# Unstructured Lumigraph Rendering

Unstructured Lumigraph Rendering. Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler and Michael Cohen. SIGGRAPH 2001.

Presented by Hong Wu

Discussion led by Dai Bui

# Unstructured Lumigraph Rendering



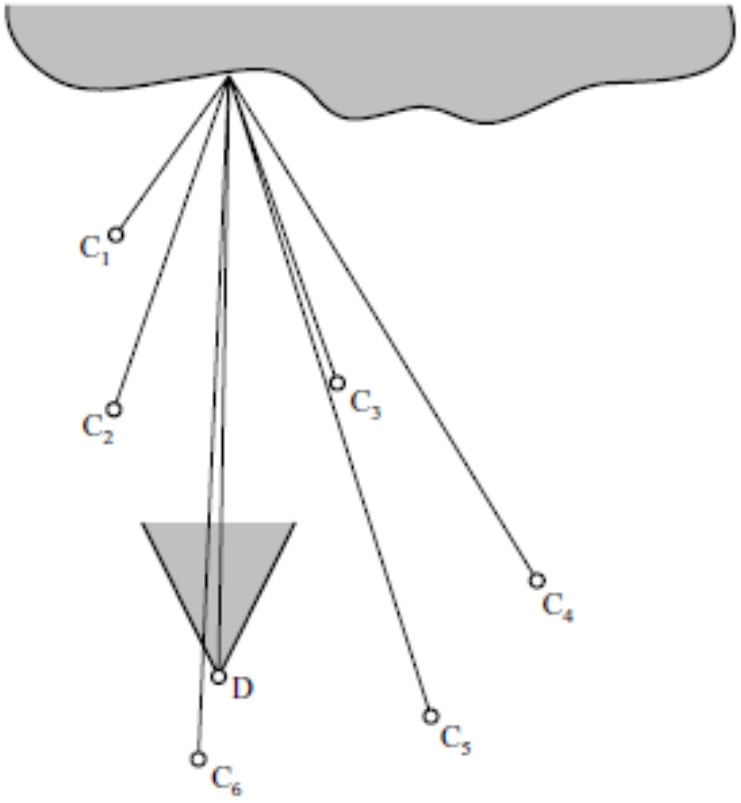
# Goals

- Combine the following desirable properties

1. Use of geometric proxies
2. Epipole consistency
3. Minimal angular deviation/ Continuity
4. Resolution sensitivity
5. Equivalent ray consistency
6. Real-time

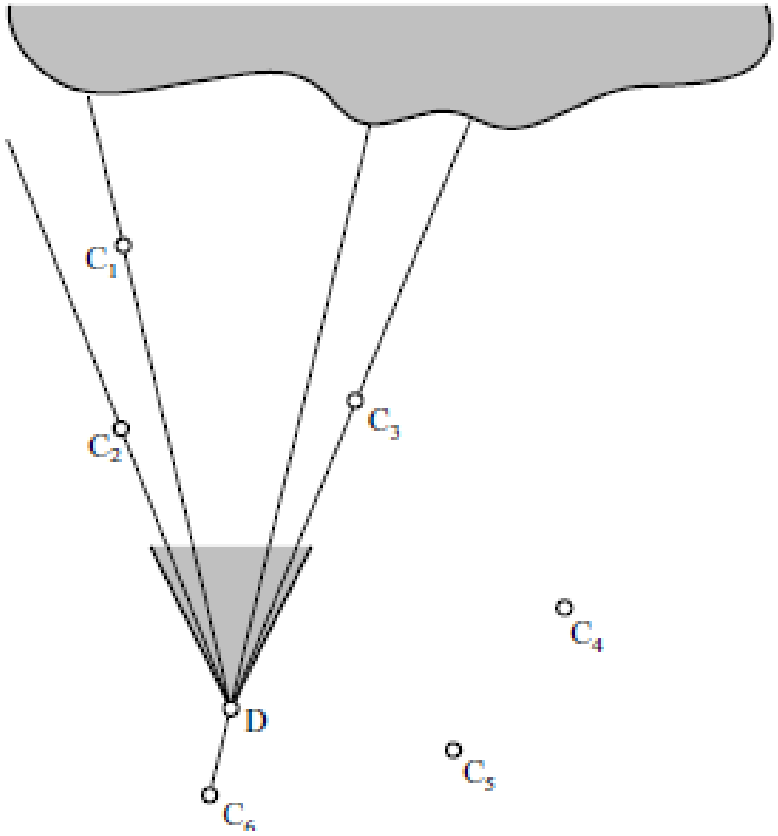
# Geometric Proxies

- **Unstructured input:** Accept input images from cameras in general position. No camera array is required.



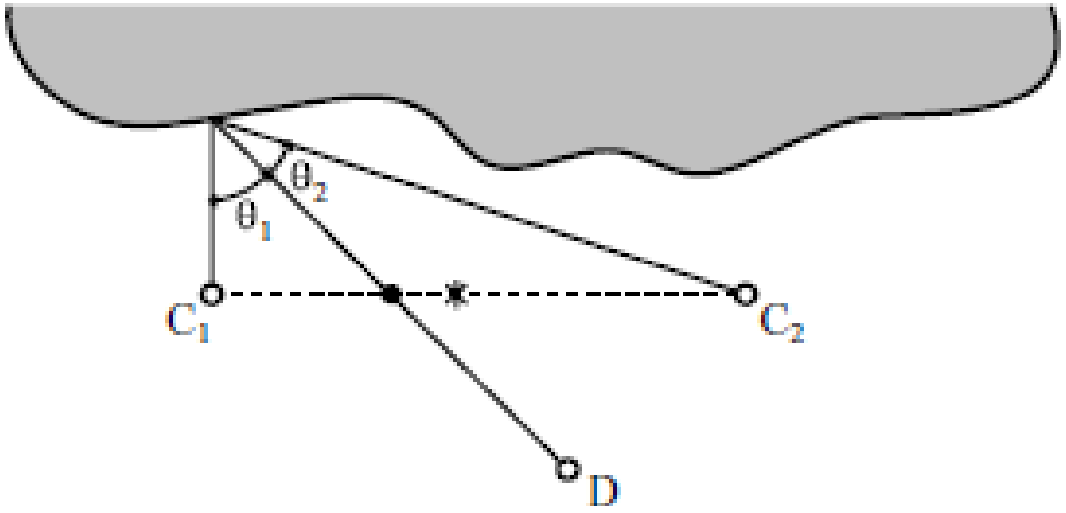
# Epipole Consistency

- A camera, which is in the field of view of the new camera, needs to be emphasized most in reconstruction.
- In this case,  it is C1, C2, C3, and C6.



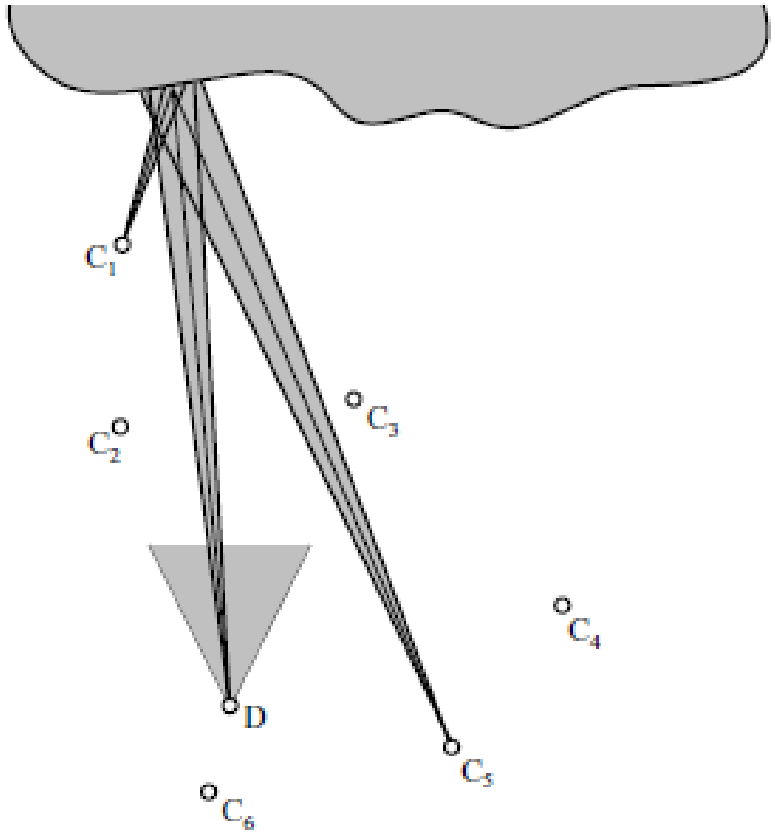
# Minimal Angular Deviation

- What is the most relevant camera?
  - Light fields algorithm
  - Minimal angular deviation/ Continuity



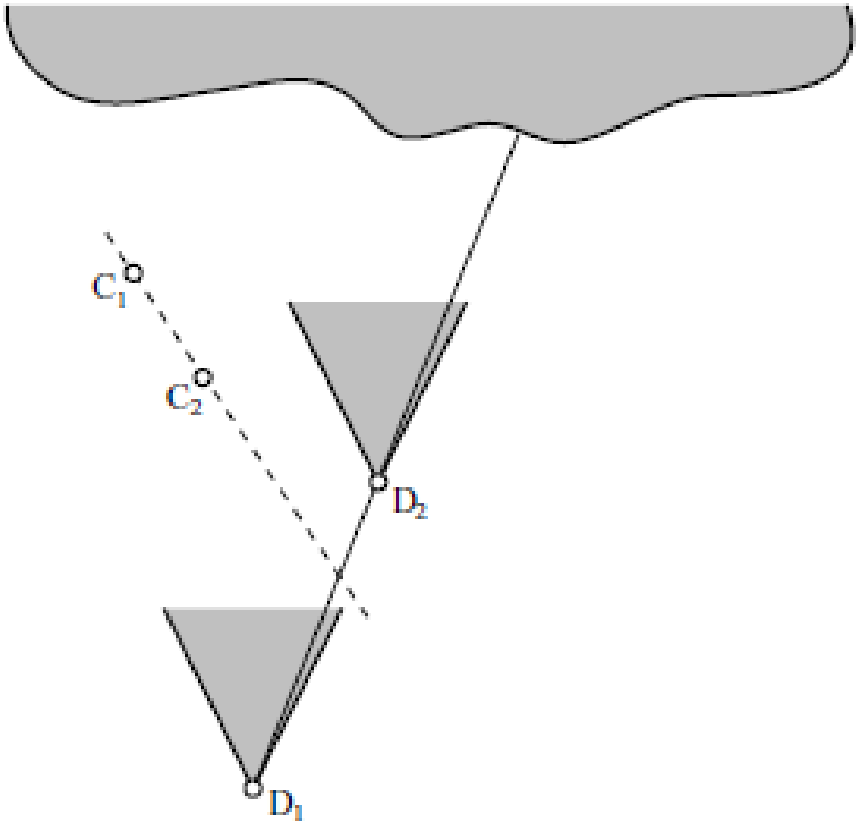
# Resolution Sensitivity

- Resolutions of the same proxy are different related to the distance between the proxy and camera.



# Equivalent Ray Consistency

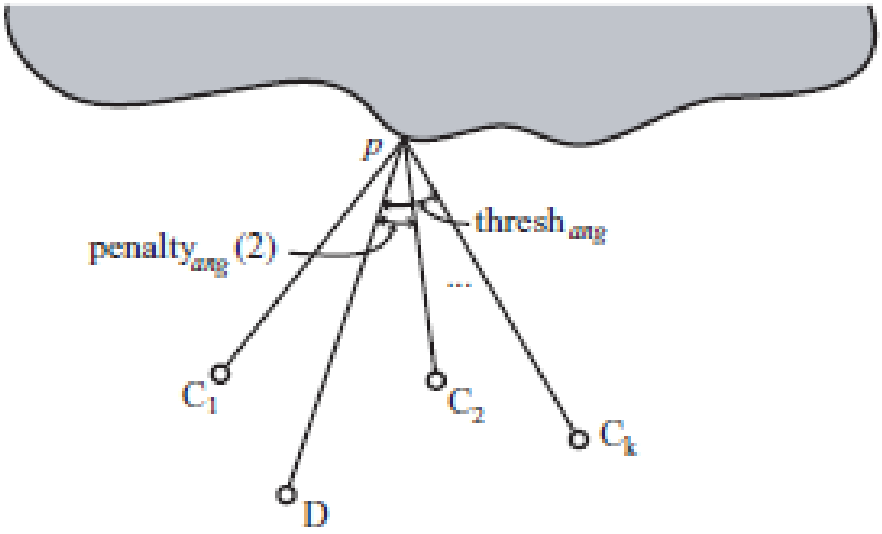
- The reconstruction of the same ray should be consistent.



# Proposed Algorithm

1. Angular penalty
2. Resolution penalty
3. Field-of-View penalty
4. Triangulation and Rendering

# Angular Penalty

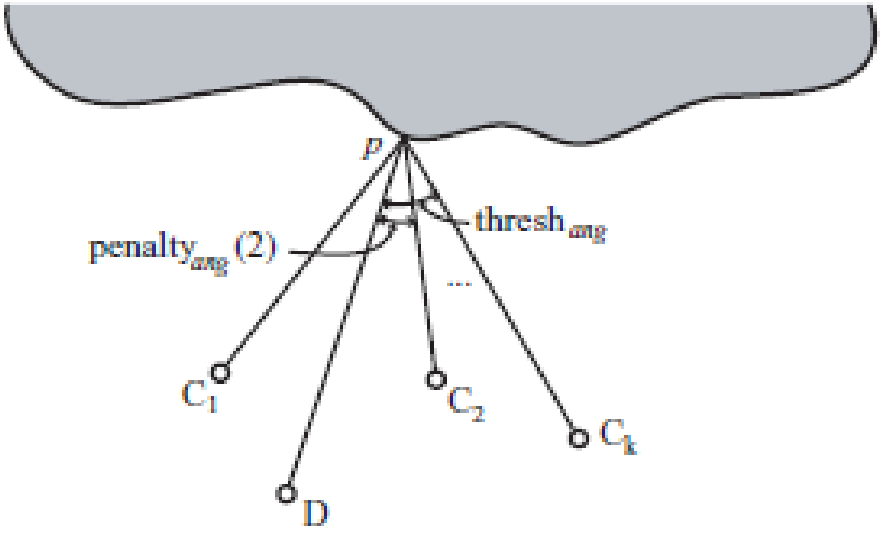



$$w_{ang}(i) = 1 - \frac{\text{penalty}_{ang}(i)}{\text{thresh}_{ang}}.$$

$$\tilde{w}_{ang}(i) = \frac{w_{ang}(i)}{\sum_{j=1}^{k} w_{ang}(j)}.$$

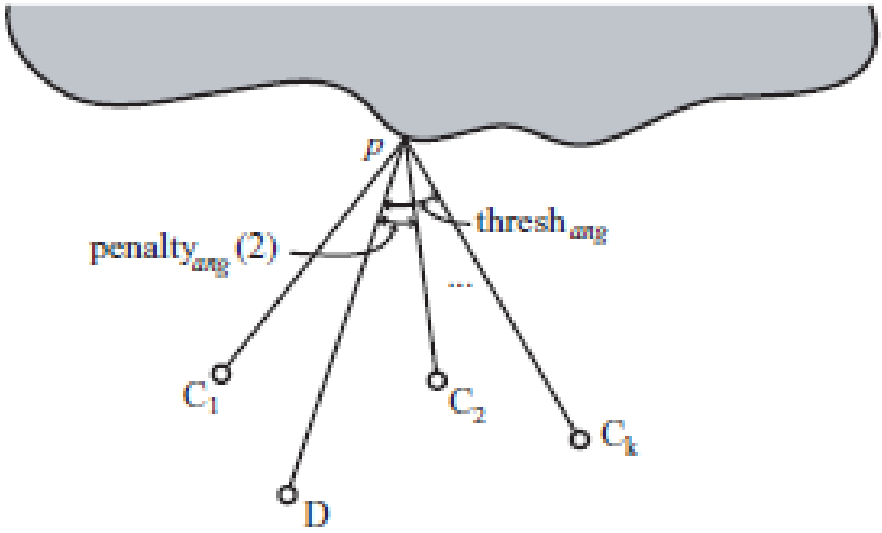
# Resolution Penalty

$$\text{penalty}_{res}(i) = \max(0, \|p - C_i\| - \|p - D\|)$$



# Field-of-View penalty

$$\text{penalty}_{fov}(i) = \begin{cases} 0 & \text{if } r_i \text{ within field-of-view} \\ \infty & \text{otherwise} \end{cases}$$



# Combined Penalty

$$\begin{aligned}\text{penalty}_{comb}(i) &= \alpha\ \text{penalty}_{ang}(i) + \beta\ \text{penalty}_{res}(i) \\ &+ \gamma\ \text{penalty}_{fov}(i)\end{aligned}$$

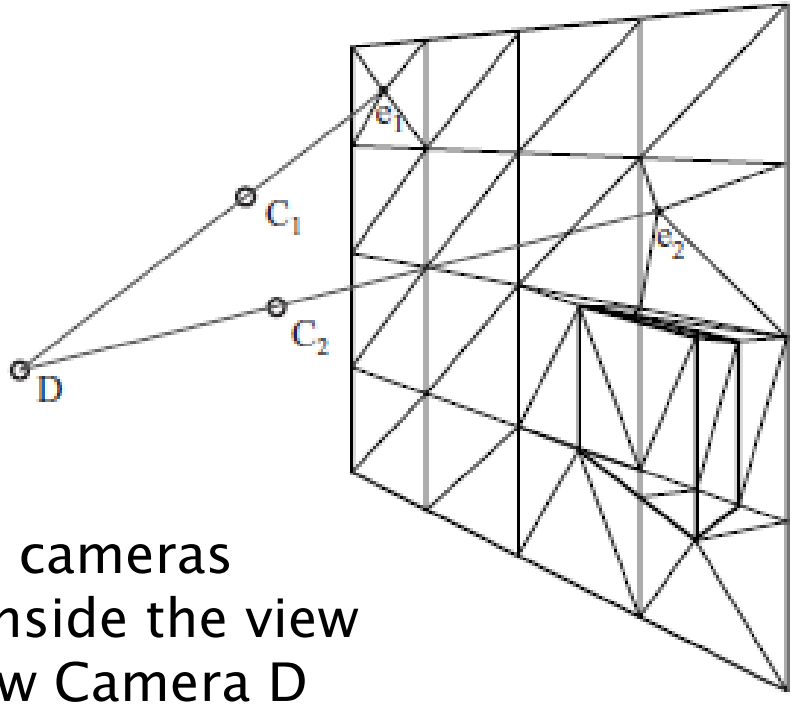
Define the combined weight $\tilde{w}_{comb}(i)$ based on K-nearest cameras

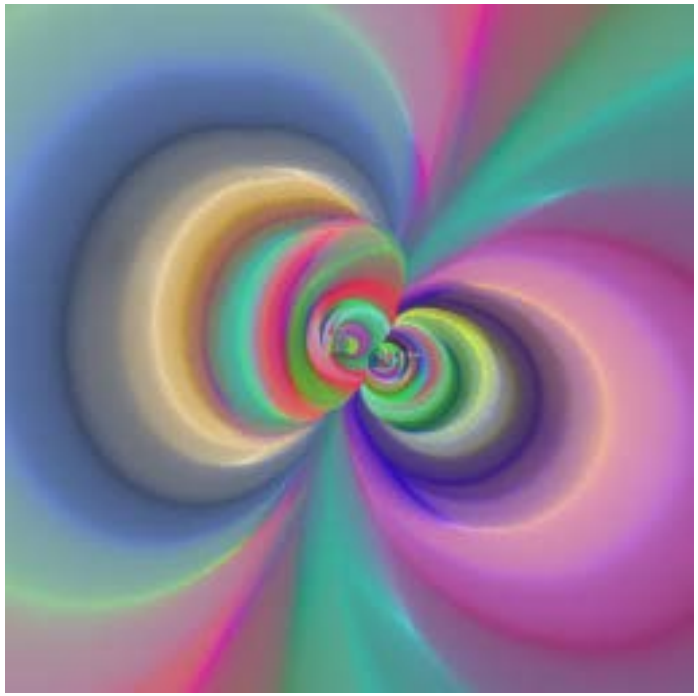
$$\tilde{w}_{comb}(i) = 1 - \frac{\text{penalty}_{comb}(i)}{\text{thresh}_{comb}}.$$

$$\tilde{w}_{comb}(i) = \frac{w_{comb}(i)}{\sum_{j=1}^{k} w_{comb}(j)}.$$

# Triangulation and Rendering

1. Draw rectangle grid



2. Search cameras (C1,C2) inside the view of the new Camera D
3. Line D-Cn to get the position e on the plane.
4. Triangulation

# Experiment

- Hallway
- A camera is mounted on a robot.
- The position of the photo is recorded.
- The robot continue move along the hallway

# Q & A

# Discussion

- What are the differences between this approach and:
  - Light fields
  - Lumigraph
- What can we use this for?


Thank you!