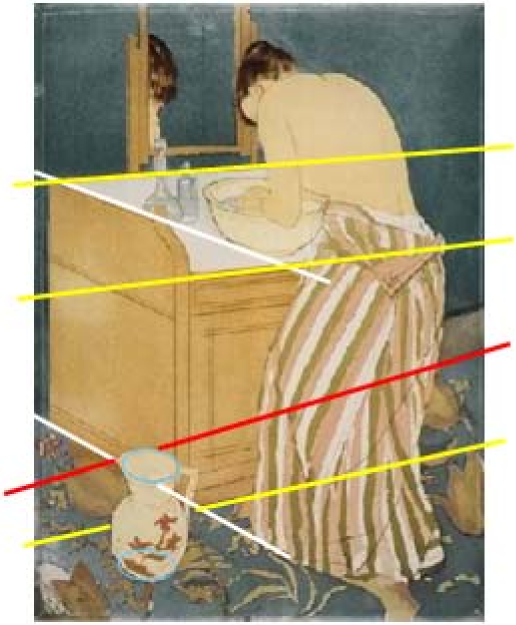
3D Screen-space Widgets for Non-linear Projection

Patrick Coleman, Karan Singh (Univ of Toronto)
Nisha Sudarsanam, Cindy Grimm (Washington Univ in St. Louis)
Leon Barrett (Univ of Calif, Berkeley)

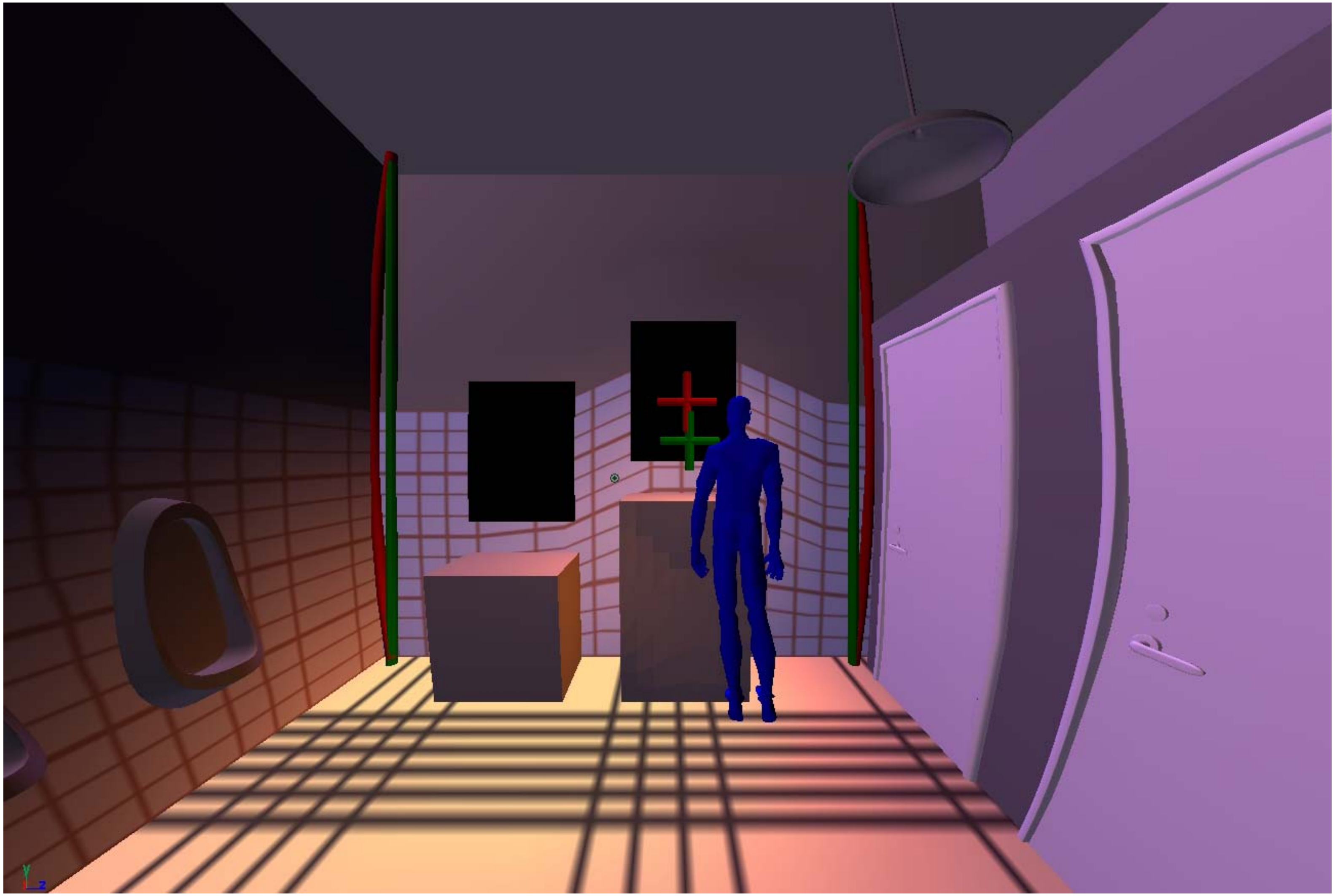
What is non-linear perspective?

- Perception uses *locally* linear perspective
  - Depth, placement in scene
- Fovea only encompasses a small number of degrees
  - 3D sense built out of saccades
- Artists use this fact to make better use of 2D canvas
  - Local perspective maintained
  - Continuity between local perspectives



Marie Cassett

What does this mean?



Mechanics

- Define more than one camera C_i
- Define region of influence of each camera w_i
- Use blended combination of cameras
  - Different camera for each vertex
    - (Dual of free-form deformation)
  - Blend matrices, projected point, camera parameters…

$$v' = (\sum w_i C_i)v$$

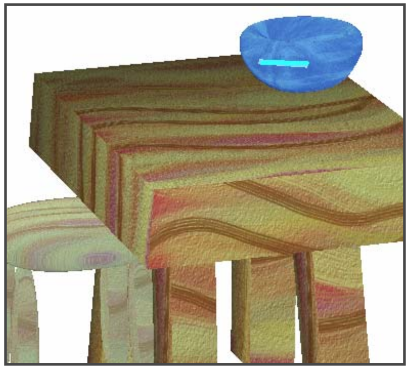
Karan Singh, A Fresh Perspective, Graphics Interface 2002

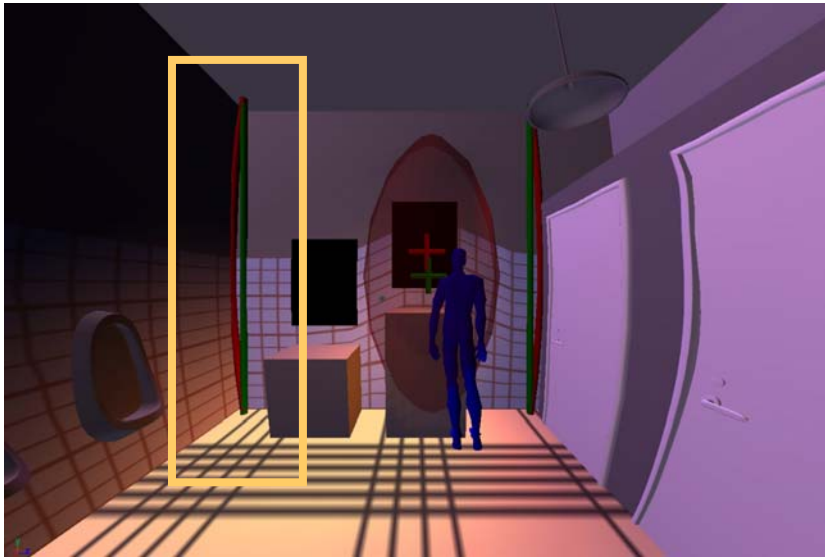
It’s all in the user interface…

- Each camera has 11 degrees of freedom
  - 6 for pose (position, orientation)
  - 5 internal (zoom/focal length, center of projection, skew, aspect ratio)
- Using n cameras implies 11n parameters…
  - One mouse

Some observations

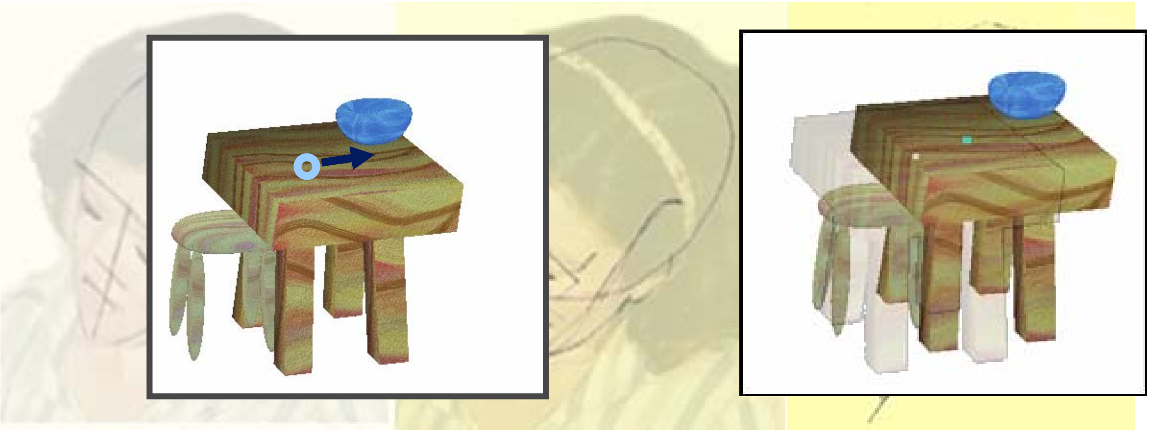
- Scene should have some coherency
  - Dominant (default) view
- Other cameras are small, local changes to default view
  - Bow the wall out
- Changes happen in screen space
  - Can be sketched
  - Simple geometry





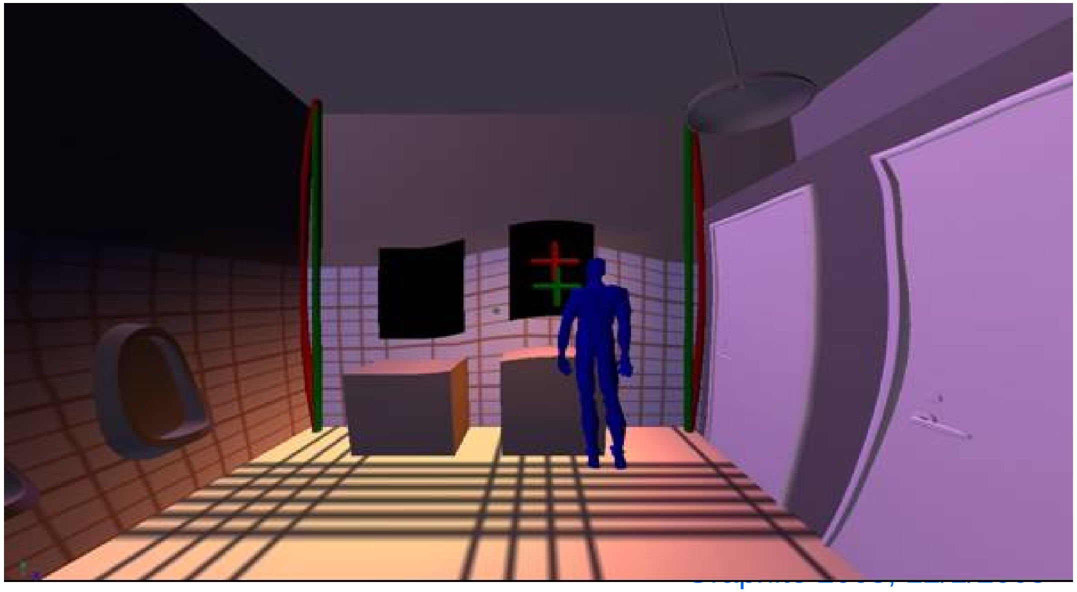
Basic approach

- Use geometric proxies
  - Lines, points, boxes
- Image-space change controls camera change
  - Point moves, camera pans
  - Also controls region of influence of new camera



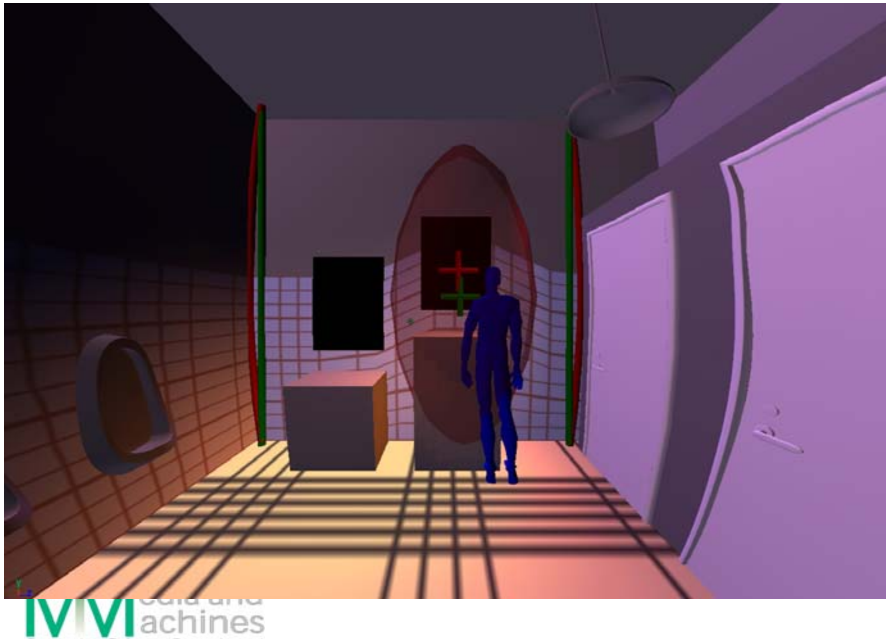
Flow

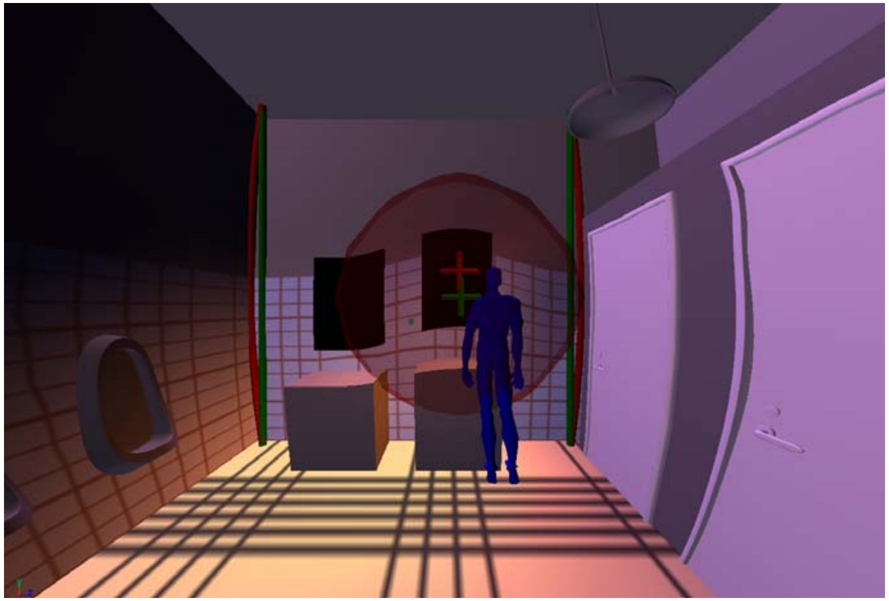
- User picks default view (may pick more than one)
- Draws geometric proxies
  - Defines 3D and 2D geometry
- User edits 2D proxies
  - System solves for new cameras
  - Displays result



Changing weight of camera

- User can then edit the region of influence of each camera
  - 3D implicit volume



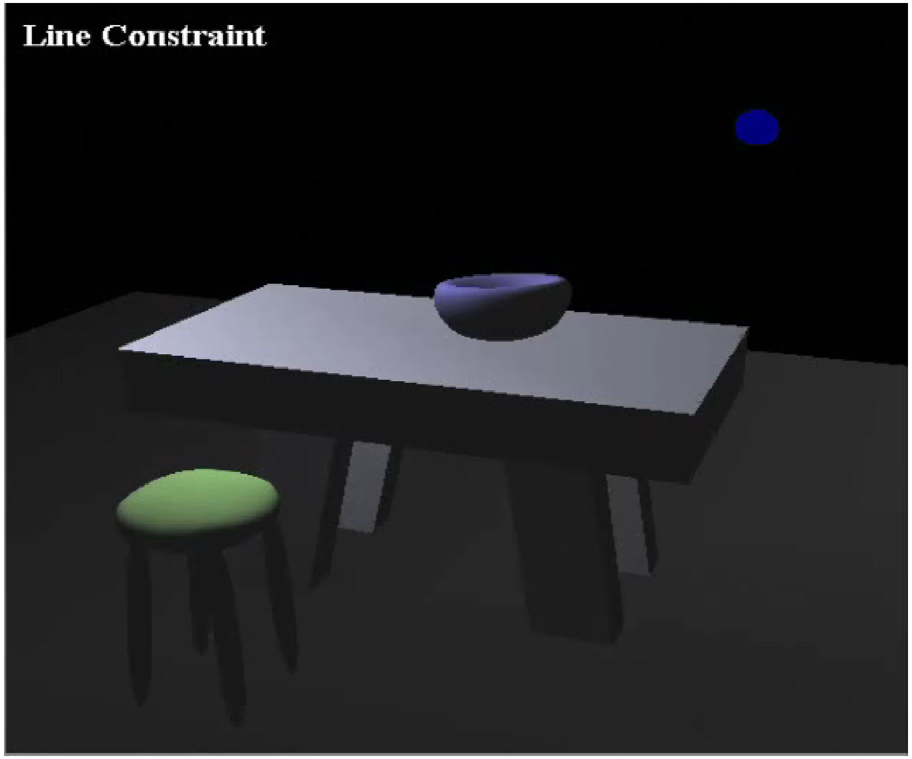


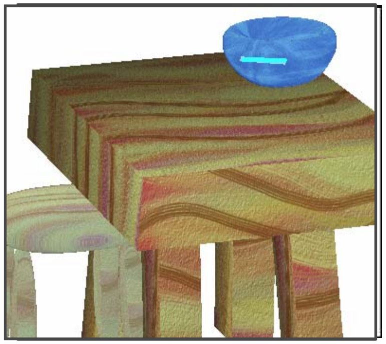
Remainder of talk

- Description of geometric proxies
  - Simple (lines, points)
  - Combined
  - Special purpose (fish-eye, panorama)
- Mechanics of camera solving

Simple proxies

- Point
  - Causes camera pan
- Line
  - Moving causes pan
  - Changing orientation rotates camera
  - Changing size changes zoom

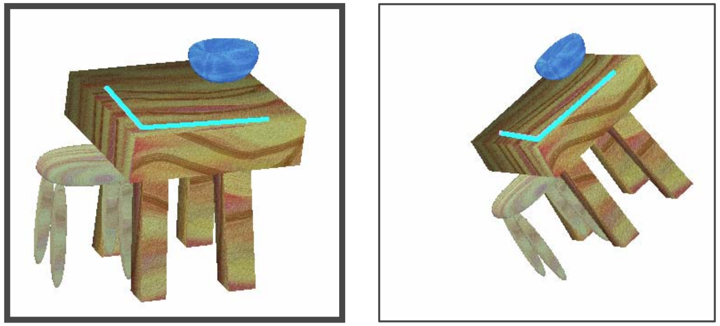




Complex proxies

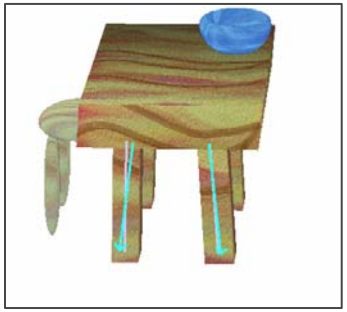
- Wedge (two lines)
  - Position, orientation, size as before
  - Angle changes perspective

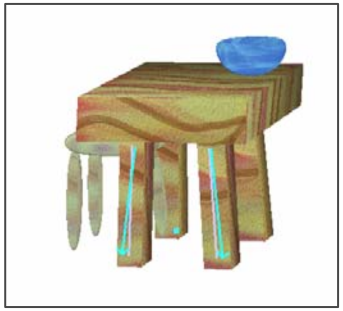
Wedge Constraint

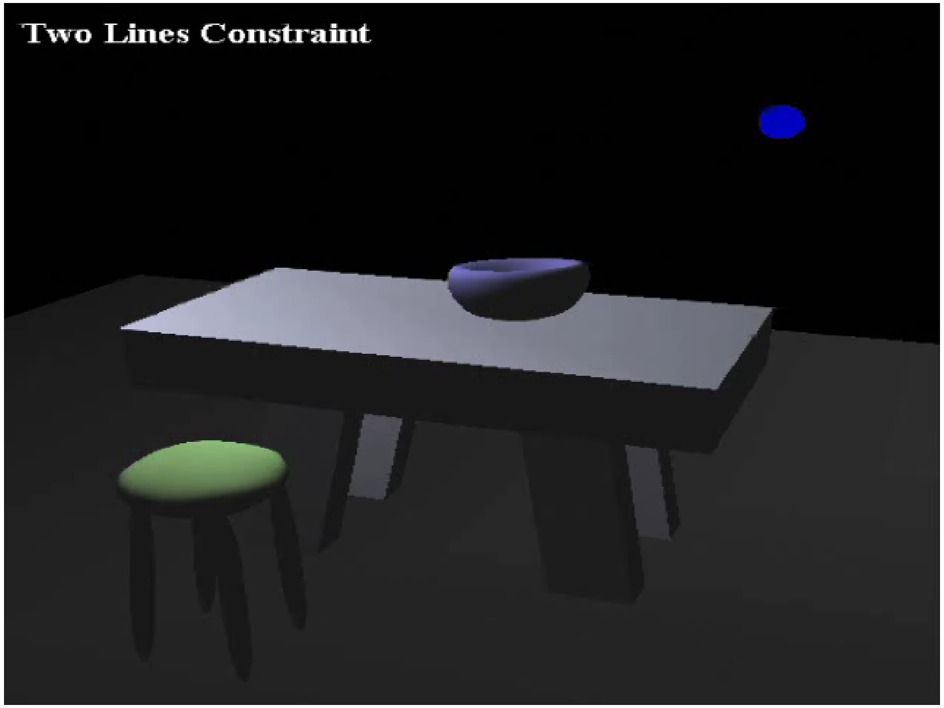


Complex proxies, cont

- Two lines
  - Position, orientation, size as before
  - Changing relative size (rotation)
  - Changing relative angle (perspective)

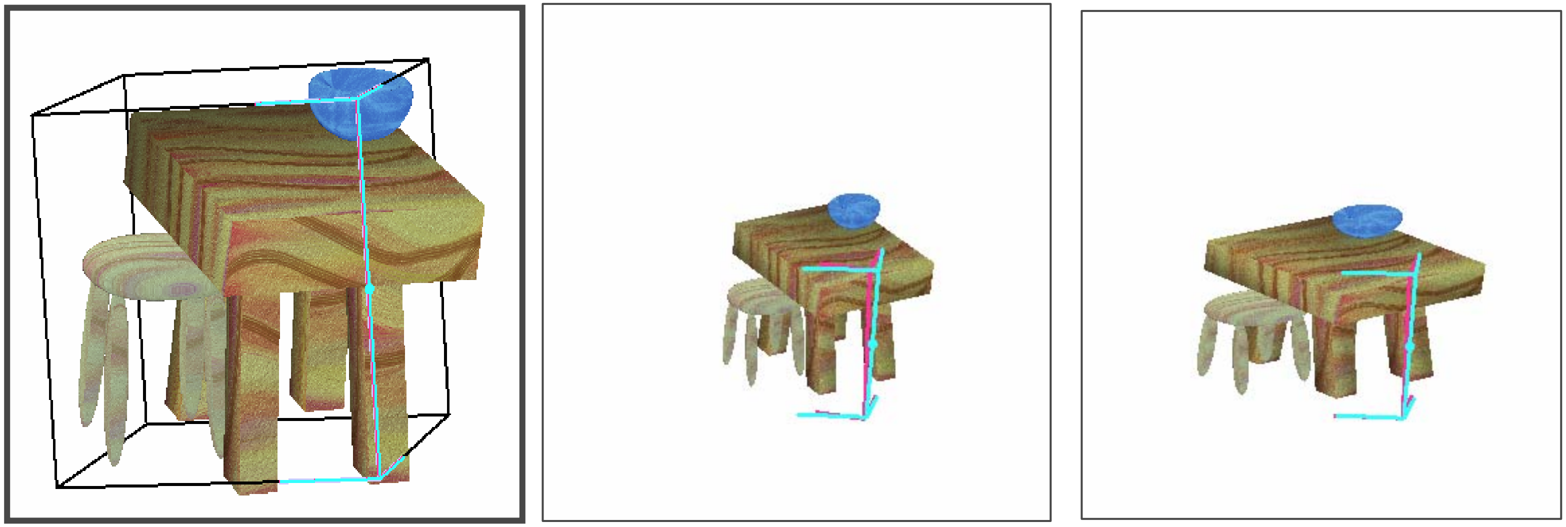






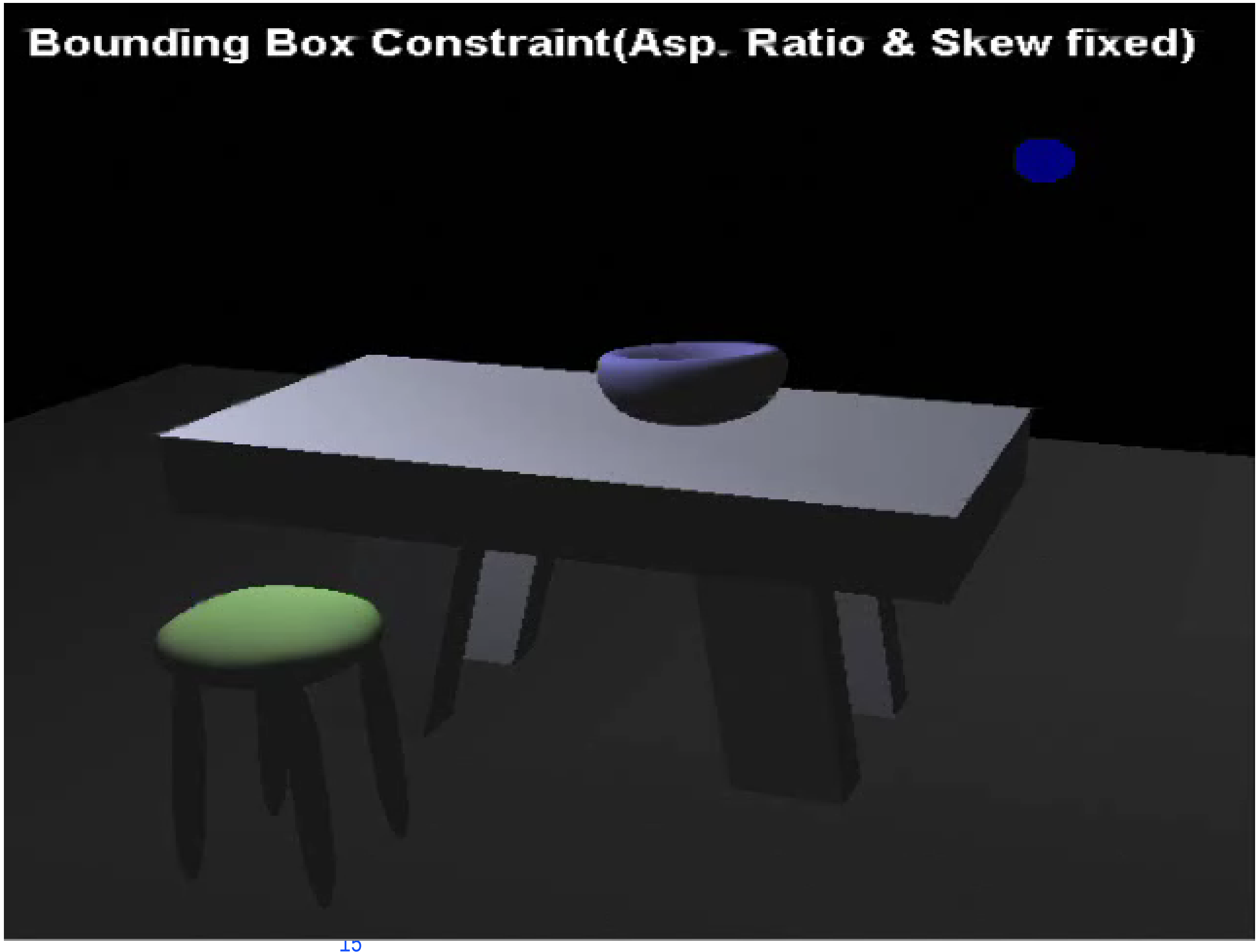
Complex proxies, cont

- Cube edge



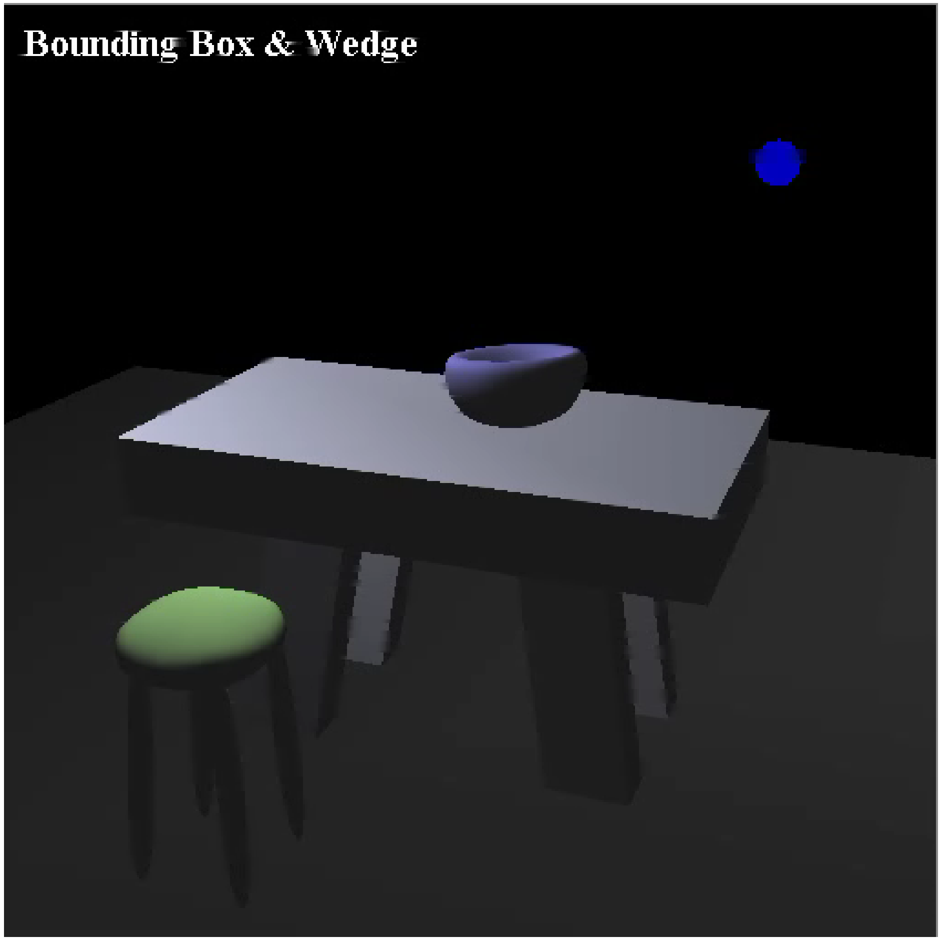
Complex proxies, cont.

- Bounding box
  - Size: zoom
  - Position: pan



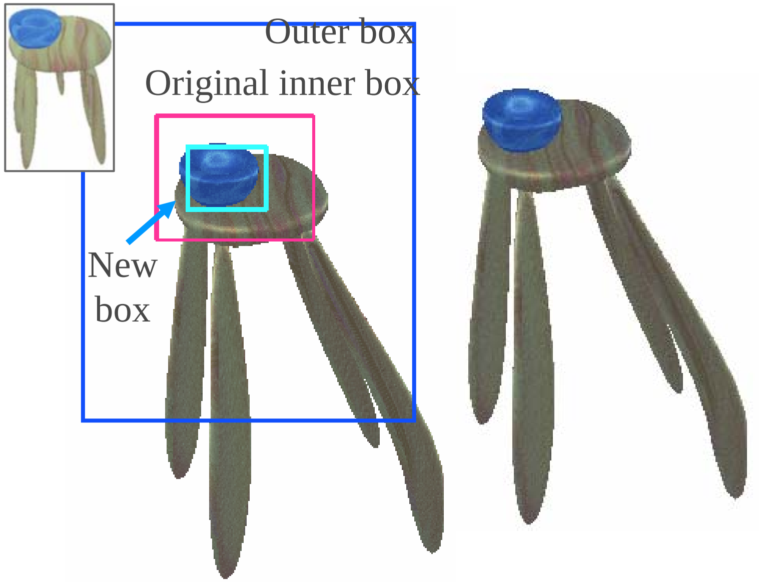
Mixing proxies

- Wedge plus bounding box
  - Wedge controls orientation
  - Bounding box controls size, position
- Wedge and wedge
- Line and wedge
- …
- Still solves for single camera



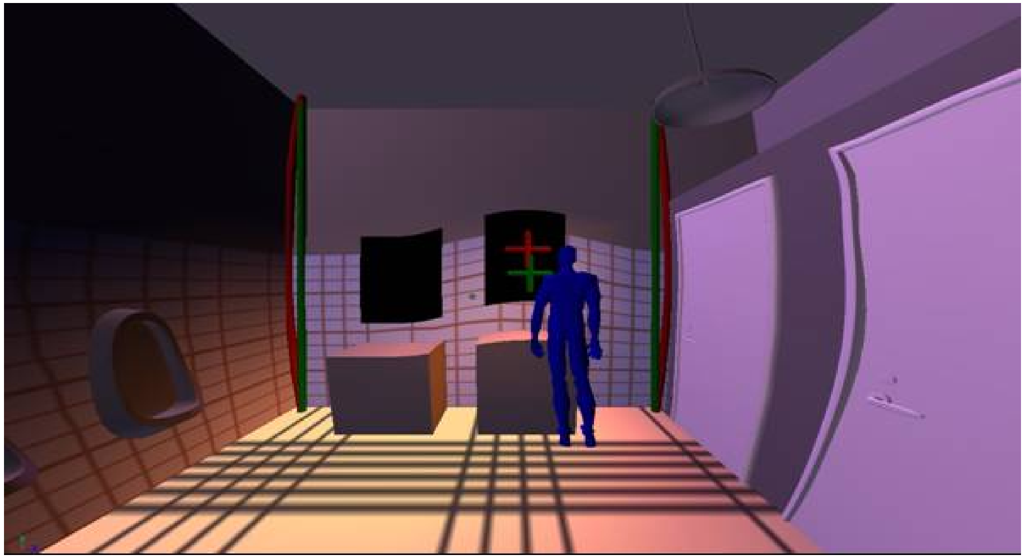
Continuous camera change

- Fish-eye
  - Two boxes, outer controls region of influence
  - Inner controls amount of zoom
  - Zoom smoothly



Continuous camera change

- Line to curve
  - Sequence of position, orientation changes
    - Line segments
  - Project point to line to determine how much to pan



Washington University in St. Louis Media and Machines Computer Science Department

Solver

- Proxy + edit defines allowable camera changes
  - E.g., pan allows only translation in film plane
- Proxy defines error metric
  - E.g., point constraint is distance of projected point from desired image point
- Find camera that minimizes error metric
  - Simplex, or amoeba, solver

Inverse kinematics approach:
Through the Lens Camera Control, Gleicher, Siggraph 1992

Camera degrees of freedom

- Translate in film plane direction
  - Proxy moved in image plane
- Focal length
  - Change in scale
- Translate in/out
  - Proxy changed perspective
- Rotate/spin around look vector
  - Proxy rotated in film plane
- Rotate left/right, up/down
  - Asymmetric change in proxy

User control

- Camera parameters to interpolate
  - Skew, center of projection, aspect ratio
- Importance of matching each geometric proxy
- Region of influence of camera
- Grouping of proxies

Summary

- Non-linear projection difficult to control
- Tool box for specifying camera changes
  - Image-based
  - Default view editing
- Proxies also provide natural region-of-influence
- Still cumbersome

Future work

Sketch-based, global widgets

www.fraps.com

www.fraps.com