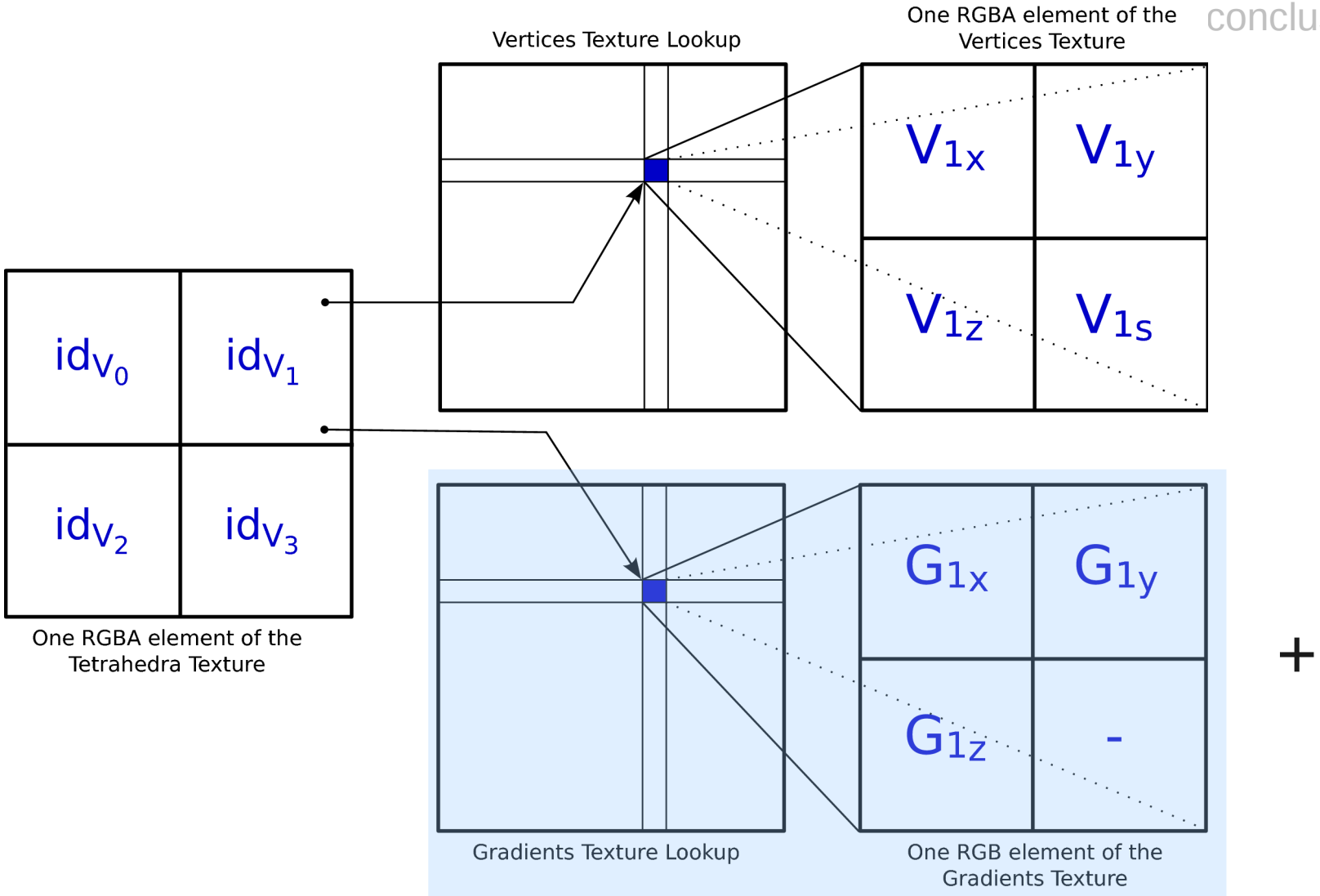
* MAXIMO, A., MARROQUIM, R., FARIAS, R., ESPERANÇA, C. **“GPU-Based Cell Projection for Large Structured Datasets”**, *in Proceedings of GRAPP (INSTICC)*, pp. 312-322, Barcelona, Spain, 2007.

algorithm III



IPTINT

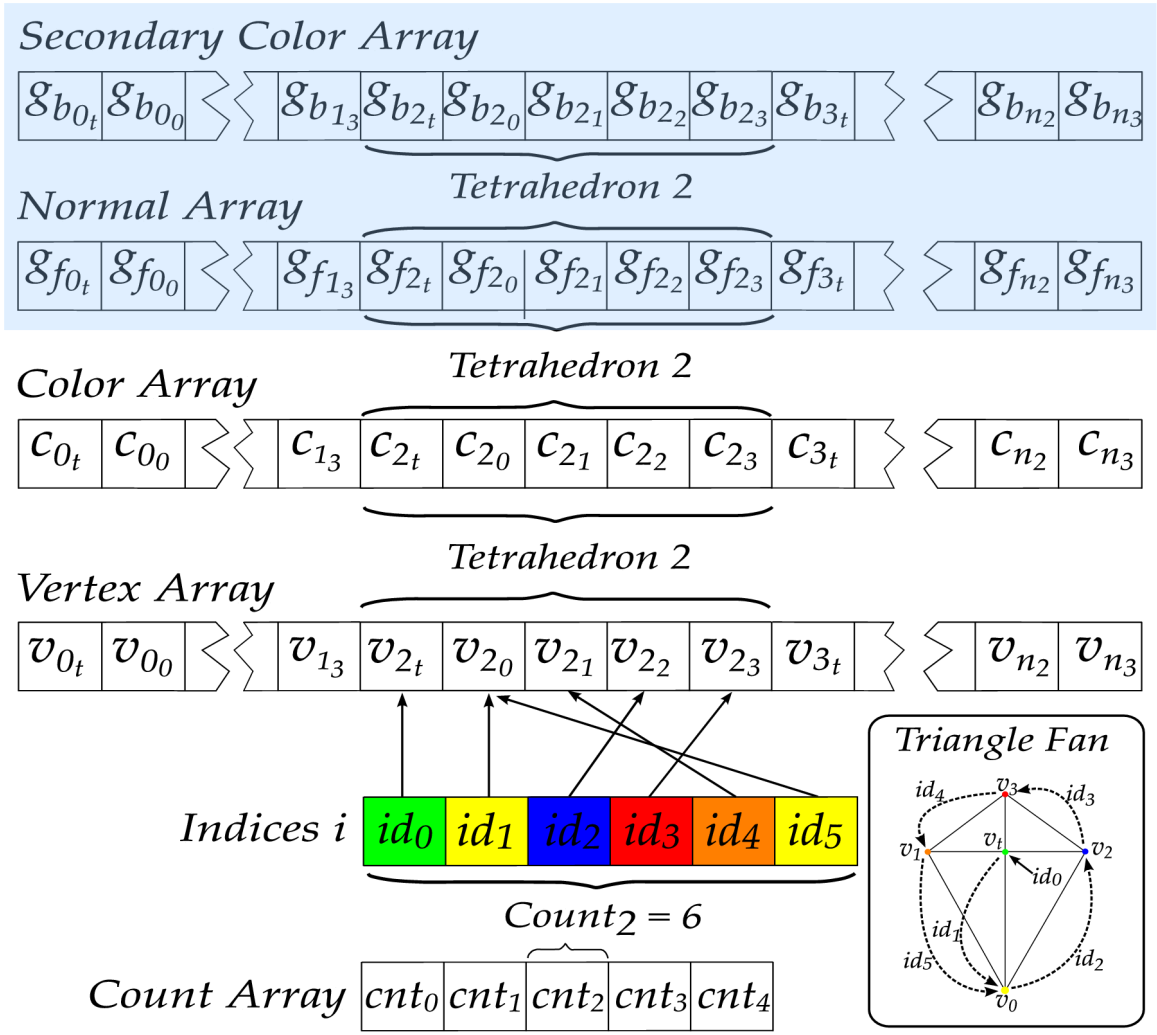
data structures



IPTINT

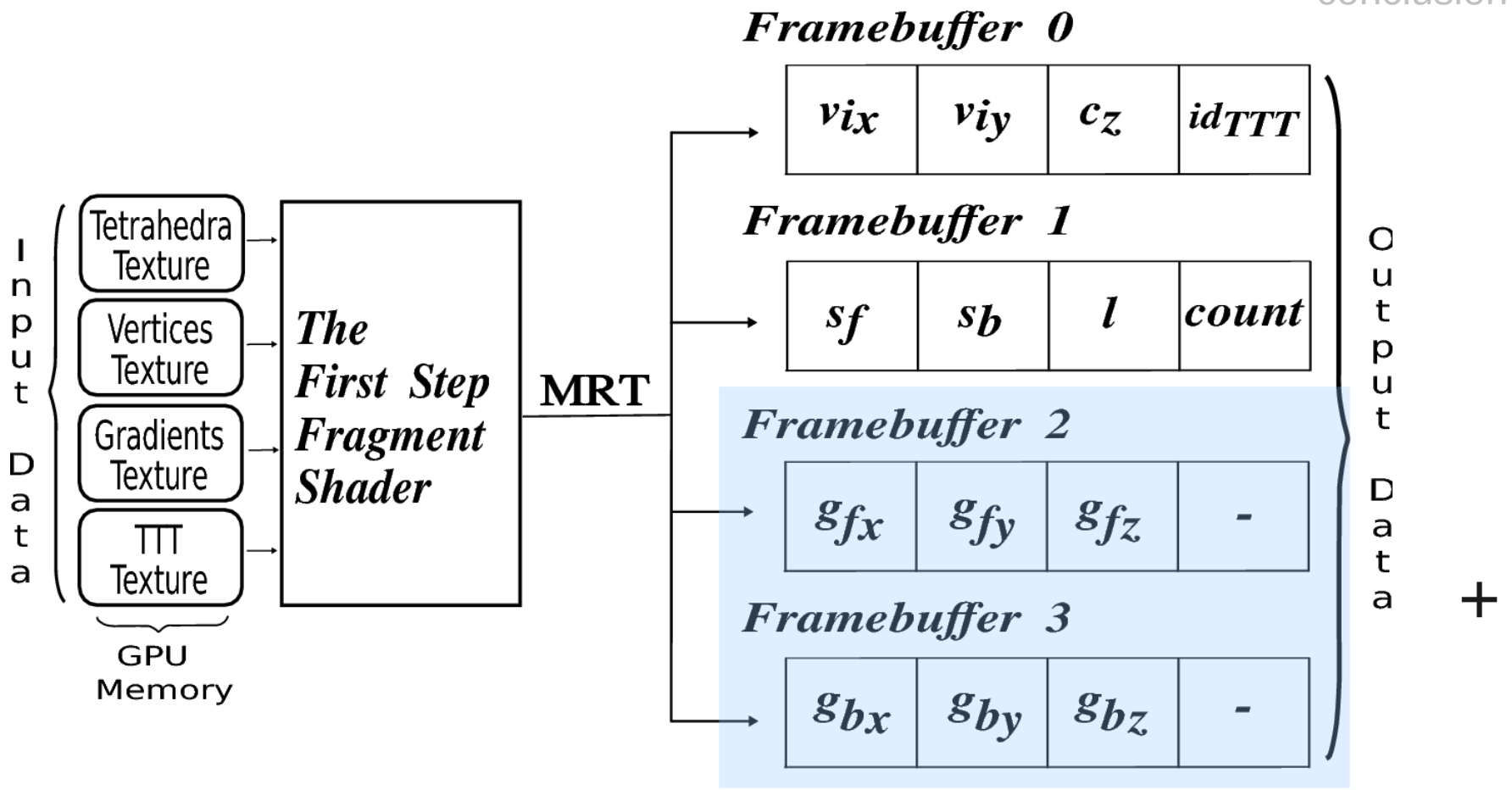
data structures

+



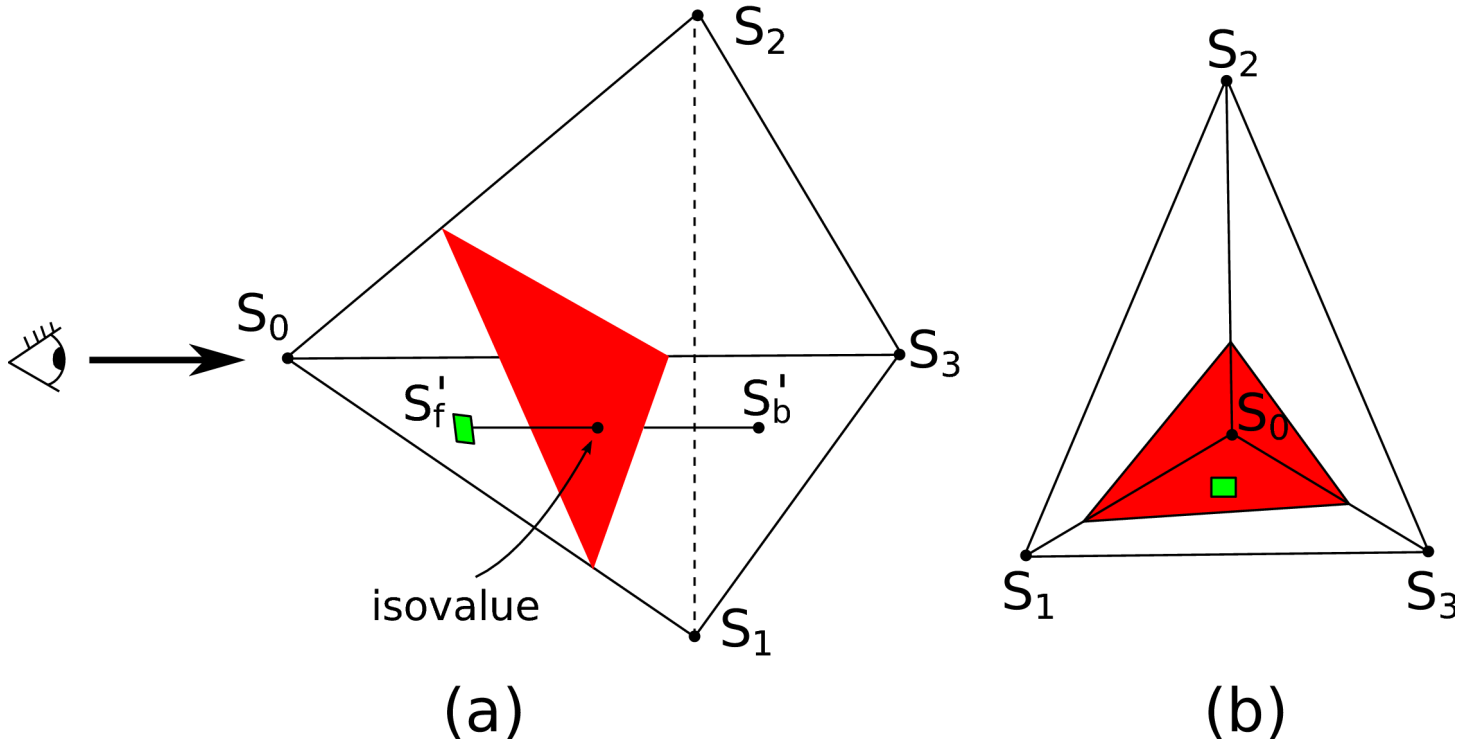
IPTINT

first step



IPTINT

iso-surface rendering



normal ~ interpolated gradient

IPTINT

results :: timing comparisons

Algorithm	blunt	post
PTINT	10.76 *fps*	4.35 *fps*
IPTINT	4.97 *fps*	2.09 *fps*
GATOR	4.07 *fps*	1.51 *fps*
VICP (GPU)	5.20 *fps*	1.93 *fps*
VICP (CPU)	1.82 *fps*	0.57 *fps*
VICP (Balanced)	4.10 *fps*	1.11 *fps*
HARC	4.47 *fps*	8.63 *fps*
HARC (INT)	4.94 *fps*	5.93 *fps*
HAVIS	2.36 *fps*	0.79 *fps*
HAVS (k=2)	6.09 *fps*	3.09 *fps*
HAVS (k=6)	3.45 *fps*	2.09 *fps*

50% slower than original PTINT

GPU hybrid rendering

~2x faster

IPTINT

results :: types of rendering

Dataset	Size		*Basic*		*+INT*		*+ISO*	
	# Verts	# Tet	*fps*	*M Tet/s*	*fps*	*M Tet/s*	*fps*	*M Tet/s*
blunt	40 K	187 K	12.75	2.38	10.76	2.01	4.97	0.93
comb	47 K	215 K	11.12	2.38	8.98	1.92	3.71	0.79
post	110 K	513 K	4.91	2.51	4.35	2.23	2.09	1.07
spx+	150 K	828 K	4.68	2.55	4.52	2.47	1.23	1.02
fuel	262 K	1.25 M	26.01	2.10	22.99	1.86	9.95	0.80
neghip	262 K	1.25 M	3.59	2.34	3.11	2.03	1.27	0.83

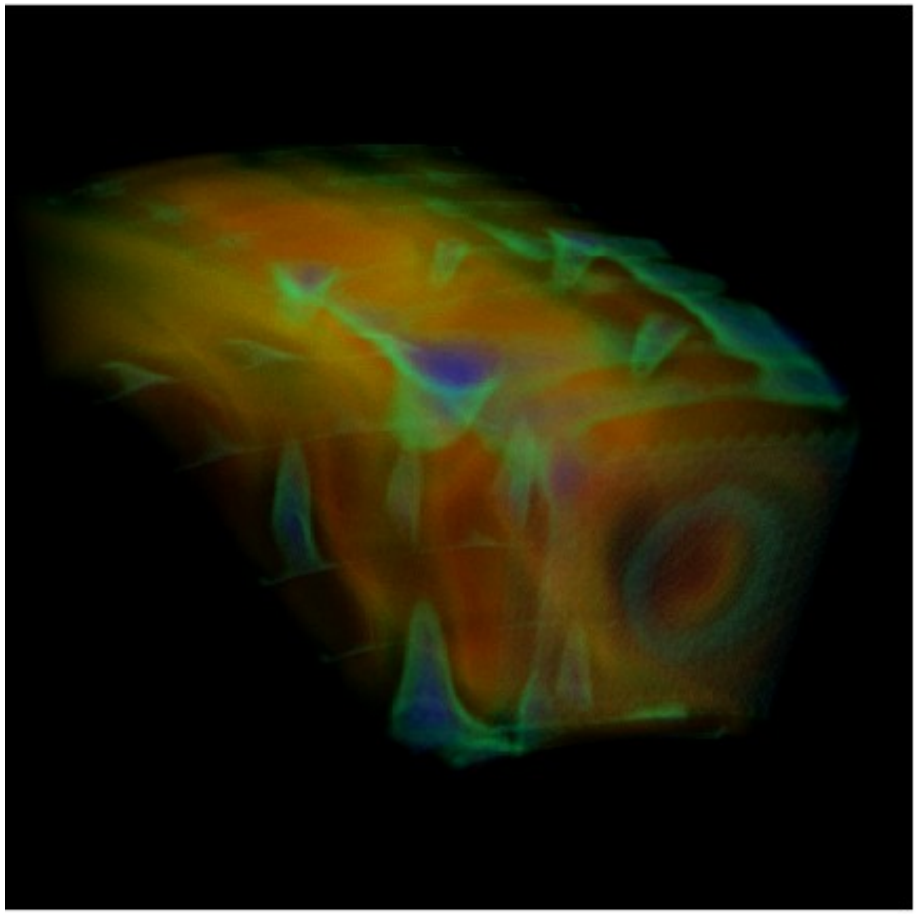
Original PT in GPU

Original PTINT

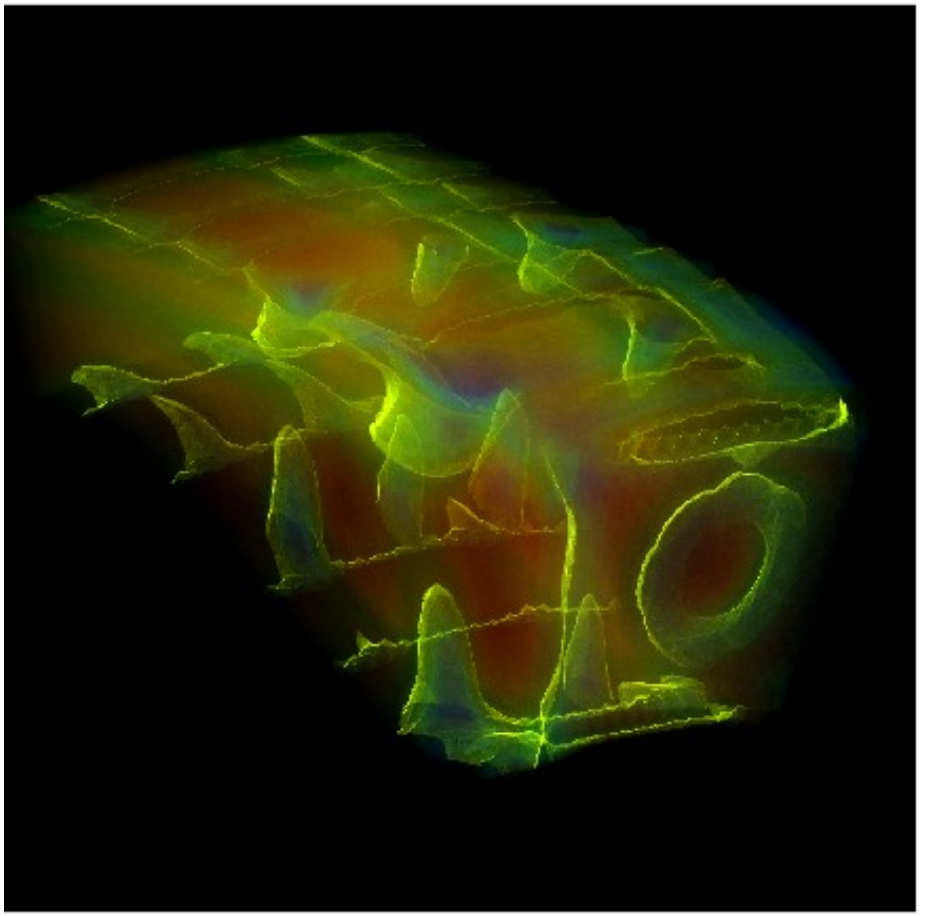
IPTINT

IPTINT

results :: renderings



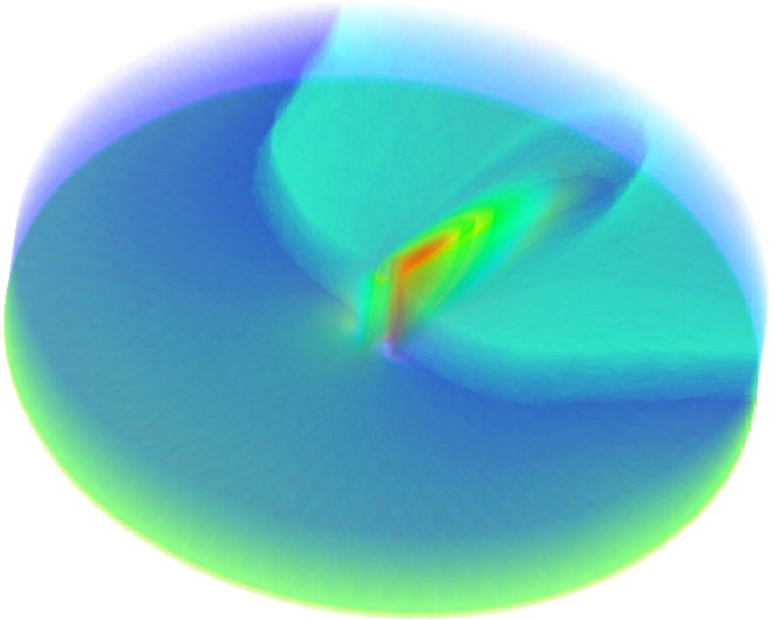
comb (215 K Tets)
+int @8.98 fps



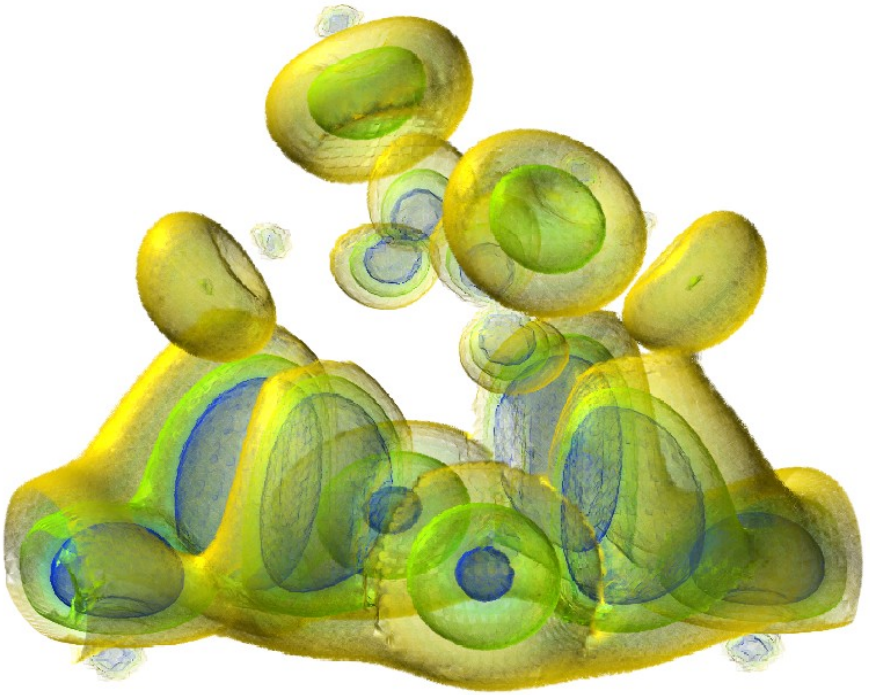
comb (215 K Tets)
+int +iso @3.71 fps

IPTINT

results :: renderings



post (513 K Tets)
+int @4.35 fps



neghip (1.25 M Tets)
+int +iso @1.27 fps

IPTINT

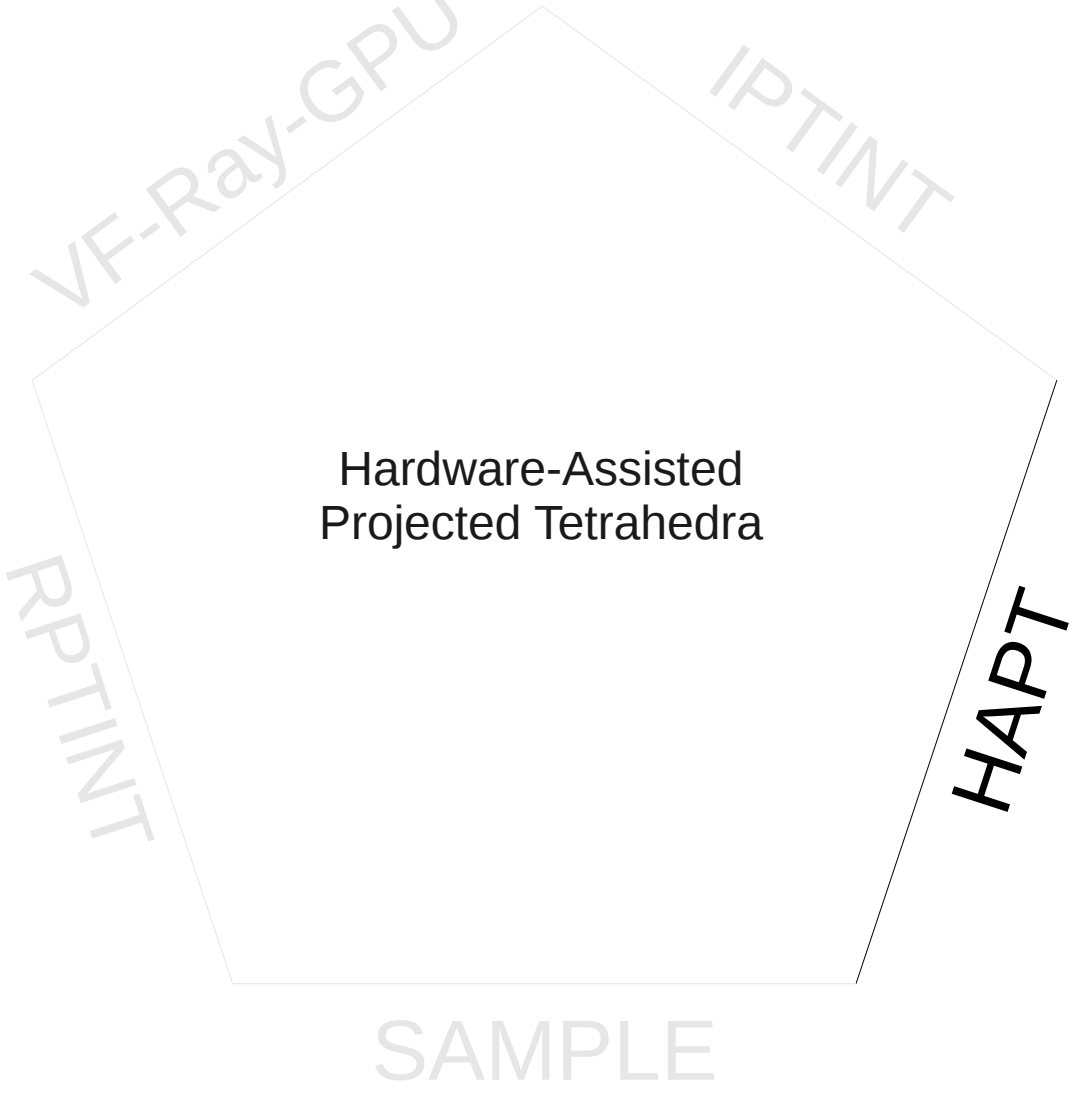
Improved Projected Tetrahedra with Partial Pre-Integration *

code & publication

http://code.google.com/p/ptint

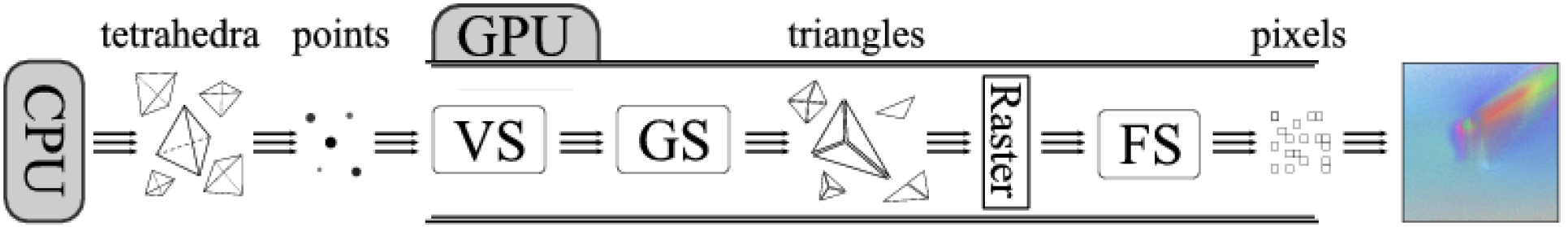
* MARROQUIM, R., MAXIMO, A., FARIAS, R., ESPERANÇA, C. **“Volume and Isosurface Rendering with GPU-Accelerated Cell Projection”**, *Computer Graphics Forum (Best Paper selection of SIBGRAPI 2006)*, v. 27, pp. 24-35, 2008.

algorithm IV



HAPT

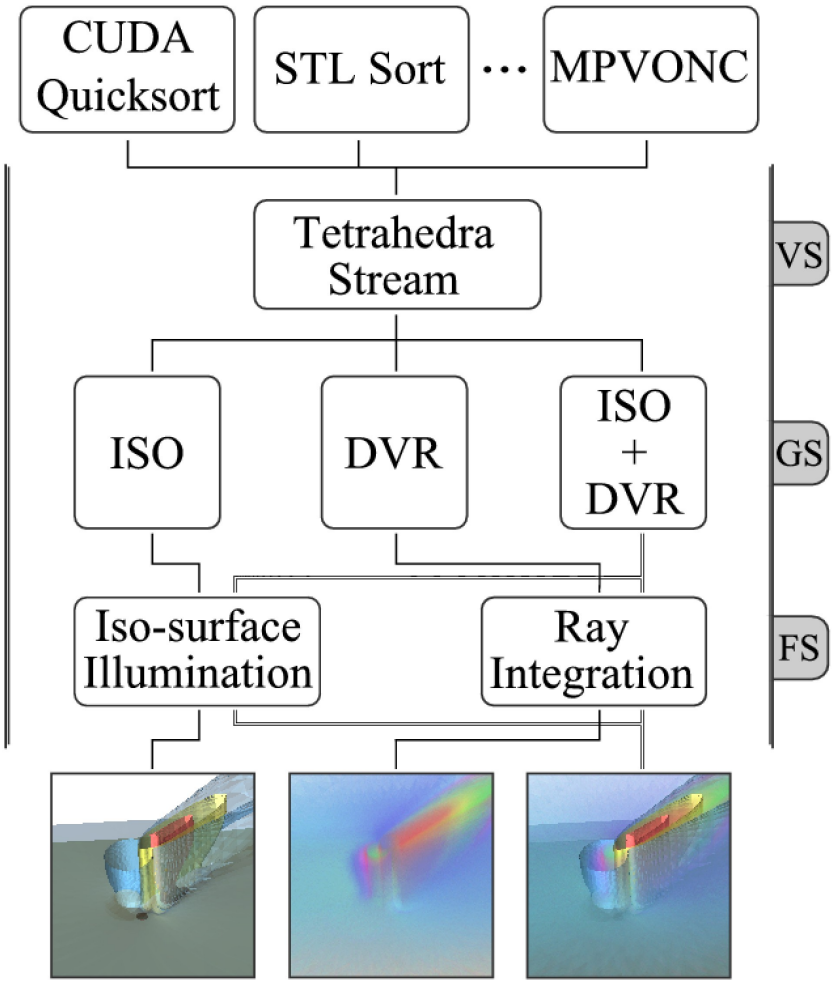
pipeline

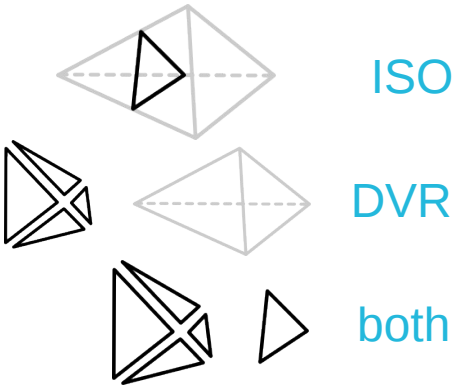


HAPT

framework

easy change of its modules





HAPT

results :: performance measures

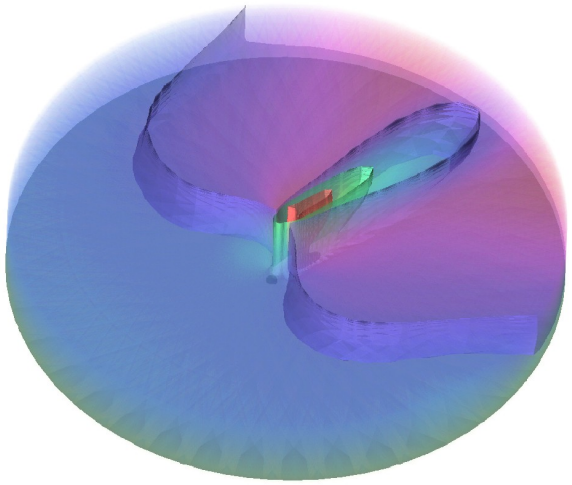
Dataset	Size		*DVR*		*ISO*		*DVR + ISO*	
	# Verts	# Tet	*fps*	*M Tet/s*	*fps*	*M Tet/s*	*fps*	*M Tet/s*
blunt	40 K	187 K	19.2	3.59	25.5	4.78	7.7	1.44
post	110 K	513 K	8.1	4.15	11.9	6.10	3.0	1.51
spx+	150 K	828 K	7.4	6.11	8.2	6.76	1.9	1.57
delta	211 K	1 M	4.5	4.52	6.0	6.01	1.5	1.51
torso	168 K	1.08 M	5.6	6.08	7.2	7.78	1.7	1.82
fighter	256 K	1.40 M	4.2	5.83	5.0	7.06	1.1	1.60
turbjet	212 K	1.01 M	17.5	17.67	n/a	n/a	n/a	n/a

good frame rates
1 M cells dataset

HAPT

results :: sorting

Dataset	Maximum Error	Average Error	Different Pixels
blunt	1.961%	0.4069%	6.04%
post	2.353%	0.4245%	33.13%
spx+	1.569%	0.3985%	8.13%
delta	5.098%	0.5895%	14.25%
torso	1.176%	0.3933%	1.51%
fighter	1.569%	0.3943%	2.02%



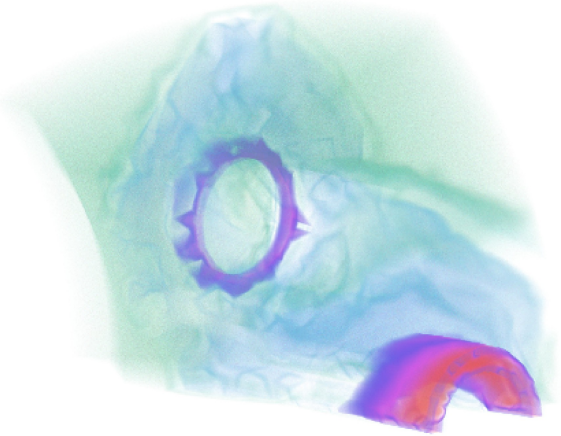
Centroid
x
MPVONC

1/3 of the pixels

0.4% ~ 1 unit [0, 255]

HAPT

results :: timing comparisons



spx (828 K Tets)
partial pre-integration

Algorithm	*Sort*	*Draw*	*fps*	*M Tet/s*
HAPT^Q	0.03 s	0.09 s	7.4	6.11
HAPT^B	0.04 s	0.09 s	6.9	5.73
HAPT^S	0.08 s	0.09 s	5.4	4.50
HAPT^M	0.13 s	0.09 s	4.4	3.61
HAVS^2	0.09 s	0.11 s	5.0	4.14
HAVS^6	0.09 s	0.12 s	4.7	3.94
PTINT	0.19 s	0.20 s	2.4	2.06
GATOR	0.08 s	0.83 s	1.1	0.93
HARC	n/a	0.22 s	4.6	3.82
HARC (INT)	n/a	0.28 s	3.5	2.90

50% faster

HAPT

results :: timing comparisons

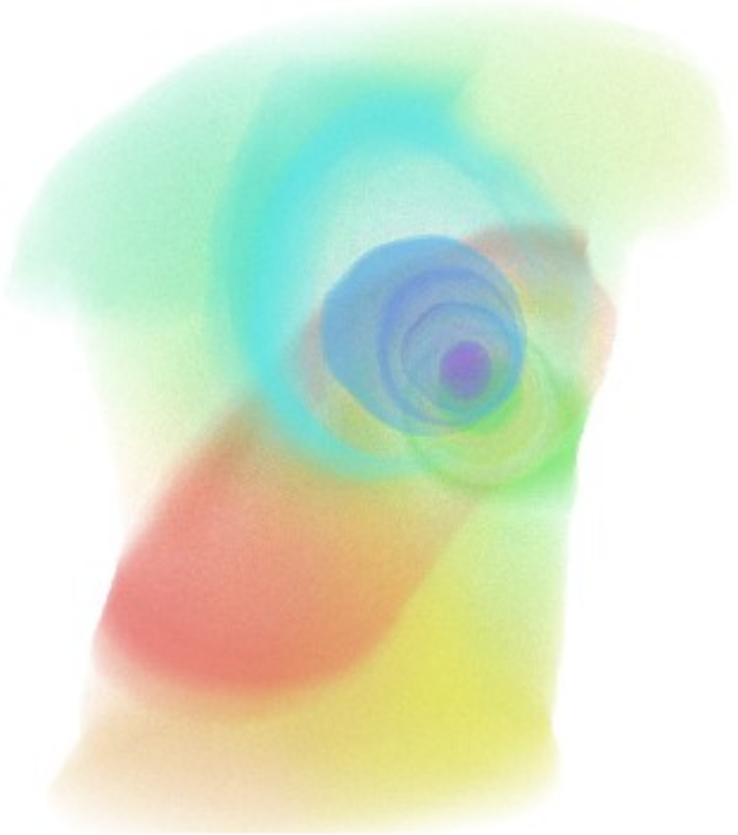
Algorithm	"torso" 1082 K Tet		"fighter" 1403 K Tet	
	fps	*M Tet/s*	*fps*	*M Tet/s*
HAPT^{Q}	5.6	6.08	4.2	5.83
HAPT^{B}	4.3	4.68	3.6	5.09
HAPT^{S}	3.9	4.25	2.9	4.10
HAPT^{M}	1.6	1.73	1.2	1.62
HAVS^{2}	3.7	4.01	2.9	4.12
HAVS^{6}	3.3	3.60	2.7	3.89
PTINT	1.3	1.47	0.9	1.31
GATOR	0.7	0.76	0.4	0.56
HARC	4.8	5.19	3.8	5.33
HARC (INT)	3.9	4.22	3.0	4.21

PT in GPU adaptations

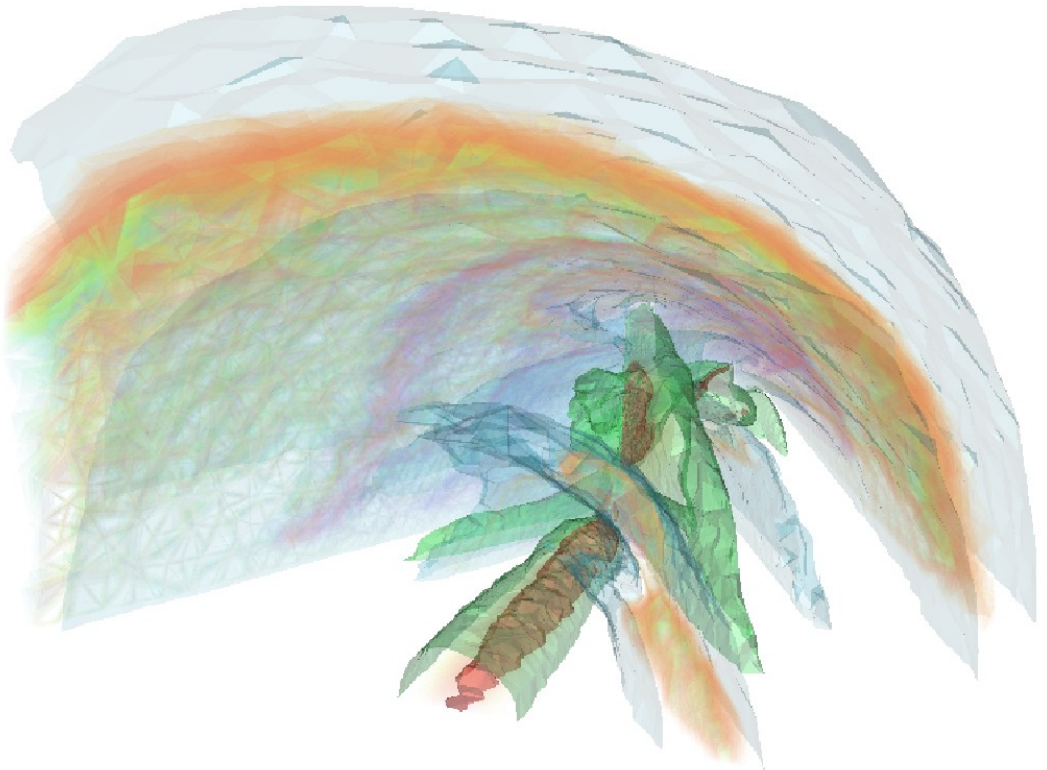
faster even with exact sorting

HAPT

results :: renderings



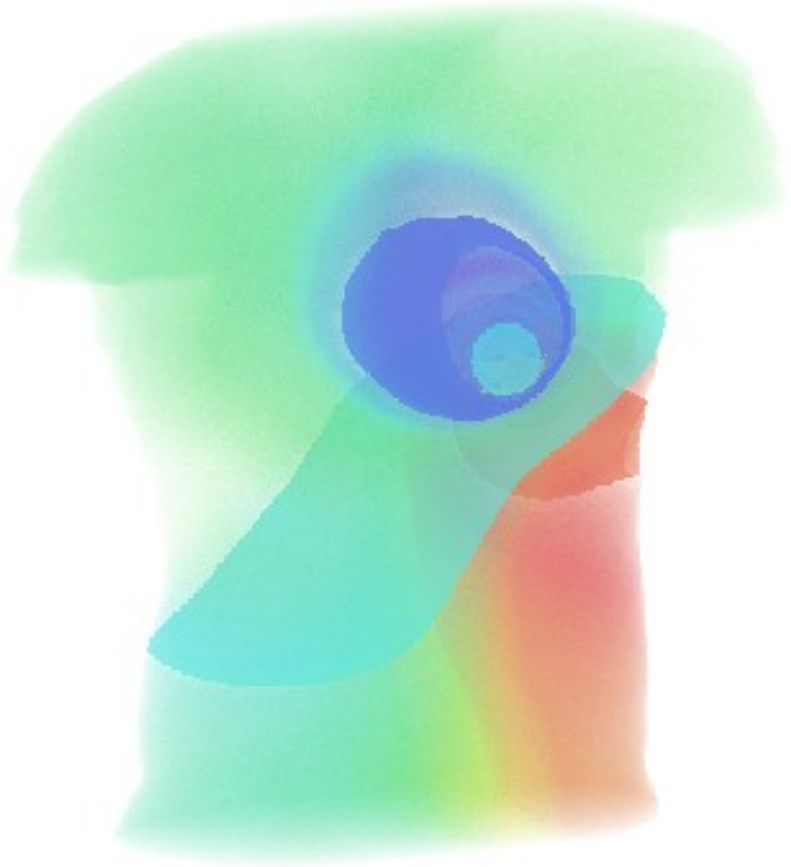
torso (1.08 M Tets)
DVR @5.6 fps



fighter (1.4 M Tets)
DVR+ISO @1.1 fps

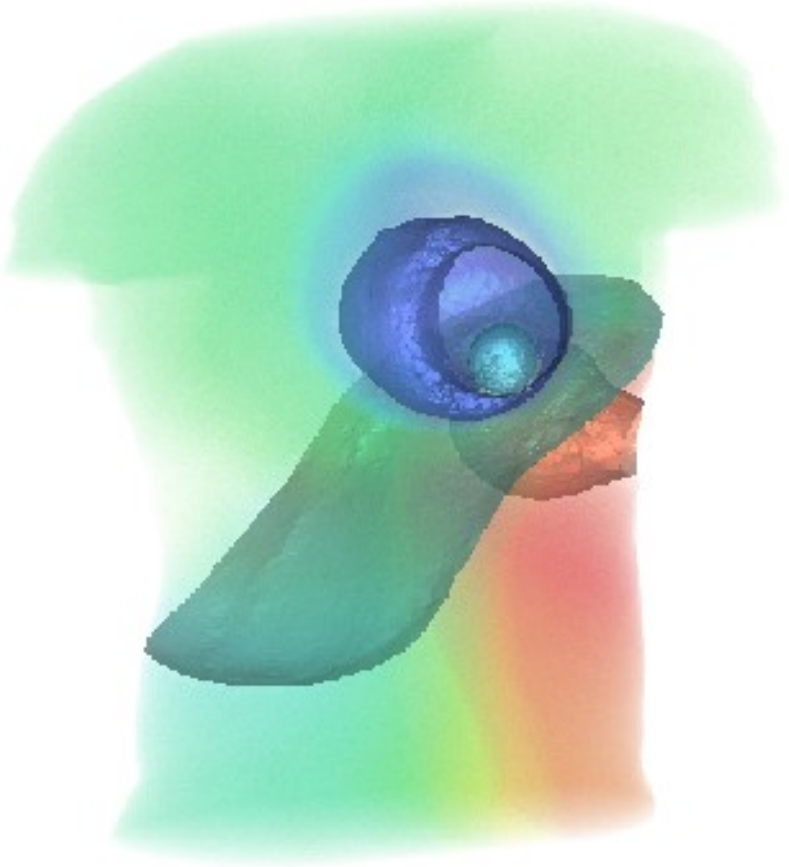
HAPT

results :: renderings



DVR+ISO
without lighting

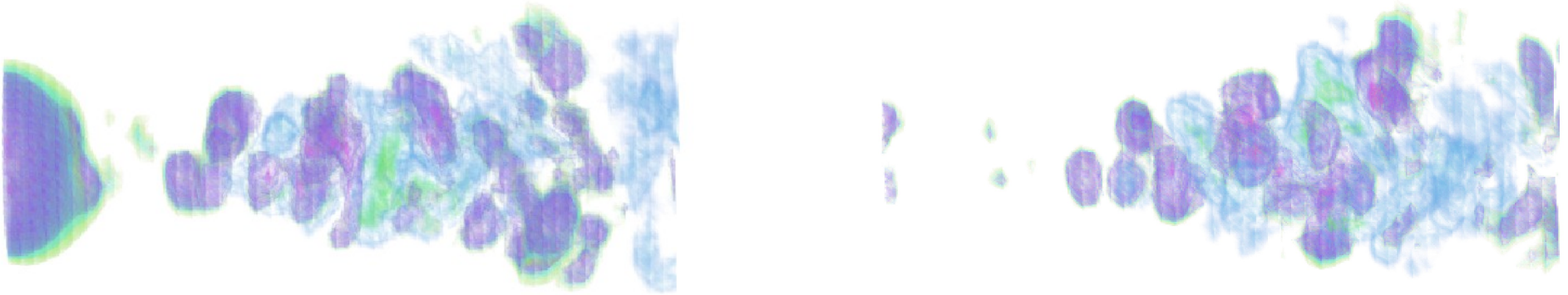
torso (1.08 M Tets)



DVR+ISO
with lighting

HAPT

results :: renderings



turbjet (1 M Tet per frame)
150 frames @17.5 fps

HAPT

Hardware-Assisted
Projected Tetrahedra *

code & publication

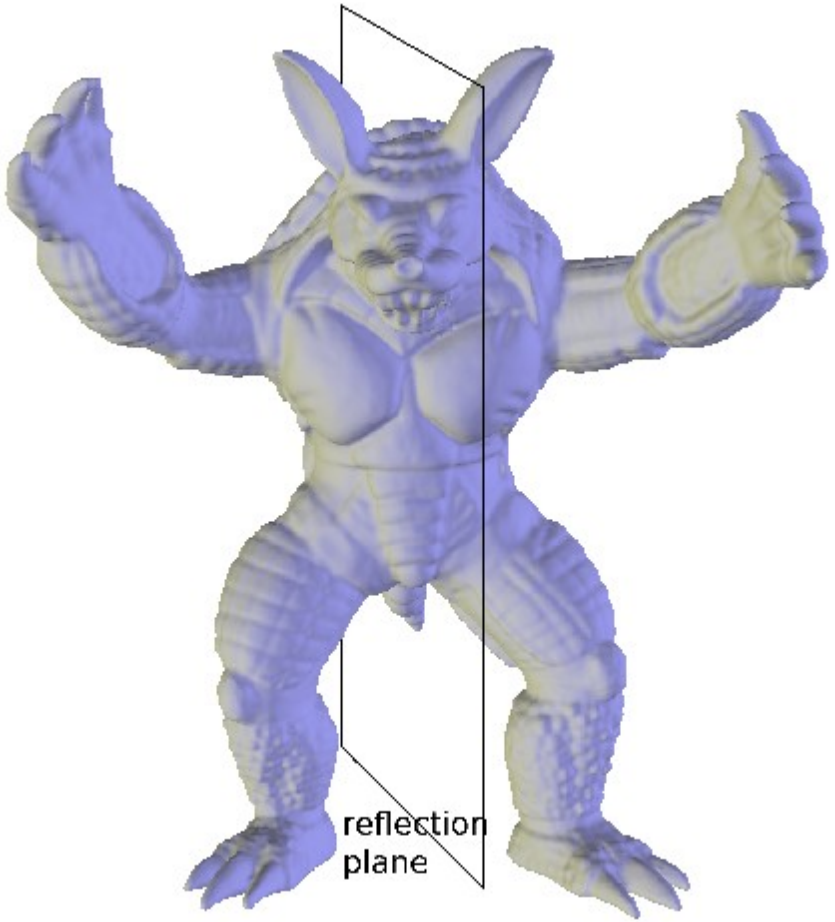
http://code.google.com/p/hapt

* MAXIMO, A., MARROQUIM, R., FARIAS, R. **“Hardware-Assisted Projected Tetrahedra”**, *in Proceedings of Eurographics/IEEE Symposium on Visualization (Computer Graphics Forum special issue)*, v. 29, pp. 903-912, Bordeaux, France, 2010.

mesh processing

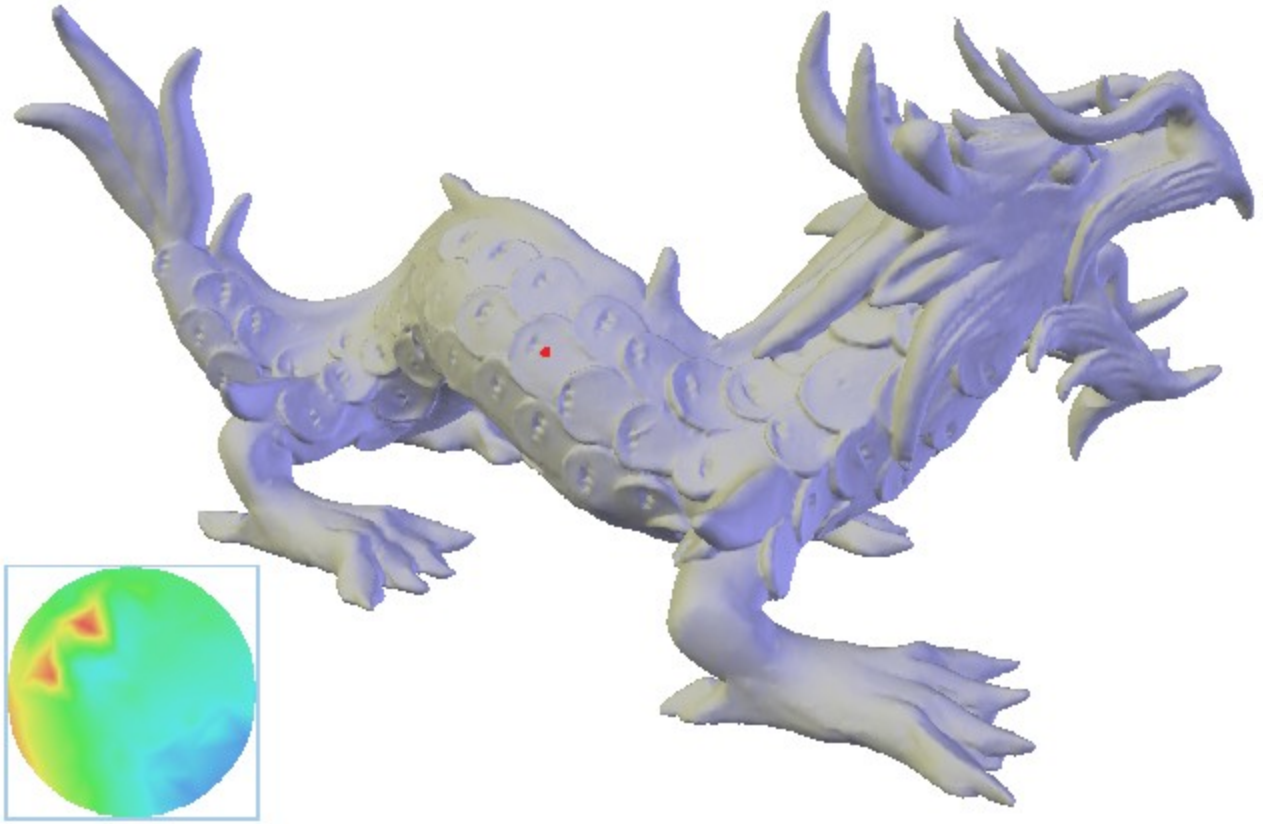
basis

symmetry

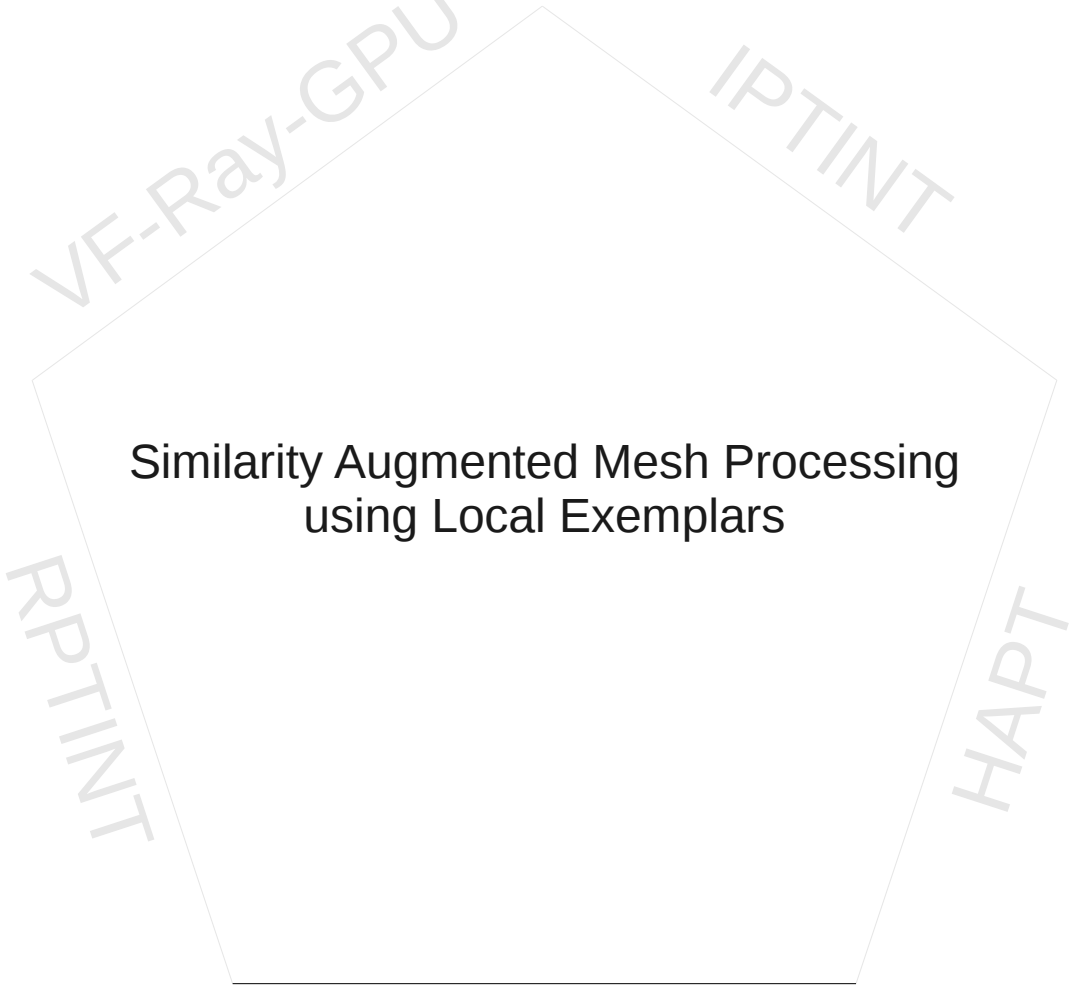


basis

similarity descriptor

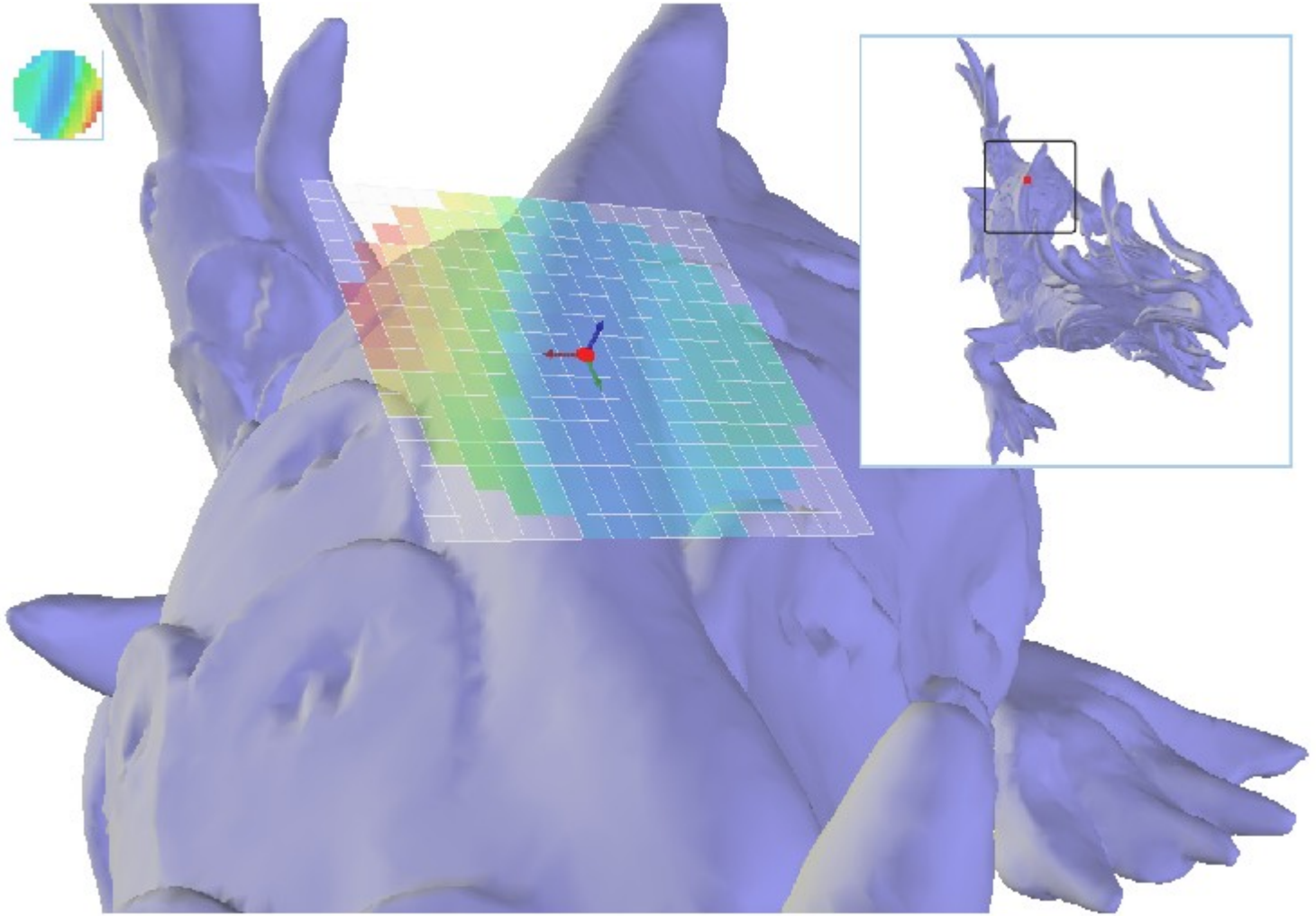


algorithm V



SAMPLE

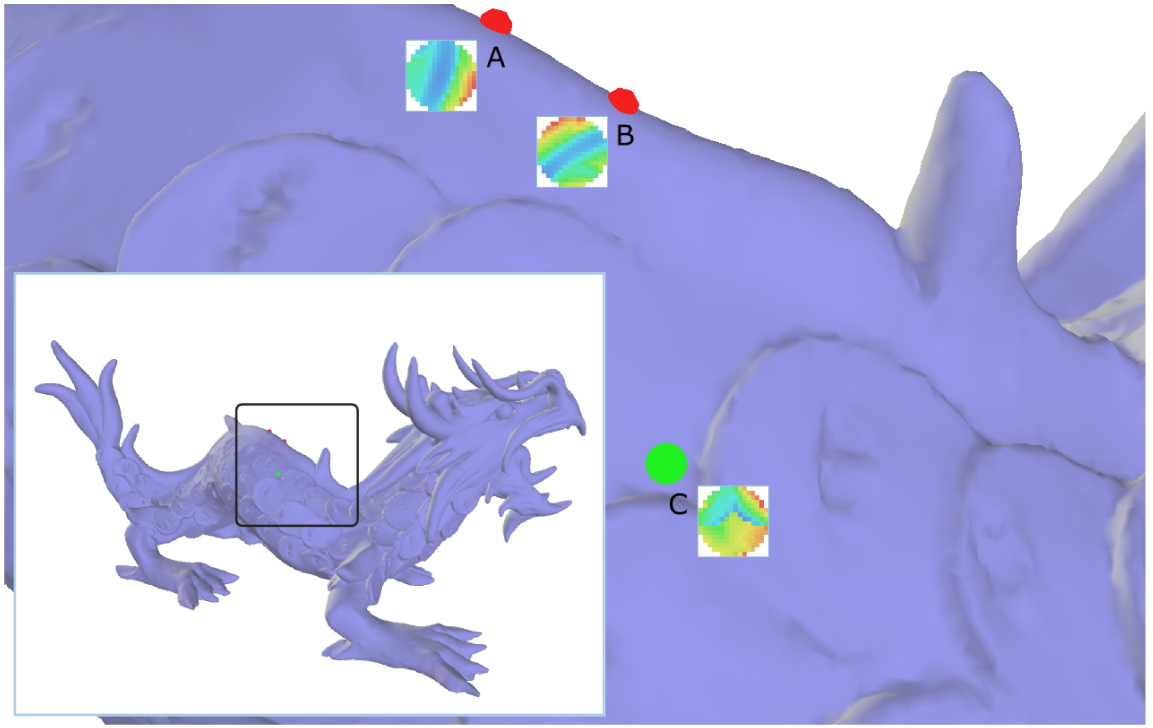
local heightmap descriptor



SAMPLE

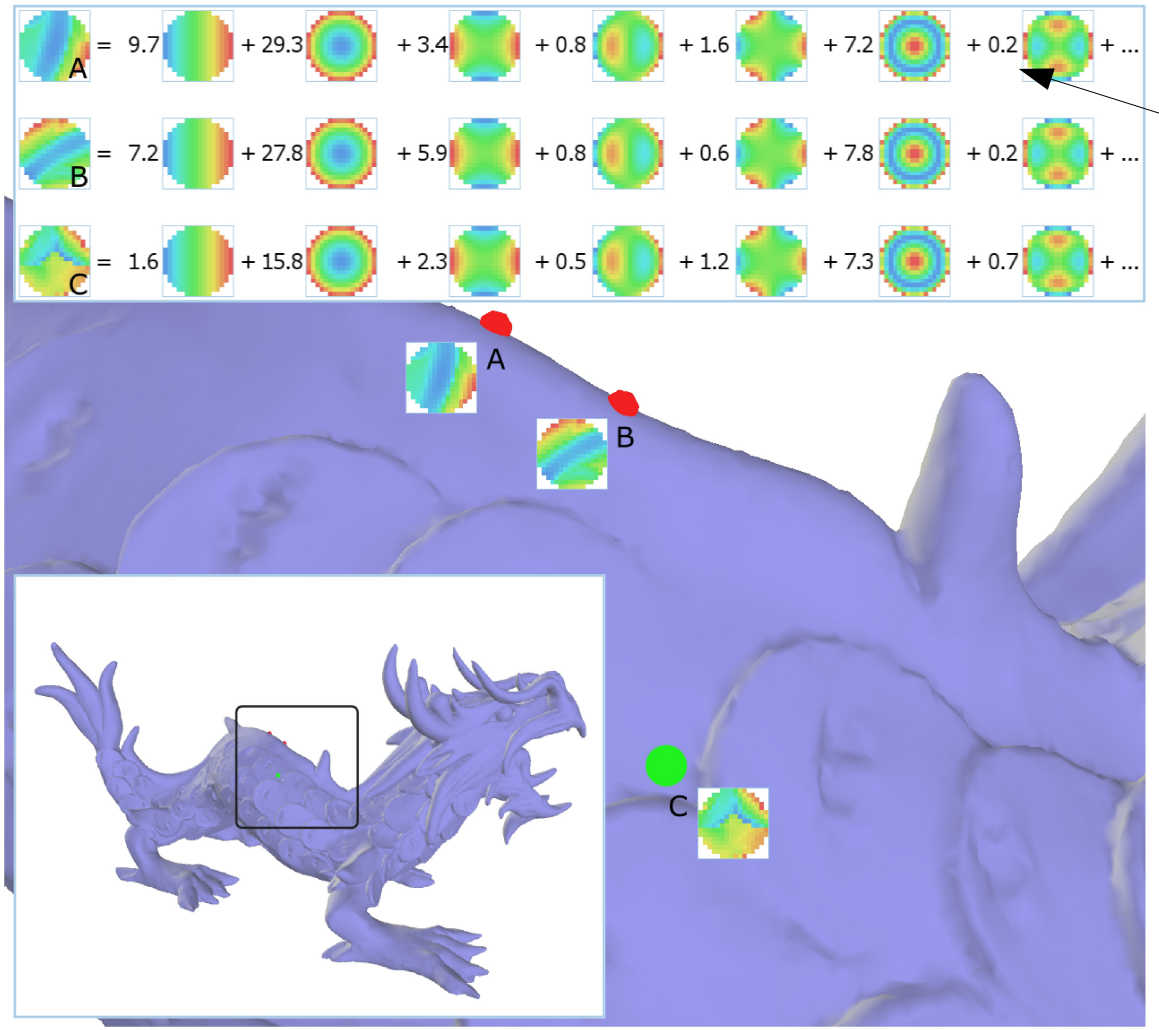
Heightmaps

A and B
not similar



SAMPLE

Zernike expansion



Zernike coefficients

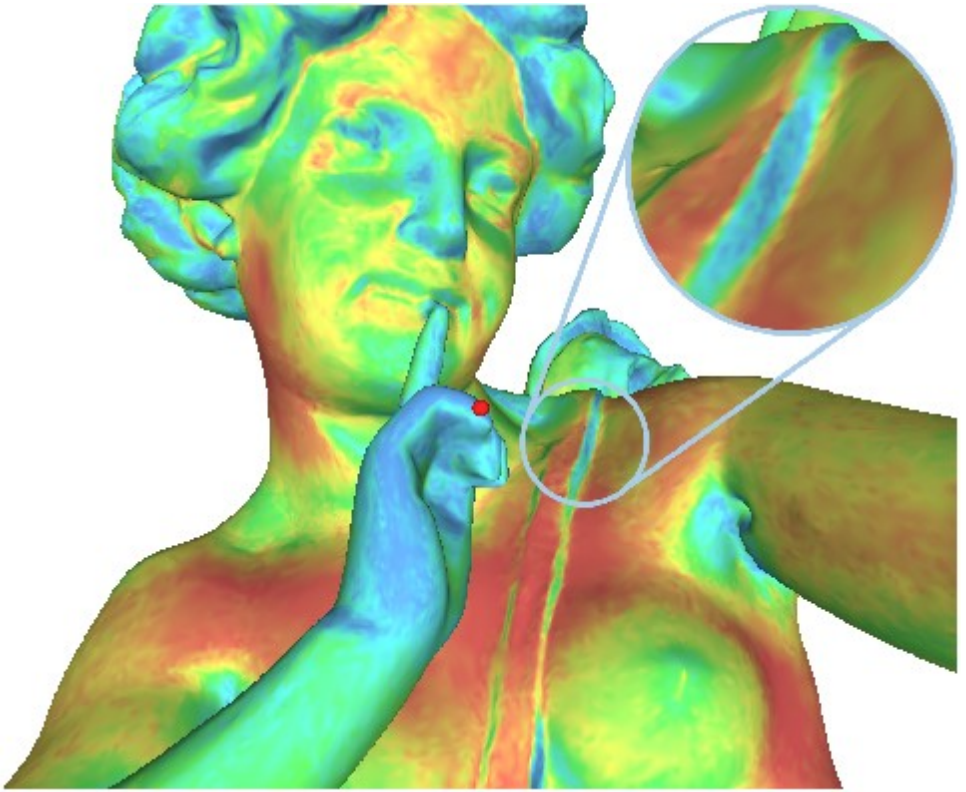
A and B similar

C not ssimilar

SAMPLE

vertex-based similarity

heightmaps / Zernike coefficients

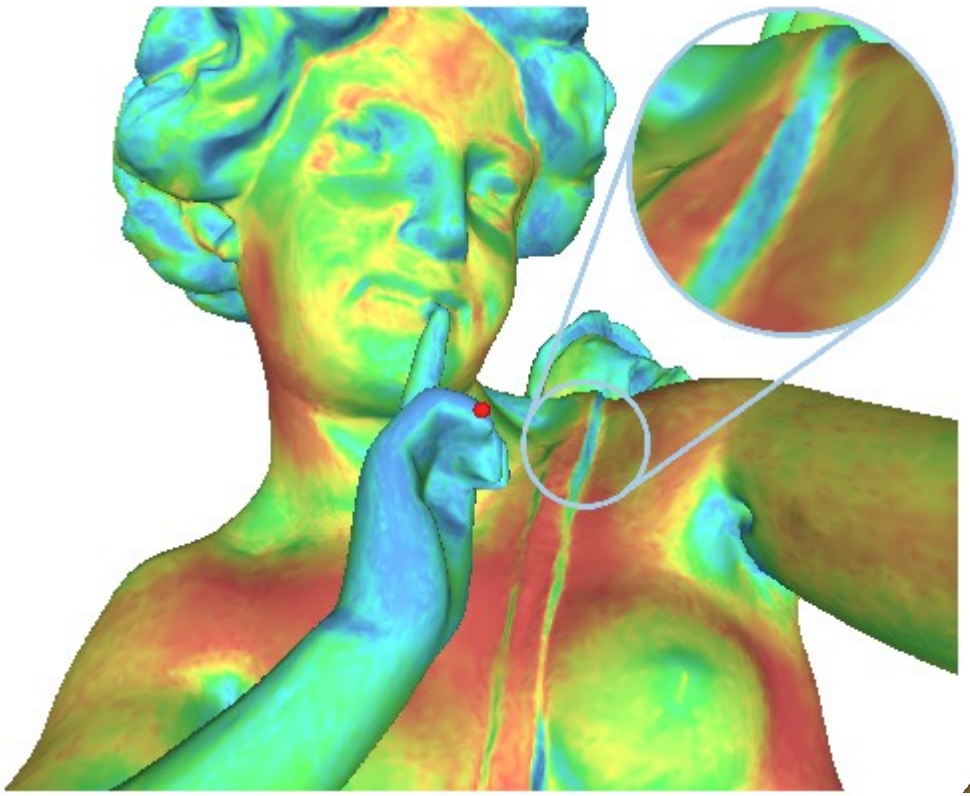


vertex-based descriptor

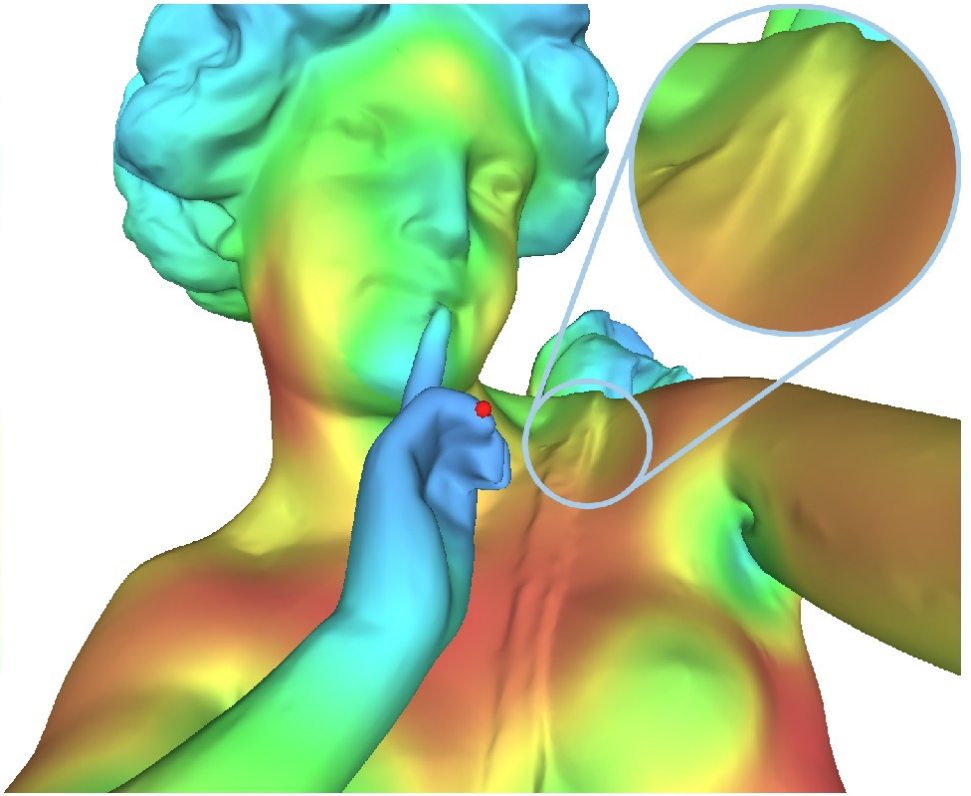
SAMPLE

region-based similarity

heightmaps / Zernike coefficients / Gaussian weights



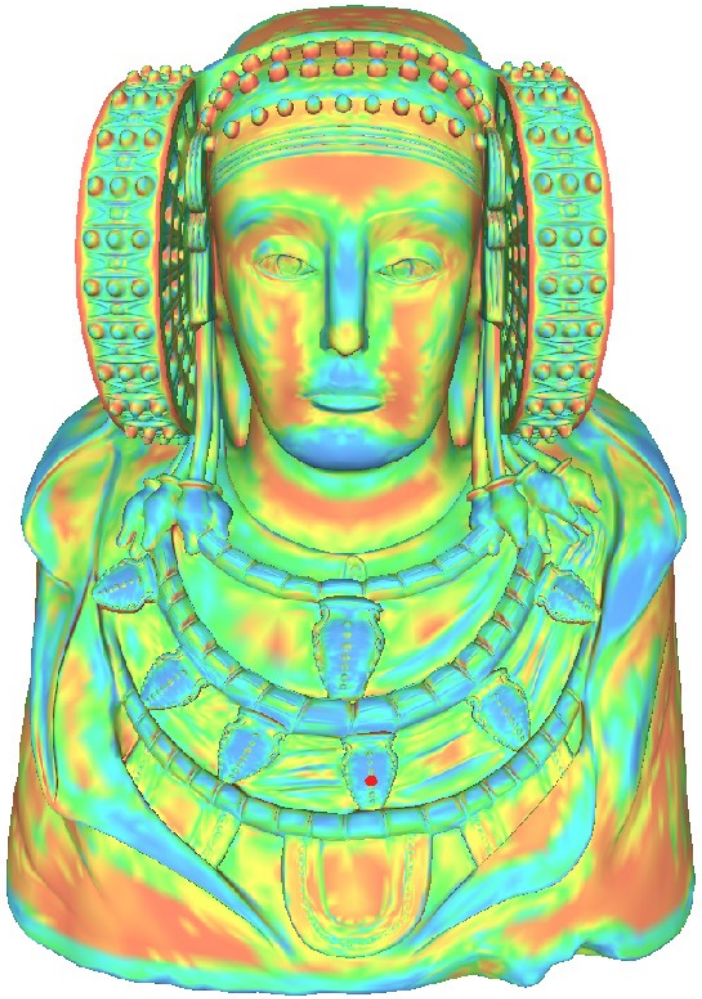
vertex-based descriptor



region-based descriptor

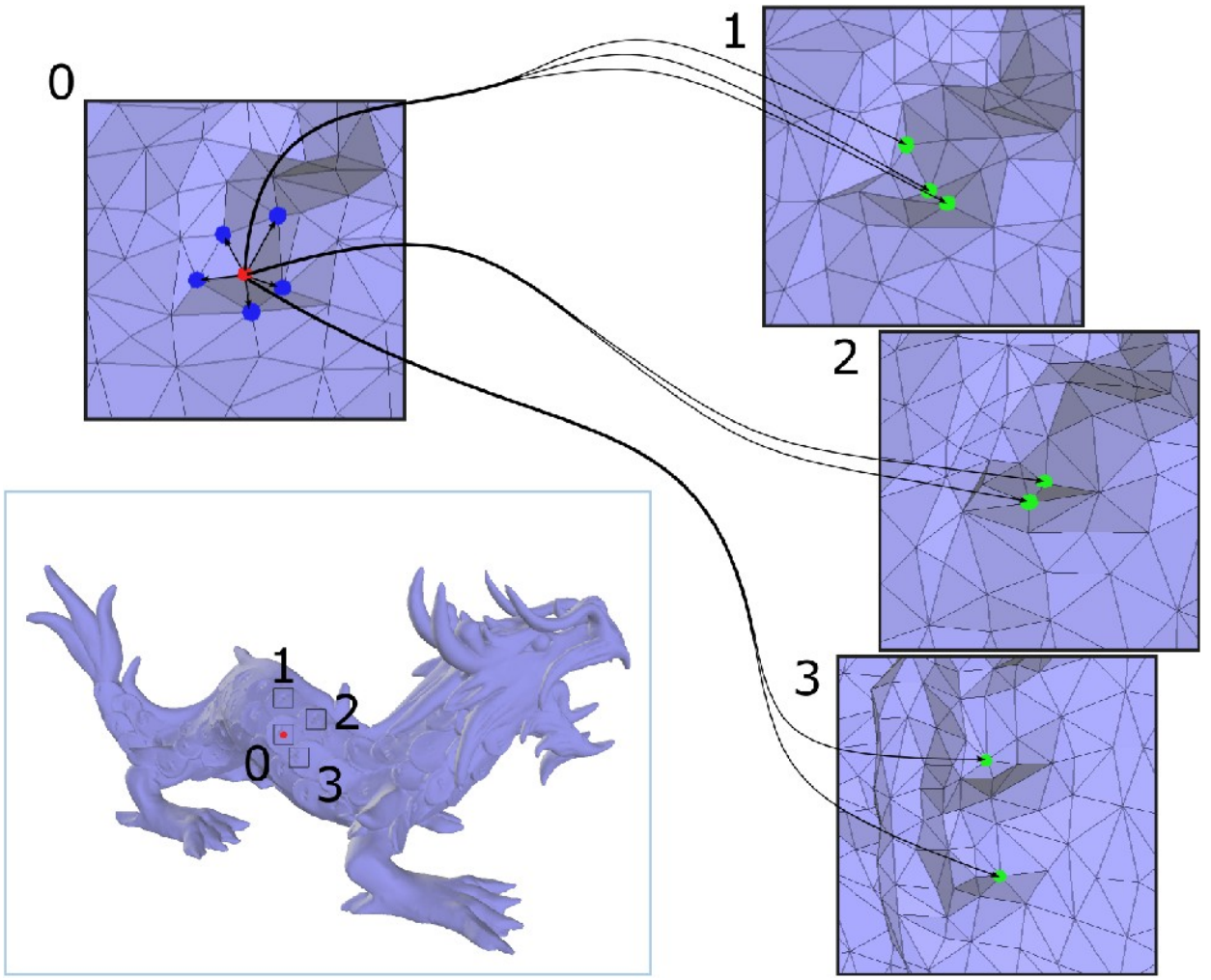
SAMPLE

region-based similarity



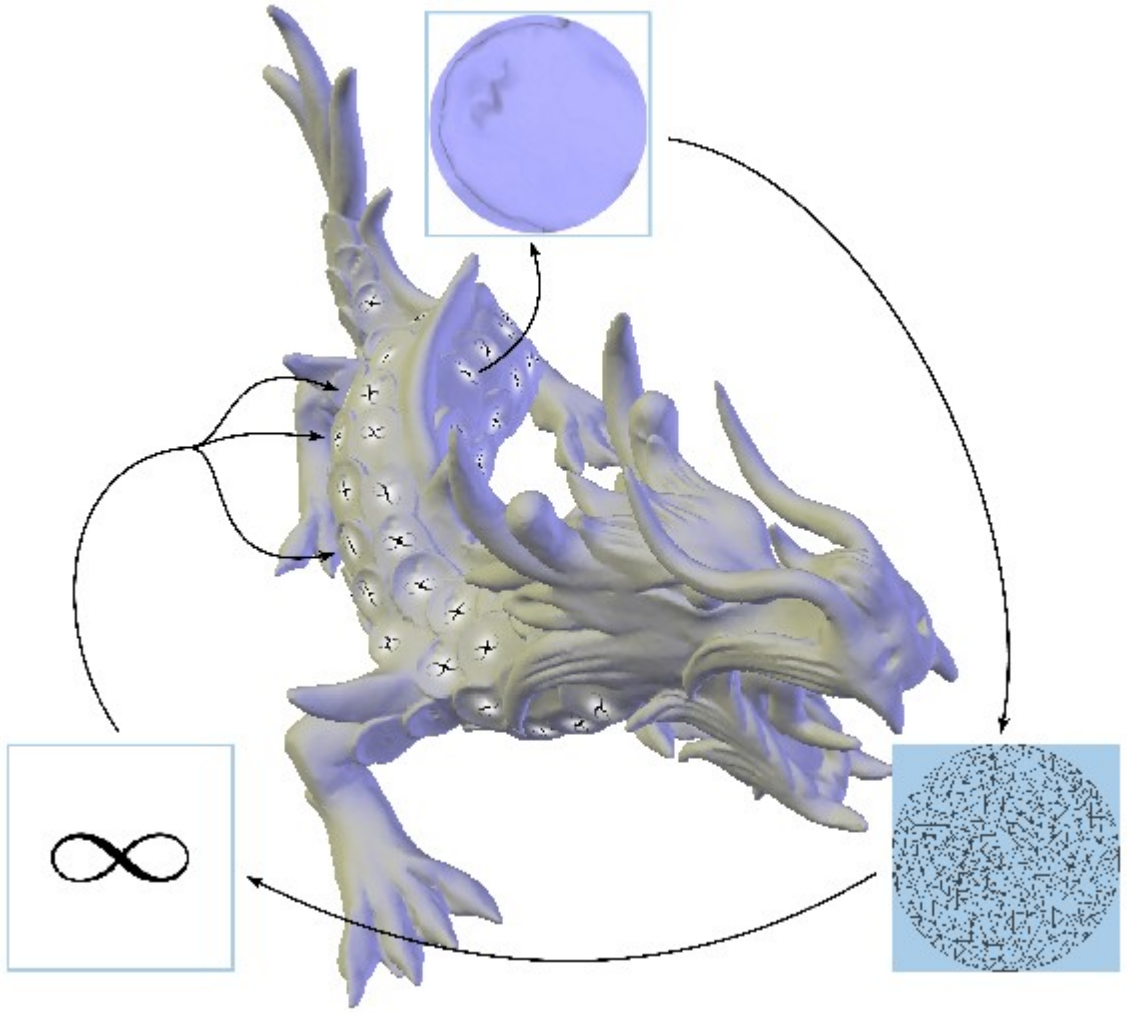
SAMPLE

propagating mesh processing



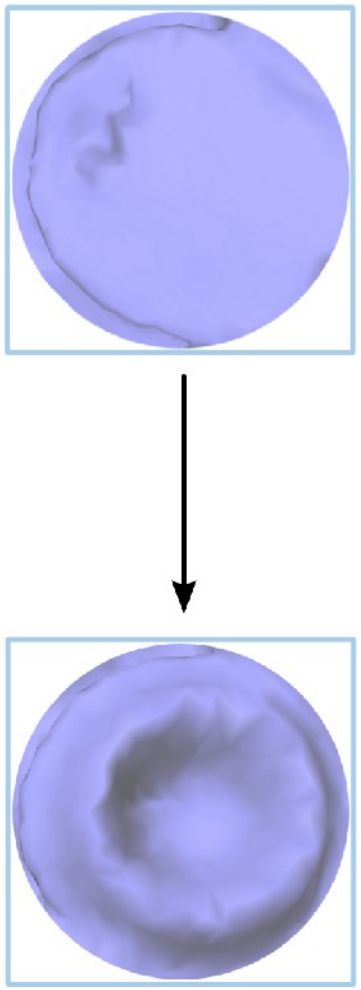
SAMPLE

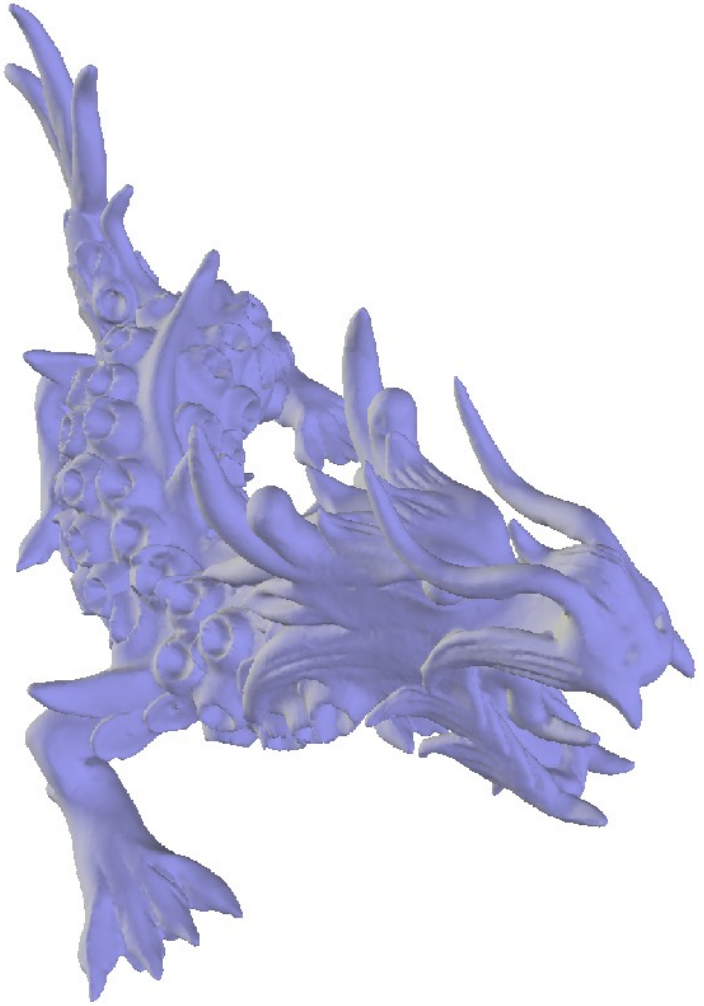
application :: mesh parameterization



SAMPLE

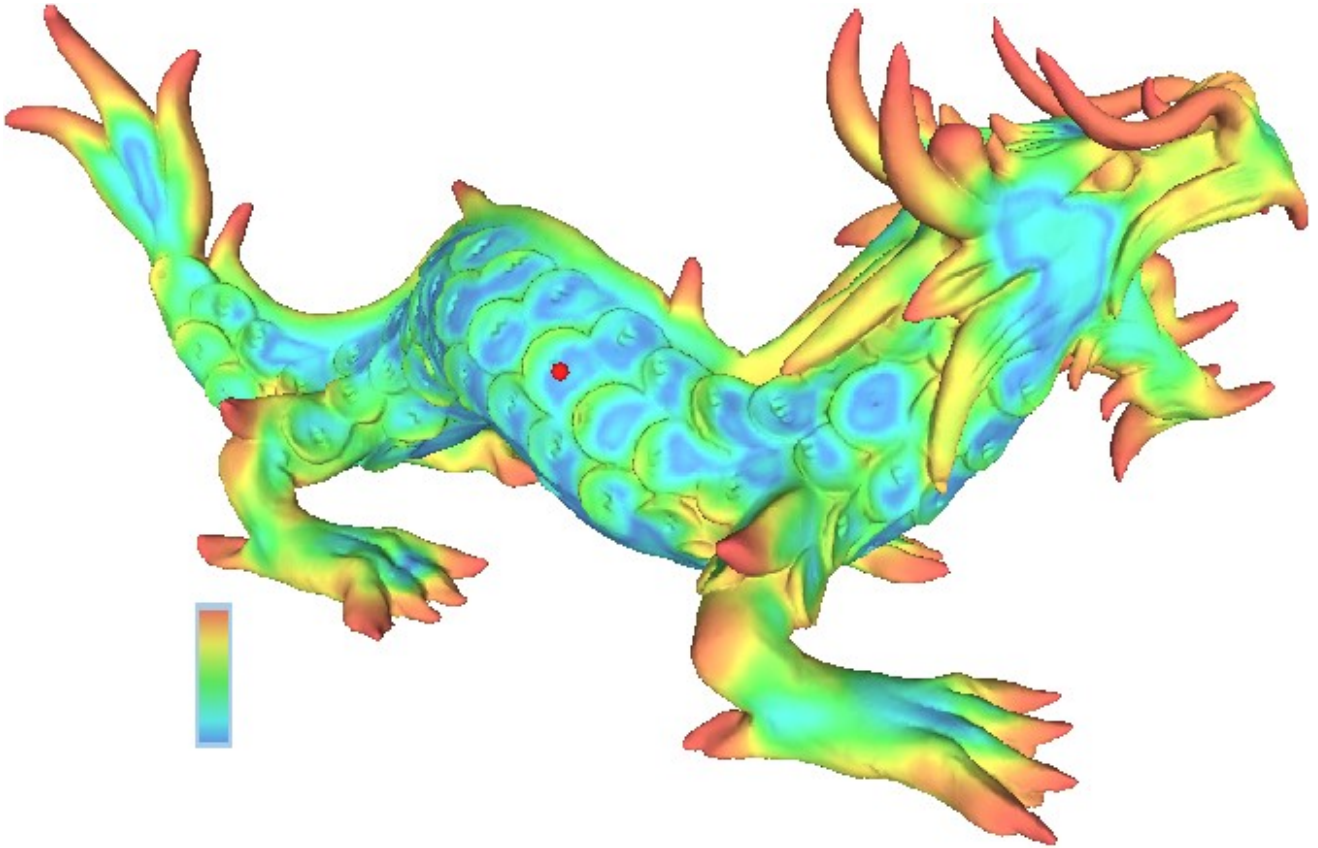
application :: detail transfer





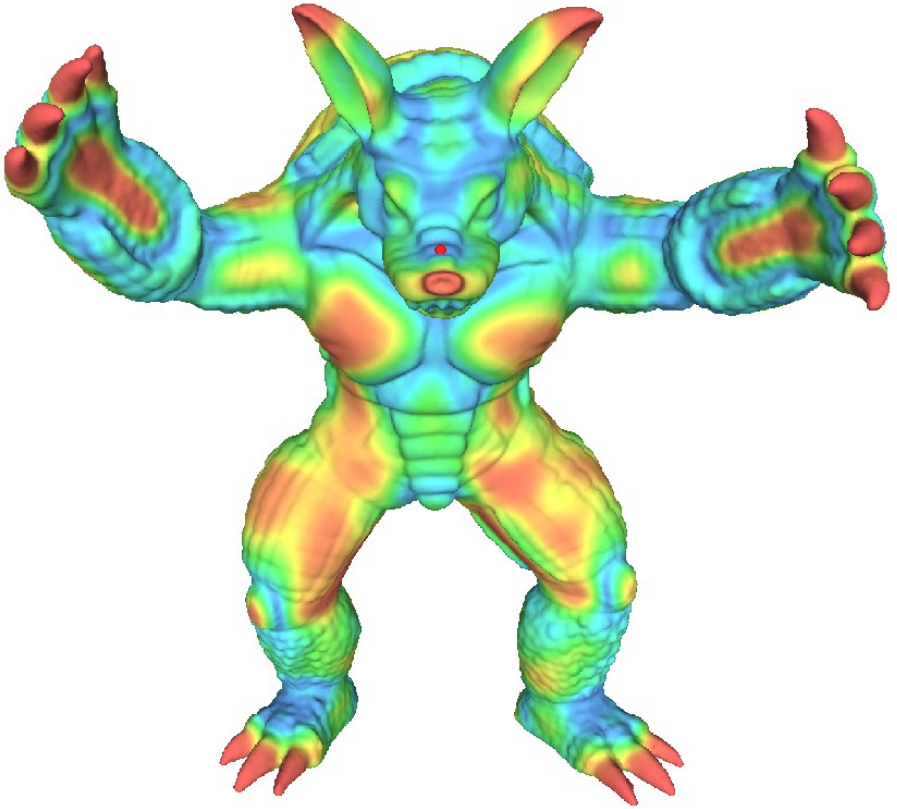
SAMPLE

results :: dual space



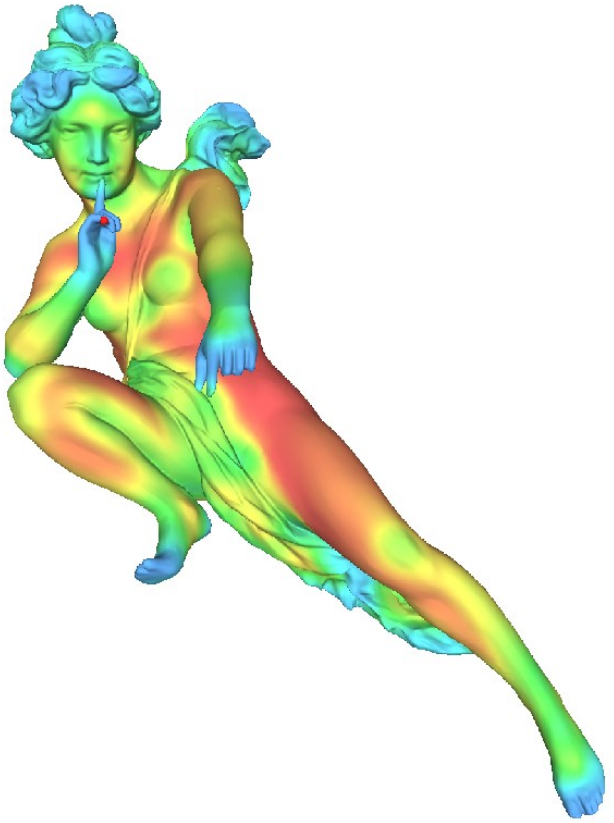
SAMPLE

results :: reflective symmetry



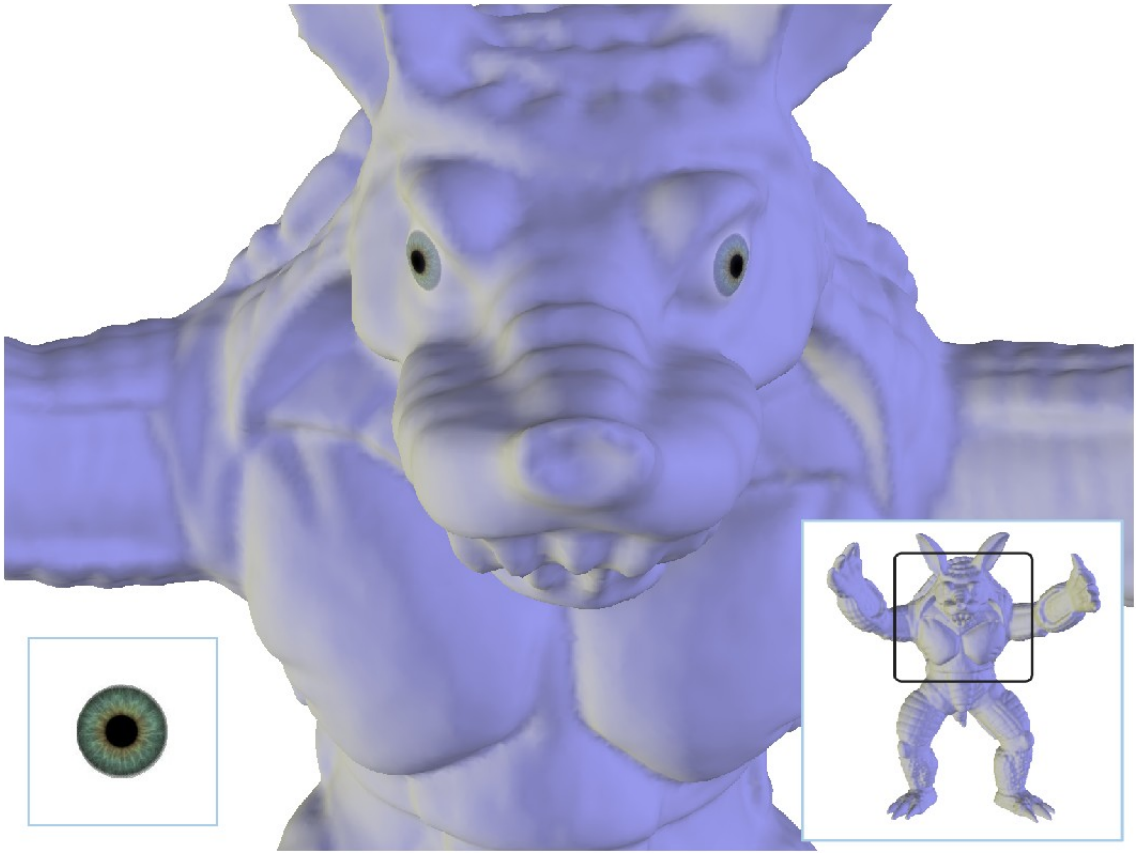
SAMPLE

results :: symmetries



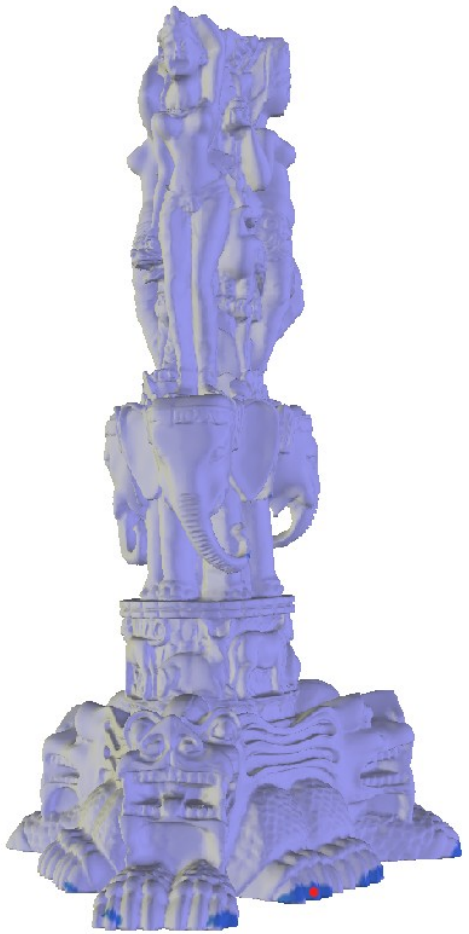
SAMPLE

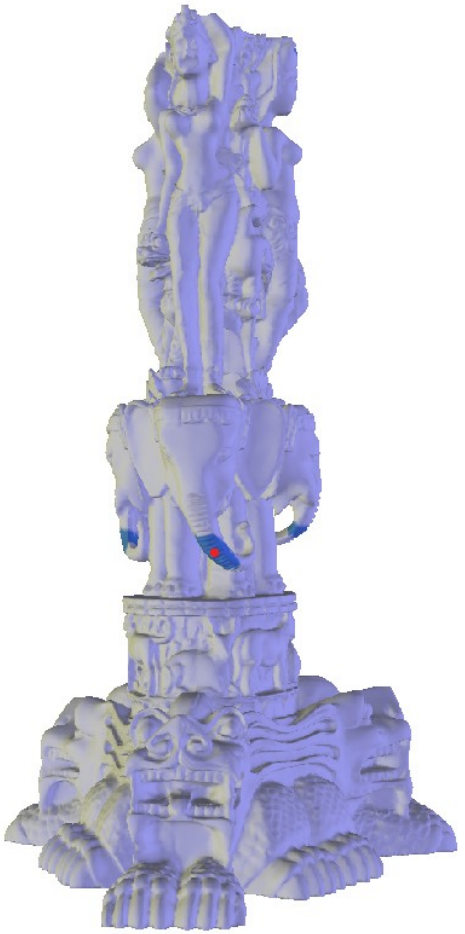
results :: immediate neighbor

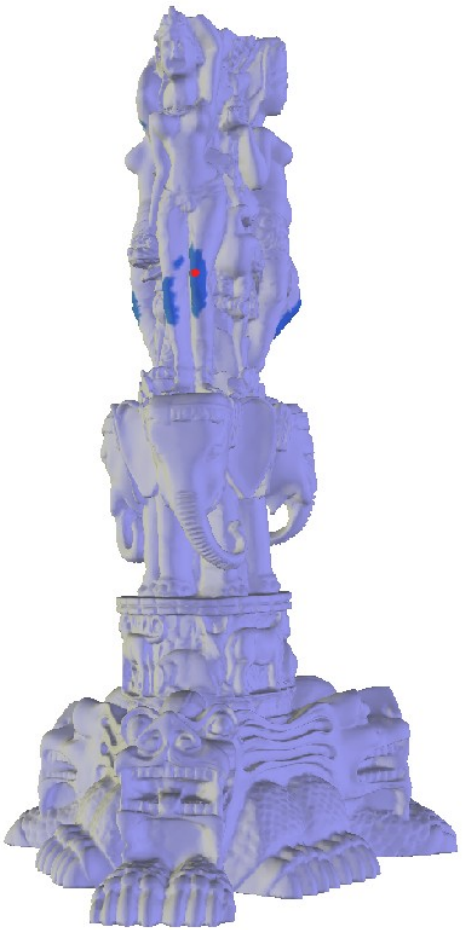


SAMPLE

results :: neighbors







SAMPLE

results :: timings

Model	# Verts	# Faces	Heightmap	Zernike	Gaussian
Dama	106 K	207 K	3.20 min	5.53 sec	2.47 min
Asian Dragon	108 K	216 K	2.56 min	5.54 sec	1.83 min
Thai Statue	125 K	250 K	2.98 min	6.58 sec	2.07 min
Armadillo	172 K	345 K	6.12 min	9.01 sec	5.34 min
Angel	237 K	474 K	17.76 min	12.15 sec	18.45 min

~ linear #vertices
pre-computation time

less than one second
running time

SAMPLE

Similarity Augmented Mesh Processing using Local Exemplars *

publication :: submitted

code not available yet

* MAXIMO, A., PATRO, R., VARSHNEY, A., FARIAS, R. **"A Robust and Rotationally Invariant Local Surface Descriptor with Applications to Non-local Mesh Processing"**, *submitted to Graphical Models*, May, 2010.

conclusion

contributions

algorithms

Volume Rendering

- Memory-Efficient Ray Casting
- Regular Data Specialization
- Efficient Cell Projection
- Flexible Framework
- Using the GPU

Mesh Processing

- Local Heightmap Descriptor
- Rotationally Invariant
- Region-based Comparison
- Dual-Space Concept
- Processing Propagation

reproducible research

http://code.google.com/u/andmax

publications

MAXIMO, A., RIBEIRO, S., BENTES, C., OLIVEIRA, A., FARIAS, R. **“Memory-Efficient GPU-Based Ray Casting for Unstructured Volume Rendering”**, *in Proceedings of VG-PBG (Eurographics Association)*, pp. 155-162, Los Angeles, California, USA, 2008.

VF-Ray-GPU

RIBEIRO, S., MAXIMO, A., BENTES, C., OLIVEIRA, A., FARIAS, R. **“Memory-Aware and Efficient Ray-Casting Algorithm”**, in Proceedings of *SIBGRAPI (IEEE Computer Society)*, pp. 147-154, Belo Horizonte, Minas Gerais, Brazil, 2007.

VF-Ray

MAXIMO, A., MARROQUIM, R., FARIAS, R., ESPERANÇA, C. **“GPU-Based Cell Projection for Large Structured Datasets”**, *in Proceedings of GRAPP (INSTICC)*, pp. 312-322, Barcelona, Spain, 2007.

RPTINT

MARROQUIM, R., MAXIMO, A., FARIAS, R., ESPERANÇA, C. **“GPU-Based Cell Projection for Interactive Volume Rendering”**, in Proceedings of *SIBGRAPI (IEEE Computer Society)*, pp. 147-154, Manaus, Amazonas, 2006. *Best Paper.*

PTINT

publications

MARROQUIM, R., MAXIMO, A., FARIAS, R., ESPERANÇA, C. **“Volume and Isosurface Rendering with GPU-Accelerated Cell Projection”**, *Computer Graphics Forum (Best Paper selection of SIBGRAPI 2006)*, v. 27, pp. 24-35, 2008.

IPTINT

MAXIMO, A., MARROQUIM, R., FARIAS, R. **“Hardware-Assisted Projected Tetrahedra”**, *in Proceedings of Eurographics/IEEE Symposium on Visualization (Computer Graphics Forum special issue)*, v. 29, pp. 903-912, Bordeaux, France, 2010.

HAPT

MAXIMO, A., PATRO, R., VARSHNEY, A., FARIAS, R. **“A Robust and Rotationally Invariant Local Surface Descriptor with Applications to Non-local Mesh Processing”**, *submitted to Graphical Models*, May, 2010.

SAMPLE

Thank You!

André Maximo

Advisor: Ricardo Farias

(CNPq Sandwich) Advisor: Amitabh Varshney

andmax@cos.ufrj.br

http://www.lcg.ufrj.br/Members/andream

July 23, 2010
Rio de Janeiro, Brazil