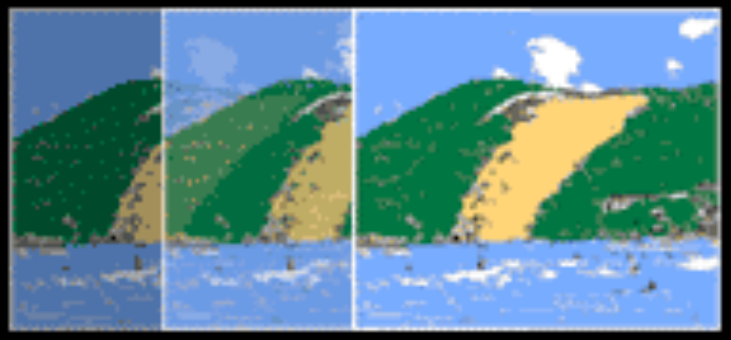
GPU-Based Volume Rendering of Unstructured Grids

Module 5:

Hardware Ray Casting (HRC) and Time-Varying Scalar Fields (TVSF)

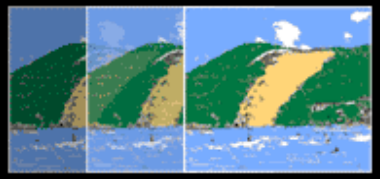
Fábio F. Bernardon
UFRGS



SIBGRAPI 2005

Natal - RN - Brazil

XVIII Brazilian Symposium on Computer Graphics and Image Processing



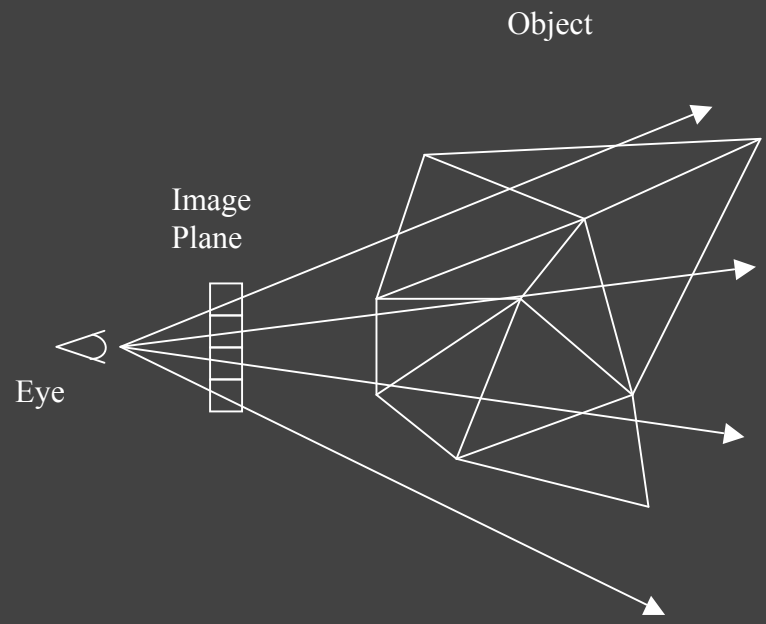
SIBGRAPI 2005

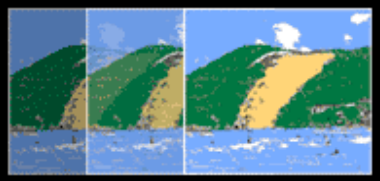
5: HRC

Hardware Ray Casting

5: HRC

- Garrity (Workshop on Vol. Vis. 90)
  - Ray tracing irregular volume data
    - Calculate entry/exit point on each iteration
  - Mapping functions based on scalar fields
    - Use the Gradient vector



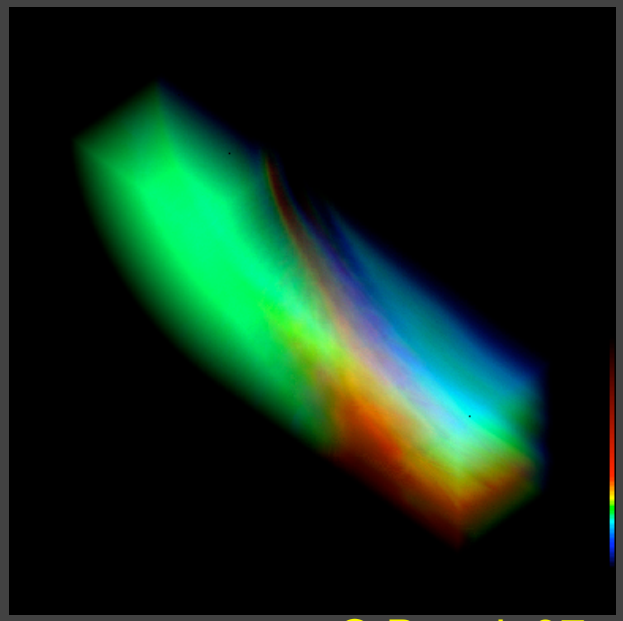


SIBGRAPI 2005

5: HRC

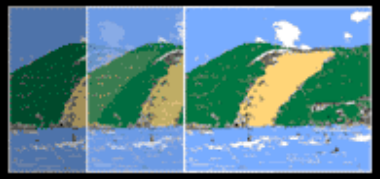
- Bunyk et al. (Dagstuhl 97)
  - Simple and fast ray casting
  - Boundary faces

© Weiler 03



© Bunyk 97

- Weiler et al. (Visualization 2003)
  - GPU Ray Casting for tetrahedral meshes

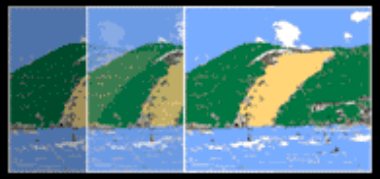


SIBGRAPI 2005

5: HRC

- Bernardon et al (to appear)
  - Performance Improvements
  - Depth peeling
  - Tiling

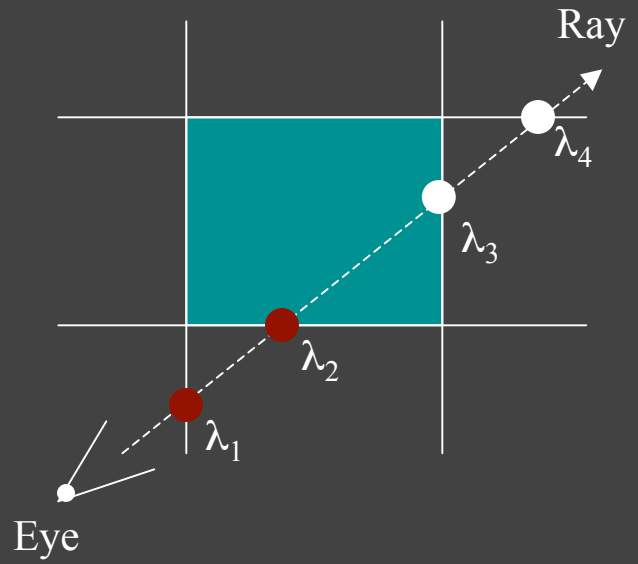
© Bernardon



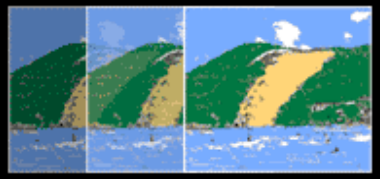
SIBGRAPI 2005

5: HRC - Intersection

- Cell intersection
- Entering: $r \cdot n_{t,i} < 0$
- Exit: $r \cdot n_{t,i} >= 0$

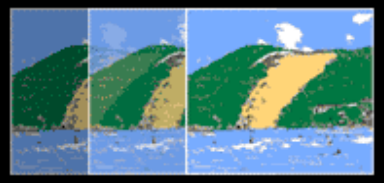


$$\ddot{e}_i = \frac{(v_{t,4-i} - e) \cdot n_{t,i}}{r \cdot n_{t,i}}, i \in \{1..4\}$$



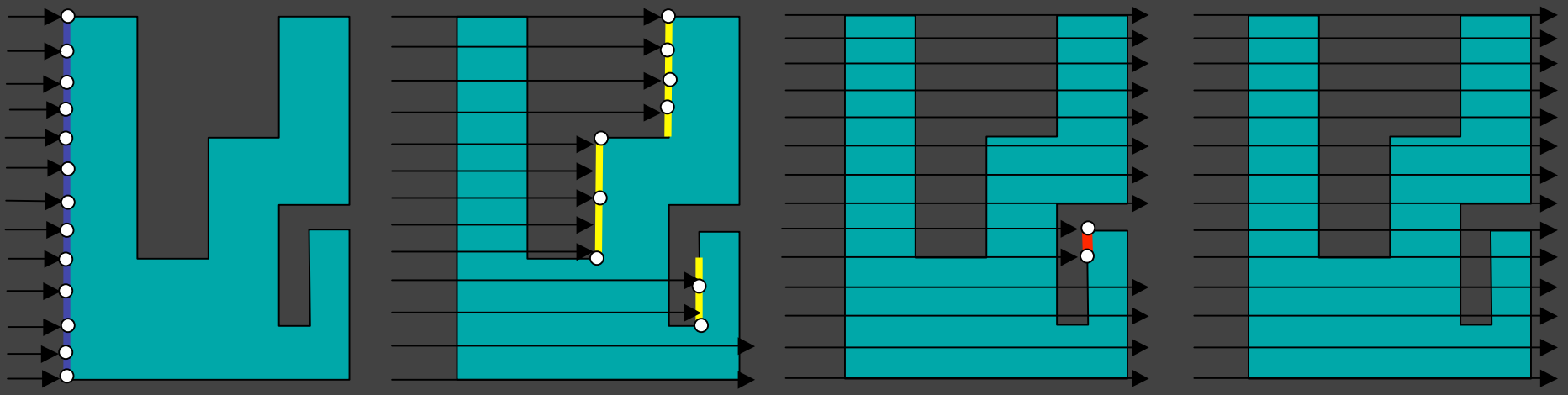
5: HRC

- What happens if the mesh is not convex?
  - Ray must reenter
    - More ray casting
    - Convexification
    - Depth peeling

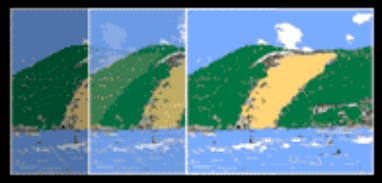


SIBGRAPI 2005

5: HRC – Depth Peeling

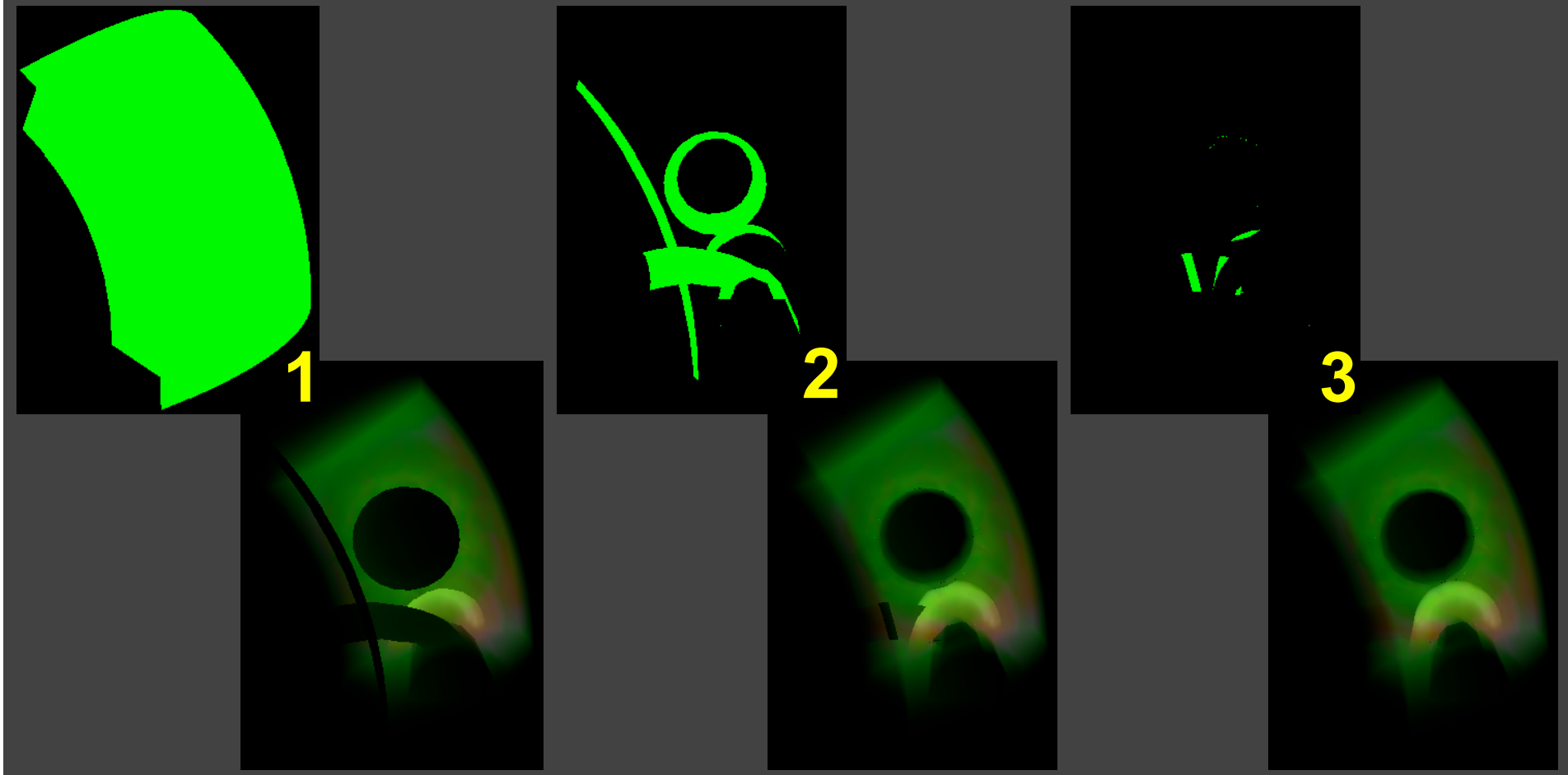


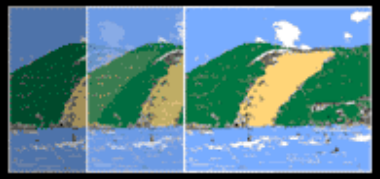
- All mesh reentries



SIBGRAPI 2005

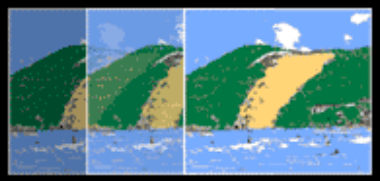
5: HRC – Depth Peeling





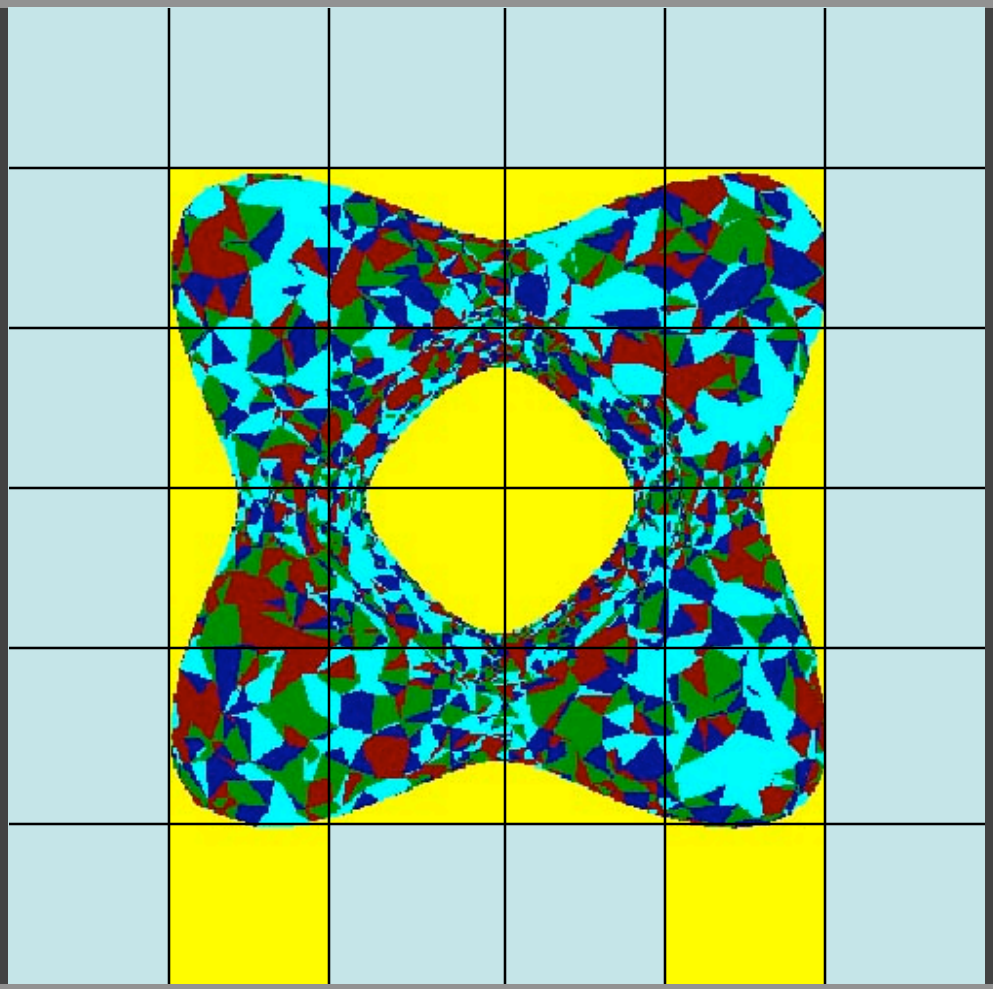
5: HRC

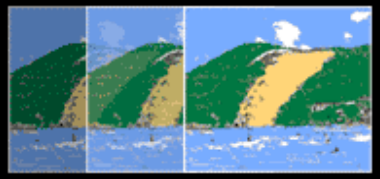
- What else can we do to gain performance?
  - Tilling
  - Mesh representation



SIBGRAPI 2005

5: HRC - Tiling

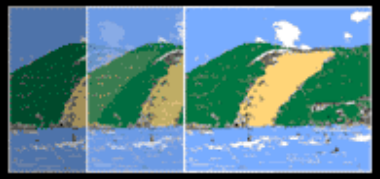




SIBGRAPI 2005

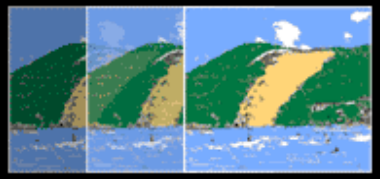
5: HRC - Tiling

39	39	40	41	41	41
74	196	166	41	41	41
104	510	510	510	197	41
214	592	429	469	197	41
39	75	285	285	105	41
39	40	41	TIME(891) 41	#pOcc: (10) 41	41



5: HRC

- How do we store the information?
  - Weiler used mesh encoded in 3D textures
  - 2D texture representation faster

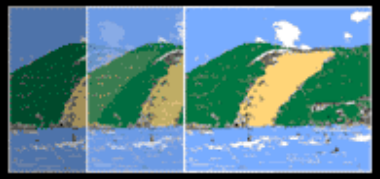


SIBGRAPI 2005

5: HRC – Data Structures

Mesh Data	Texture Format	Coordinate		Data			
		u	v	r	g	b	a
Vertices	F32x4	t_u	t_v	$V0.x_t$	$V0.y_t$	$V0.z_t$	$V3.x_t$
Vertices	F32x4	t_u	t_{v+dv}	$V1.x_t$	$V1.y_t$	$V1.z_t$	$V3.y_t$
Vertices	F32x4	t_u	t_{v+2dv}	$V2.x_t$	$V2.y_t$	$V2.z_t$	$V3.z_t$
Neighbor Index	F32x4	t_u	t_v	$t_u(a_0)$	$t_v(a_0)$	$t_u(a_1)$	$t_v(a_1)$
Neighbor Index	F32x4	t_u	t_{v+dv}	$t_u(a_2)$	$t_v(a_2)$	$t_u(a_3)$	$t_v(a_3)$
Gradient + Scalar	F32x4	t_u	t_{v+2dv}	$g.x_t$	$g.y_t$	$g.z_t$	s_t

Mesh Data	Texture Format	Coordinate		Data			
		u	v	r	g	b	a
Current Tetra	F32x4	Screen Pos		t_u	t_v	λ	$S(e+\lambda r)$
Color and Opacity	F32x4	Screen Pos		R	G	B	A

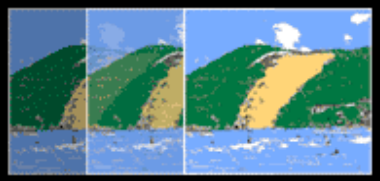


SIBGRAPI 2005

5: HRC – Data Structures

- Extra texture for each depth level generated

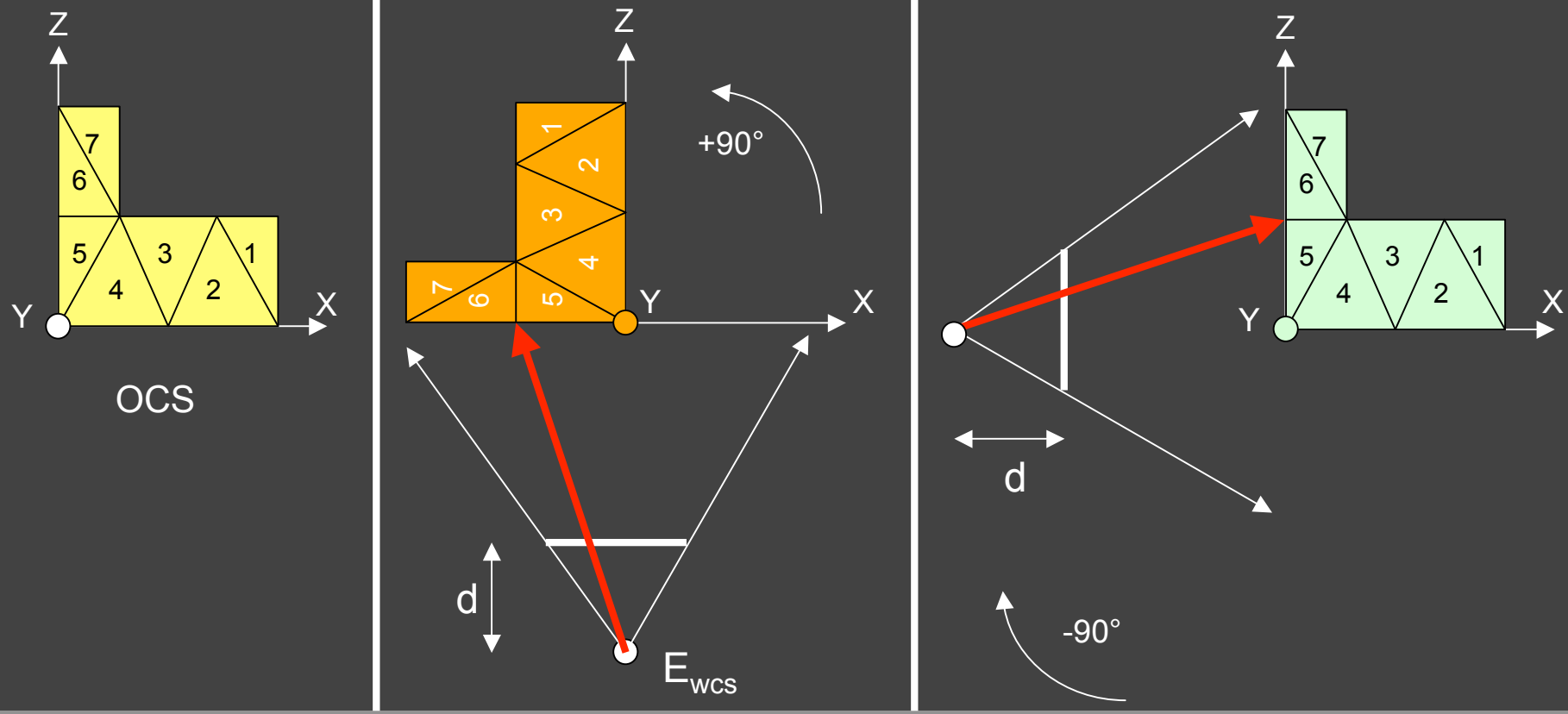
Mesh Data	Texture Format	Coordinate		Data			
		u	v	r	g	b	a
Entering Tetra	F32x4	Screen Pos		t_u	t_v	λ	$S(e+\lambda r)$

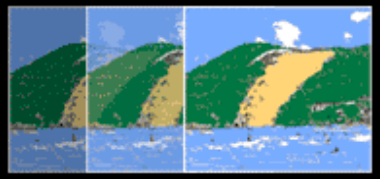


SIBGRAPI 2005

5: HRC

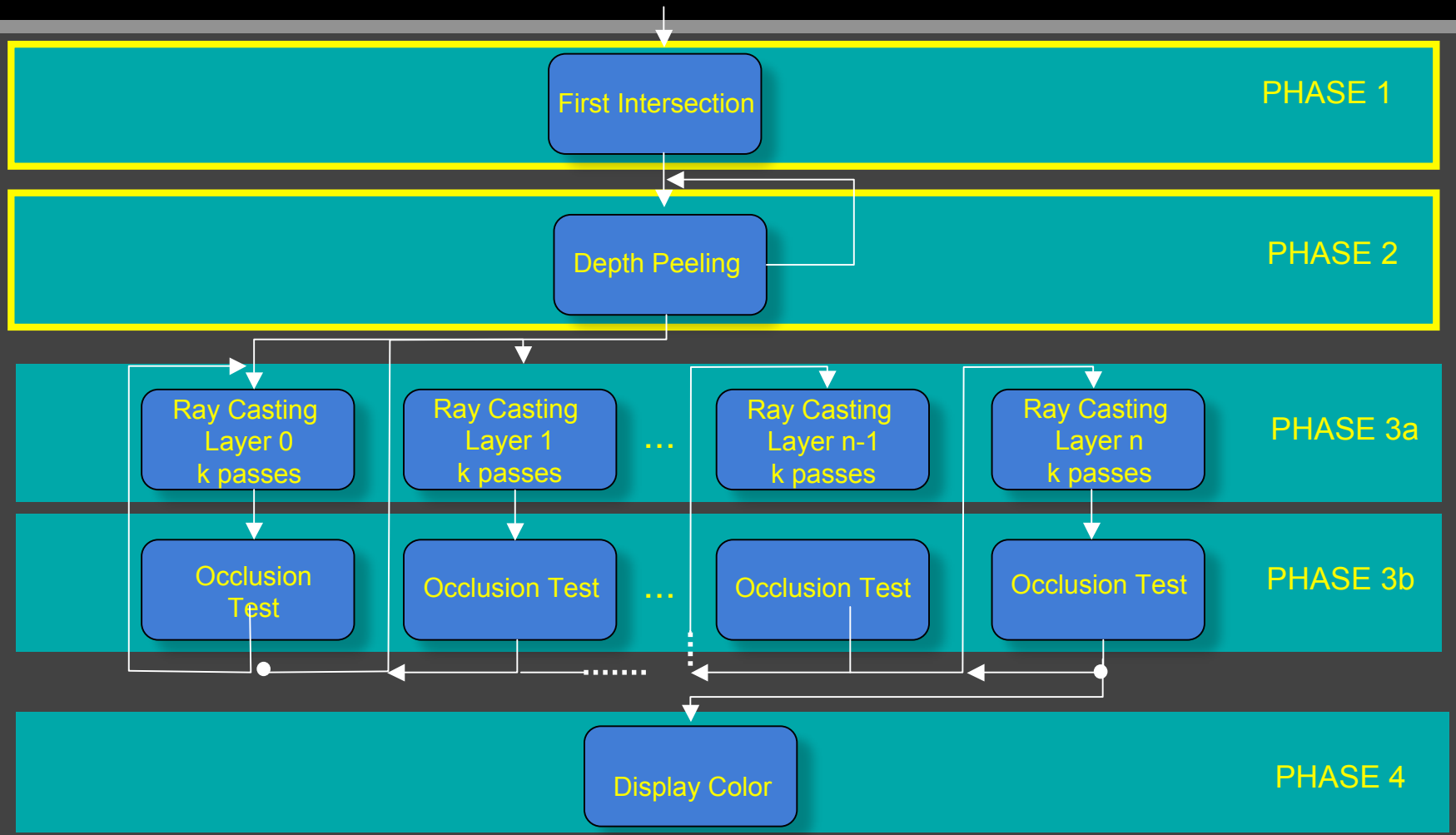
- Coordinate systems

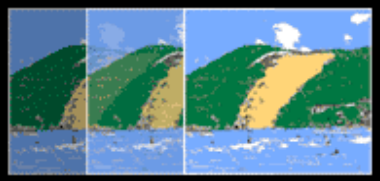




SIBGRAPI 2005

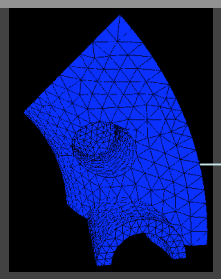
5: HRC - Implementation

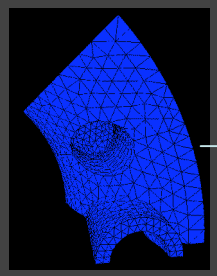


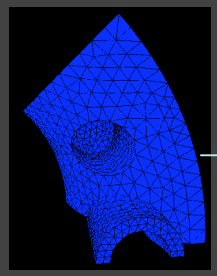


SIBGRAPI 2005

5: HRC - Implementation







Boundary Faces

First Intersection

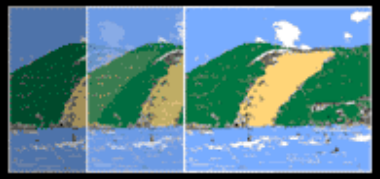
Depth Peeling Pass 1

Depth Peeling Pass n

Current Cell 0

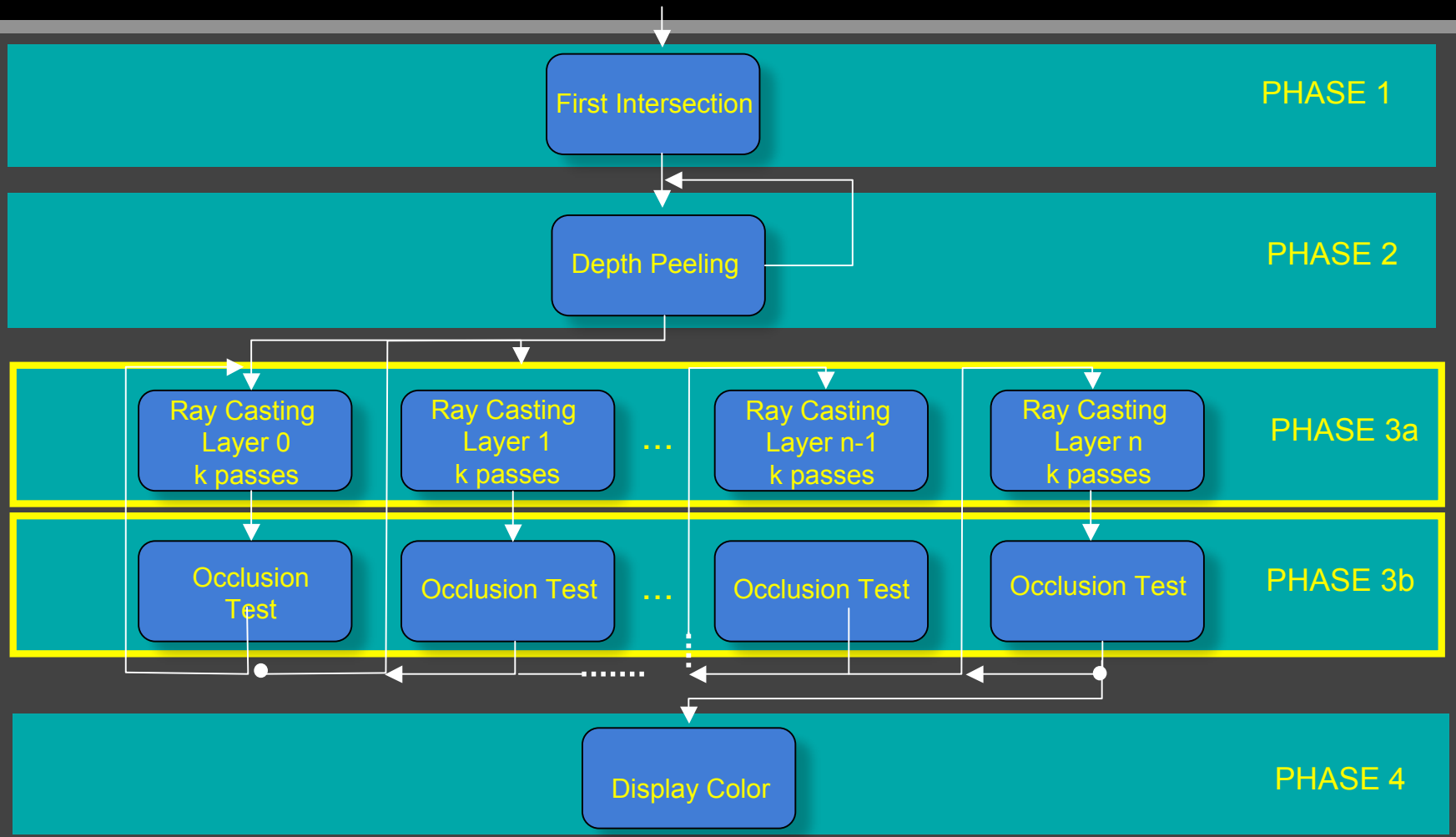
Current Cell 1

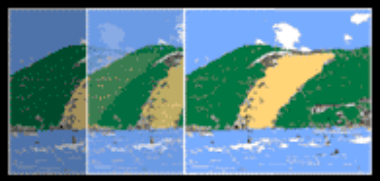
Current Cell n-1



SIBGRAPI 2005

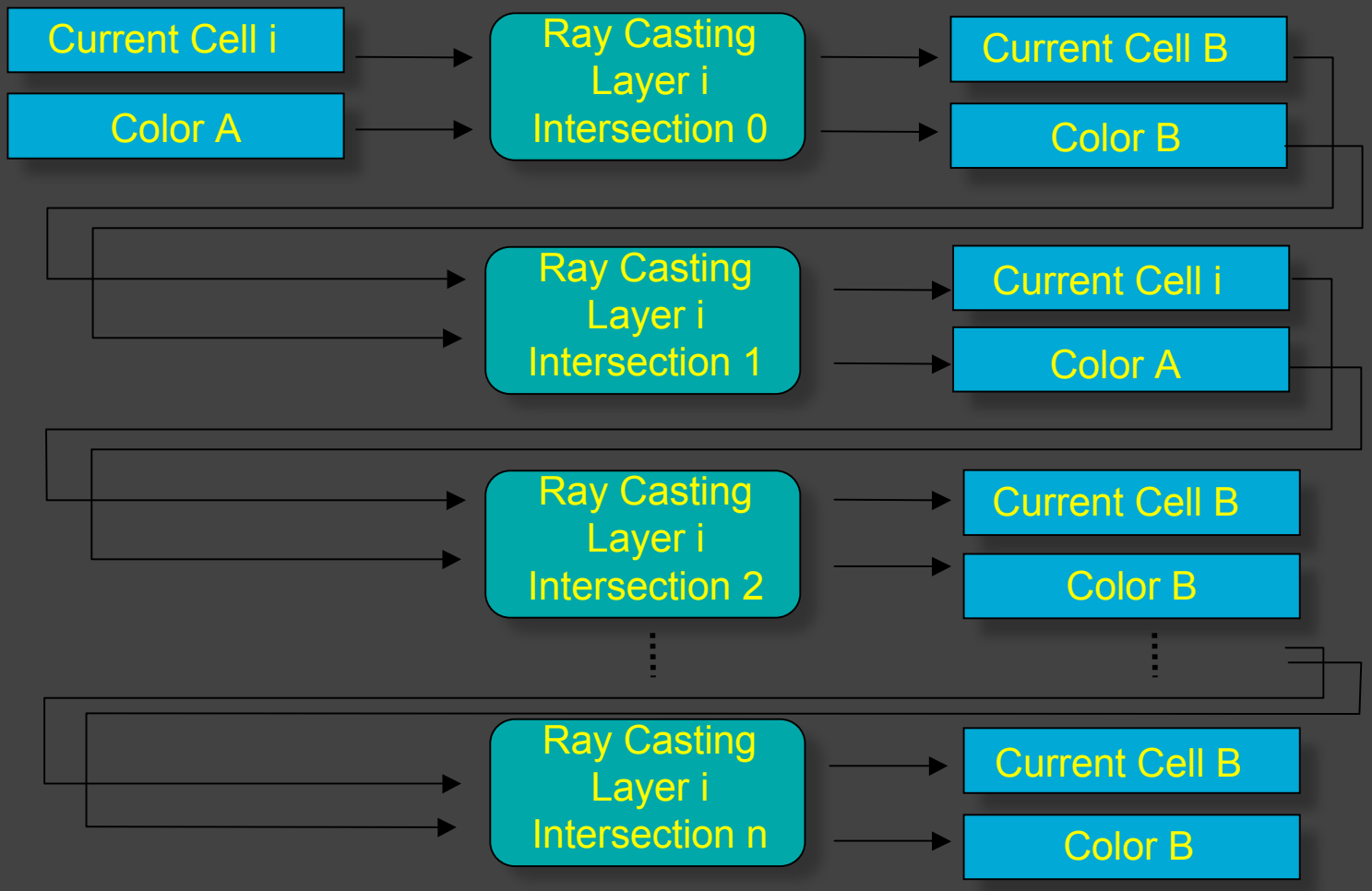
5: HRC - Implementation

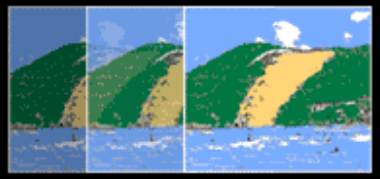




SIBGRAPI 2005

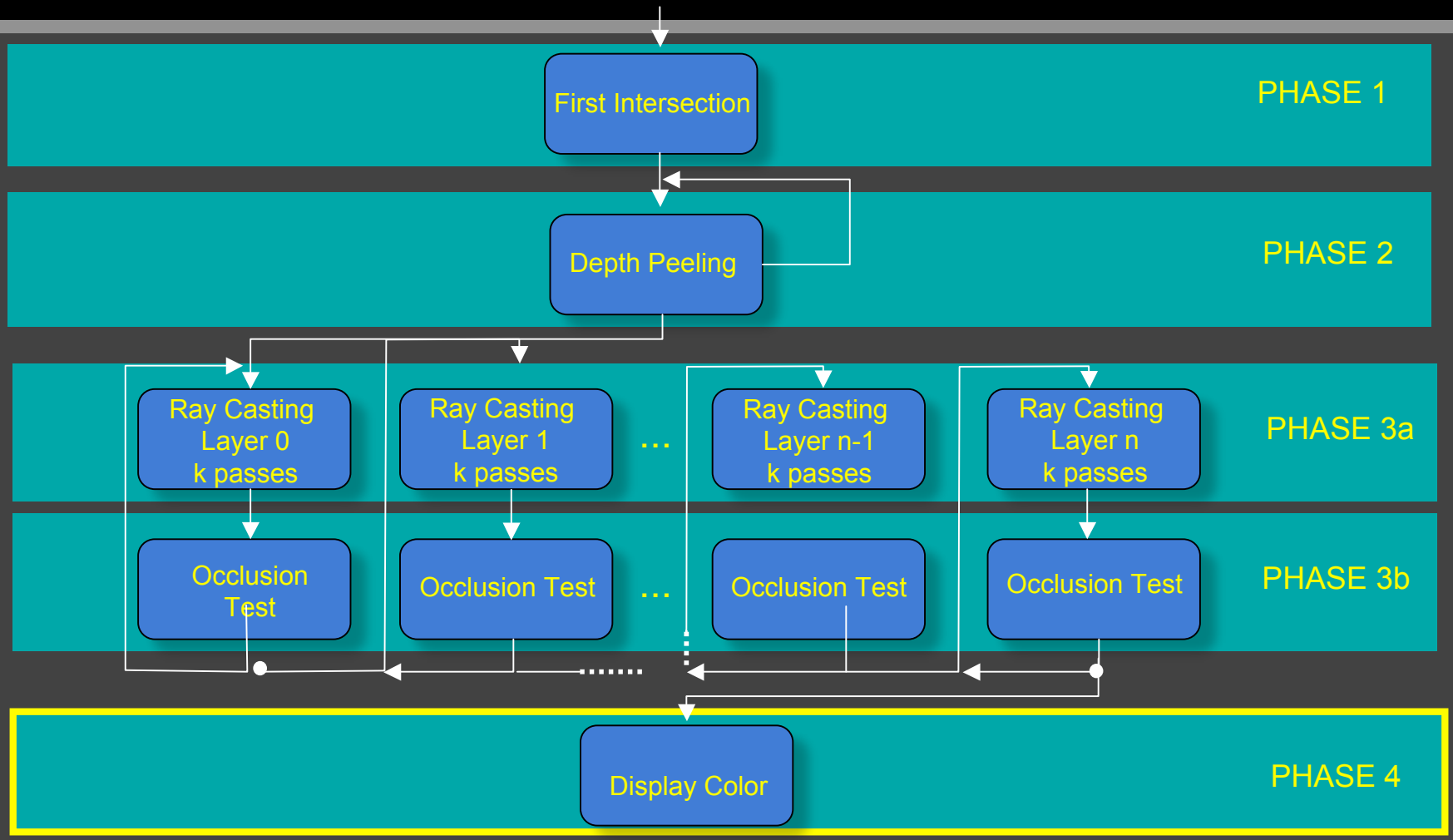
5: HRC - Implementation

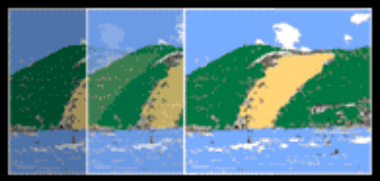




SIBGRAPI 2005

5: HRC - Implementation





SIBGRAPI 2005

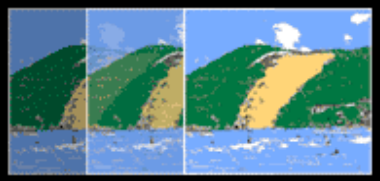
5: HRC - Results

Mesh	1x1	2x2	3x3	4x4	5x5	6x6
Spx1	297-516	312-516	312-515	297-500	313-547	329-609
Blunt	94-1203	94-1047	94-1062	94-1031	109-1032	109-1047
F117	156-484	157-453	157-453	156-437	172-438	172-469

GeForce 6800GT, 256 MB, AGP 8X

PIV 2.8 GHz, 1 GB RAM, WinXP (time in ms)

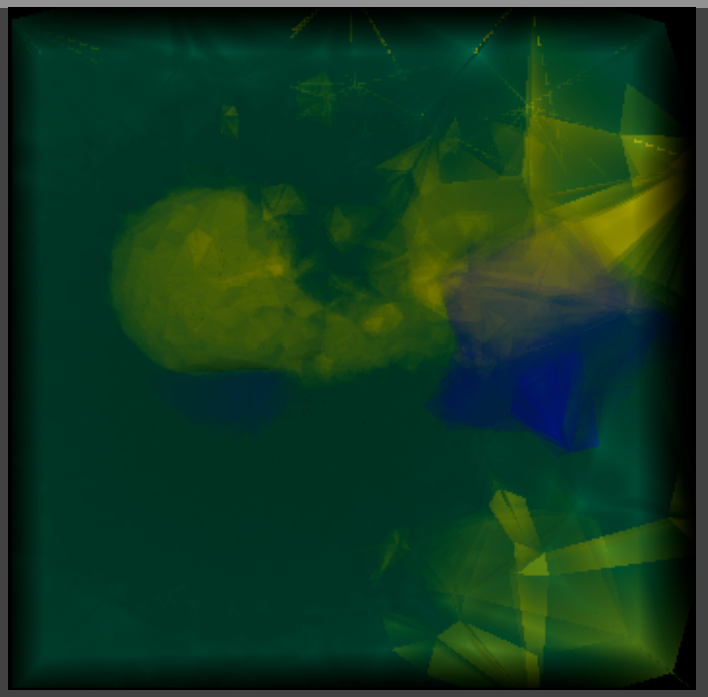
Mesh	Tets	Min FPS	Max FPS	Min Tets/s	Max Tets/s
Spx1	103K	1.88	3.38	195K	350K
Blunt	187K	0.93	10.64	173K	1.99M
F117	240K	1.83	3.77	438K	906K



SIBGRAPI 2005

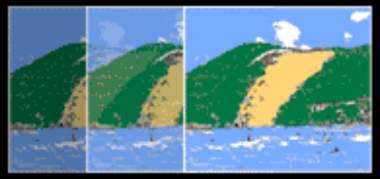
5: HRC - Conclusions

- Performance
- Good for pixel processor
  - Parallel + memory access



© Bernardon

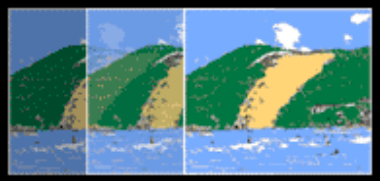
- http://www.cs.utah.edu/~csilva/software/gpu_volume_ray_casting.zip



SIBGRAPI 2005

5: TVSF

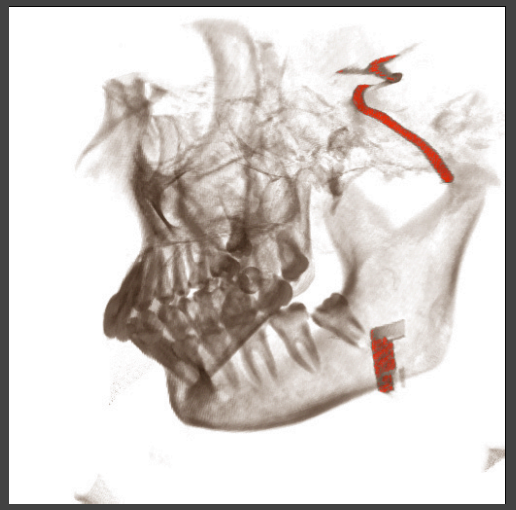
Time-Varying Scalar Fields



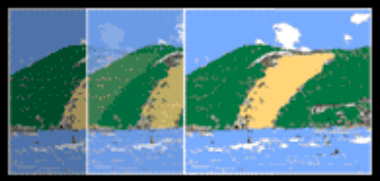
SIBGRAPI 2005

5: TVSF

- Schneider (Visualization 2003)
  - Compression technique for volume rendering of structured grids
    - Static data
    - Dynamic data



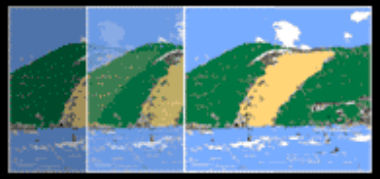
© Schneider 03



SIBGRAPI 2005

5: TVSF

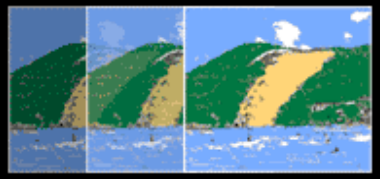
- Scalar data associated with vertices changes over time
- Temporal data arranged as a 64^{th} dimensional vector
- Decomposition + Quantization



SIBGRAPI 2005

5: TVSF

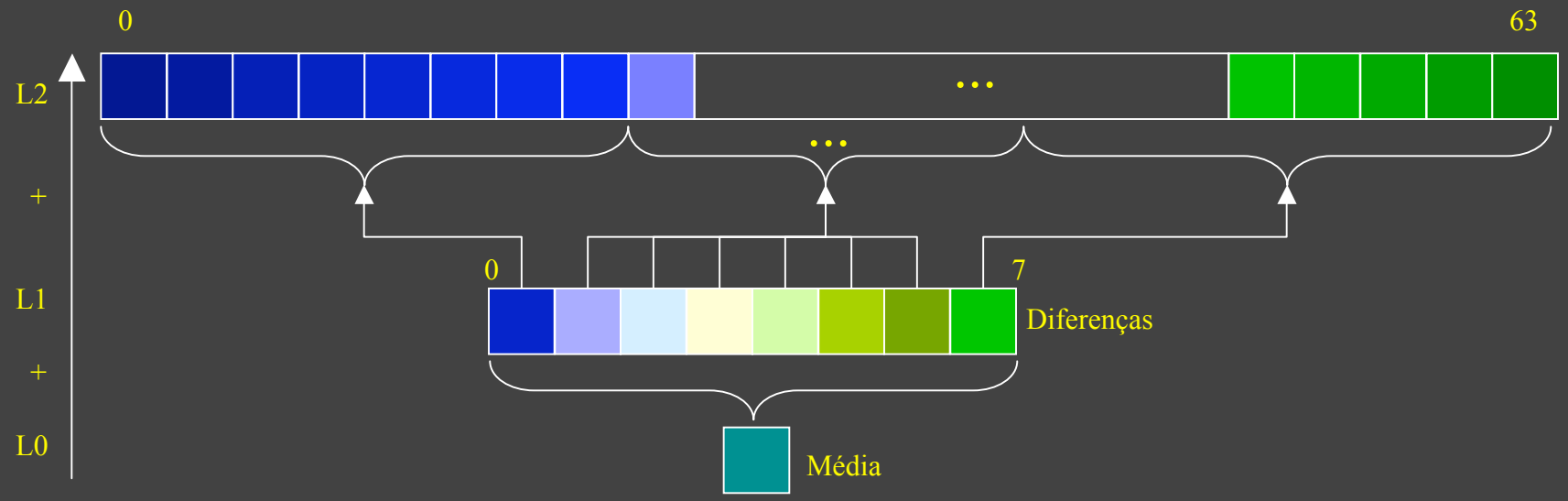
- Compression produces per scalar:
  - Medium
  - 2 indices
- Tables with 256 entries
  - 8 and 64 values per entry

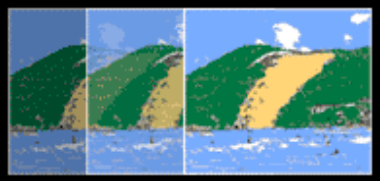


SIBGRAPI 2005

5: TVSF - HDD

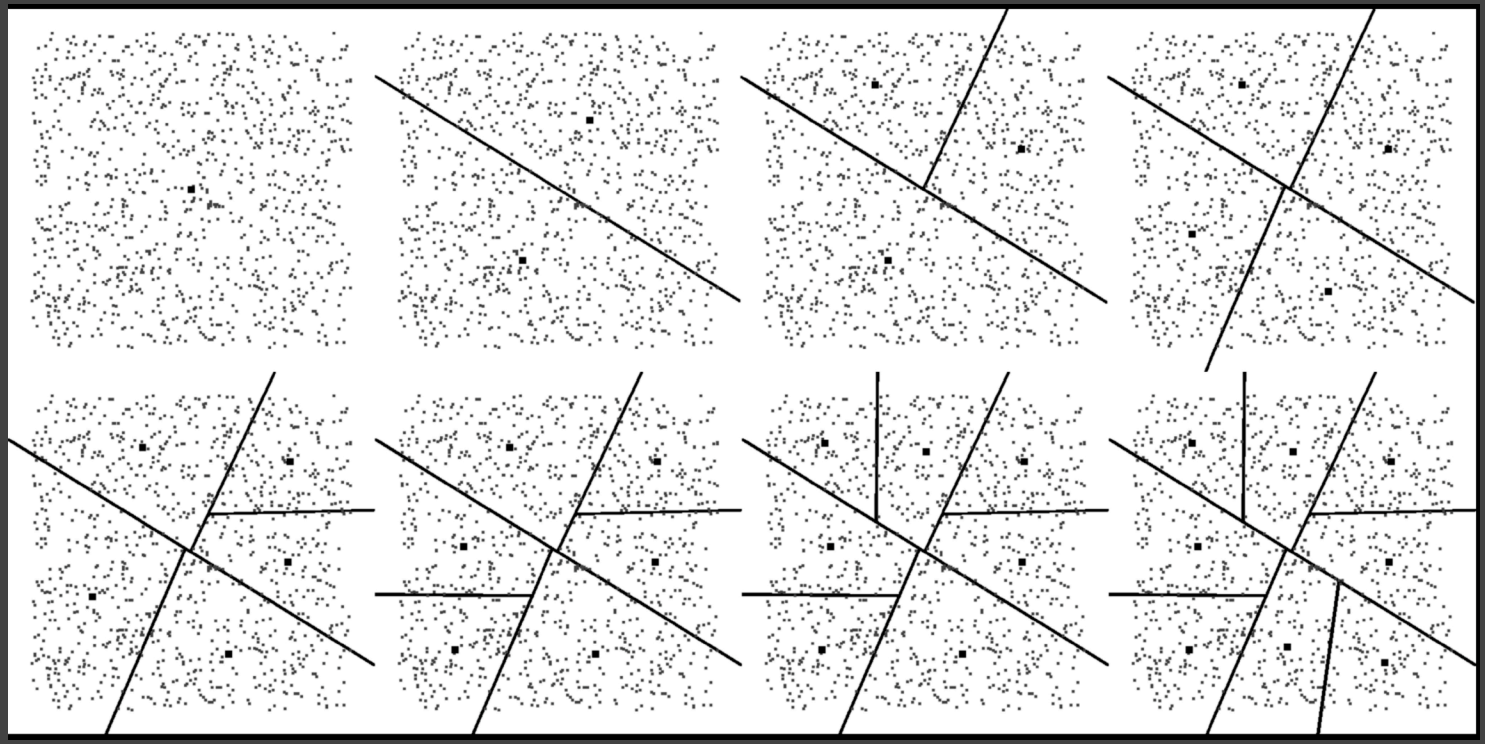
- Hierarchical Data Decomposition (HDD)



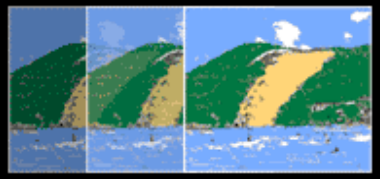


5: TVSF - VQ

- Vector Quantization (VQ)



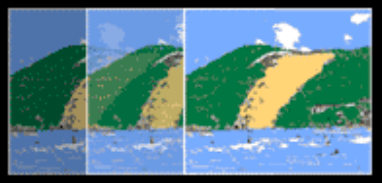
Schneider 2003



SIBGRAPI 2005

5: TVSF - Compression

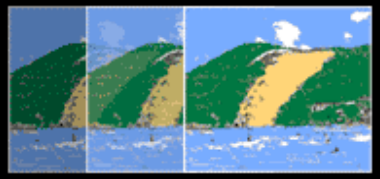
Mesh	Vertices	Tetrahedral	Time
SPX	19K	12K	64
SPX1	36K	101K	64
SPX2	162K	808K	64
BLUNT	40K	183K	64
TORSO	8K	50K	360
BRAIN	68K	387K	120



SIBGRAPI 2005

5: TVSF - Compression

Mesh	Original Size	Compressed Size	Compression Ratio	SNR Min	SNR Max	Error Max
SPX	4.75M	300K	16.21	39.44	42.08	0.0041
SPX1	9.00M	504K	18.29	39.45	41.96	0.0045
SPX2	40.50M	1.97M	20.57	39.24	41.88	0.0091
BLUNT	10.00M	552K	18.55	41.70	44.36	0.0046
TORSO	11.25M	1008K	11.43	20.53	28.12	0.0017
BRAIN	31.87M	1.73M	18.38	2.96	10.24	1.0632

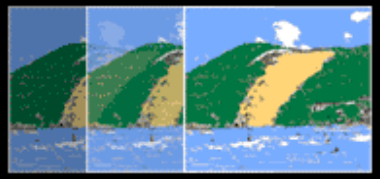


SIBGRAPI 2005

5: TVSF - HRC

Mesh Data	Texture Format	Coordinate		Data			
		u	v	r	g	b	a
Vertices	F32x4	t_u	t_v	$V0.x_t$	$V0.y_t$	$V0.z_t$	$V3.x_t$
Vertices	F32x4	t_u	t_{v+dv}	$V1.x_t$	$V1.y_t$	$V1.z_t$	$V3.y_t$
Vertices	F32x4	t_u	t_{v+2dv}	$V2.x_t$	$V2.y_t$	$V2.z_t$	$V3.z_t$
Neighbor Index	F32x4	t_u	t_v	$t_u(a_0)$	$t_v(a_0)$	$t_u(a_1)$	$t_v(a_1)$
Neighbor Index	F32x4	t_u	t_{v+dv}	$t_u(a_2)$	$t_v(a_2)$	$t_u(a_3)$	$t_v(a_3)$
Gradient Matrix	F32x4	t_u	t_{v+2dv}	a_{11}	a_{12}	a_{13}	a_{14}

- Static



SIBGRAPI 2005

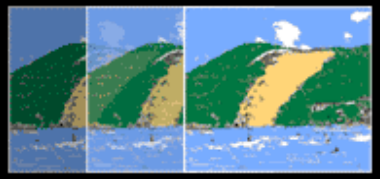
5: TVSF - HRC

Mesh Data	Texture Format	Coordinate		Data			
		u	v	r	g	b	a
Gradient Matrix	F32x4	t_u	t_v	a_{21}	a_{22}	a_{23}	a_{24}
Gradient Matrix	F32x4	t_u	t_{v+dv}	a_{31}	a_{32}	a_{33}	a_{34}
Gradient Matrix	F32x4	t_u	t_{v+2dv}	a_{41}	a_{42}	a_{43}	a_{44}

- Static

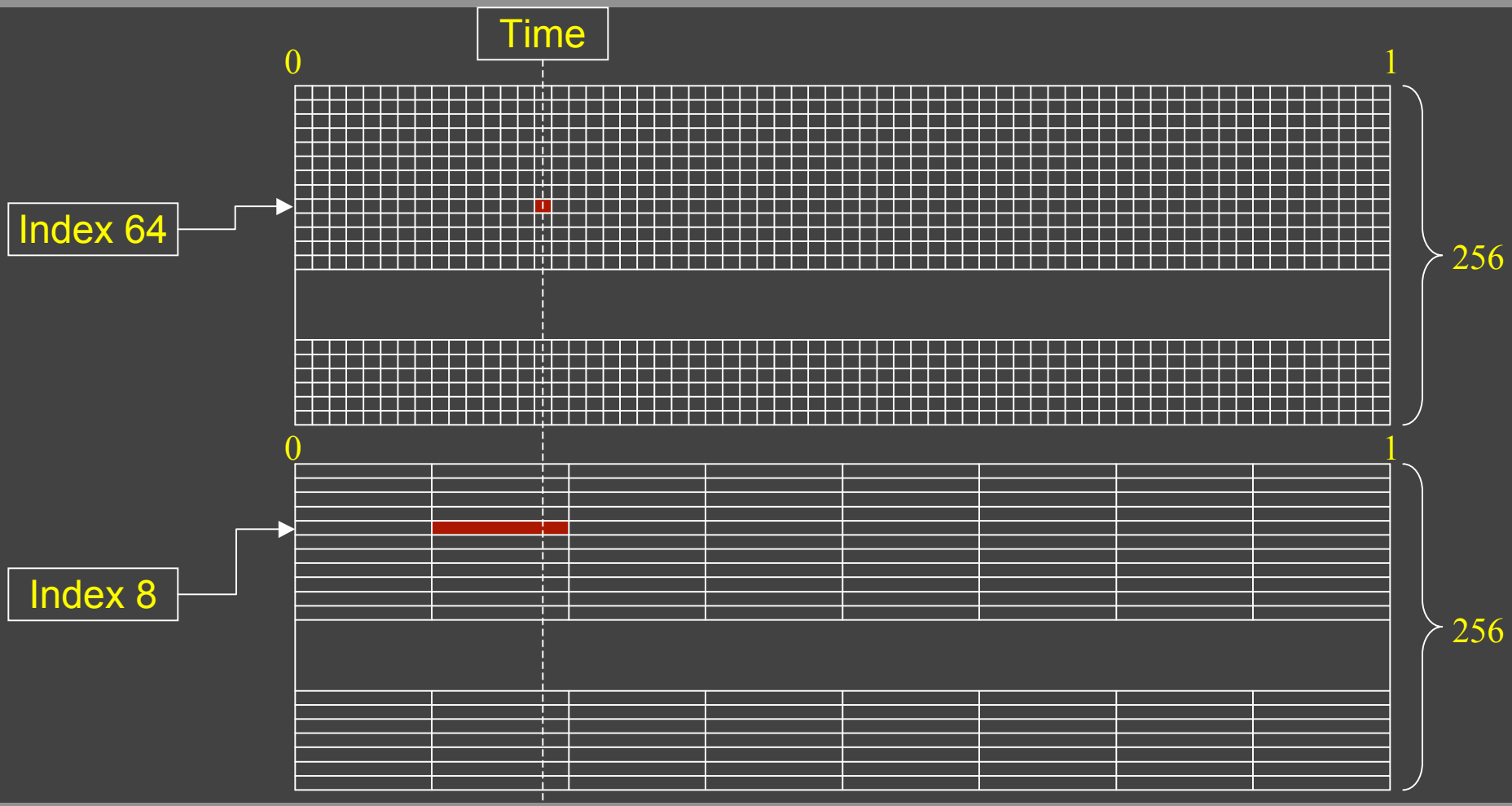
Compression Data	F32x4	t_u	t_v	$m_{0,t}$	$i8_{0,t}$	$i64_{0,t}$	$m_{3,t}$
Compression Data	F32x4	t_u	t_{v+dv}	$m_{1,t}$	$i8_{1,t}$	$i64_{1,t}$	$i8_{3,t}$
Compression Data	F32x4	t_u	t_{v+2dv}	$m_{2,t}$	$i8_{2,t}$	$i64_{2,t}$	$i64_{3,t}$

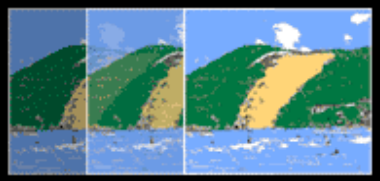
- Dynamic



SIBGRAPI 2005

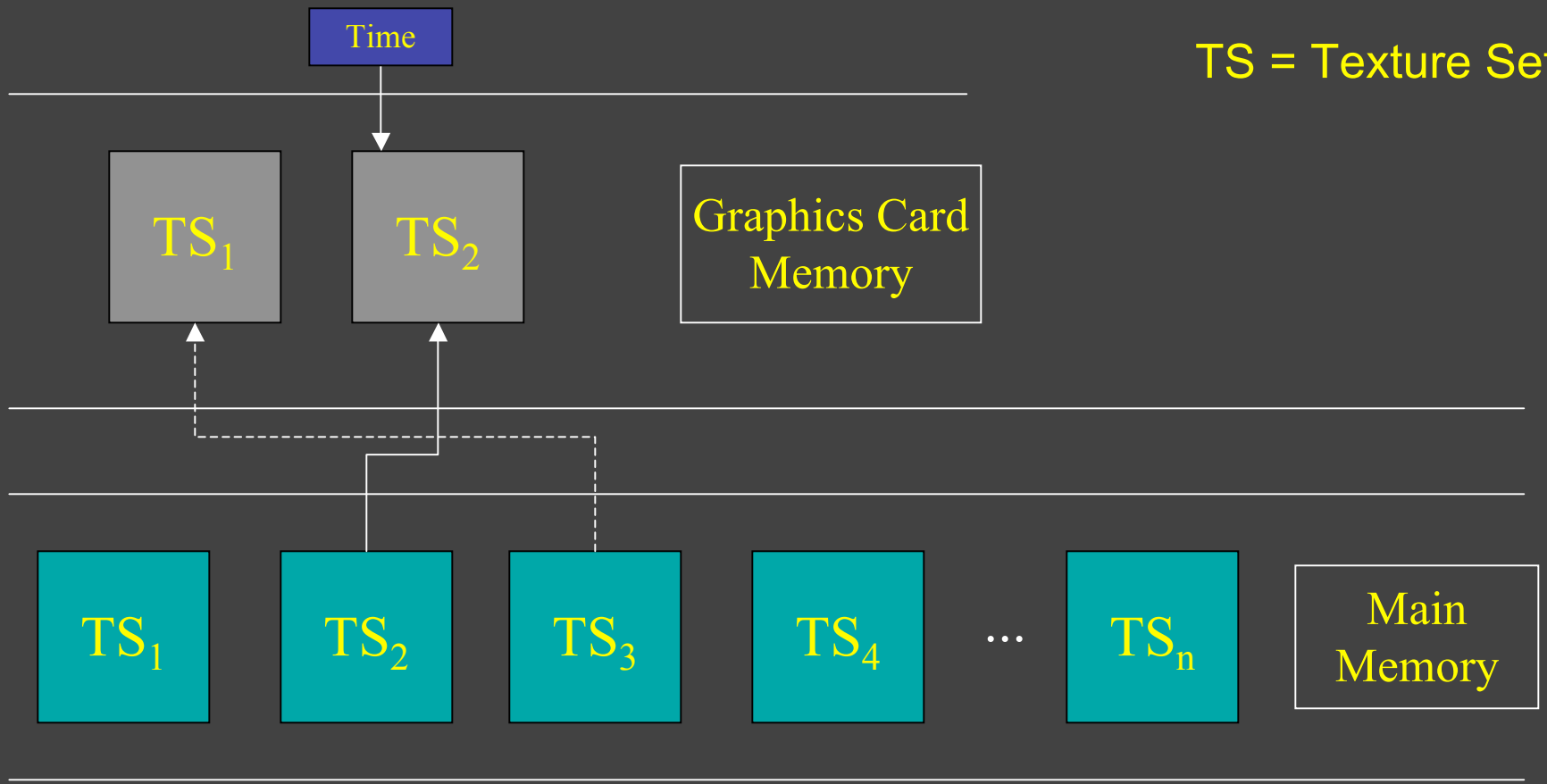
5: TVSF - HRC

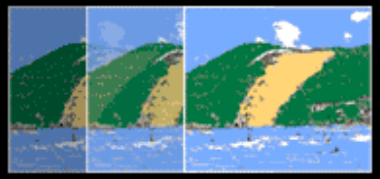




SIBGRAPI 2005

5: TVSF - HRC



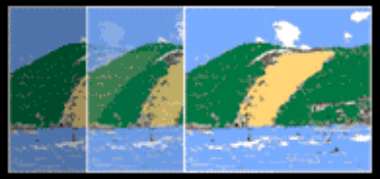


SIBGRAPI 2005

5: TVSF - HRC Results

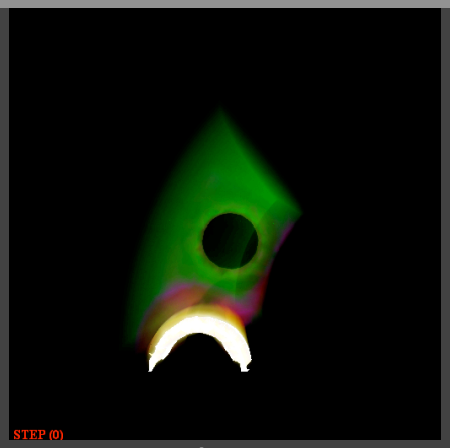
- Time in ms

Mesh	Static Min Time	Static Max Time	Dynamic Min Time	Dynamic Max Time
SPX	156	265	203	235
SPX1	297	500	406	672
BLUNT	94	1062	125	1125



SIBGRAPI 2005

5: TVSF - HRC Results



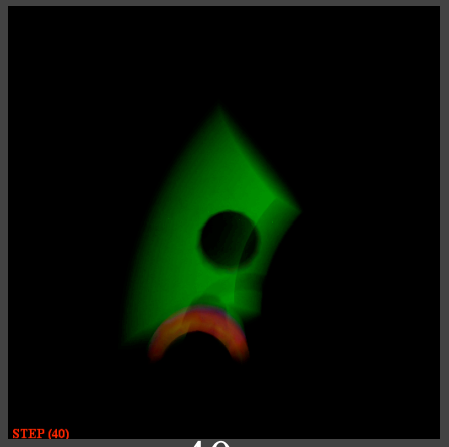
0

8

16

24

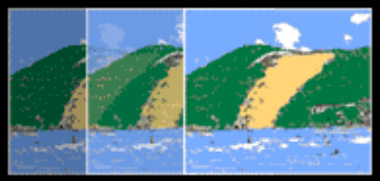
32



40

48

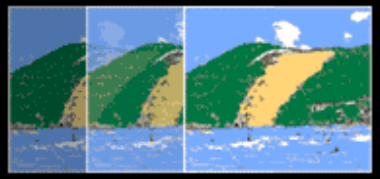
56



SIBGRAPI 2005

5: TVSF - HAVS

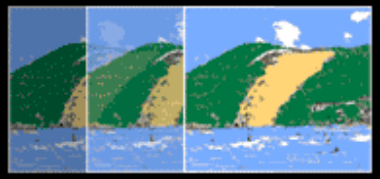
- Three approaches
  - Reconstruction at the vertex processor
    - Too slow due to small texture cache size
  - Reconstruction at the pixel processor
    - Too many information to send
  - CPU reconstruction
    - Good



SIBGRAPI 2005

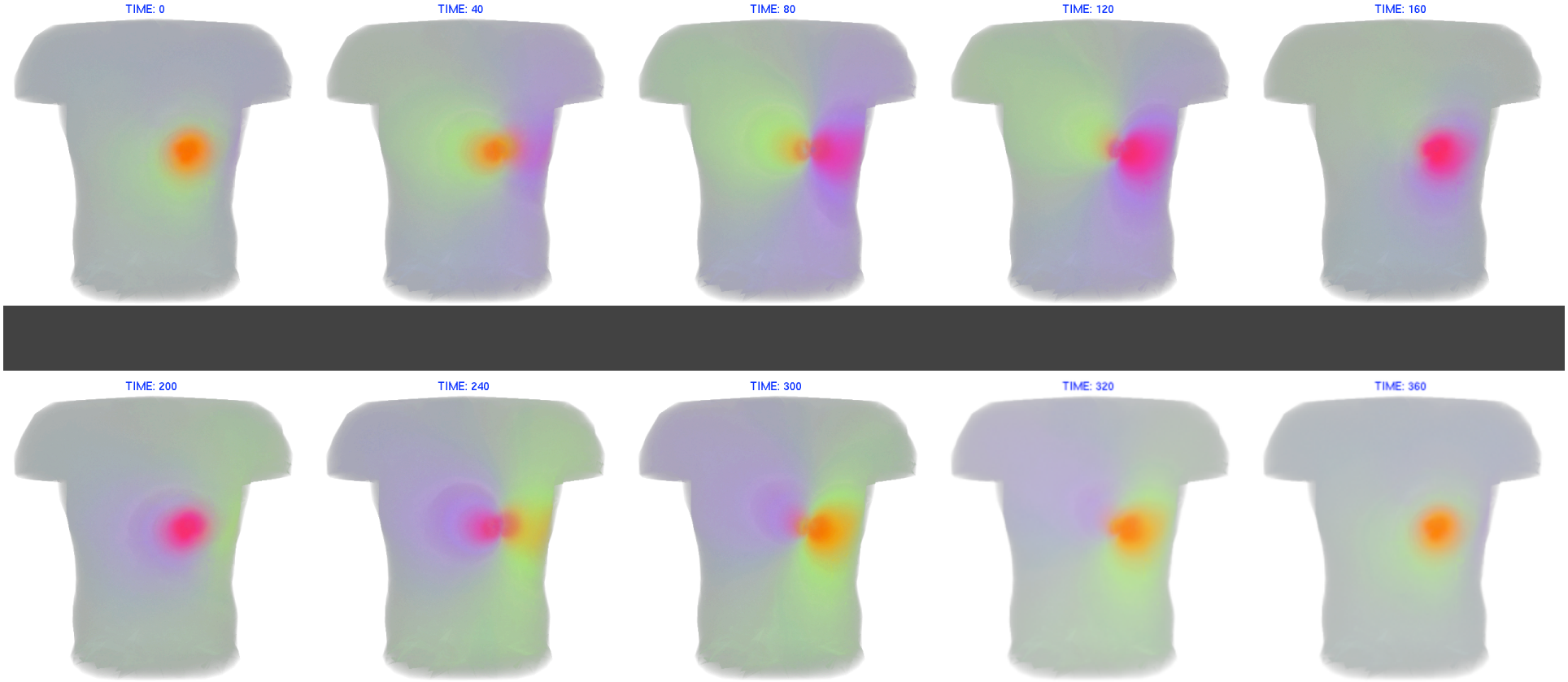
5: TVSF – HAVS Results

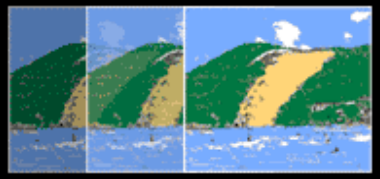
Mesh	Static Min Time	Static Max Time	Dynamic Min Time	Dynamic Max Time
SPX	31	47	31	47
SPX1	109	125	110	125
SPX2	703	813	1016	1157
BLUNT	156	312	218	266
TORSO	62	79	62	79
BRAIN	438	500	578	625



SIBGRAPI 2005

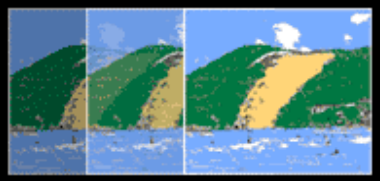
5: TVSF - HAVS Results





SIBGRAPI 2005

5: TVSF - HAVS Results



SIBGRAPI 2005

5: TVSF - Conclusions

- Compressed data
- Low performance loss for HRC
- HAVS datasets:
  - Small: had no performance changes
  - Larger: up to 40% performance loss
    - Use GPU to reconstruct data