Special Applications

- If we assume a specific application, many image-based rendering tools can be improved
  - The Lumigraph assumed the special domain of orbiting small objects
- Two applications stand out:
  - Architecture, because people wish to capture models of cities
  - Faces, because there is no other good way to do it, and pictures of faces are essential to various fields (movies, advertising, and so on)

Hybrid Image/Geometry for Architecture

- Most buildings:
  - Are made up of common, simple, architectural elements (boxes, domes…)
  - Have lots of implicit constraints like parallel lines and right angles
- We can exploit the simple geometry and constraints to simplify the image-based rendering problem
- Hybrid approaches build simple geometric models from images, then texture them with the same images

Façade Debevec, Taylor, Malik 1996

- Start with a sparse set of images of a building (from one to tens of images)
- With an interactive *photogrammetric modeling program*, a user builds an approximate 3D model of the building
- Generate *view dependent texture maps* with the images
- Use *model-based stereo* to reconstruct additional detail (doorways, window ledges…)
- Render from any view (assuming images see all the surfaces)

Photogrammetric Modeling

- User specifies which parametric *blocks* make up a model, and the constraints between them
- User marks edges on the model and corresponding edges in images
- The system determines camera locations and model parameters using minimization algorithm
- Result: A 3D model of the approximate geometry
  - The blocks used determine the accuracy of the model
  - Details can be left out – later stages will catch them

View Dependent Textures

- The images can be projected onto the 3D model to determine which parts of the images correspond to which parts of the model
  - Hardware projective texture mapping can make this very fast
- More than one image may see a point on the model
  - Blend the image values
  - Use weights that favor images from cameras closer to the viewing direction (alpha blending in hardware)
- Some points may be seen in no image – use hole filling (or creative photography)

Model-Based Stereo

- Blocks do not capture all the details, such as sunken doorways
  - This introduces errors in the new renderings
  - View-dependent texture mapping helps if there is a view close to the right direction
- The approximate model gives many major hints to an automatic shape-from-stereo algorithm
  - Find correspondences between points in different images
  - Add depth information to the texture

Other Systems

- Other systems do a fully automated job
  - Find correspondences from "stable" features
  - Compute 3D location of some points/edges
  - Fit planes to the data
  - Feedback to improve results
- Works well in architectural environment (where flat surfaces abound)
- Must deal with occlusion – in most environments buildings hide each other in different views

Rendering Faces

- Producing realistic, expressive animated faces is (probably) the holy grail of rendering
- It is hard for many reasons:
  - Skin has extremely complex interactions with light
  - The face deforms due to a vast number of muscles and complex tissues
  - Must manage: hair (including facial), teeth, tongue, eyes, ears
  - The mapping from emotions to facial expression is complex
  - Most of all, viewers are exceptional at spotting problems

Overview

- A broad range of approaches:
  - Physiologically- and physically-based models
  - Pure image based approachs
- Current state-of-the-art depends on which application:
  - Capturing and then reproducing (in 3D) a fixed sequence
  - Producing static 3D models of faces, real and imagined
  - Dubbing speech with correct mouth and facial motions
  - Producing 3D facial animations
- Reference: Parke & Waters "Computer Facial Animation"

Why Imaged-Based?

- Rendering realistic looking skin from first principles is currently intractable
- Taking pictures of a person and applying it to a model is much simpler
- Some pure image based models may entirely avoid an accurate geometric model
- The only plausible current technology for capturing moving faces is video (or film)
- Key technology: Cyberware head scanners capture geometry and texture of a static face

Geometry, Physics and Images

Lee, Terzopoulos and Waters 1995

- Use a Cyberware scanner to get cylindrical depth information and textures
- Fit a pre-defined mesh to the data
  - Use landmarks like corners of eyes, mouth, etc
- Attach muscles and estimate skeletal structures
- Animate using a spring-mass model
- Render 3D model with captured texture
- Geometry for teeth, eyes, hair and shoulders

Video Rewrite

Bregler, Covell, Slaney 1997

- Aim: Take video of a person and make them say something else, with correct mouth and facial movements
  - For example, do much better film dubbing
- Works at the level of phonemes and visemes: the audio and visual subunits of speech
  - A training phase identifies visemes with phonemes
  - A reconstruction phase synthesizes new video by inserting a reconstructed viseme sequence into an existing video background

Training Overview

- Take video and audio of someone speaking, and break it into pieces
  - Uses Hidden Markov Models (HMM) for audio segmentation
    - Phonemes are standard from speech community
    - Different models for men and women
  - Use vision techniques to estimate and correct for changes in head pose
  - Output is a set of triphones (three visemes in a row) with associated visemes sequences

Synthesis Overview

- Take new speech and break it into phonemes (use training HMM)
- Search for appropriate sets of visemes
- Adjust timing of visemes to match audio timing
- Blend neighbors together to get seamless mouth motion
- Remove mouth and jaw from background video
- Estimate pose, morph visemes, and blend them into the background

Making Faces

Guenter, Grimm, Wood, Malvar, Pighin 1997

- Aim: Take video of someone, and generate a representation that allows reconstruction from any view
- Take video with multiple cameras of a person speaking with colored dots glued all over their face
- Track correspondences between dots to determine dot motion over time
- Generate a 3D model by scanning the head (with the dots)

Making Faces (cont)

- For each frame, compute model vertex motion according to the (known) locations of the dots
  - Two stage fit: first determine motion for a grid of points, then determine vertex motion
- To render, simply texture map the 3D fitted model
  - Textures come from the video streams
  - Dots must be removed from the textures, and the resulting holes filled in
  - Blend textures into one view independent texture
- Brute force and not real time

Facial Expressions From Photographs

Pighin, Hecker, Lischinski, Szeliski, Salesin 1998

- Aim: Generate static 3D models of various facial expressions, and then morph between them to animate
- Five pictures of each expression for each face
- Fit a standard 3D model to the pictures
  - Uses hand marked features (lots of them)
  - For good morphing, marked features should correspond to facial structures (eyes, mouth, eyebrows, etc)
  - Allows textures to be mapped onto vertices
  - Use view dependent textures when rendering

Facial Expressions From Photographs

- Outcome of fitting is a mapping from facial expression to deformations of the model
- Blend different deformation to get different expressions
  - Simple transitions to go from one expression to another
  - Different blends for different regions to get combinations of expressions (eg: forced smile)
- Deformations can be mapped from one person to another
- Animations generated by specifying sequence of expressions

A Morphable Model

Blanz, Vetter 1999

- Create a space of possible face geometries and textures
  - Points in the space are vectors of vertex locations and colors at vertices
- Apply statistical techniques to describe probable faces, to generate completely new faces
- Find regions in the space corresponding to particular facial features
  - Mark some faces with the given features (eg all males)
  - Determine which direction to move from a neutral face to get that feature

A Morphable Model (cont)

- Match a face to the model using iterative search
  - Find which point in the space corresponds to a given face
  - Start with a human specified guess
  - Repeatedly move the point according to the difference between a rendered image and the photograph
- Add new faces to the model using 3D scans
  - First find closest face in existing model
  - Improve the model point using optical flow, applied iteratively and hierarchically in difficult cases

Acquiring the Reflectance Field

Debevec, Hawkins, Tchou, Duiker, Sarokin, Sagar 2000

- Aim: Determine how a face's appearance changes under different lighting
- All previous work assumes fixed lighting
- First consider re-rendering from fixed views:
  - Use a *light stage* to capture images from fixed views of a face lit from various directions
  - To render with novel lighting:
    - Weigh each captured image by the strength of the incoming illumination
    - Composite weighed images

Acquiring the Reflectance Field

- For rendering with novel lighting and novel views
  - Use a model for how skin reflects
    - Specular scattering from surface oil layer
    - Diffuse reflection from subsurface scattering
    - Fit some parameters using data from the forehead and polarizing filters
  - Separate diffuse and specular components using color matching
  - Generate 3D model using structured light
  - Project images onto model and re-render using skin illumination model
    - Blends images from both original camera locations