Haptic rendering: How do we touch an object?

Subhasis Chaudhuri
IIT Bombay
www.ee.iitb.ac.in/~sc
IIT Roorkee, March, 2014

Outline
• Introduction
• Virtual reality & tele-robotic applications
• What is haptic rendering?
• Various examples
• Conclusions and homeworks
• References for those who needs.

Introduction
• Haptics is the science of touch
• Human haptics - study of human sensing and manipulation through touch.
• Machine haptics - study of design, construction, and use of machine to replace or augment human touch.
• Computer haptics - deals with models and behavior of virtual objects together with rendering algorithm for real time interaction.

About Touching
• Unlike others, touch is an active sensing mechanism.
• Based on force feedback
• Two types – kinesthetic and tactile
• Our study is on kinesthetic aspect.

Haptic Devices
• Consumer products:
  – Mouse
  – Joystick
  – Steering wheel
• Professional devices:
  – CyberGrasp
  – CyberForce
  – Phantom

Virtual Reality System
• Haptic interface (Haptic device)
  ◦ Electro-mechanical system
• Simulated Graphical objects
  ◦ Contain shape and other properties
• Haptic rendering algorithm
  ◦ Connects the first two components
Telepresence and Teleaction: Overview

Operator with Human-System Interface

Communication of Multiple modalities

Teleoperator in remote Environment

Telepresence makes task execution over barriers possible (distance, scaling, matter)

Haptic Rendering

- The process associated with generating and displaying the touch and feel of virtual object to human operator through a force reflecting device

Illustration of simple rendering

Reaction force follows spring law $F = kx$
- $k$: stiffness of object
- $x$: depth of penetration
- Direction of force along surface normal

Problems with Single Point Interaction

- Thin objects
- When HIP is equidistant from two faces

God Object Rendering

- Define additional variables to represent the virtual location of the haptic interface (god-object, ideal haptic interface point (IHIP), proxy)
- In free space, HIP and IHIP are co-located
- When HIP moves into an object, the IHIP remains on the surface
- IHIP computed such that its distance from the HIP is minimized

God Object Method (Contd.)

- Define a polygonal mesh of the bounding surface
- Location computation using Lagrange multipliers
  - $x$, $y$, $z$: coordinates of IHIP
  - $x_p$, $y_p$, $z_p$: coordinates of HIP
  - Constraints added as planes
  \[
  Q = \frac{1}{2} (x - x_p)^2 + \frac{1}{2} (y - y_p)^2 + \frac{1}{2} (z - z_p)^2. \quad (1)
  \]
  \[
  A_x x^2 + B_x y + C_x z - D_x = 0 \quad (2)
  \]
- Problems moving the proxy across sharp edges and sinking in small gaps.
(a) Mesh model of hand. (b) Visually rendered hand. Sphere represents the position of the haptic device in the 3D environment. (Data courtesy: http://www.zbrushcentral.com/showthread.php?t=66847)

God-object renderer. Green represents proxy and red the HIP

Home work #1

- Given a 3D mesh for an object, find an efficient way to detect collision.
- If you are given a 3D dense point cloud instead of a mesh, how would you detect the collision?

Surface Properties

- Stiffness
- Friction
- Smoothness
- Texture
- Rigidity

Issues

- How to handle point cloud data?
- Can we handle data sparsity?
- How to make sure that the proxy does not sink?
- How to render it faster than 1 ms?
- How does one handle scale of an object?
- How do we avoid dynamic meshing?

DISTANCE FIELD BASED OBJECT RENDERING
Data (P, N) Generation
- Hold a haptic device and try to feel the object. Force and position values are sampled.
- Render sparse data using a haptic device.

Haptic Scanning of Object
- Advantage-
  - scanned 3D points also provide the normal direction.
- Disadvantage-
  - Scanned points are highly sparse

Volume Reconstruction
- The problem: Given a sparse set of augmented points S(P,N), find the bounding surface and hence the distance field over a uniform grid of voxels.
- Distance field is computed in regular 3D grid of voxel from available haptic data through sphere packing. Computational advantage is due to following lemma.
- **Lemma 1.** When a single point and the direction of normal at that point is given only one additional point is needed to uniquely define the sphere on which they both lie.

Circle Embedding of Point Set
- Explanation in 2D with circle instead of sphere
- \( P_1, P_2, P_3, P_4 \) are arbitrary points in the set, the radius of the embedded circle between points \( P_1 \) and \( P_2 \) is
- \( r_{ij} = \frac{1}{2} |a_{ij}| \sec(\theta_{ij}) \)
- \( C_{ij} = P_i - r_{ij} \hat{a}_{ij} \)
  Where \( a_{ij} \) is the vector connecting the points \( P_i \) to \( P_j \).
- The radius of the embedded circle at \( P_i \) is
  \[ r_i = \min_{j \neq i} r_{ij} \]
  and \( C_i \) is the corresponding center

Circle Embedding of Point Set
- Point set is mapped on to a regular 2D grid
- Embed circle at each point
- Take the union of spheres (\( \odot \))
Distance Field Computation
Distance at a point $x$ is approximated as
\[
\Phi(x) = \begin{cases} 
0 & \text{if } x \notin \partial (\mathbb{D} \ominus nB) \\
& \text{boundary of } n\text{-times eroded set} \\
& \text{B is a unit radius structuring element} 
\end{cases}
\]

Problems with Sparse Sampled Data
- Circle ABC leaks out as there is no sampled point near P.
- Also, may not be topologically connected.

Haptic Rendering
- **Collision detection**
  
  If $X_h$ is the HIP, no collision detected if $\Phi(X_h) = 0$ when rendered force $F(X_h)=0$
- **Force computation**
  
  If $\Phi(X_h) > 0$, reaction force is proportional to the gradient of the computed distance field.

\[
F(X_h) = k |\Phi(X_h)| \frac{\nabla \Phi(X_h)}{|\nabla \Phi(X_h)|}
\]

Sample Results
- **Implementation**
  
  - in visual C++ in a windows XP platform with a CORE 2QUAD CPU @ 2.66GHZ with 2GB RAM.
- **Data set generation**
  
  - created virtual objects using polygonal meshes
  - Objects scanned manually with a 3-DOF haptic device
- **Reconstruction of objects**
  
  - reconstructed in a fixed 300X300X300 regular grid.
- **Haptic rendering**
  
  - force calculation takes only 0.02ms (less than 1ms).

Sampling Polygonal Mesh
- Model of a horse with 5K polygon

- (a) Polygonal mesh of horse (courtesy www.turbosquid.com)
- (b) mesh model with a different viewing angle

Reconstructed Object
- Model
  
  - Horse (5079 polygon)
- Number of sampled points
  
  - 1836
- Grid size
  
  - 59X157X178
- Number of inside grid nodes
  
  - 2,74,075
- Time required
  
  - 11.7 Seconds

- (a) Pre-processed haptic samples, as obtained after scanning, representing the object.
- (b) Zero-isosurface points of the reconstructed horse for visual rendering.
Cross-section of Distance Field

- White pixels are near the boundary of the object. Darkness increases as we move inside the object.

Distance field at selected cross-sections of the horse along XY, YZ and XZ planes respectively.

More results

- Model: Bomberman (2480 polygon)
- Number of sampled points: 3,751
- Grid size: 133X189X132
- Number of inside grid nodes: 11,30,637
- Time required: 2 m 43 s

(a) A .obj model of Bomberman (courtesy: www.turbosquid.com) (b) sampled sparse point set after pre-processing (c) zero-isosurface points of the reconstructed object for visual modeling, (d) the distance field for a horizontal cross-section of Bomberman, and (e) the same for a vertical cross-section.

More results

- Model: Hydrant (944 polygon)
- Number of sampled points: 3,756
- Grid size: 131X188X109
- Number of inside grid nodes: 5,91,775
- Time required: 44 Seconds

(a) A .obj model of a water hydrant (courtesy www.turbosquid.com) (b) The sampled sparse point set after pre-processing (c) the zero-isosurface points of the reconstructed object (d) the distance field for a vertical cross-section of the reconstructed object, and (e) the same for a horizontal cross-section.

More results

- Model: Owl (7652)
- Number of sampled points: 3,851
- Grid size: 188X184X89
- Number of inside grid nodes: 8,27,617
- Time required: 1 minute

(a) A .obj model of an owl (courtesy www.top3dmodels.com) (b) The sampled sparse point set after pre-processing (c) the distance field for a vertical cross-section of the object, and (d) the zero-isosurface of the reconstructed object.

Homework #2

- May have a non-smooth/bumpy experience.
- Would smoothing $\Phi(x)$ help?
- Reconstruction (off-line) is a bit slow but interaction is fast.
- How to handle varying stiffness?
- Can we make it proxy-based?
Motivation

1. Access of cultural heritage objects to visually impaired people
2. Better realistic and immersive experience to sighted persons
3. Issues – scalability and stereoscopic display

Proposed Method

Rendering

1. Graphic Rendering using quad mesh
2. Haptic Rendering
   Proxy based

Data Courtesy: www.archibaseplanet.com

Data Generation

Gaussian Pyramid - Gaussian low-pass filtering followed by down sampling to obtain depth image data at different level of details

The value of each pixel at level $l$ is given by

$$g(i,j) = \sum_{m=0}^{M} \sum_{n=0}^{N} w(m,n)g(i,j)[2m, 2n, l]$$

Results

Data Courtesy: www.archibaseplanet.com

Data Courtesy: www.gatech.edu/projects/large_models

Data Courtesy: www.archibaseplanet.com

Data Courtesy: www.archibaseplanet.com
**Introduction**

- **Objective**: Given a certain amount of clay develop a system for haptic pottery constrained to volume conservation.

- **Key points**:
  1. External or internal interaction with tool.
  2. Height of clay volume may increase due to interaction.
  3. Mechanical stability aspects of the clay model is also looked into.

**Bulk Clay Model**

- Clay body is initially assumed to be composed of N cylinders vertically stacked up.
- The number of cylinders may increase during interaction.
- The $i^{th}$ cylinder has an external radius of $R_i$ and internal radius of $r_i$.

**Structure of the haptic tool**

- Radially symmetric haptic tool.
- Implemented by revolving a 2D curve about the y-axis.
- The indices top and bottom gives the interaction of the haptic tool with the clay body.

**Collision Detection**

- Collision is of two types: internal and external.
- Condition for collision detection are easily checked using distances.
- Demarcating radius used to differentiate between external and internal collision.
Clay redistribution

- On collision with the tool the radii are modified as
  \[ R_{i}^{nw} = R_{i} - \epsilon_{x} s_{i}, \]
  \[ r_{j}^{nw} = r_{j} + \epsilon_{x} s_{j}. \]
- Distribute removed clay following Rayleigh distribution.
- Movement of the haptic tool decides direction of clay redistribution.

Force Feedback

- Force feedback is comprised of
  1. Frictional force
  2. Spring force due to deformation.

Mechanical Stability

- Stress at each cylinder level is calculated.
- If stress exceeds certain threshold value then the structure collapses.
- Further interaction is with the remaining clay in such a case.

Results

- A 3 degree of freedom Falcon haptic device from NOVINT was used for the interaction.
- Clay preservation inaccuracy: 0.44% of initial volume.
- User rating: 4.0/5.0

Motivation

- Detect hand tremor at an early stage.
- Accelerometry affects the dexterity and EMG is invasive.
- Haptic devices are now affordable and accurate
- As a gaming device your (grand)kids will have it at home
- Virtual environment can be altered as and when required.

Hand Tremor Detection in a Virtual Haptic Environment

(Caltech, June 2013)
Hand Tremor
- Hand tremor: unintentional and rhythmic movement
- Tremor types:
  - Essential (4-12 Hz)
  - Parkinson’s (3-7 Hz)
  - Physiological (3-30 Hz)

Methodology
- Define a virtual haptic task.
- Subject is asked to perform the task.
- Collect the velocity information.
- Pre-process data to remove task related motion.
- Perform spectral analysis of data.
- Peak, if any, shows tremor.

Haptic task specification

1. Moving a rigid object
- Contacts are impulsive
- Generates transients
- Pin-ball type motion
- Effect of mass can be studied

2. Handling Soft, deformable object
- Smooth contact with object
- Prolonged interaction
- High computational cost
- Stereoscopic display for better hapto-visual immersion
3. Interacting with variable density medium

- To study nature of resting/static tremor
- Is there an effect of loading?

3.17 Hz 0.0097

8 2.93 Hz 0.0073

12 3.17 Hz 0.0037

Validation of Method

Using accelerometer based motion sensor

No tremor

Tremor

Power spectral density plots

Results for rigid object manipulation

<table>
<thead>
<tr>
<th>Mass</th>
<th>Tremor frequency</th>
<th>Amplitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>3.17 Hz</td>
<td>0.0097</td>
</tr>
<tr>
<td>8</td>
<td>2.93 Hz</td>
<td>0.0073</td>
</tr>
<tr>
<td>12</td>
<td>3.17 Hz</td>
<td>0.0037</td>
</tr>
</tbody>
</table>

Results of static tremor with loading

Tremor frequency vs. Spring constant plot
Conclusions

- Haptics is an interesting research area
- How efficiently can you move the proxy during the interaction?
- Computation has to be fast enough
- How do you render a deformable object?
- How to handle the communication issues?
- What all new applications can you develop?

References


References (Cont.)

References (Cont..)


References (Cont..)