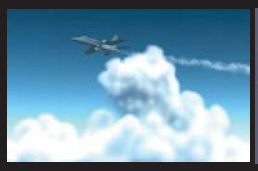
Modeling of Clouds

Modeling of Clouds

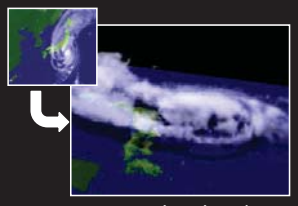
- Procedural approach
  - fractals, noise functions, etc.
- Difficult to create realistic shapes
  - fractals/noises are useful for small-scale details
  - overall shapes need to be specified by the user



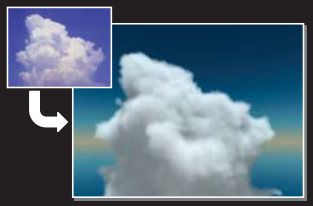
[Gardner 1985] [Nishita 1996] [Schpok 2003]

Image-based Modeling

- Use of a single photograph
- Not to reconstruct the same clouds
- Using the photo as a guide to synthesize similar clouds



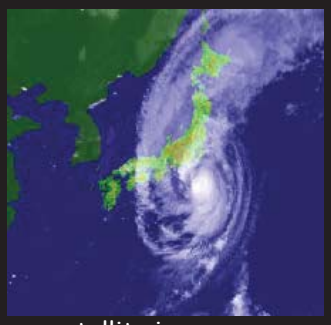
Large-scale clouds using satellite images



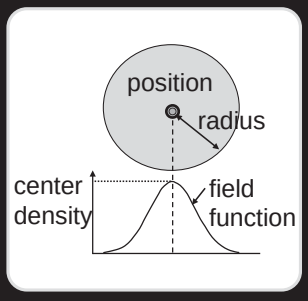
Cloud from a single photograph

3D clouds from satellite images

- Density distribution using metaballs
- Optimizing parameters of metaballs



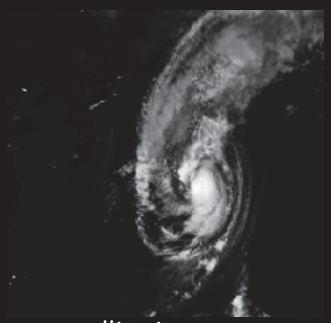
satellite image



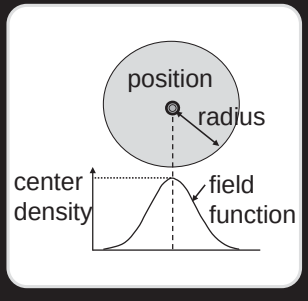
metaball

3D clouds from satellite images

- Density distribution using metaballs
- Optimizing parameters of metaballs



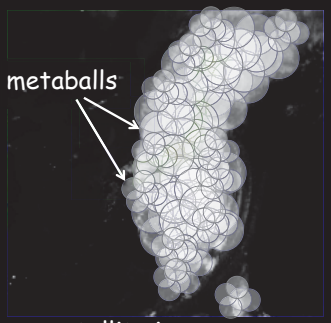
satellite image



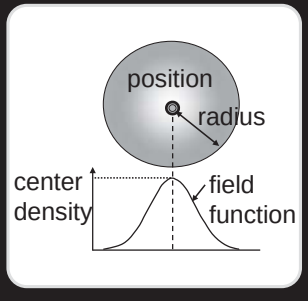
metaball

3D clouds from satellite images

- Density distribution using metaballs
- Optimizing parameters of metaballs



satellite image



metaball

3D clouds from satellite images

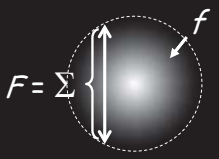
- Objective function

intensity of pixel k of input satellite image

$$\sum_{k=1}^{n} | I_k - J_k |^2 \rightarrow \min$$

cumulative density of metaballs overlapping at pixel k

$$J_k = \sum_{j=1}^{m} q_j F(| \mathbf{p}_k - \mathbf{c}_j |, R_j)$$



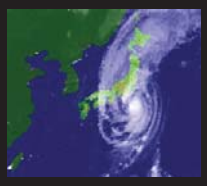
metaball

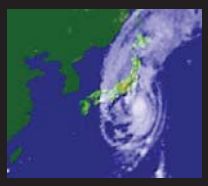
3D clouds from satellite images

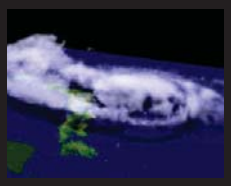
- Optimization process

1. Create new metaball
2. Determine center position
3. Determine radius and density
4. Calculate error
5. If error < ε then stop, otherwise go to 1

Examples


satellite image

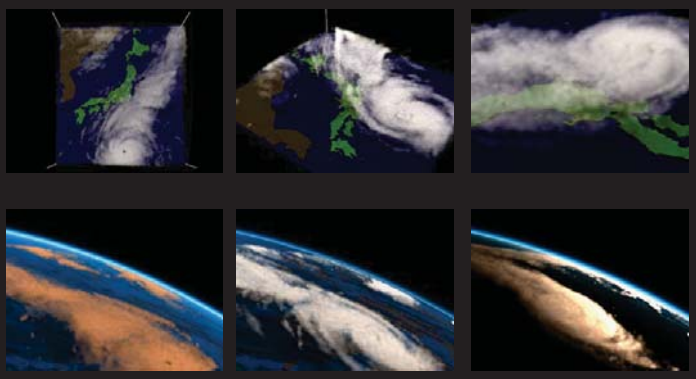

synthesized clouds


different viewpoint

No. of metaballs: 8506
Computation time: 13 sec.
Rendering time: 3.7 min.

computer: SGI Indigo2 MAXIMUM IMPACT
(195 MHz MIPS R10000)

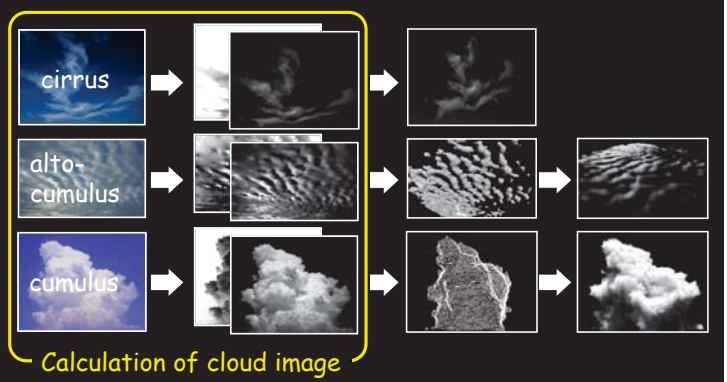
Examples



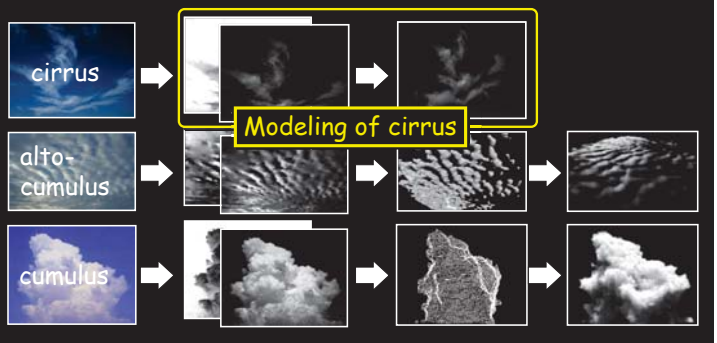
video video

Clouds from a single photograph

- Three types of clouds



Overview of Our Method

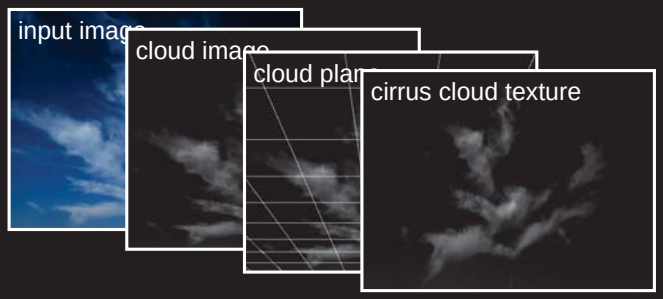


Modeling of Cirrus

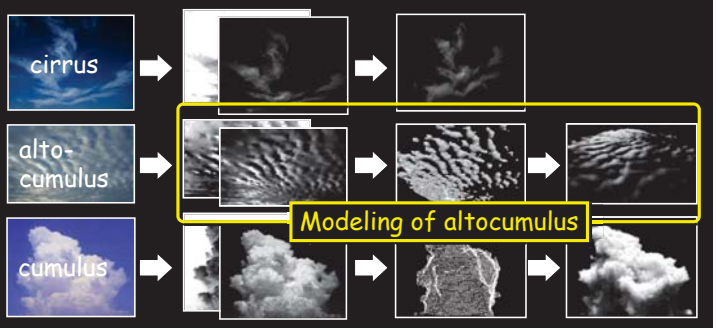
- Cirrus clouds
  - Thin and no self-shadows
  - Two-dimensional texture

Modeling of Cirrus

- Use of cloud image as 2D texture
  - Removing effect of perspective transformation by specifying cloud plane

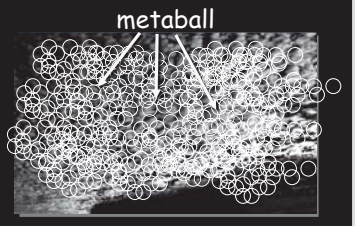


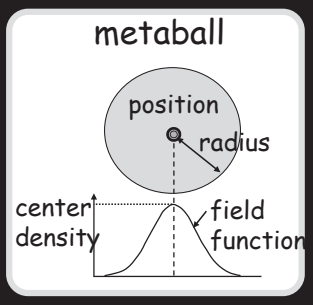
Overview of Our Method



Modeling of Altocumulus

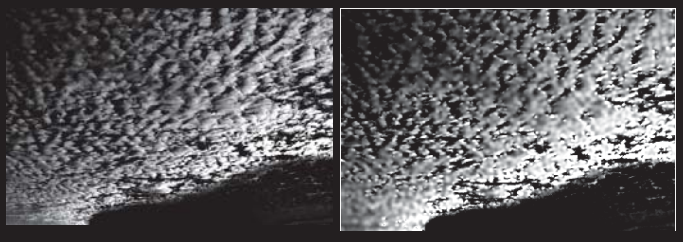
- Altocumulus
  - Thin but with self-shadows are observed
  - Using Metaballs to define three-dimensional density distribution





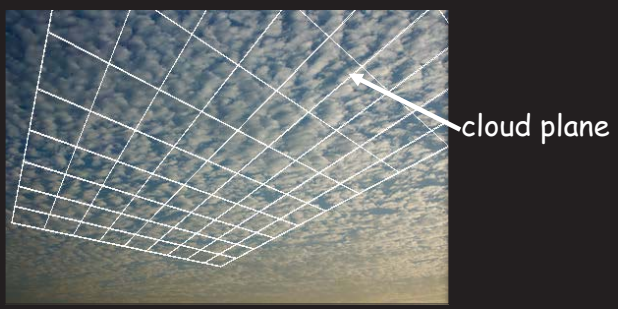
Optimizing Metaball Density

- Example



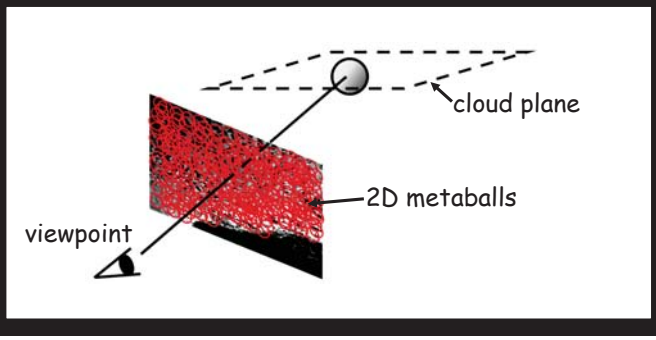
Specifying Cloud Plane

- Interactive specification
  - Orientation of cloud plane and viewing angle

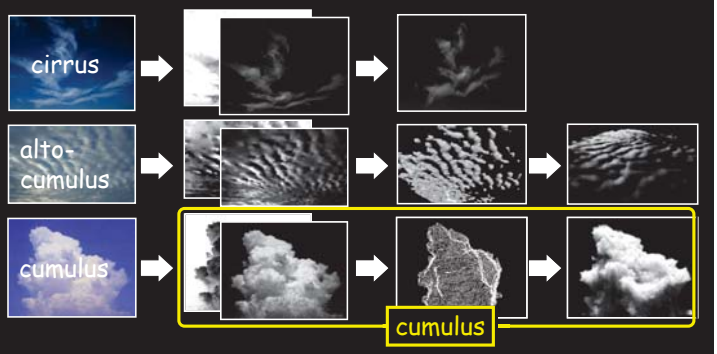


Projecting Metaballs

- Projecting metaball center onto cloud plane
- Scaling metaball radius in proportion to distance from viewpoint

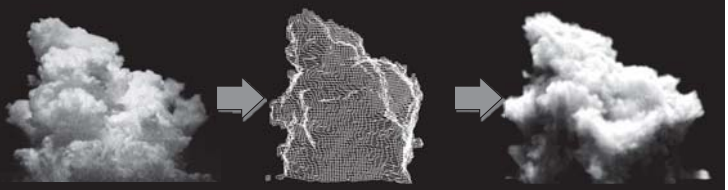


Overview of Our Method



Modeling of Cumulus

- Cumulus
  - Generating surface shape
  - Calculating densities inside surface shape

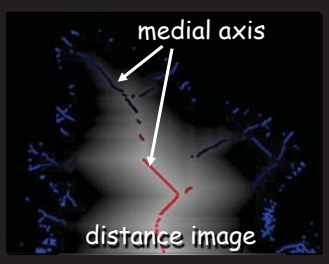


Computing Surface Shape

- Converting cloud image into binary image

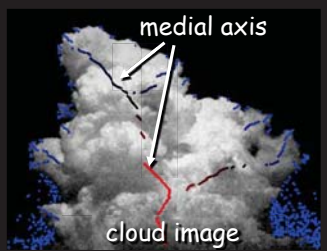
Computing Surface Shape

- Distance transform of binary image
- Extracting medial axes
  - Pixels where distances are local maxima



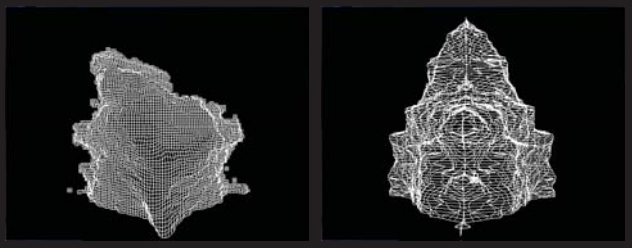
Computing Surface Shape

- Use distance at medial axis as thickness of clouds
- Propagate thickness by optimization



Computing Surface Shape

- Constructing surface shape
  - Assuming symmetric shape with respect to image plane



Computing Density Distribution

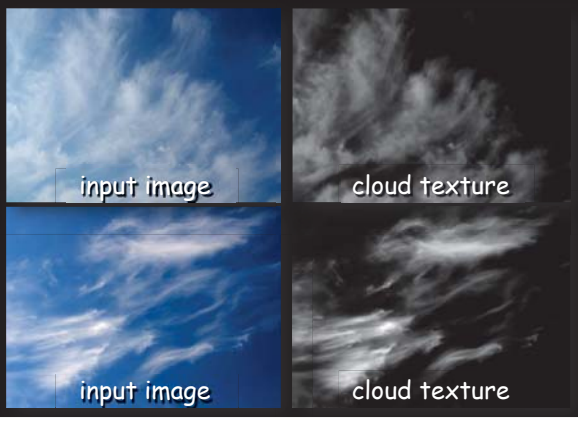
- Calculating bounding box of surface shape
- Subdividing bounding box into grid
- Calculating density at each grid point

Results

- Computer
  - CPU: Intel Corei7 (3.33 GHz)
  - Main memory: 4GB
  - GPU: NVIDIA GeForce GTX 295
- Computation time
  - within 10 seconds

Results

- Cirrus

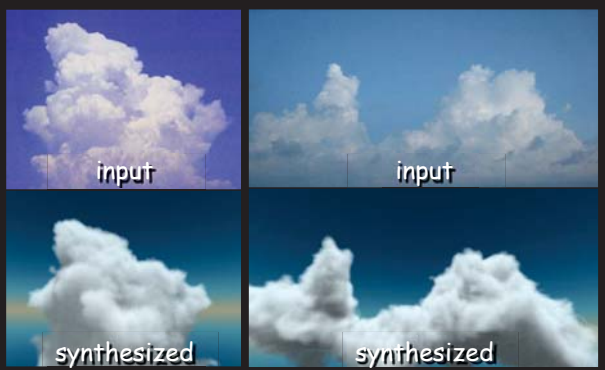


Results

- Altocumulus

Results

- Cumulus



Results

- Modeling process

Results

- A cloud scene