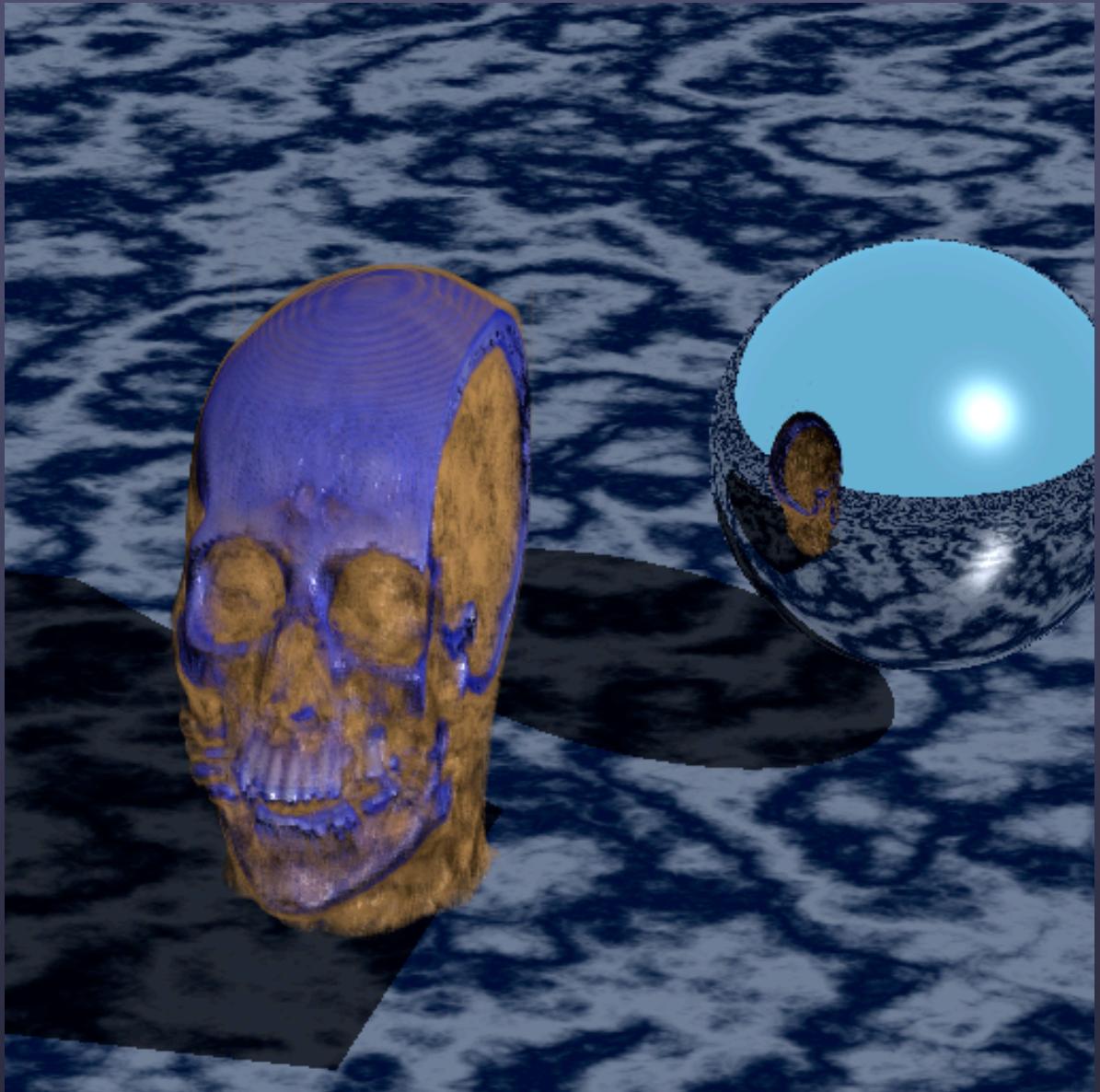


# Volume Rendering Clarification

March 23, 2005

CS6620 Spring 05

- Program 8 questions?



CS6620 Spring 05

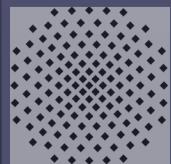
# Complex optimizations

- Use adaptive stepping
- Use adaptive grid
- Make a faster way to skip over transparent space (maybe adaptive grid)

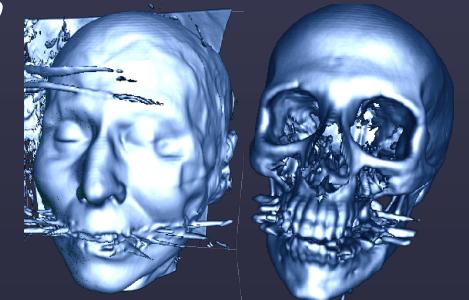
# Possible improvements

- Use a better transfer function
- Change shadow infrastructure to allow attenuated shadows
- Allow self-shadowing by tracing shadow rays at each point
- Combine above two effects
- Full scattering model (nice for clouds, but slow)

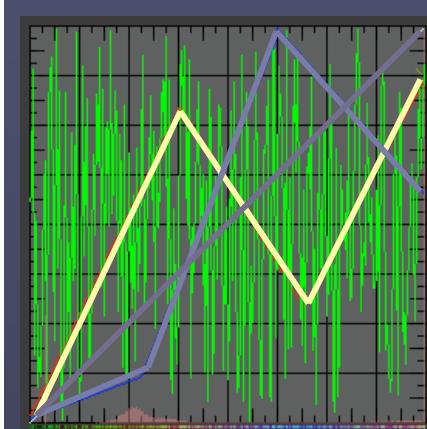
The next several slides are from  
this talk:  
**High-Quality Pre-Integrated Volume  
Rendering Using Hardware  
Accelerated Pixel Shading**



**Klaus Engel, Martin Kraus, Thomas Ertl**  
*Visualization and Interactive Systems Group*  
*University of Stuttgart, Germany*

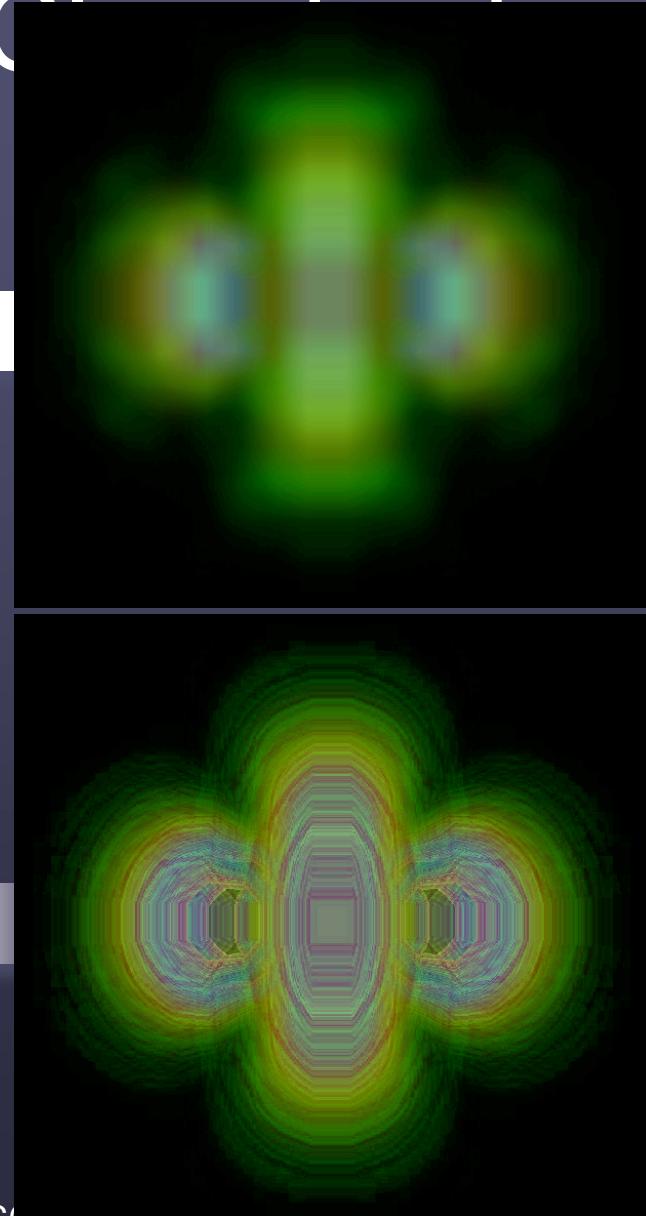


# Volume Classification



classification

interpolation



CS6620 Spring 05

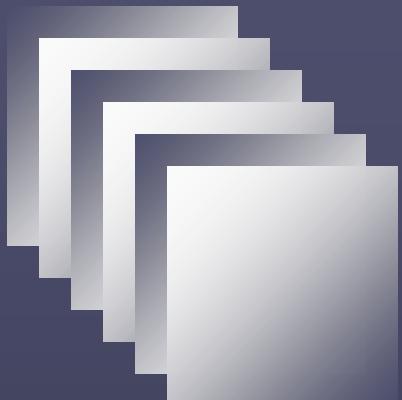
Pre-classification

Post-classification

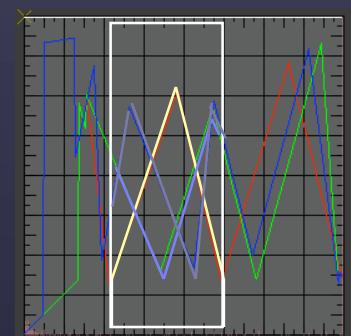
# Pre-integrated Volume Rendering - Idea

slice-by-slice

slab-by-slab

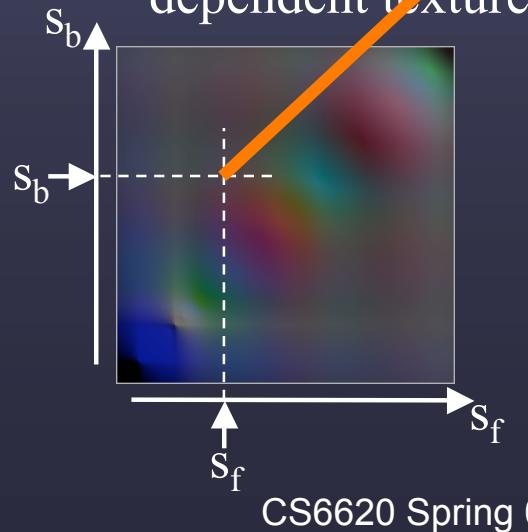


pre-integrate all  
possible combinations



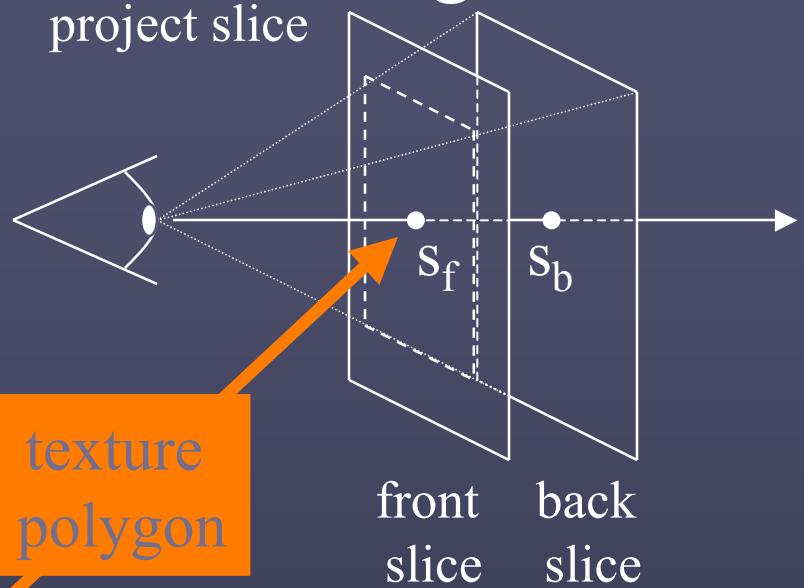
$S_f$      $S_b$

fetch integral from  
dependent texture



CS6620 Spring 05

project slice

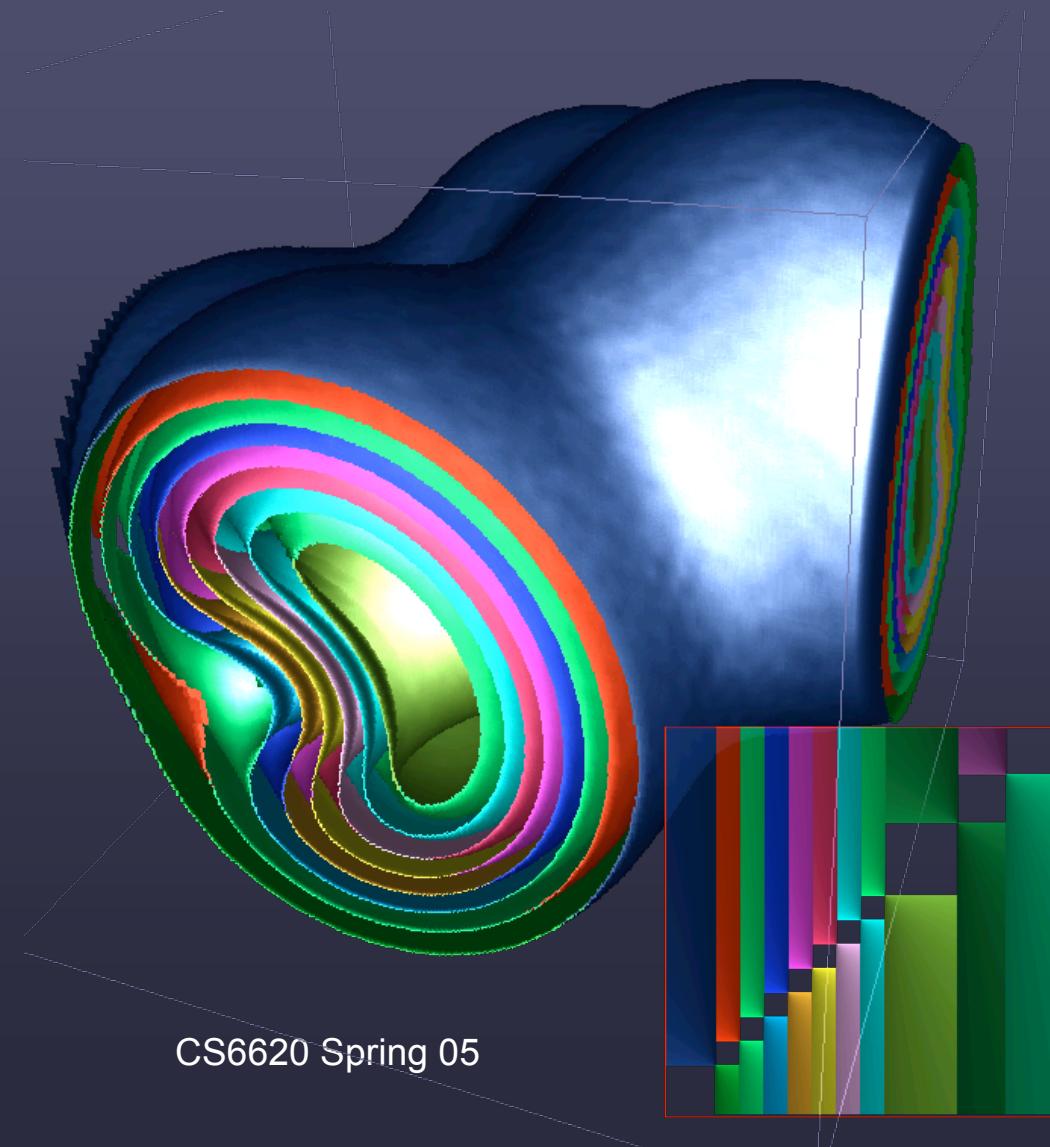


texture  
polygon

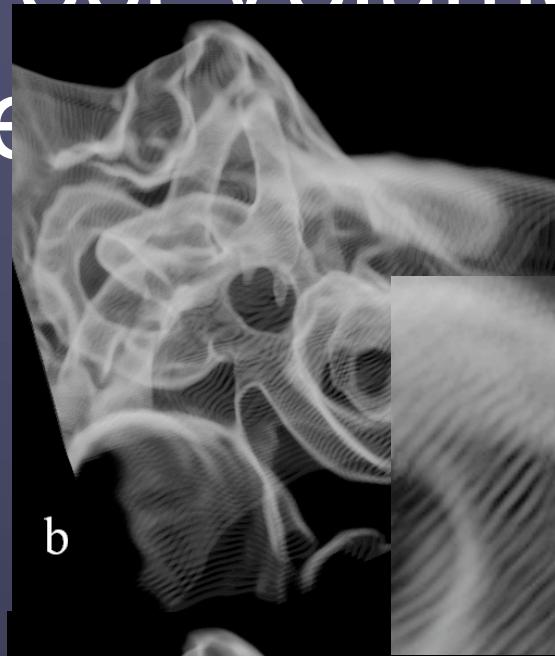
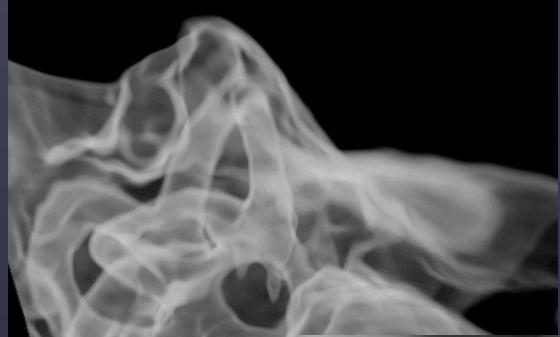
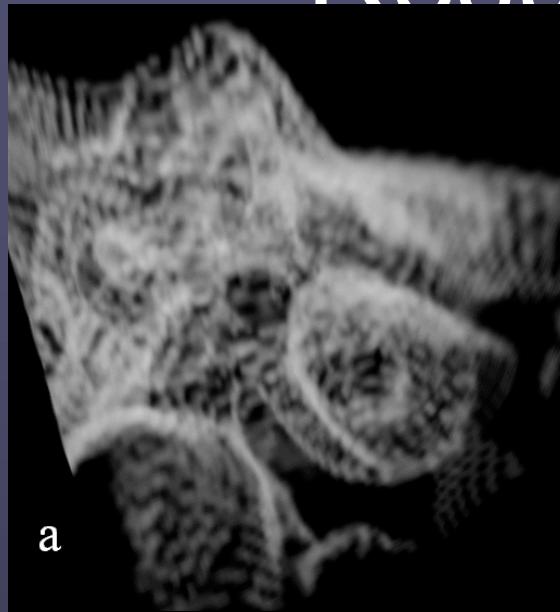
front slice    back slice

hardware-accelerated  
implementation  
on NVidia GeForce3 chip

# Example

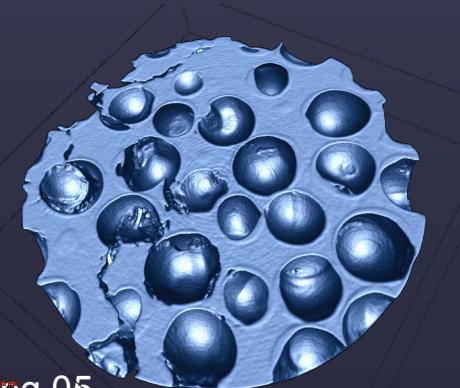
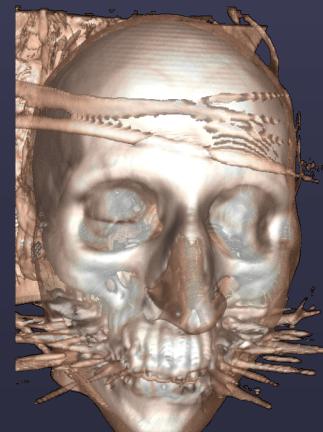
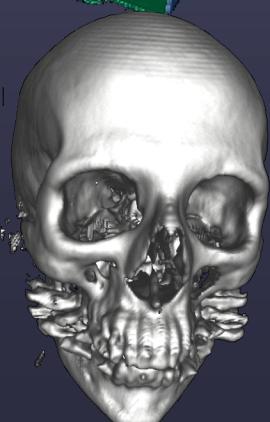
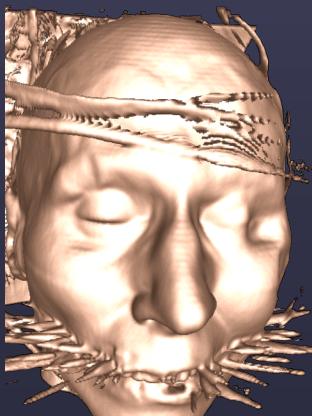
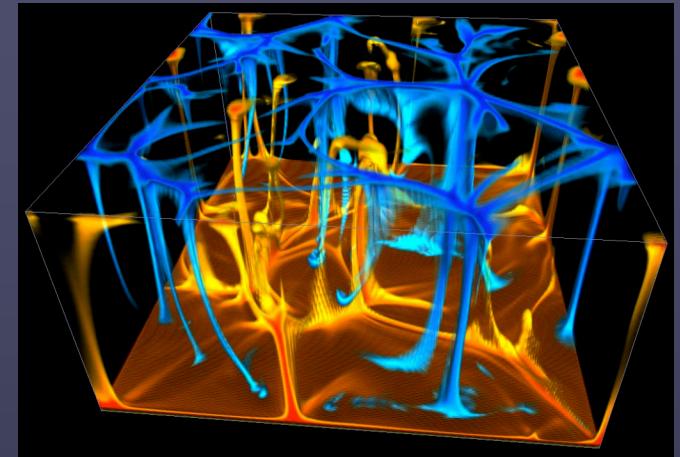
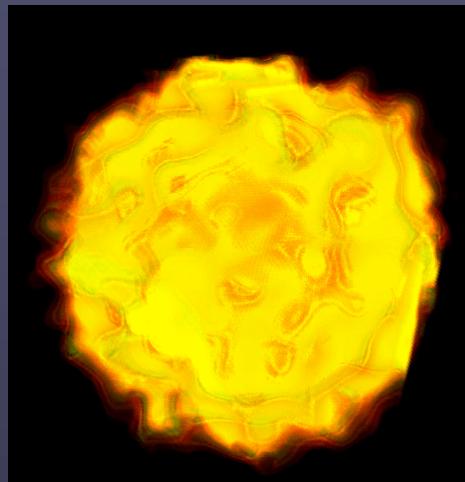
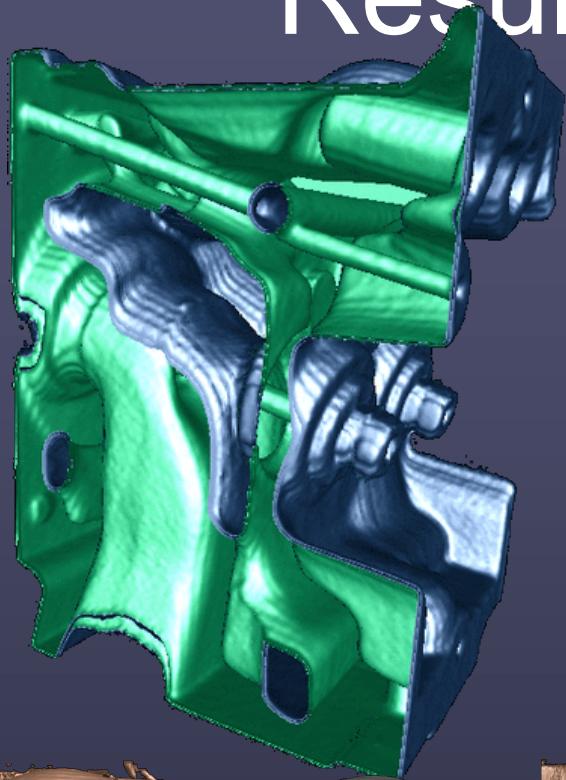


# Results – Direct Volume Render



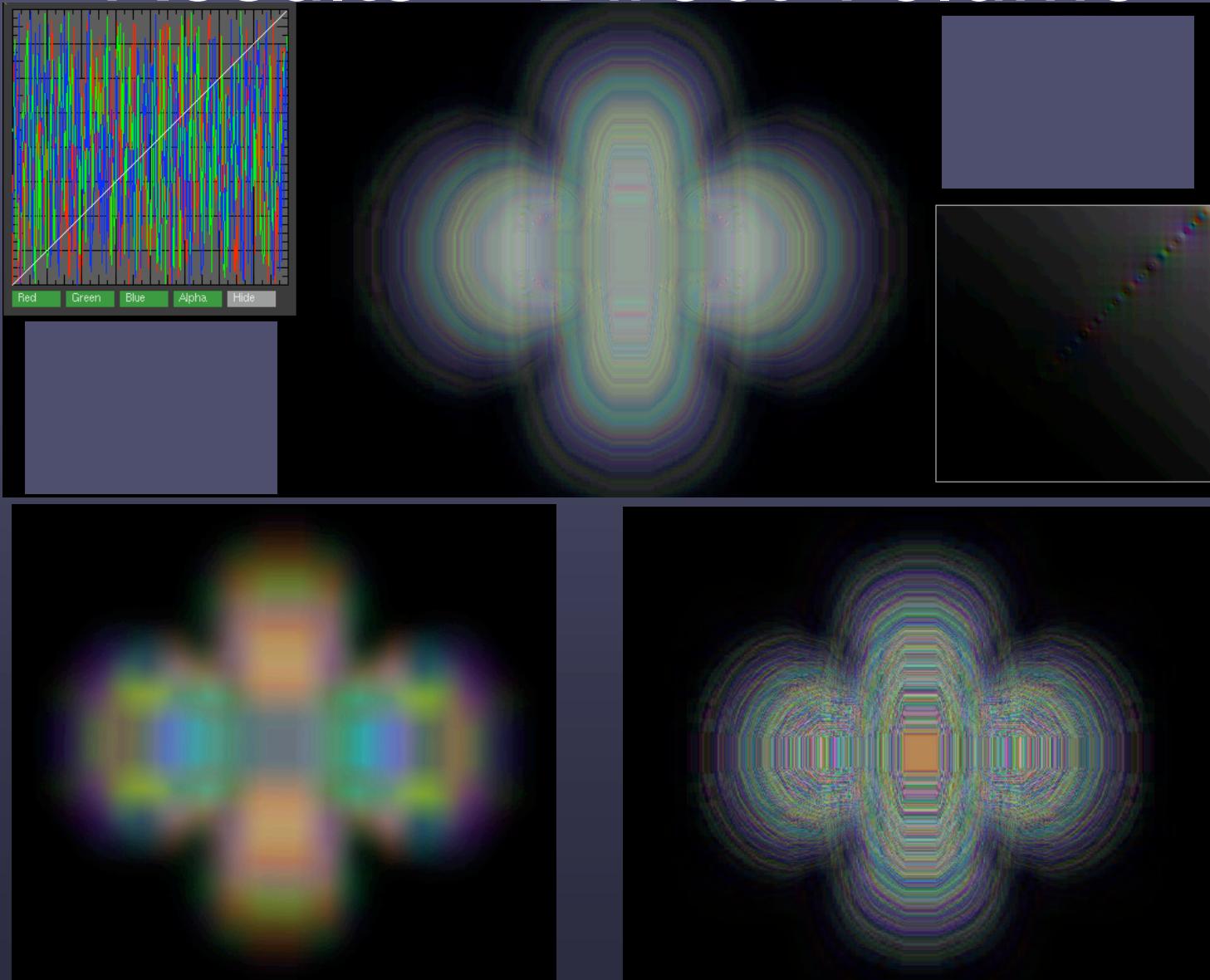
128 slices  
pre-integrated

# Results - Isosurfaces



CS6620 Spring 05

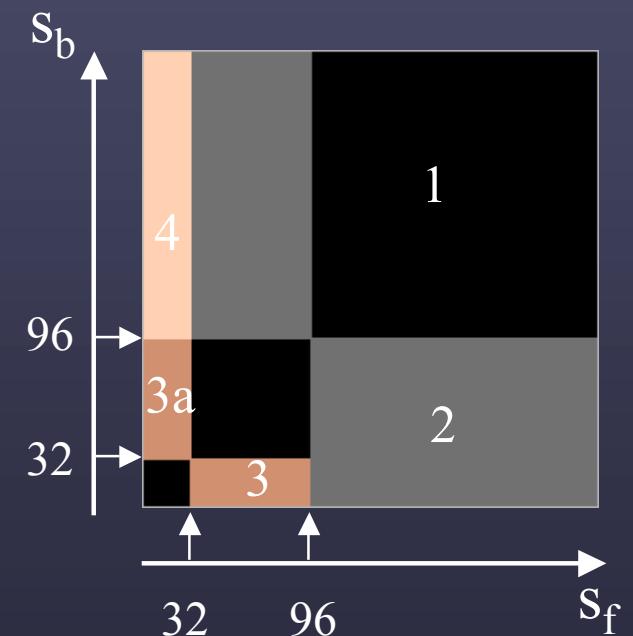
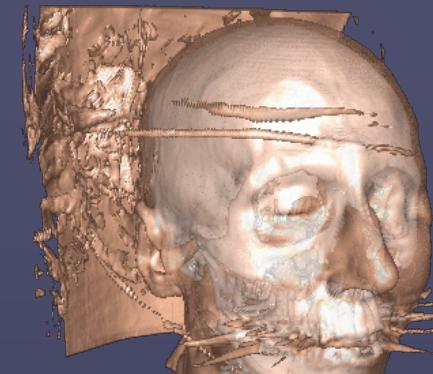
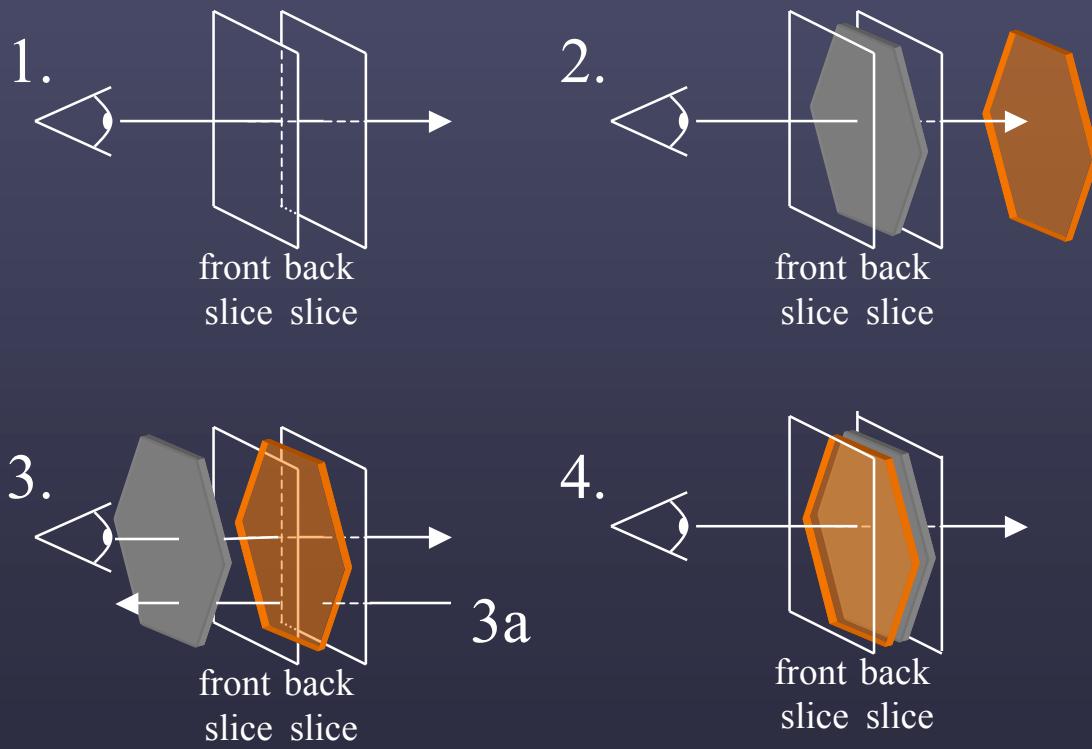
# Results – Direct Volume



CS6620 Spring 05

# Isosurfaces

Isosurfaces:  
particular dependent texture



C:\ENGEV\PRE-IN~1\SIMPLE.EXE

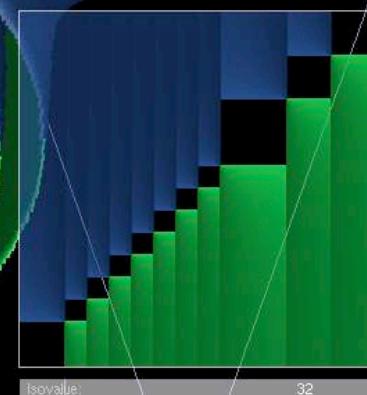
Isovalue 1/10: 32  
Transparency: 1.000000  
Mode: IsoSurface  
Resolution: 1/1 (off)  
PMode: ProjectBackSliceToFrontSlice  
Lighting: Diffuse  
Light Intensity: 0.980000

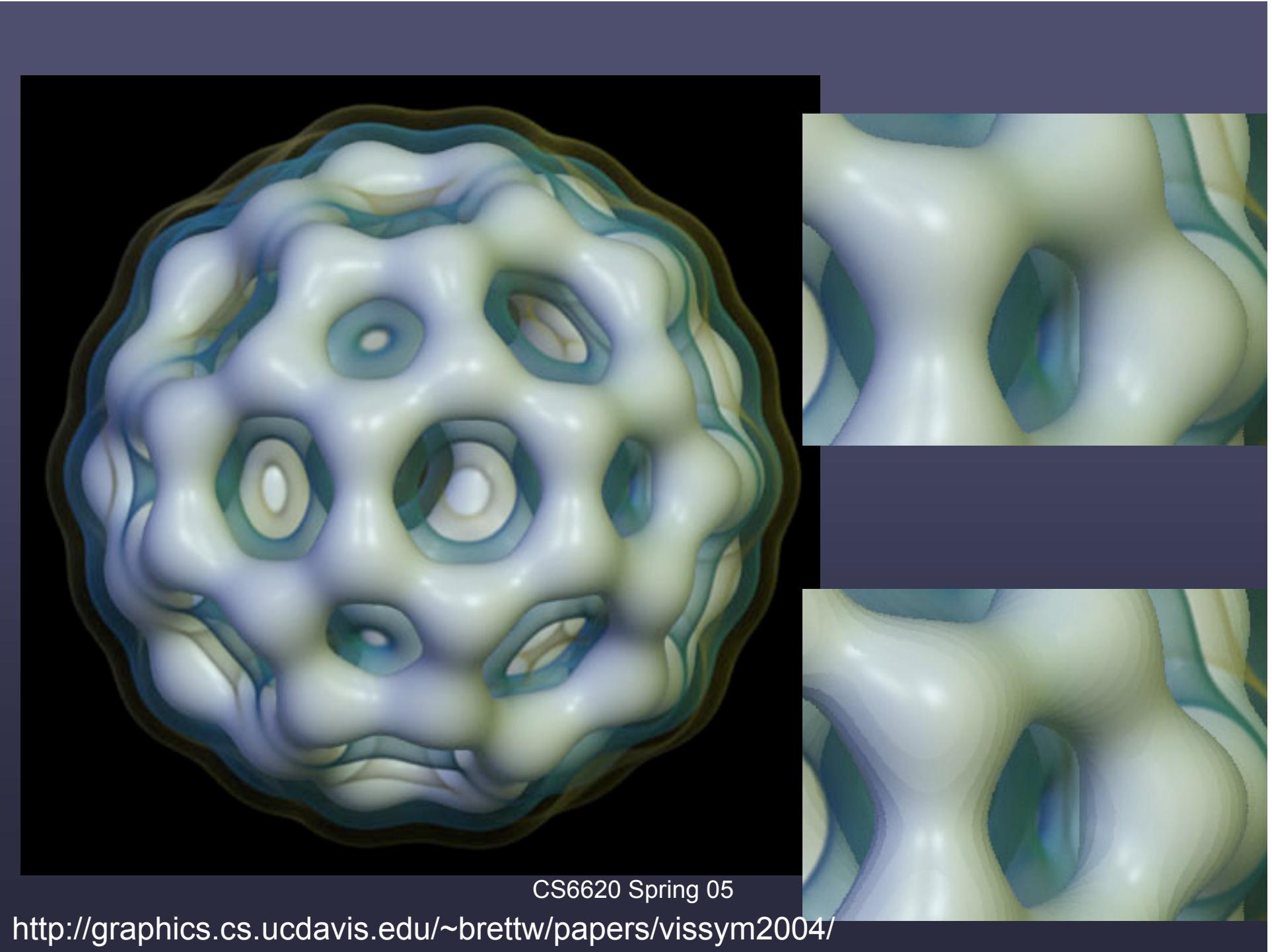
Press 'h' for help  
Compressed RGBA textures: No  
Correct Transparency: No  
Classification: Pre-Integrated

## Two-sided lighting

IsoSurf Direct VolShd

Pre-Integrated Volume Renderer V1.3 – Frame rate: 5.681818





CS6620 Spring 05

<http://graphics.cs.ucdavis.edu/~brettw/papers/vissym2004/>

# Course roadmap

- Almost 2/3 done
- Hopefully been fun so far
- Fun part is just beginning
- Remaining assignments:
  - Instancing (lecture on Friday)
  - Monte carlo ray tracing (your choice of glossy reflections, motion blur, depth-of-field, soft shadows, blurry transparent objects, etc.)
  - Final project (your choice, possible contest)

# Final project

- Be thinking about ideas for your final project
- No new features need to be implemented
- But, this is your chance to show off!

# Rendering contest

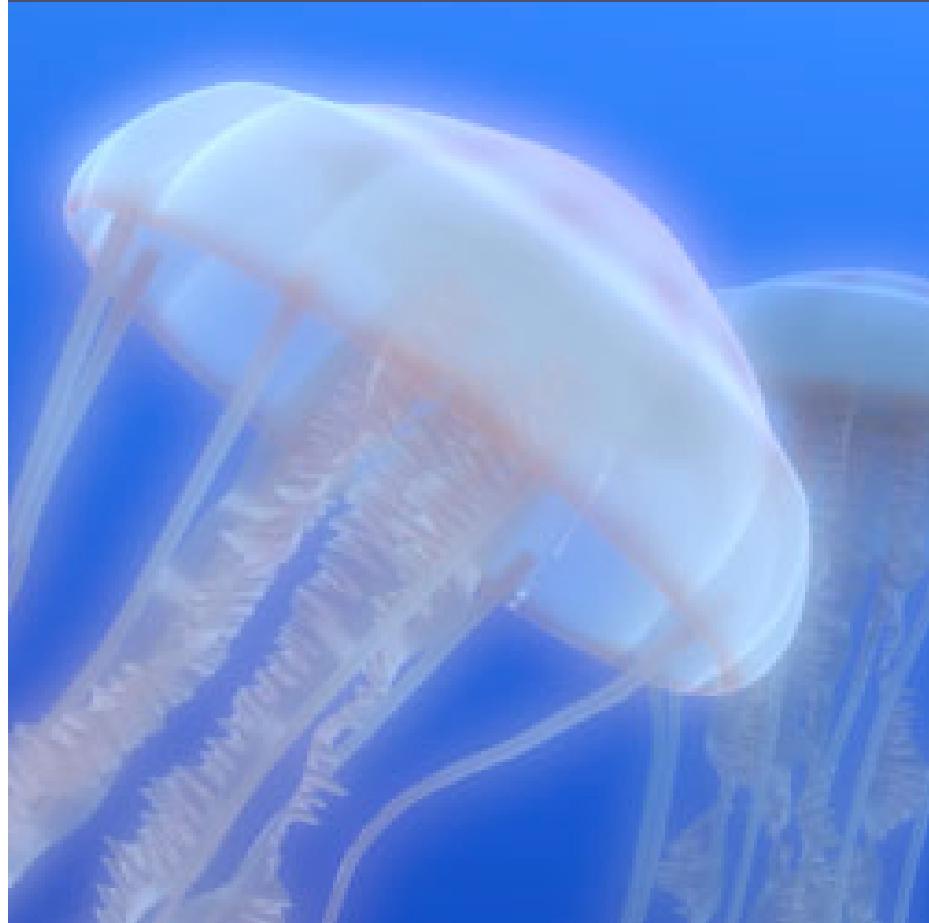
- For the final project images, you can optionally enter a rendering contest
- Prizes will be given to the best image as judged by an independent panel

# Final project ideas

- Stanford does a similar thing:

<http://graphics.stanford.edu/courses/cs348b-competition/>

- The Internet Ray Tracing Competition  
([www.irtc.org](http://www.irtc.org))



CS6620 Spring 05

<http://www-graphics.stanford.edu/courses/cs348b-competition/cs348b-04/>



CS6620 Spring 05

<http://www-graphics.stanford.edu/courses/cs348b-competition/cs348b-03/>

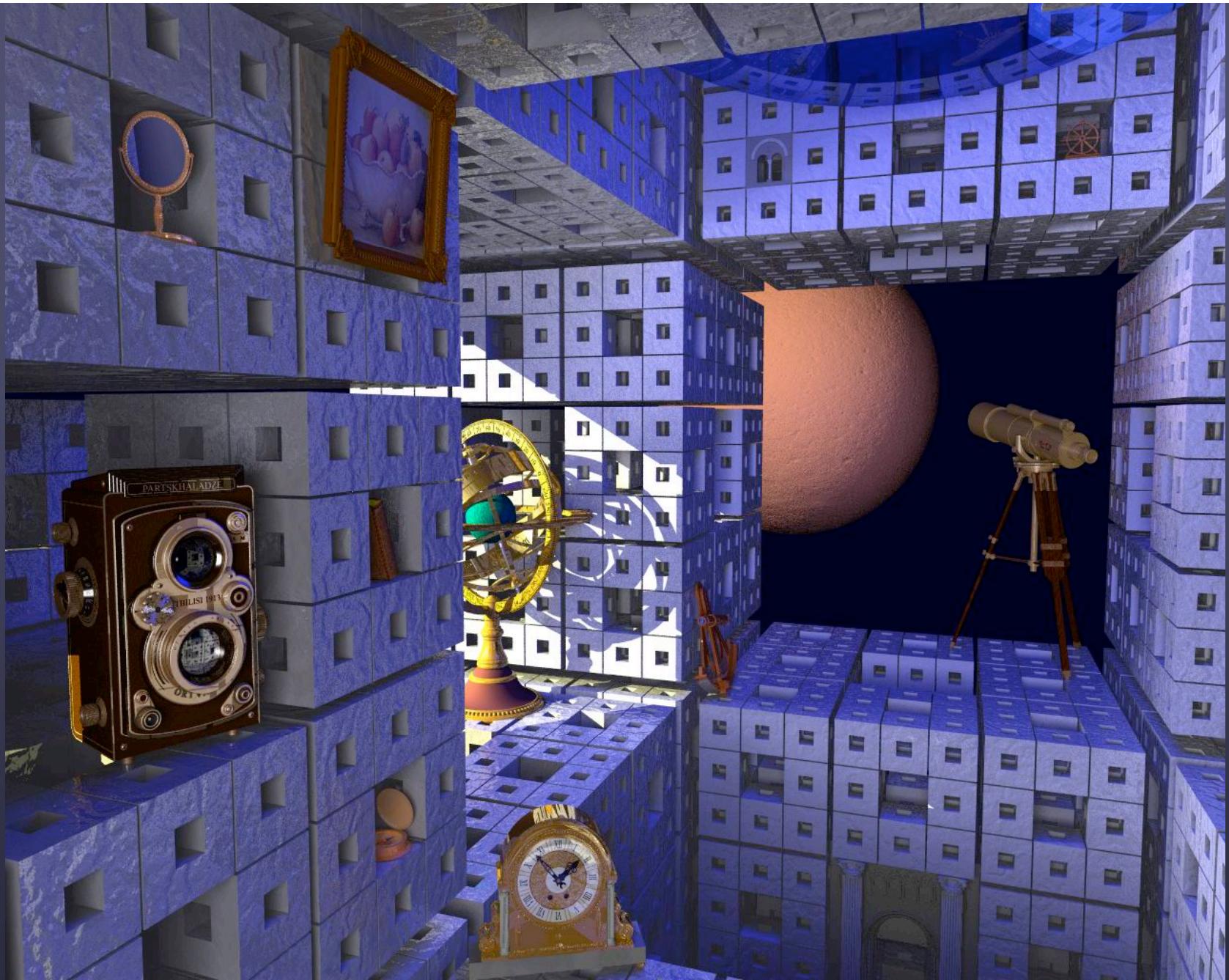
<http://www-graphics.stanford.edu/courses/cs348b-competition/cs348b-03/>





[www.irtc.org](http://www.irtc.org)

Where is my Muse ?  
© 2004 - LightBeam



CS6620 Spring 05

[www.irtc.org](http://www.irtc.org)

# Your idea here

- You now have a very full-featured ray tracer
- By the final project you will be able to make very realistic images
- Possibilities limited only by your imagination and artistic ability
- Start thinking about your ideas!