Tangible Interaction with Intelligent Virtual Agents

Seminar Advances in HCI
Lecturer: Matthias Loskyll
24.01.2008
Chair: Peter Feld
Supervisor: Martin Rumpler
Overview

- Introduction
- COHIBIT/VirtualConstructor
  - Installation
  - Authoring Interactive Performances
  - Advantages and Disadvantages
  - System Architecture
- Summary
Introduction

- last week: multimodal interaction
- tangible interaction: person interacts with the system through the physical environment
  - e.g. manipulating instrumented objects by moving, rotating, etc.
- interaction with intelligent virtual agents:
  - How do the agents know what’s happening?
  - How can we bridge the gap between real and virtual world?
  - How can we get co-presence?
COHIBIT/VirtualConstructor

- COHIBIT = Conversational Helpers in an Immersive exhibit with a Tangible interface
- Edutainment exhibit for theme parks
- Installed in VW Autostadt in Wolfsburg
- Two life-like virtual characters
- 3D car puzzle as tangible interface
- Easy to use, simple and robust
Installation
Virtual Agents

- life-sized conversational agents
- team of agents because:
  - less obtrusive information exchange / interaction
  - attracting potential visitors
- multimodal: speech, gestures, body language
- act as
  - guides: giving hints how to complete construction
  - commentators: comments on visitors’ actions and task state
  - motivators: encouraging to continue
  - tutors: giving background information
- mainly reactive, but can take initiative
- context-aware behavior
Tangible Interface

- 10 RFID-tagged car-model pieces
- four categories of pieces:
  - two front-ends
  - one driver’s cab
  - two middle parts
  - five rear ends for convertible, coupé, compact car, van and SUV
- can be placed on workbench with five adjacent areas in either direction
- complete car consists of at least three parts
Issue: Combinations

- 802.370 different combinations
- impossible to address each
- BUT: do not over-generalize!
- Classification scheme:
  - Car completed
  - Valid construction
  - Invalid configuration
  - Completion impossible
  - Wrong direction
Construction Code

- consider category of element:
  - **F** for front element
  - **C** for cockpit
  - **M** for middle element
  - **R** for rear element
- empty positions between pieces: hash symbol
- example: F#M
- construction code + used rear element => car type
- if invalid configuration, only local context evaluated
- example: FR#: F must be followed by C!
System Modes

- **Off-Mode:**
  - when no user is present
  - guides try to attract attention of passers-by
  - small talk between two agents, idle time actions

- **On-Mode**
  - welcome visitors
  - explain purpose of the exhibit
  - comment user’s actions and state of construction
  - give information about car parts and possible combinations
    => information + entertainment
On-Mode Phases

- **Visitor arrives**
- **Welcome Phase**: Hello and welcome to our car construction exhibit. My colleague and I will assist you during the construction process.

- **Visitor places rear end**
- **Construction Phase**: Oh! This looks funny! I guess we should help. Even if you want to build a compact car, you need a driver's cab between the front and the rear end.

- **Visitor completes car assembly**
- **Completion Phase**: Congratulations! You have built a Sedan, a comfortable four-door car. Very popular in the eighties!
Authoring Interactive Performances

- interactive presentation created with SceneMaker authoring suite
- separation of content and narrative structure
- three concepts:
  - **scenes**: dialog content written in screenplay-like language
  - **sceneflow**: linking scenes in a graph with SceneMaker => narrative structure
  - **transition event rules**: transition in a sceneflow defined in a XML-based rule language
Scenes

- pieces of user-edited dialog
- can contain tags to control agent’s non-verbal behavior
  - gestures
  - facial expressions
  - system control commands (e.g. pictures, videos, camera)
  - context information (e.g. current piece, weather, time, etc.)

Carina: [progress][+turn2visitor] The car you’ve build is called a roadster.
[picture car1.jpg] [happy] It’s a very sportive car with two doors.
Adrian: Well, [happy] this car is stylish!
Carina: [+nod]

- 696 pre-scripted scenes
- generation at runtime would also be possible
Issue: Variation

- characters shouldn’t repeat themselves
- solutions:
  - make scene selection dependent on current context
  - blacklisting: block played scenes for some time period
  - 2 – 9 variations for each scene (scene groups)
  - decomposition of long scenes into smaller sections, reassembling at runtime
SceneFlow

- represented as cascaded Finite State Machines
- consisting of nodes and edges
- scenes attached to nodes and edges
- two node types:
  - Scenenodes
  - Supernodes
- edge types:
  - Conditional-edge
  - $\varepsilon$-edge
  - Probabilistic-edge
  - Interrupt-edge
- currently 205 nodes, 35 Supernodes and 306 edges
Sceneflow
Transition Event Rules

- condition-action rules operating on input data
- mapping of visitors’ actions onto transition events
- update context variables
- define a path through sceneflow
- four types:
  - Visitor appeared/disappeared: trigger ON-/OFF-Mode
  - Car completed/disassembled: construction -> completion
  - Piece taken/placed
  - Piece upheld
Issue: Concurrent Actions

- multiple users could move multiple objects
- same action can trigger various transition event rules
- solution:
  - assign priorities to transition event rules
  - example:
    - visitor puts piece on workbench, completing construction
    - piece_placed event
    - car_completed event (higher priority)
- agents should react instantly
- BUT: interruption shouldn’t destroy dialog
- solution: transitory scenes, e.g. “Hold on for a second!”
Pros and Cons

Advantages:
- plan-based approach would be confusing
- graphical programming
- after compilation, direct test in system possible
- no programming skills needed
- separation of content and narrative structure
- combination of pre-scripted and generated scenes possible

Disadvantages:
- no parallel execution of scenes possible
- not applicable for multimodal dialog systems
The exhibit
Summary

- COHIBIT: an interactive edutainment exhibit
- intelligent virtual agents act as guides, commentators, motivators and tutors
- tangible interaction via instrumented car pieces
- scenes and sceneflow modeled with SceneMaker tool
- issues and solutions:
  - number of combinations -> current orientation, construction code, local context
  - need for variation -> context memory, blacklisting, grouping
  - concurrent actions -> transition event rules with priorities
Thank you!
Any questions?