Interactive object driven Multi-party VR framework with time scaling

23rd January 2018

Nisha Jain
Research Fellow, IMI, NTU
Supervisor: Prof. Nadia Magnenat Thalmann, Director, IMI, NTU
Motivation

• Realistic Immersive Platform for object driven interactions
  – VR Simulation
  – Multi-party framework
  – Autonomous agents
  – Moods and behaviors of computer agents
SOA : VR Simulations

[1] Kinect, Oculus and Smartphone to build a VR platform game.


Simple setup ✓ Action Limitations ×

Comprehensive Actions ✓ Complex setup, Latency issues ×


Realistic Immersive Platform: Requirements

• Multi-party Framework

• No delays and latency issues

• Realistic behaviors and motions of autonomous agents
Problem I: Requirements for Multi-party

- Comprehensive setup: Simple and generic
- Multi-party framework: Interaction using objects
- Full body capture
- VR devices:
  - HMD: reconstructing the head rotations.
  - Two hand controllers: reconstruction of hand position and rotation. *Restricts the actions and movements of the user.*

- We develop a comprehensive multi-party virtual reality platform which captures the users by integrating data from a head mounted device and kinect.
Multi-party VR Framework

Problem 1

Problem 2

Problem 3
Interactions in the multi-party VR pipeline

Problem 1
World Controller
- World State
- Latency Management

Object Control
- Broadcast world state change
- Notify Objects state
- Action
- Contact of object with agents

Problem 2
Agents
- Agent State
- Agent Characteristics

Problem 3
Autonomous Agents

Human/Real Agents
Illustrative Example:
Multi-player Volleyball game

• Combination of autonomous players and avatars.
• World : game environment which comprises of the court, the players and the ball.
• Agents -> Players.
• Object Controller -> ball controller.
• World state : Positions of all players (autonomous and avatars) in the game and their respective interactions with the ball.
• Interactions between the players and ball is notified to the ball controller.
Interactions between avatars
Interactions between avatars

Making a serve in the Volleyball game.

User makes different actions while playing Volleyball.
Evaluations:
Problem 1 (In Progress)

• Impressiveness of the proposed framework.

• We create a User survey based on the questions from IEQ [1] and PQ [2] questionnaire.

Need for Latency Management

<table>
<thead>
<tr>
<th>Avatar1’s view</th>
<th>Avatar2’s view</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Timestamp $t_1$" /></td>
<td><img src="image2" alt="Timestamp $t_1$" /></td>
</tr>
<tr>
<td><img src="image3" alt="Timestamp $t_2$" /></td>
<td><img src="image4" alt="Timestamp $t_2$" /></td>
</tr>
<tr>
<td><img src="image5" alt="Timestamp $t_3$" /></td>
<td><img src="image6" alt="Timestamp $t_3$" /></td>
</tr>
</tbody>
</table>
Problem II: Latency issues

- **VR devices:**
  - HMD: researched and can be handled by their respective SDKs.

- **Network setup:**
  - Synchronization of the objects: as each user has a different view.
  - VR environment: if the views are not in sync, it reduces the quality of the immersive experience and makes communication between avatars difficult.

- **We propose a synchronization technique for multiple interaction amongst avatars using objects.**
Latency Management
Target based time scaling

Problem 1
Problem 2
Problem 3
Latency Management

Avatar 1’s view

Timestamp $t'_1$

Avatar 2’s view

Timestamp $t'_2$

Timestamp $t'_3$
Multi-party VR
Evaluations:

Problem II (In Progress)

• User experience survey - to evaluate our synchronization method.

• Participants will complete a questionnaire after they experienced the simulation with and without the time scaling.

• For every question asked, each participant decides whether they found SM1: Simulation 1 or SM2: Simulation 2 better.
Conclusion

• Our setup is generic yet simple and utilizes combination of Oculus and Kinect.
• Client-server architecture : modularity of our proposed scheme allows easy addition of multiple avatars (clients) in the virtual world.
• Time synchronization scheme : enhances the user experience in the virtual environment.
• Our platform is flexible and can be extended to different VR applications.
Problem III:
Realistic immersion of Virtual Human
(Future Work)

• Realistic motions of autonomous players
  – Moods and behaviors
  – Realistic reactions

• Group Decision making
  – Mental simulation – predict user possible decisions in real time (RL)
  – Learning of utilities of certain actions within certain contexts
  – Maximize group utilities
Thank you!!

Q & A ??
Autonomous Agent Computations
Real User (Avatar) Processing

Real Agent

Object Control

Action Processing

Pose Analysis

Force Estimation

Avatar

Avatar Controller

Skeleton Capture

Calibration Skeleton Data
Skeleton Data

Back
SOA : Multi-party VR

• Distributed architecture implementation.
• User study to demonstrate high spatial presence with low cyber sickness effects.

[2] 3D virtual pets game using Photon Unity Network and Google cardboard
• FSM – simulation process
• Accelerometer and google card board
• Survey for user experience.
• Basic 3D models and simulation process.