Zoomable User Interfaces

Chris Gray

cmg@cs.ubc.ca

Prologue

- What / why.
- Space-scale diagrams.
- Examples.

Introduction to ZUIs

- What are they?
- Why would we want to use them?
  - Physics based rather than metaphor based.
  - Reduce information required to process for navigation.
  - Semantic aspects.
  - More intuitive to search for an object in a large data set.
  - Less cognitive overhead during navigation.

Semantic aspects

- We can substitute representations when we zoom in or out.
  - Some types can’t be represented in the interface.
- Like icons.

Physics Based Metaphor

- Uses physics concepts rather than abstract concepts.
- Folders and the desktop metaphor are “dead metaphors”.
- Zoomable interfaces scale better than the metaphor-type interfaces.

Space-Scale Diagrams

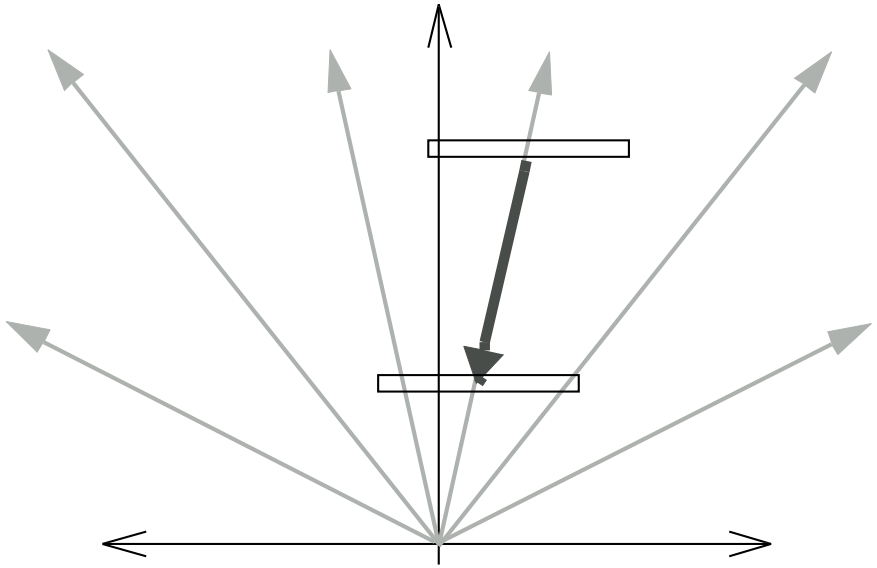
- Formalism to describe different zoomable interfaces.
- Easy method to calculate trajectories between points in a ZUI.
- Can be 2-d or 1-d.

Basic math

- We have (x, z) as the world coordinates and (u, v) as the view coordinates.
- x is the absolute position in the world, u is the position in the window.
- Both z and v represent the magnification.
- Formal relationship is $u = xz$ and $v = z$ (or $v = \log(z)$).

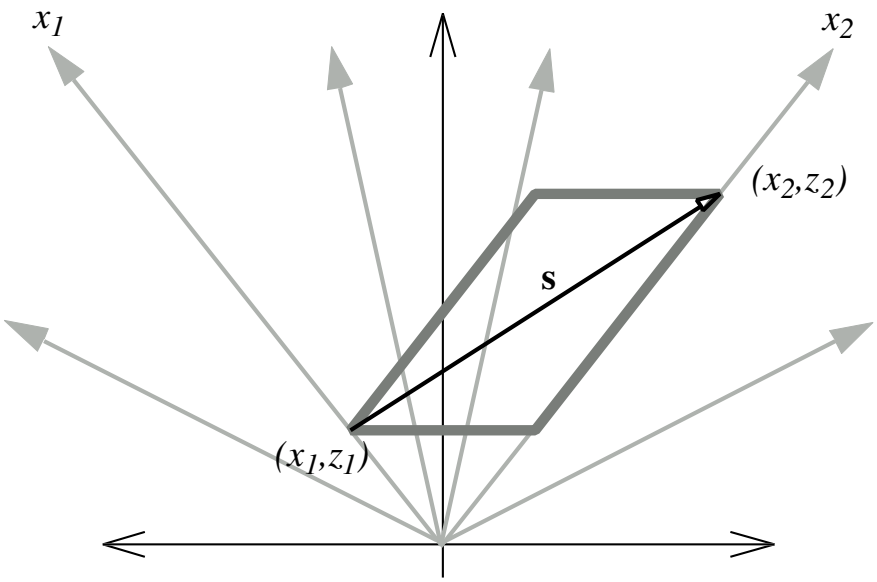
Zoom

- The most obvious application.
- Trajectory simply in the v direction along a ray from $(0, 0)$.

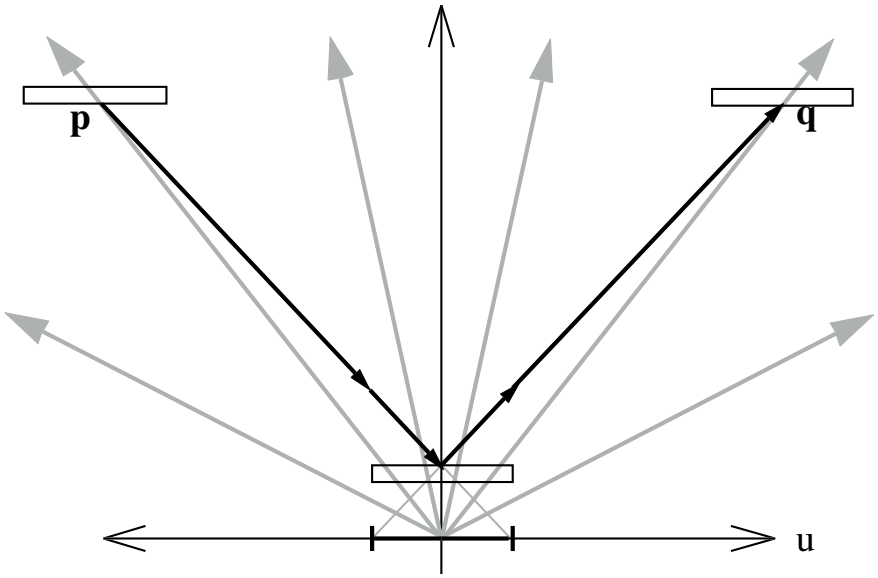


Joint pan and zoom

- For example, moving viewing window from California to Chicago.
- Chicago moving away from window exponentially fast as zoom occurs.



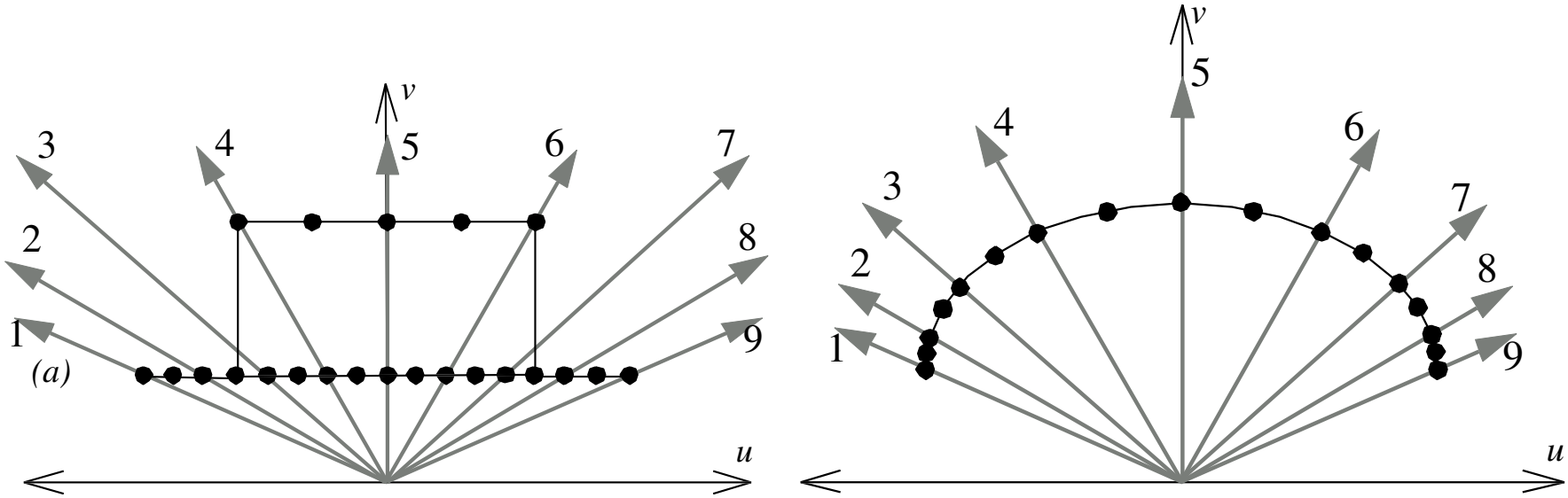
Zoom around a window



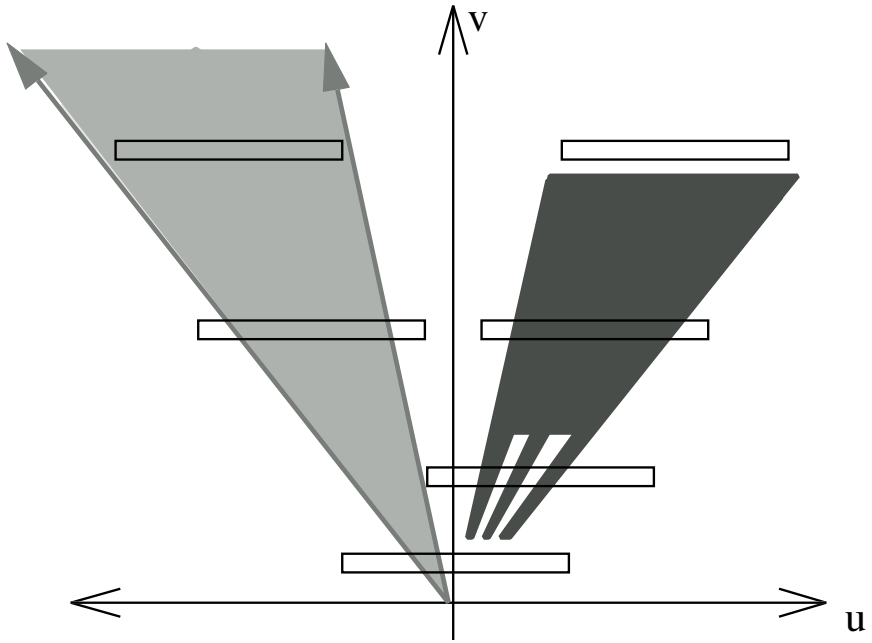
- Pan and zoom out from the old point and then back in to the new point.

Non-traditional zooms

- "Warps" and fish-eye views



Semantic zooming



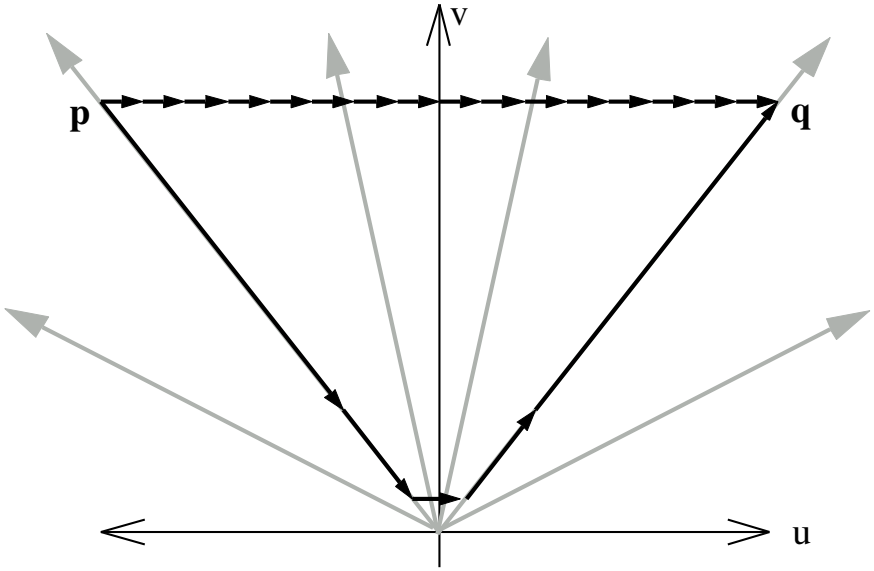
- Representing things changing as they get closer to the camera.

Measuring cognitive overhead

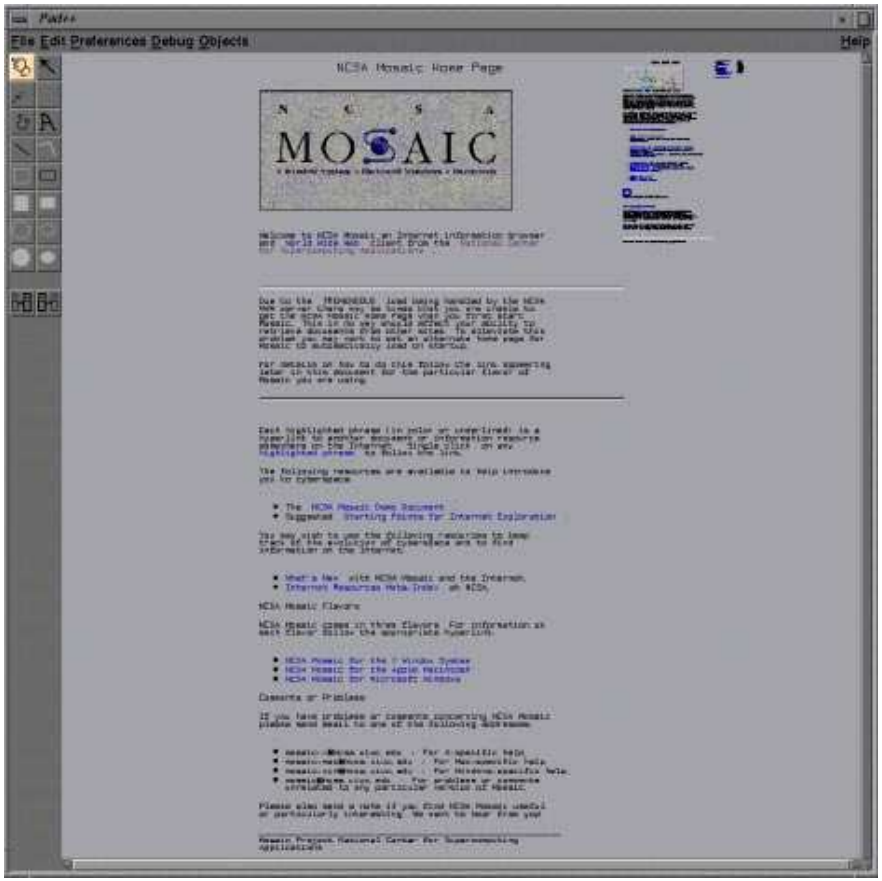
- Want a “movie which compresses well”.
- Scaling is a logarithmic activity, whereas panning must be linear.
- Combining a zoom and a pan can be simpler for the user to understand than just panning.

cont'd

- Space-scale diagrams can be used to find the path which causes the least overhead for the user.



Pad++



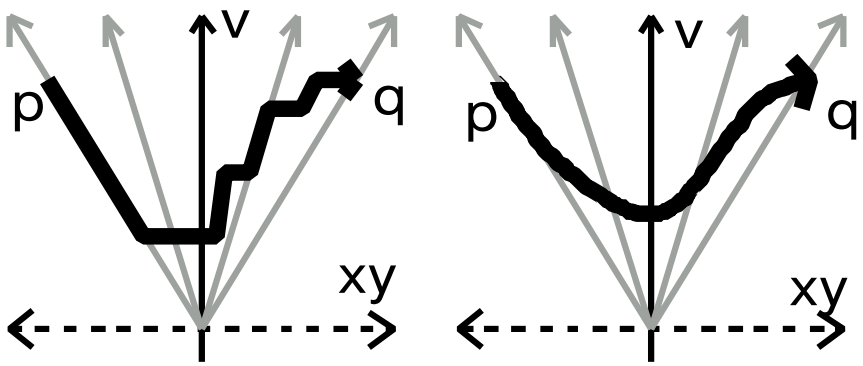
- Example zoomable interface.
  - Library, not actual application.
- Mouse buttons control zoom.
- Used to implement
  - Web browser
  - File browser
- Superseded by Jazz.

Efficiency Considerations

- Only load the part of the database that can be seen.
- Use different levels of detail.
  - If a detail can't be seen, there is no point in attempting to display it.
- Clipping.
- Refinement.
  - Render at low resolution while moving and refine the image when still.

Speed-dependent automatic zooming

- Uses semantic zooming when possible.
- Attempts to maintain a constant information density.
- Can zoom more smoothly than traditional ZUI.



Tested applications

- Web browser.
  - Preferred over traditional.
  - Can reduce some areas of documents more than others.
- Image viewer
  - Can only reduce the size of the images.

Things not Mentioned

(but which should be)

- Clustering.
  - Similar objects should be grouped together so that zooming in shows the similarities.
  - Treemaps.
- PDAs

Conclusions

- Zooming as a user interface tool is a useful idea if nothing else.
- Automatic zooming can be better than manual in some situations.
- Space-scale diagrams can be a good way to design zoomable interfaces.
- You will find true love on Flag Day.

References

[1] Bederson, B., Hollan, J., “Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics”, Proc UIST, 1994.

[2] Furnas, G., Bederson, B., “Space-Scale Diagrams: Understanding Multiscale Interfaces”, Proc SIGCHI, 1995.

[3] Igarashi, T., Hinckley, K., “Speed-Dependent Automatic Zooming for Browsing Large Documents”, Proc. UIST 2000.