Digital Interactive Game Interface Table Apps for iPad

Supervised by:  Professor Michael R. Lyu
Student:        Ng Ka Hung (1009615714)
                 Chan Hing Faat (1009618344)

Year 2011 – 2012 Final Year Project

Department of Computer Science and Engineering
The Chinese University of Hong Kong
Agenda

Introduction & background

Project Idea

Our work

Conclusion
Augmented Reality
What is AR?

Combination of reality + Computer generated graphics

Interactive & digitally manipulable
What is AR?

Enhance reality and impressiveness

 Mixed Reality (MR)

Reality  Augmented Reality (AR)  Augmented Virtuality (AV)  Virtual
AR types

Marker-less

- GPS
- Digital compass
- Camera assisted
AR types

Marker-based
- Camera
- Analyze marker
- e.g. QR code
AR examples

Applications

- Geo-navigation
  Compass, etc.

- Informative
  Stores

- Translation
  Direct view
Applications

- Samplers
  As an interactive prototype
  Product Advertisement
Digi.T.able

A project supervised by Prof. Michael Lyu in 2007
Background

Digi.T.able

- a multi-purpose interactive table
- allows players in different places to play games by real objects
- Board games (e.g. Chinese chess, uno)
- Action games (e.g. snooker, air hockey)
Digi.T.able

allows players in different places to play games by real objects
Share a same common space

Implementation on iPad
i.Digi.T.able

2 iPad shares a common AR space to play a game
Objectives

**i.Digi.T.able**

- Track the real-object mark and determine the camera’s position
- Display simple objects on virtual space depends on real space scenes
- Exchange position information between 2 iPad clients
- Implement a simple AR game on iOS platform (iPad)
How we do that

Design

4 main components

- Marker tracking module
- Network connection module
- Virtual world construction module
- Game engine
Marker tracking

The Marker Tracking Module is the agent that directly communicate with the Qualcomm AR SDK. It is a finite state machine keep analyzing data from camera.
Design

Marker tracking

Detection and recognition

Image conversion → Feature points computation → Identification
Marker tracking

Image conversion
- thresholding
- convert the captured frame from colored into binary image
Design

Marker tracking

Feature points computation
- corners need to be detected
- in order to have reliable camera pose estimation
Marker tracking

Identification
- restore the effect of rotation, translation and perspective transformation
by solving a simple linear system
Marker tracking

Identification
- the positions of four corners by feature points computation and the 3D coordinates in object space of the marker’s corners are given by \((x_i, y_i, 0)\)

\[
X_i = \frac{a_1 x_i + a_2 y_i + a_3}{a_7 x_i + a_8 y_i + 1} \\
Y_i = \frac{a_4 x_i + a_5 y_i + a_6}{a_7 x_i + a_8 y_i + 1}
\]
The result vector of the linear system implies a normalized marker. By using the result vector, the system can provide users the pose information for drawing virtual 3D objects.
Marker tracking

Qualcomm AR SDK

- fetches live streaming from the device camera
- The platform consists of these components:
  - Camera
  - Image converter
  - Tracker
  - Renderer
  - Application Code
  - Target Resources
Design

Marker tracking

Application
- Query State Object
- Update App Logic
- Render Graphics

Camera

Camera Frame

Pixel Format Conversion

Converted Frame

State Object
- Render Camera Preview
- Target Snapshot
- Converted Frame

Tracker
- Image Target
- Frame Marker
- Multi Target
- Virtual Buttons
  - Detect New Objects
  - Track Detected Objects
  - Evaluate Virtual Buttons

QCAR SDK
Marker tracking

Qualcomm AR SDK
- Trackables

-right-handed coordinate system is used
Marker tracking

How to choose input images?

- Rich in detail
- Good in contrast
- No repetitive patterns

Examples

Not good

Good
Marker tracking

Examples
Marker tracking

**Qualcomm AR SDK Compare with String AR**

<table>
<thead>
<tr>
<th></th>
<th>Qualcomm AR</th>
<th>String AR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>License</strong></td>
<td>Free</td>
<td>Free for limited version</td>
</tr>
<tr>
<td><strong>Platform</strong></td>
<td>iOS, Android</td>
<td>iOS, Android (in progress)</td>
</tr>
<tr>
<td><strong>Multiple markers</strong></td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td><strong>3-rd Party Integration</strong></td>
<td>Yes, Unity3D</td>
<td>Yes, Unity3D</td>
</tr>
</tbody>
</table>
Recap

Design

4 main components

1. Marker tracking module
2. Network connection module
3. Virtual world construction module
4. Game engine
Design

Network

How to exchange data efficiently?

- Data size, data type
- Network load, frequency of update
- Accessibility
Design

Network

Consider...

- Network socket
- HTTP requests
- Game center
- Peer-to-peer
Design

Network

Connection protocol

Register phase
- gets a token
- use the token to connect again
(due to HTTP’s stateless property)

I want to connect as A

I have a key for you...

phiekurn8riyyv65t31h7ujyt73q2

phiekurn8r6iyyv65t31h7ujyt73q2
Network

Connection protocol

Register phase
- No IP involved
- Unique device identification
Network

Connection protocol

Data update phase
- present token
- update information to server
- gets update from server

Device B has some update for you...

I have some update..

Make changes...

phiekurn8r6iyyv65t31h7ujyt73q2

phiekurn8r6iyyv65t31h7ujyt73q2

UPDATED JSON
Recap

Design

4 main components
Virtual world construction

Graphics and UI

An important part to combine computer graphics and real scenes

*openGL ES (OpenGL for Embedded Systems)*

Light-weighted version of OPENGL on mobile devices
Virtual world construction

- Marker Tracking module: Local device position
- Game engine: Position of game objects
- Network connection module: Remote device/object position

Output to screen

Object shape information
Game engine

What and how you can play...

- game logic part
- Rule defined
- dependent specified game
Game engine

Event

- first delivery to the base view
- then its inherited views
Experiment

On Camera match

Objective
- investigate the effect of marker's properties e.g. size, number of features on the stability of camera match

Set up
- Place a target at a fixed position.
- Paper is pushed forward steadily 10cm
- Application replay the movement of iPad by representing it as a 3D model
- Observing the replay
- Test the stability of camera match
On Camera match-moving

Control

Control A (less feature and small size)
Control B (more feature and small size)
Control C (less feature and large size)
Control D (more feature and large size)
On Camera match-moving

Results & evaluation

- all controls, the z value keeps constant
- A and B
  - x value changes regularly, but not constantly
  - y value oscillates
- C and D
  - x value change differences are constant and small
  - Y remains constant
Experiments

On Camera match-moving

Conclusion

- A large marker gives more stable tracking
- Marker size also outweighs the number of features of the marker.

Other factors

- Movement of the iPad
- Camera Shake
  - Algorithm to minimize unnecessary effect
On Camera match-moving

Demo program
- Record the track of the movement
- Move the device
- Replay the tracked movement

Target marker
Network part

Objective
- Test stability and performance

Set up
- 2 iPads
- app installed
- 2D movement only
Experiment

Network part

Procedure

- Connect to server
- Moves **ME** label
- Observe performance of **YOU**
  on another device
Network part

DEMO
The game

A demo game

Criteria

- A game to demonstrate AR and network technique
  - Marker dependent
  - battle via network
- Should be simple and direct
The game

A demo game

Dodge ball

- 2-player battle version

SETUP

- A marker on the wall
- 2 iPads with app installed
- Server ready

Game interface

Dodge this ball

Tap to shoot a ball
The game

A demo game

DEMO

Target marker
The game

Admin view on web interface

DEMO
Conclusion

To summarize this semester ...

- We focused on tracking AR marker and analysis positional data
  - QCAR SDK
  - Network implementation
  - A simple dodge ball game
Future work

Next goals

- Stabilized camera tracking
- Network Connection
- User Interface
- Assit with iPad accessories (GPS? Gyroscope? Ambient light sensor? )
- More on Game design and implementation (snooker? chat room? )
- Investigate possibility for more clients
Q & A
The end

Thank you!