ARC Reference Modules for Real Character Representation

SIGGRAPH Web3D Meeting
Aug. 8 2012
Kwan-Hee Yoo
Chungbuk National University
• Set of principles
• Terms and their precise definitions
• ARC reference module for real character representation
• Use cases
Set of principles

• This module is made to represent real character into ARC space based on ARC Reference Model

• Representation items of real characters
  – Spatial Mapper: Embedding real characters in real space into those in ARC space. Real characters obtained from (for an example) depth camera can be animated after converting their action data to H-Anim
  – Event Mapper: connect recognized events acted by real characters to reactions due to their events (interaction between real character and ARC environment)
Terms and their precise definitions

- Real characters
- Real space
- ARC space
- BBRS: Bounding box in real space
- BBARCS: Bounding box in ARC space
- Spatial mapper
- Spatial mapping description
- H-Anim
- (Virtual & Real) Character animation

- Action – gesture, voice, face
- (Action) recognition
- Event
- Event Mapper
- Augmented mapping description
Real Character in the world

Image Information

- Background removal & Feature region computation

Image Information

- Depth Information

Depth Information

- Skeleton & depth extraction

User Information Extraction
- Color data / Depth data / Skeleton data

Filtering

Filtering
Filtering

Spatial Data

Tracking & Extraction Depth Information

BBRS (Bounding box in real space)

BBARCS (Bounding box in ARC space)

Spatial Mapper

Spatial Mapping Description

Character 3D Model

Scene Composite

Real Character+
3D Virtual Model

Static/Dynamic 3D DB

BBARCS -> BBARCS

convert

H-Anim

Real Character+
3D Virtual Model
Nodes

Nodes for real character

DeviceNode
{
    SFString[in] id
    SFString[in] type
    SFFloat[in] fov
    SFInt[in] framerate
    SFImage[out] image
    MFString[out] jointType
    MFVector[out] value
}
Nodes

Nodes for real character

BBRSNode
{
    SFString[in] id
    SFString[in] description
    SFVector[out] startpoint
    SFVector[out] endpoint
    DeviceNode[in] device
}

BBARCSNode
{
    SFString id
    SFString description
    SFVector[out] startpoint
    SFVector[out] endpoint
    // 가상공간을 명시하는데 좋은 듯 함
}
Use Cases of Spatial Mapper

General Camera Sensor for real character

<device id = "cam0" type = "camera" fov="50" framerate="30"> </device>

Depth Camera Sensor for real character

<device id = "depthcam1" type = "depthcamera" fov="50" framerate="30" joint_type = "left_hand" value = "0 0 0"> </device>

Get two points for a bounding box in real space

<bbrs id = "bbrs1" description = "movable space of real characters in real space" startpoint="0 0 0" endpoint="640 480 100" device="cam0"> </bbrs>

Get two points for a bounding box in ARC space

<bbarcs id = "bbarcs1" description = "movable space of real characters in ARC space" startpoint="30 30 20" endpoint="60 40 10"> </bbarcs>
Action Types

GESTURE_XXXX_XXXX
- Hand, Foot, Arm, Leg, Face, Eye, Full body, image
VOICE_XXXX_XXXX
etc

Events

Action Recognition

point
image
vector
Action Types

GESTURE_XXXX_XXXX
- Hand, Foot, Arm, Leg, Face, Eye, Full body, image
VOICE_XXXX_XXXX etc

Gesture Events

P = (x, y, z)
P1 = (x1, y1, z1)
P2 = (x2, y2, z2)

point
vector
image
skeleton
**Gesture Events**

- **Posture**
  - Point: \( P = (x, y, z) \)
  - Vector: \( P_1 = (x_1, y_1, z_1) \), \( P_2 = (x_2, y_2, z_2) \)

- **Gesture**
  - Rightward: \( X_{\text{POS}}, X_{\text{NEG}} \)
  - Upward: \( Y_{\text{POS}}, Y_{\text{NEG}} \)
  - Forward: \( Z_{\text{POS}}, Z_{\text{NEG}} \)

- **Directions**
  - Clockwise (CW): \( X_{\text{WAVE}} \)
  - Counterclockwise (CCW): \( Y_{\text{WAVE}} \), \( Z_{\text{WAVE}} \)

**Image**

**Skeleton**
<table>
<thead>
<tr>
<th>Action</th>
<th>Objects</th>
<th>Move</th>
<th>Definition of action types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gesture</td>
<td>Left hand (LH)</td>
<td>The x positive direction</td>
<td>GESTURE_LH_X_POS</td>
</tr>
<tr>
<td></td>
<td>Right hand (RH)</td>
<td>The x negative direction</td>
<td>GESTURE_RH_X_NEG</td>
</tr>
<tr>
<td></td>
<td>Left foot (LF)</td>
<td>The y positive direction</td>
<td>GESTURE_LF_Y_POS</td>
</tr>
<tr>
<td></td>
<td>Right foot (RF)</td>
<td>The y positive direction</td>
<td>GESTURE_RF_Y_POS</td>
</tr>
<tr>
<td></td>
<td>Left Lower Arm (LLA)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Left Arm (LA)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Right Lower Arm (RLA)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Face (F)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Eye (E)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Full body (FB)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Camera control using hand gestures</td>
<td>LEFT_HAND</td>
<td>RIGHT_HAND</td>
<td>Functions</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------</td>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td>GESTURE_LH_X_NEG</td>
<td></td>
<td>GESTURE_RH_X_POS</td>
<td>Left (Rotation) Movement of a camera in x-positive direction</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>GESTURE_LH_X_POS</td>
<td>Right (Rotation) Movement of a camera in x-negative direction</td>
</tr>
<tr>
<td>GESTURE_LH_Y_POS</td>
<td>GESTURE_RH_Y_POS</td>
<td></td>
<td>Up (Rotation) Movement of a camera in y-positive direction</td>
</tr>
<tr>
<td>GESTURE_LH_Y_NEG</td>
<td>GESTURE_RH_Y_NEG</td>
<td></td>
<td>Down (Rotation) Movement of a camera in y-negative direction</td>
</tr>
<tr>
<td>GESTURE_LH_X_NEG</td>
<td>GESTURE_RH_X_POS</td>
<td></td>
<td>Zoom In (Scaling) Decreasing distance between a model center and a camera center</td>
</tr>
<tr>
<td>GESTURE_LH_X_POS</td>
<td>GESTURE_RH_X_NEG</td>
<td></td>
<td>Zoom Out (Scaling) Increasing distance between a model center and a camera center</td>
</tr>
</tbody>
</table>
Image Events
AR/Dynamic 3D model DB

D_Model 1  D_Model 2  ...  D_Model n

Static 3D model DB

S_Model 1  S_Model 2  ...  S_Model n

Display

2D/3D Display Devices

Rendering

Scene Compositing

Action Execution

Transform D_Model connected to the action to be executed
Thank you

Kwan-Hee Yoo
khyoo@chungbuk.ac.kr