Spherical Mirror: A new approach to projection for immersive environments

Paul Bourke
Astrophysics and Supercomputing
Swinburne University, Melbourne

Introduction / motivation

- Visualisation of inherently spherical datasets in astronomy
  - Zoom and pan => lose sense of the whole
  - Map to planar surface => introduces distortion
- Upgrade small planetariums to full dome digital
- Full dome digital projection for inflatable domes, public outreach activities
- Wide angle projection for immersive gaming and virtual environments
- Museum exhibits requiring wide angle projection or dynamic ambient lighting

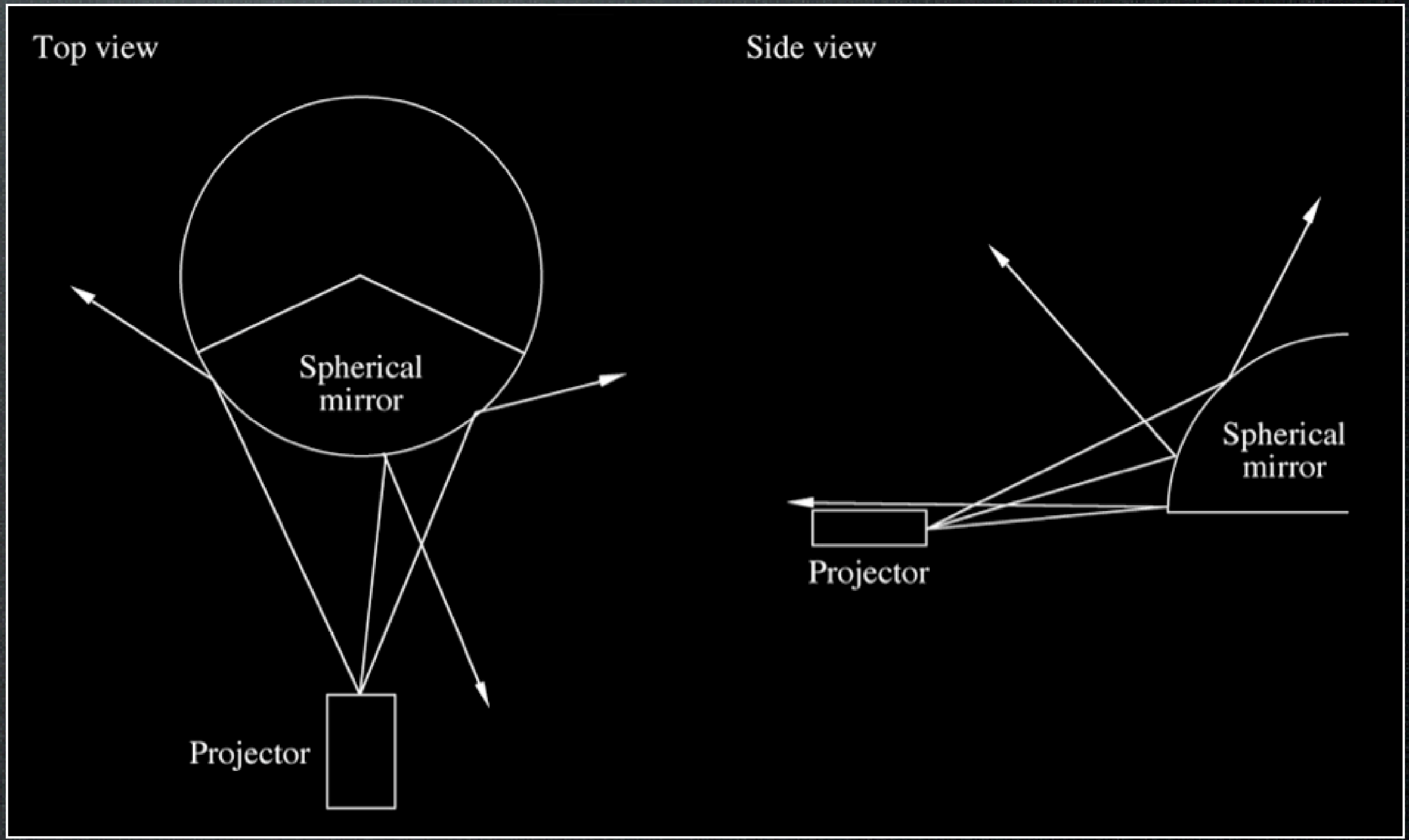
Background

- Traditional approach to full dome and surround projection
  - Multiple projectors (CRT or more recently DLP)
  - Fisheye lens and projector (single or twin)

- Issues
  - Single projector without fisheye has maximum throw of 0.8:1
  - Edge blending of tiled images
  - Portability for inflatable domes
  - Space limitations in smaller planetariums
  - Software complexity of multiple computer/projectors
  - Hardware cost prohibitive for many applications
  - Cost of ownership can be high
  - Fisheye projectors take up the “best seats in the house”

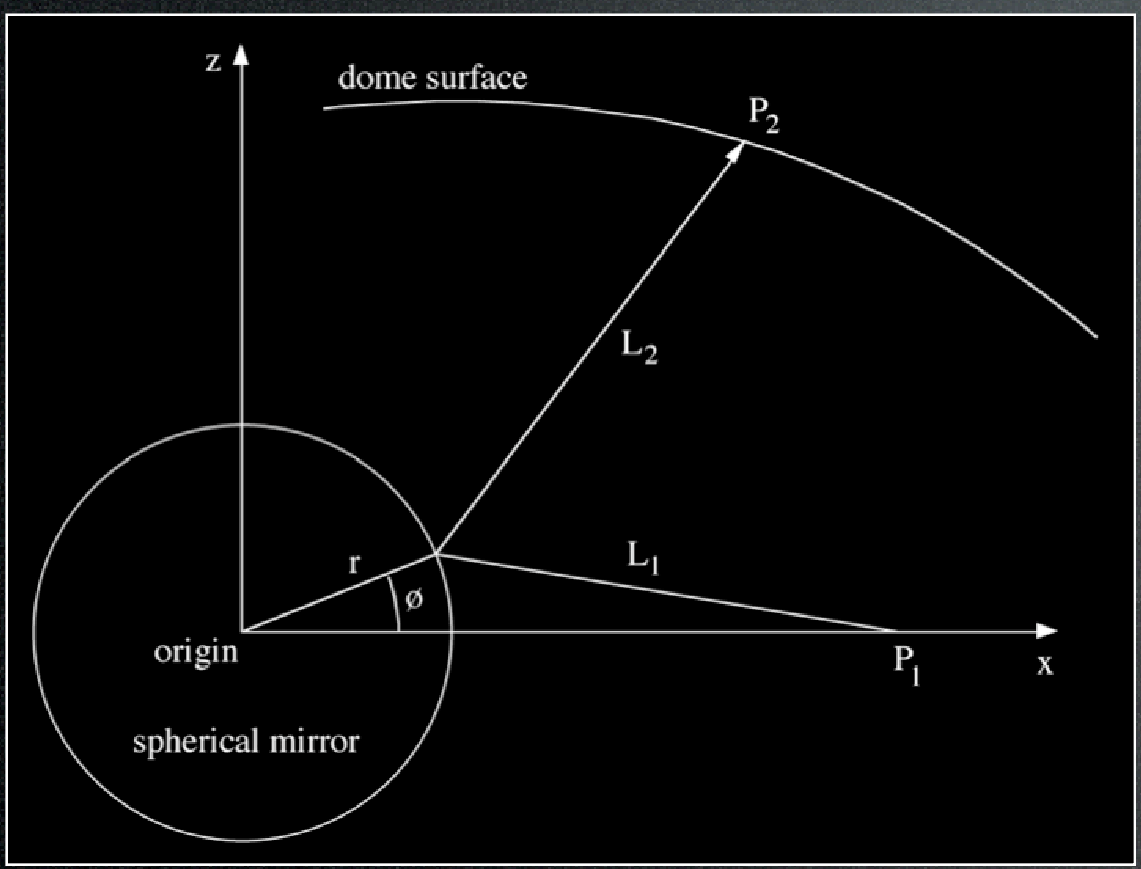
Reflection off a spherical mirror

- Scatter light over a wide angle
- Require enough visual information in projected images: fisheye, spherical panoramic, cylindrical panoramic, cubic maps.
- Distort the image in software so the projected result appears correct, note that the exact warping depends on the geometric arrangement



Geometry correction

- Given a point in the projection plane what is the corresponding point on the dome, and therefore on fisheye image/texture?
- For each point on the dome (fisheye space) what is the corresponding point on the projection plane?
- Can be turned into a 2D problem by rotating the coordinate system so a point on dome, mirror, projector lie in one plane.

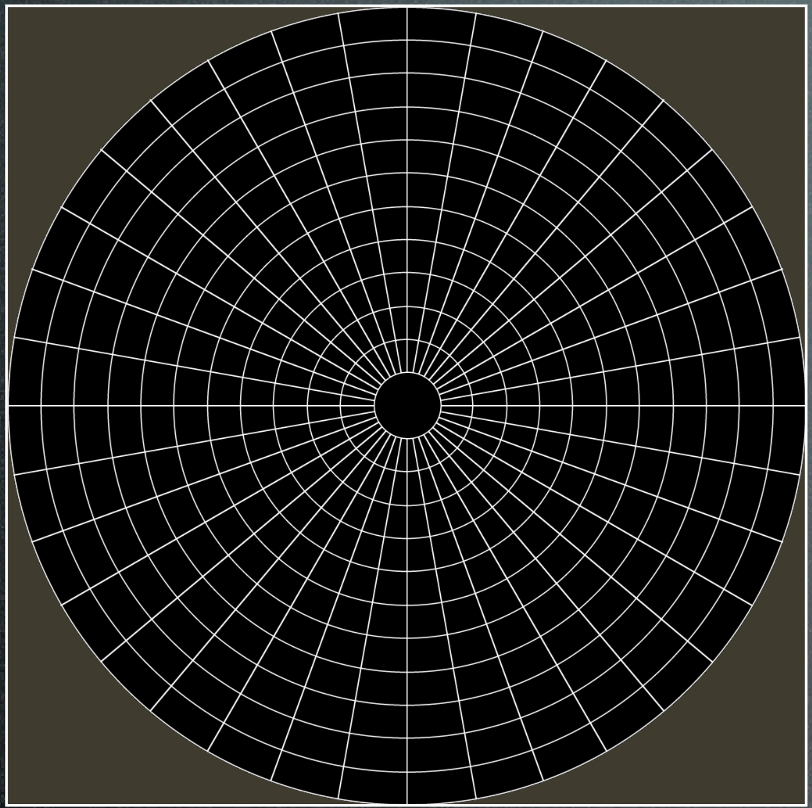


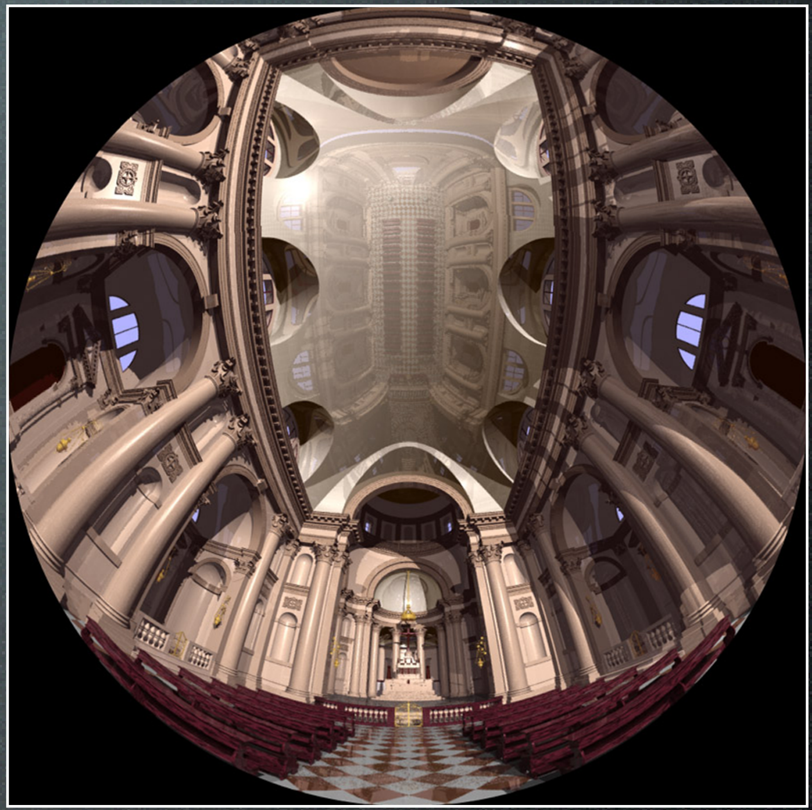
$$L_1^2 = (P_1x - r\cos(ø))^2 + (r\sin(ø))^2$$
$$L_2^2 = (P_2x - r\cos(ø))^2 + (P_2z - r\sin(ø))^2$$

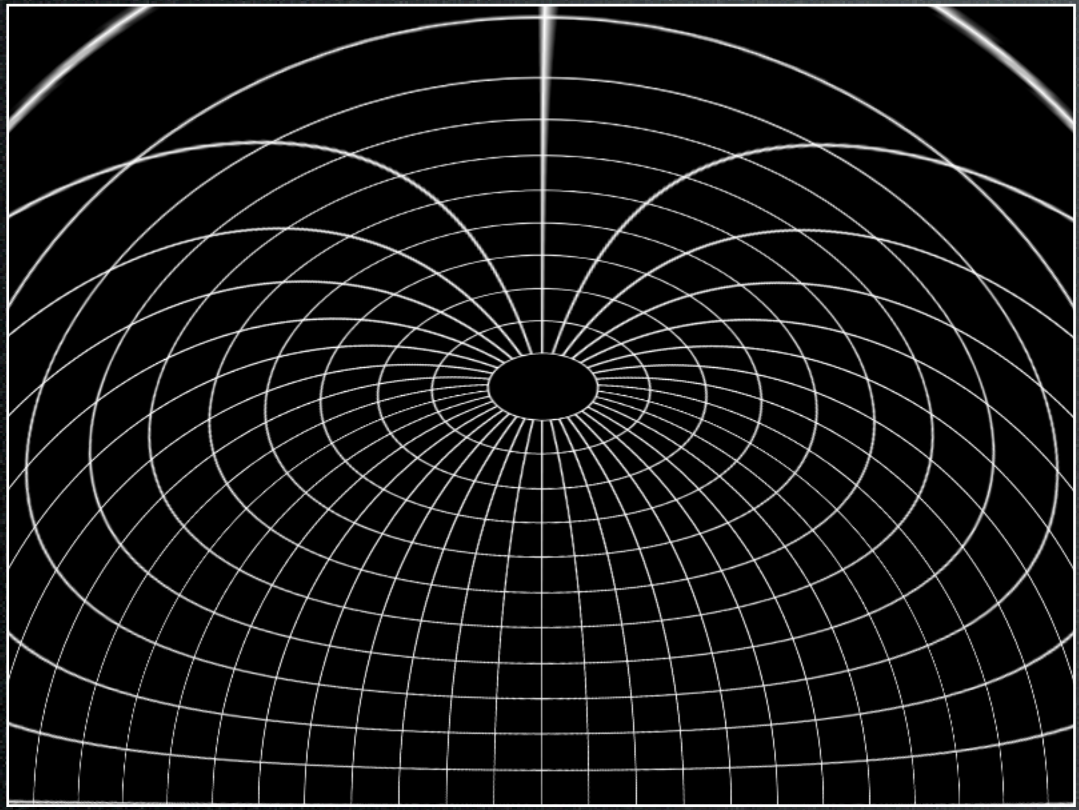
By Fermats principle light travels by shortest route, so solve by minimising

$$(L_1^2 + L_2^2)^{1/2}$$

Warped examples







Software

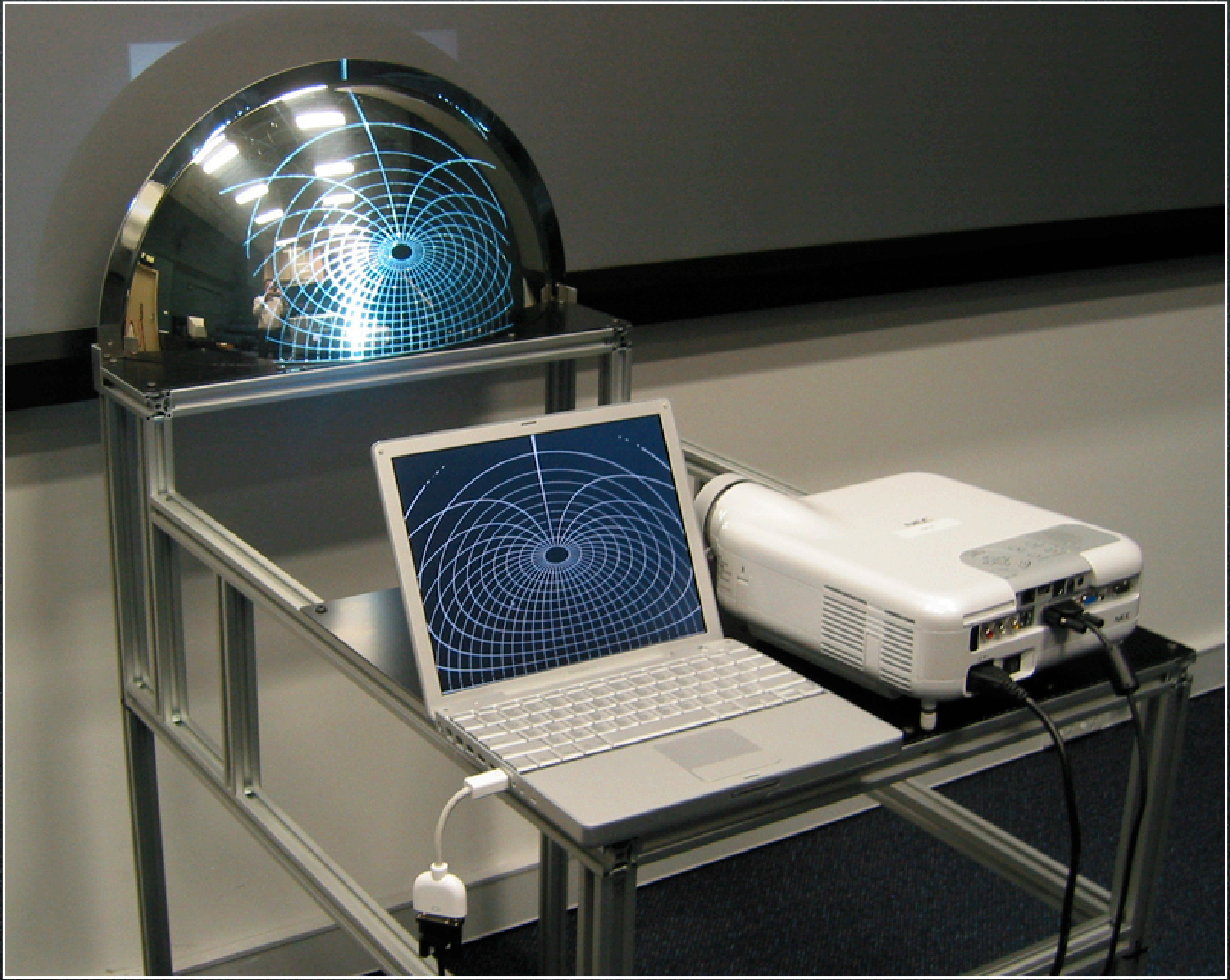
- Precomputed, simply warp movie frames
  - Use supersampling antialiasing
  - Resulting movie is limited to a particular geometry
  - Fixed resolution

- "On-the-fly" warping of movies
  For example: using QuickTime 7 and the Quartz engine
  - Map file for each geometric arrangement
  - Performance issues for higher resolution and laptops

- Interactive applications based upon OpenGL also based upon map files
  - Fisheye warping (eg: stellarium)
  - Geometry warping, requires tessellation of primitives
  - Multipass cubic textures, 4 render passes, 1 texture pass

Projector rigging

- Currently based upon XGA and SXGA+, good depth of focus
- “First surface” chroming of mirror
- Future: optimised mirror shape for a particular environment



Variation: Upright dome

ICinema, UNSW

Variation: rectangular rooms

- Can be adapted for any geometry, warping map derived from ray-casting simulations
- Installations with cylindrical geometry also underway

Summary, current and future projects

- Low cost solution for projecting into immersive spaces [SXGA+ projector resolution <$3.5K]
- Incorporating the warping into game engines for an increased immersive experience (Torque, Unreal, Unity, ...)
- Compares favourably with fisheye for a fraction of the cost
  - Hardware away from the center of environment
  - Choice of projector (resolution/brightness/contrast/....)
  - "Not precious"
- Current projects
  - Improve performance of on-the-fly warping
  - Custom/optimal mirror shapes
  - Planetarium and inflatable installations under negotiation