SIGGRAPH 2003

SAN DIEGO

Course #25

Computer Graphics for Large Scale Immersive Theaters

Spherical Image Processing: Tools and Techniques

Garland Stern

Computer Scientist

Spitz, Inc.

gstern@spitzinc.com

Modeling the theater

- PolyDome – Spitz's modeling tool

- Dome size, shape, tilt
- Projector placement
- 'map' (= source image) placement

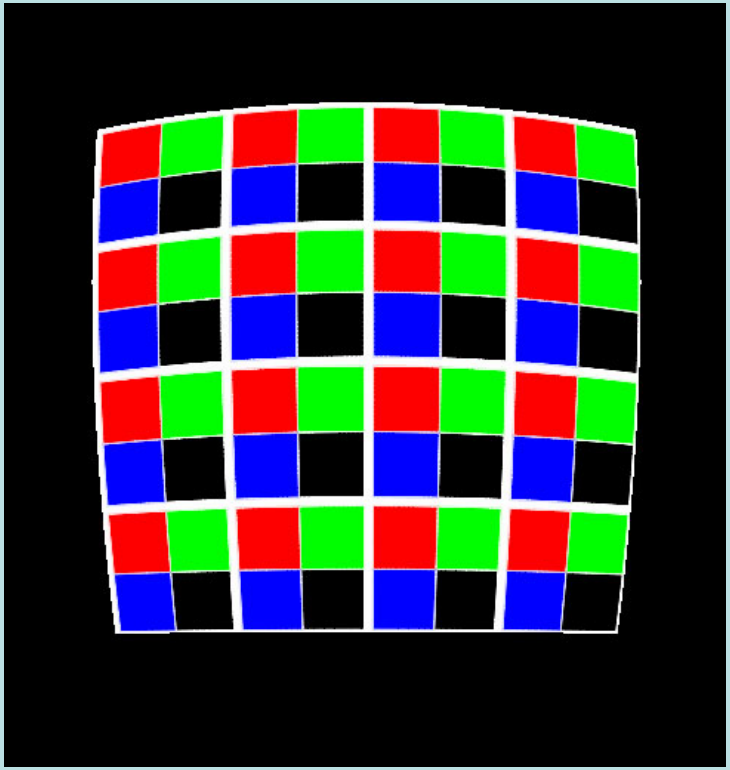
- Projectors and (animation) cameras are 'duals'

Mapping

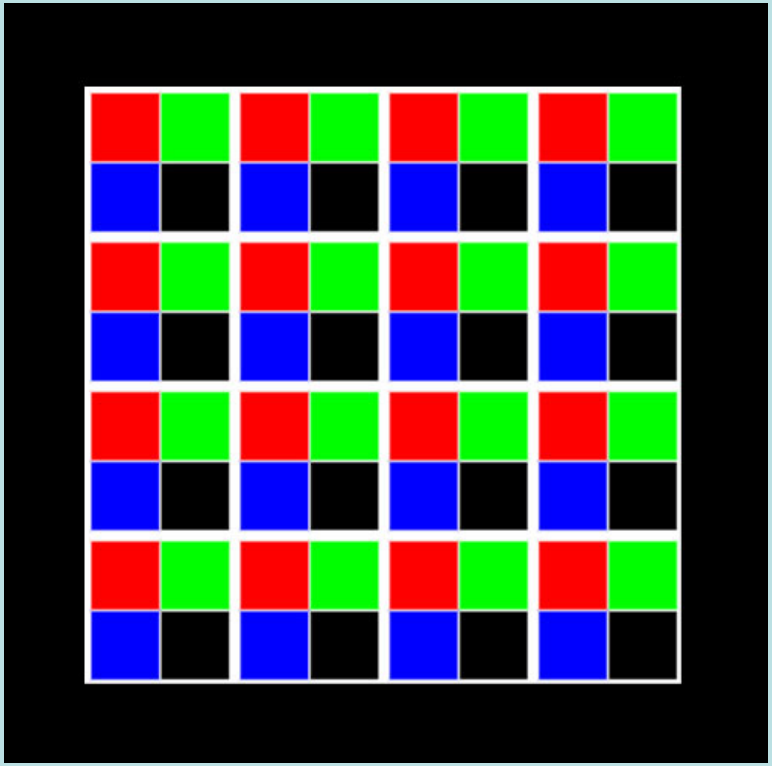
- What part of the map image does a projector see?
- Map pixel + eyepoint = screen point
  - But most maps independent of eyepoint
- Screen point + projector = slide pixel

Billboard mapping

The projector shows:



The viewer sees:

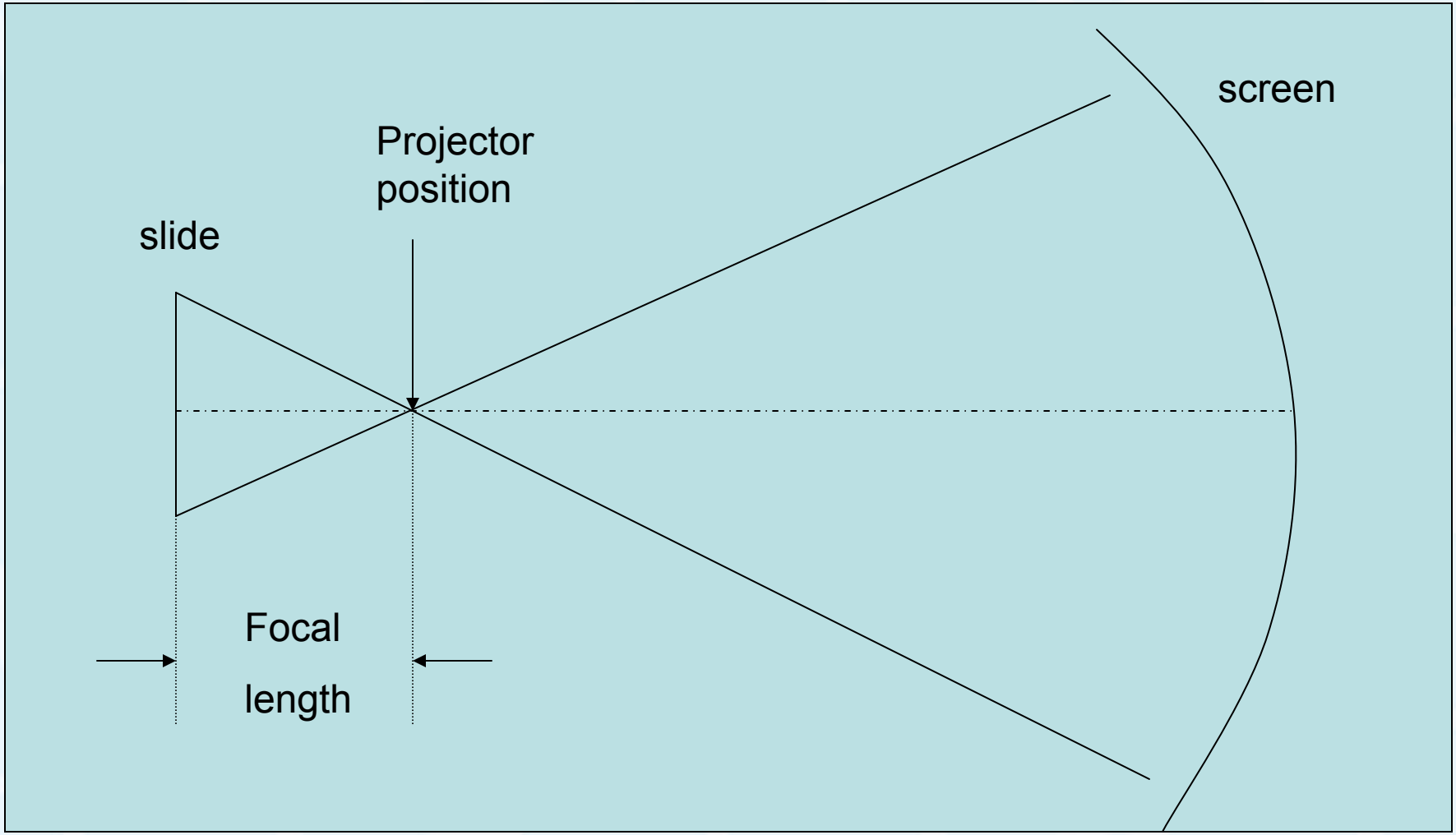


Mapping

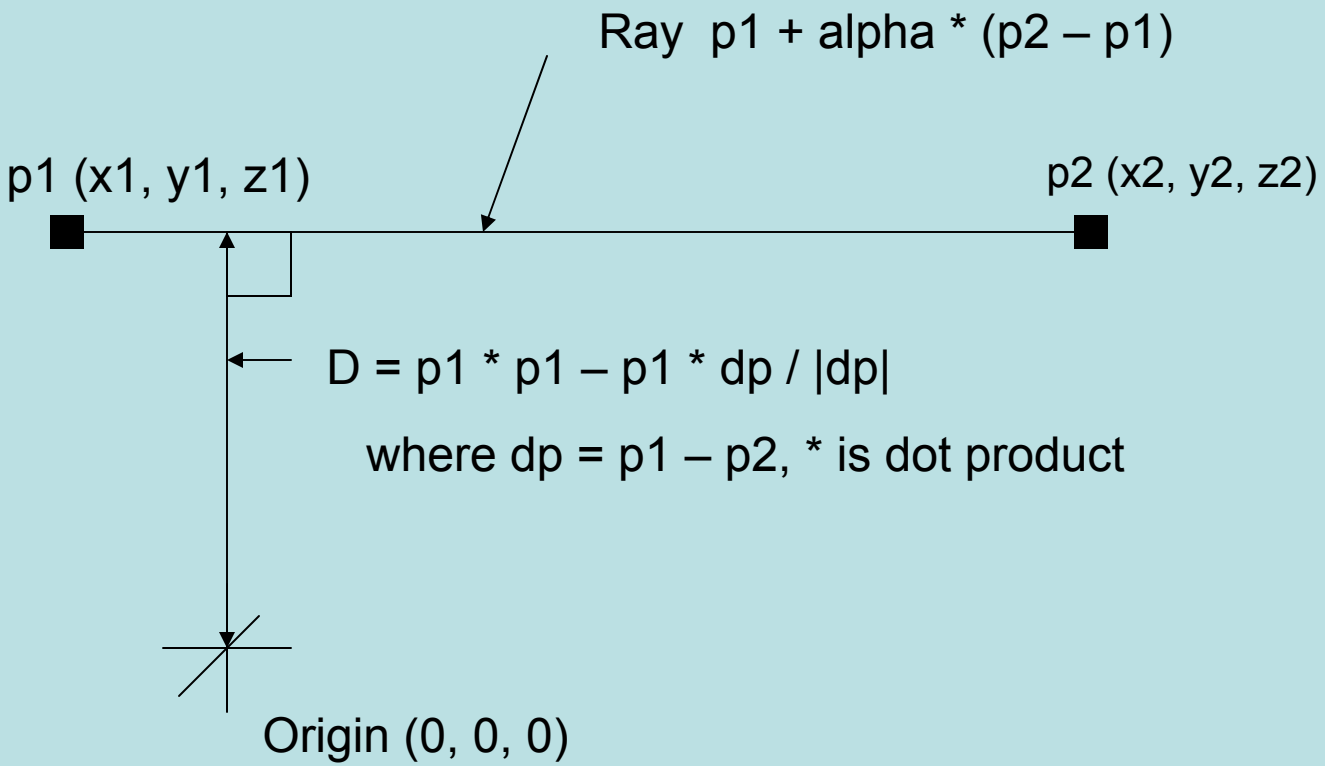
- How do we do it?

- Largely a matter of intersecting lines with a sphere

Pinhole camera model



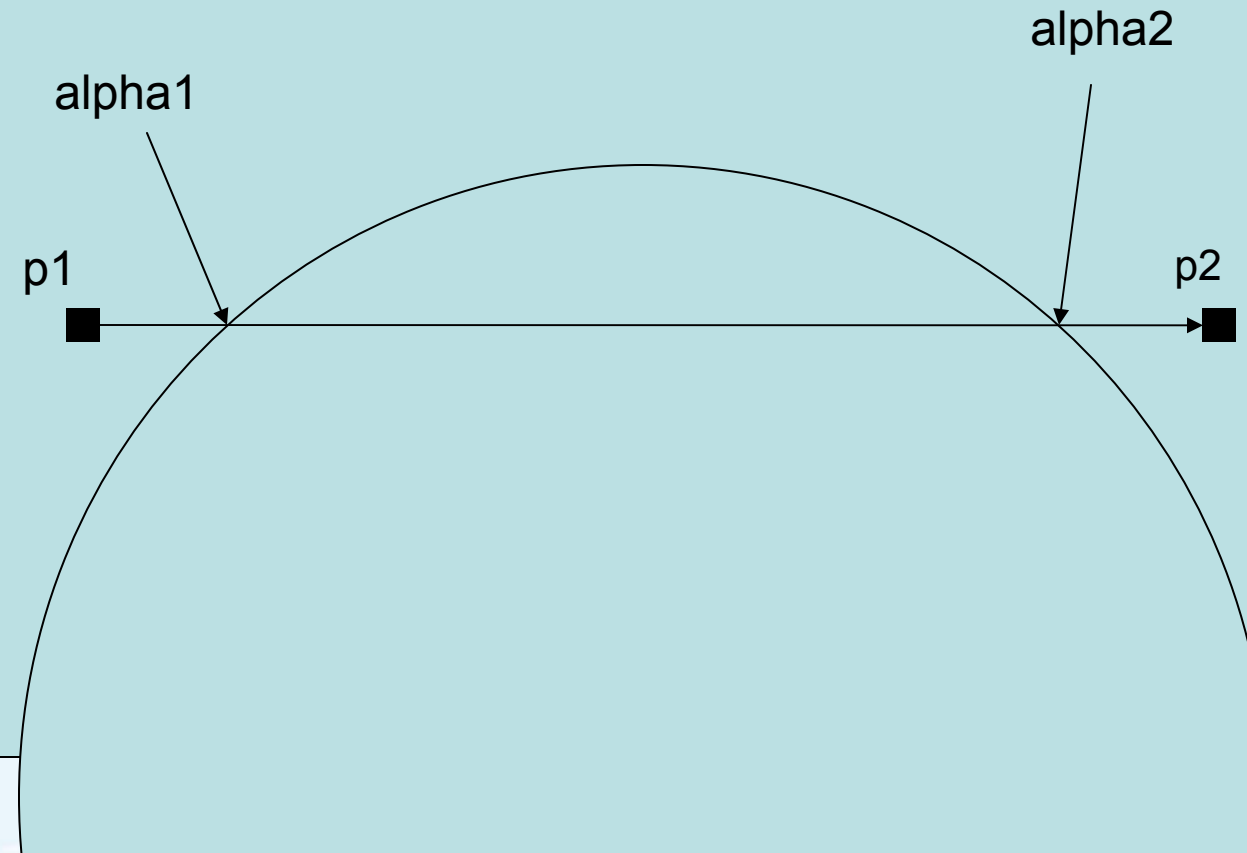
Sphere Intersections



Sphere Intersections

SIGGRAPH 2003
SAN DIEGO

alpha1 ,2 = { +/- (1 – D) – p1 * dp / |dp| } / |dp|



Stitching flat renders

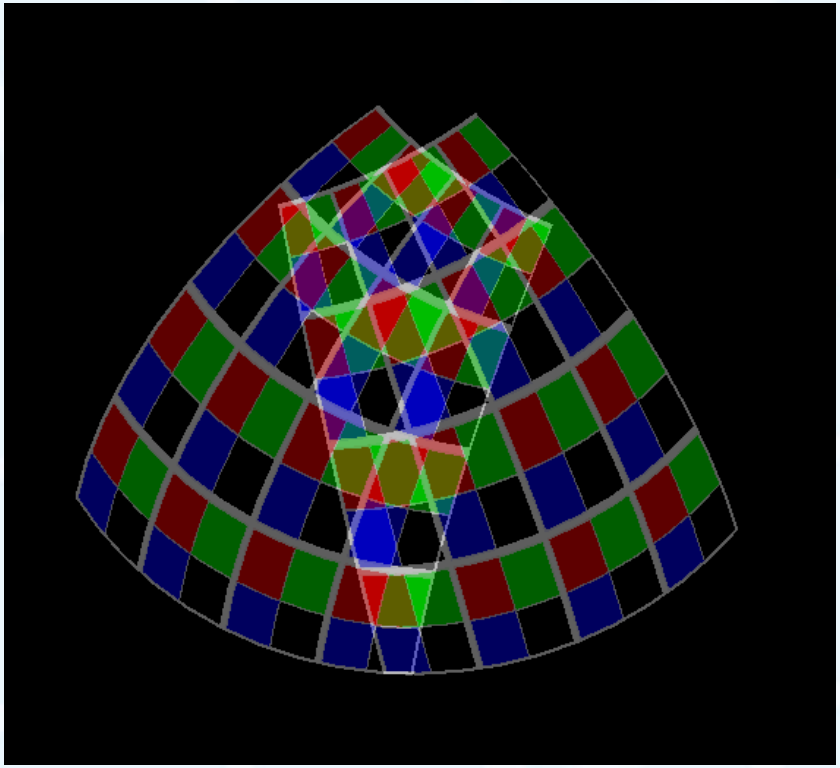
- Need enough ‘billboards’ to cover dome
- Minimize screen overlap to minimize render time
- Similar to half of the classic cubic environment map

glom

- Stand-alone remapping tool
  - Rectangular <-> polar
  - Esky pans
  - Hemi-cube stitching
  - Unsupported executable in course notes

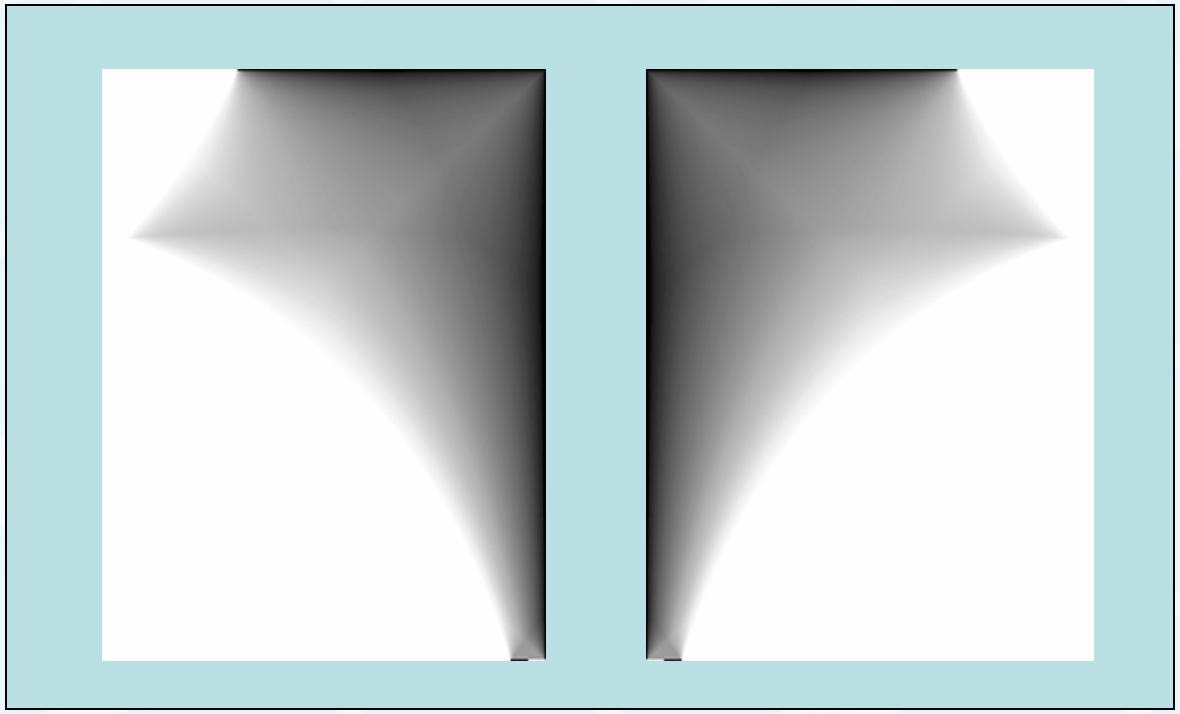
Blending

- In overlap area, image intensity is the sum of contributions from all projectors



Blending

- An under constrained problem
- Want smooth gradations falling off to black at edge of blend region

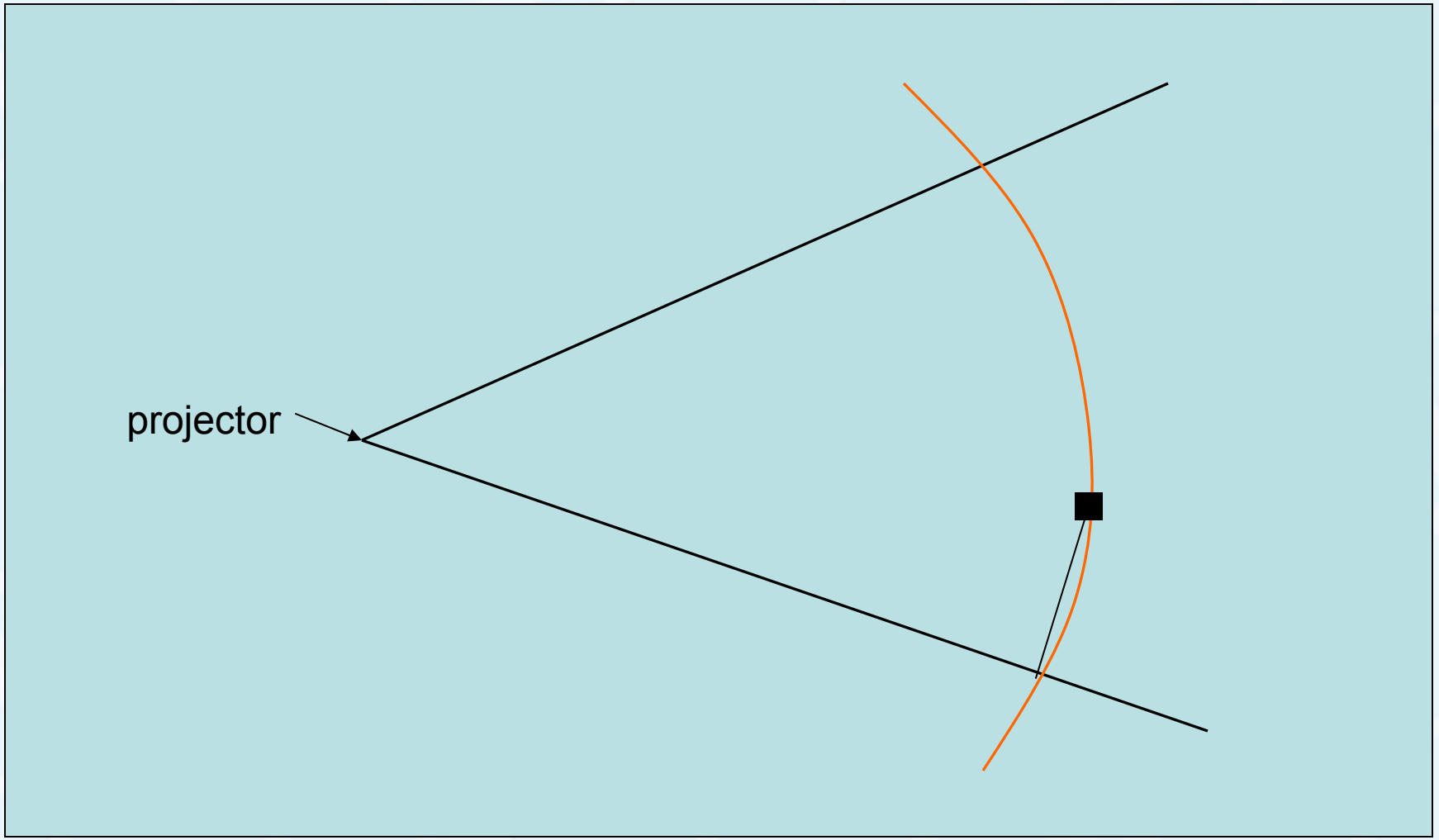


Blending

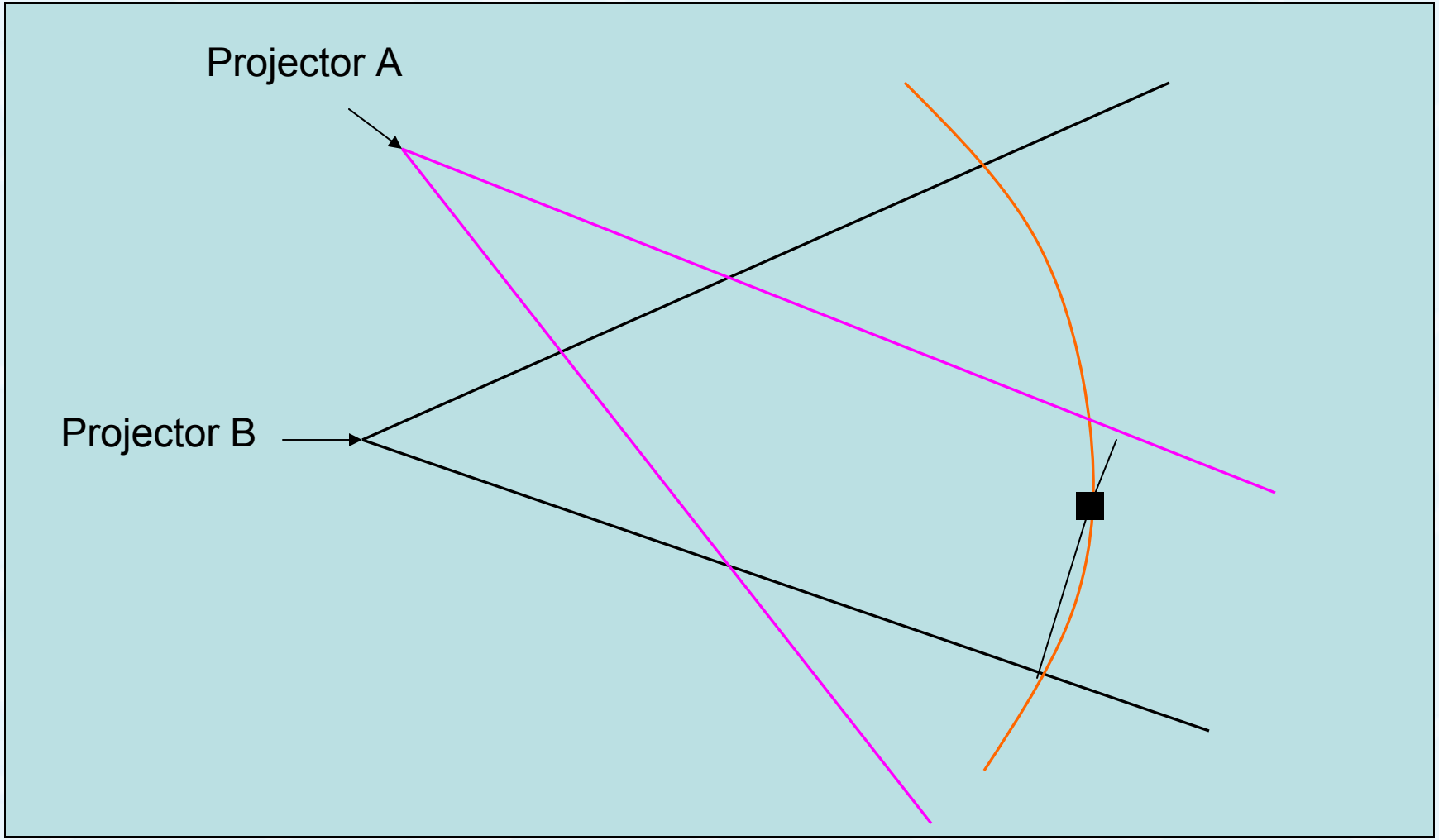
- Where is the ‘edge’ of the overlap region?
- Hard to calculate on sphere surface

- Use distance from point on sphere to nearest plane of view pyramid

Blending



Blending

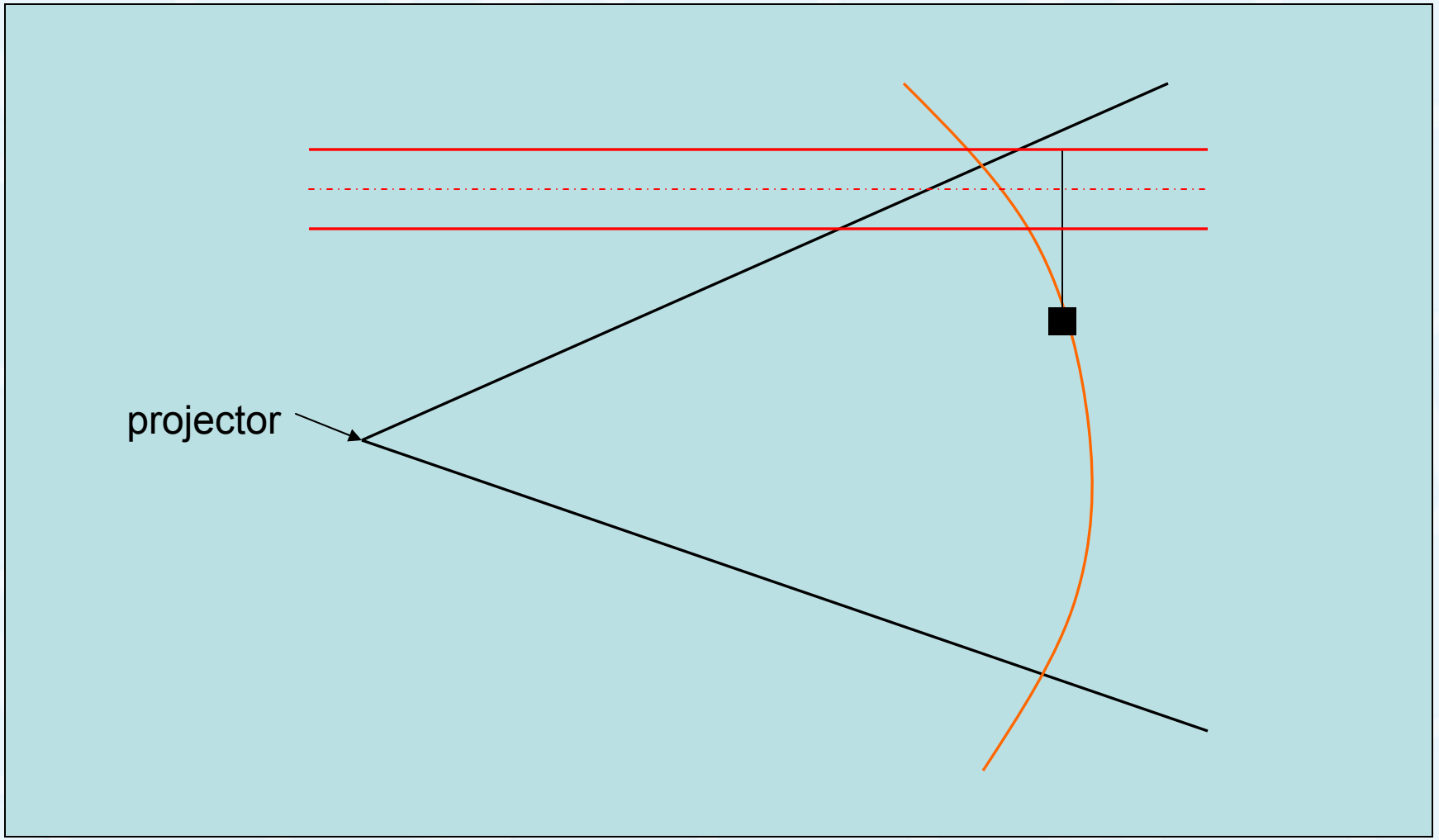


Blending

- Full overlap blends are smooth, but
  - Lens imperfections are worse at corners
  - Geometry errors appear as blurred or double images

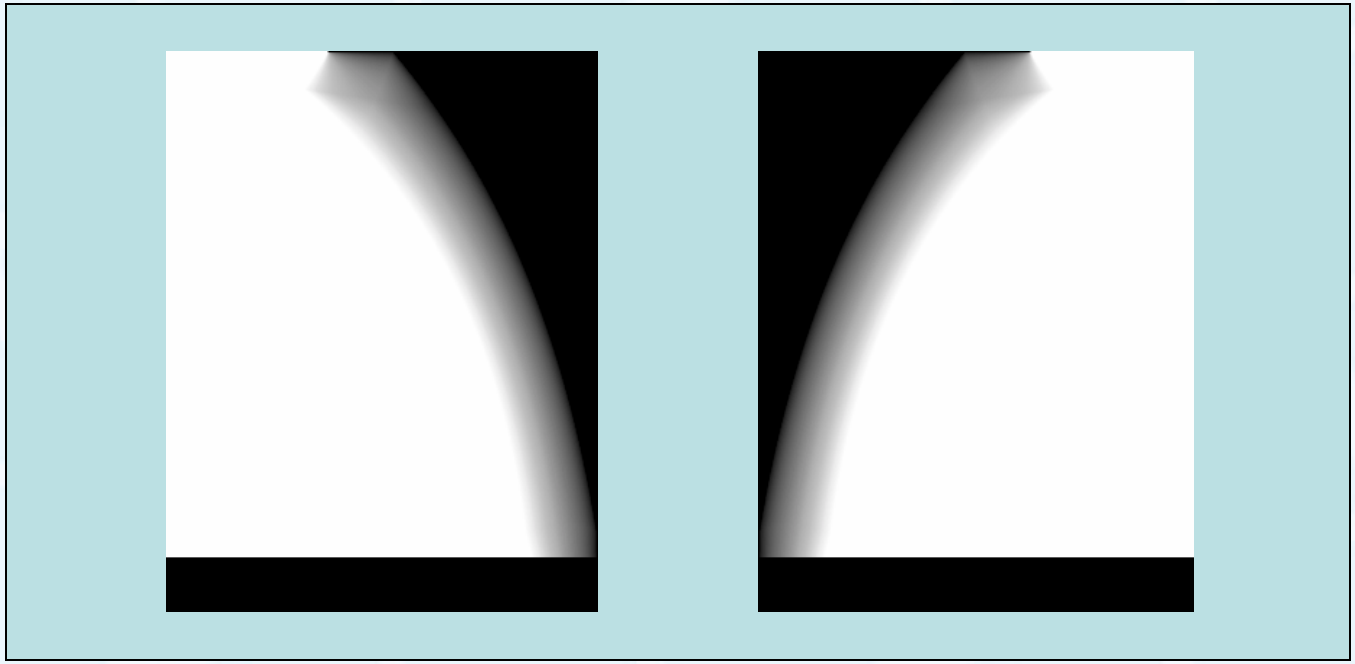
One solution – trim the view pyramid

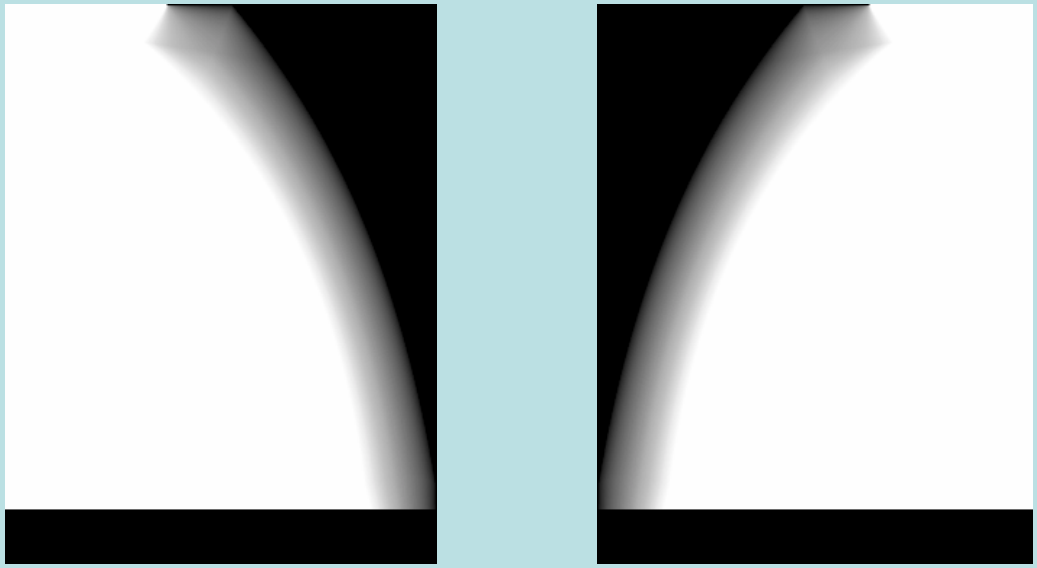
Blending



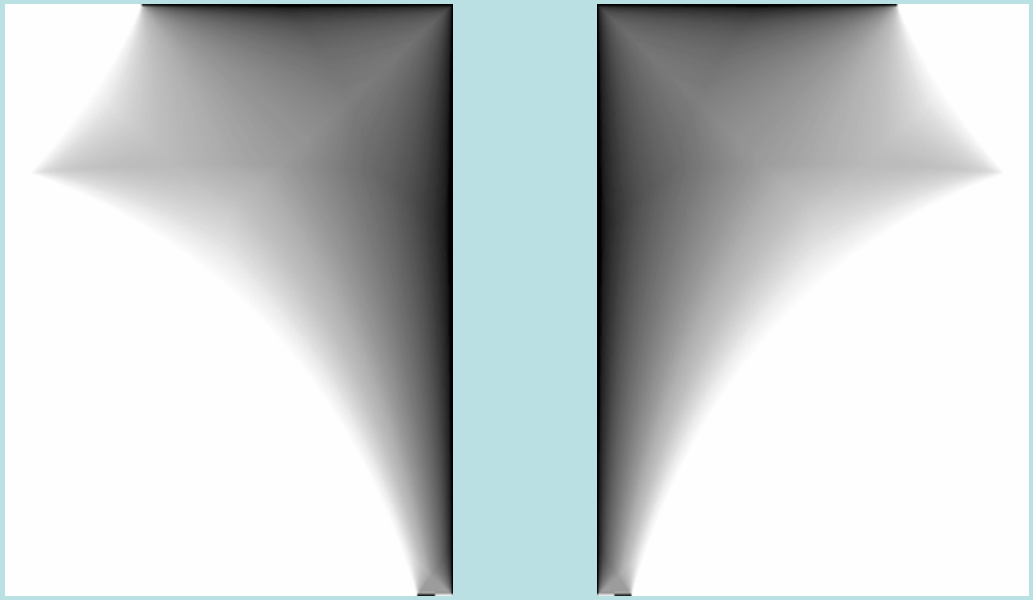
Blending

- Controlling the slab thickness determines the blend width
- Around 5 degrees is good





thin



Full
overlap

Blend width tradeoffs

- Geometry errors show up more over large areas
- Color differences show up more over small areas

Blending

- Now we have for pixels in overlap area:
  - Projector A intensity =
    Projector A intensity * mask A
    + Projector B intensity * mask B
    = Projector B intensity

But is it?

Blending

- Not quite – we have to account for intensity non-linearity in the projector

  Screen intensity =

  f(projector A intensity * mask A)

  + f(projector B intensity * mask B)

  Function f() different for film, video

ViewSphere

- Digital planetarium projector
- Spherical compositing and animation