Ludic Computing

• The study of design, construction and evaluation of software for interactive play
• Entertains and enriches the user
• Cuts across many areas of computing
Ludic Computing

We focus on **digital games** and **computer art**

- Video game AI
- Content creation for digital media
- Supporting creative expression

1. Content generation (7 lectures)
2. Ludic agents (5 lectures)
3. Players (3 lectures)

Part One

**Content Generation**

- Computational creation of **beautiful** artefacts
- What have creativity and beauty got to do with ludic computing?
  - Computers can help overcome the **content creation** bottleneck for digital media (such as video games)
  - Computers can support creative expression
Lecture 2

Computational Creativity

• Building software which takes on some of the creative responsibility in arts and science tasks
• Which AI techniques lend themselves to creative software?
  • Genetic programming, case-based reasoning, conceptual blending
• Methods for handing over creative responsibility
• How to assess progress?
  • Process vs. artefacts, measures of creativity

Lectures 3 and 4

Evolutionary Design

• Genetic algorithms and genetic programming
  • Coding schemes
  • Fitness functions and selection mechanisms
  • Evolutionary operators: crossover and mutation
• Applications
  • Evolving building designs
  • User-guided evolutionary design
  • Computer art
Lectures 3 and 4

Evolutionary Design

Lecture 5

Shape Grammars
Lecture 6

L-Systems

• Another approach to visual grammars
• Abstract art
• Vegetation/terrain generation
• City Generation (CityEngine and Subversion)

Lecture 7

Image Filters

• Transforms, compositors & filter trees
Lecture 8

Non-photorealistic Rendering

• Simulating natural media
• Brush paths
• Watercolour and pencil simulation
• Image segmentation
• Rendering colour regions
Part Two

Ludic Agents

- AI agents that support playful interactions
- Game playing agents
  - Video games, board games, ...
- They often need to act intelligently in complex and dynamic virtual worlds
- More importantly, they need to entertain

Lecture 9

Steering Behaviours

- Reactive movement in games
- Reynold’s steering paradigm
- Some basic behaviours in detail
- Combination techniques
- Emergent behaviour
Lecture 9
Steering Behaviours

Lecture 10
Pathfinding

• Using tiles, waypoints, navigation meshes
• Heuristic search over navigation graphs
• Precomputing paths
Lecture 11

Behaviour Trees

• Popular language for NPC behaviour
• BTs vs. state machines
• Specifying BTs
• Design and architecture issues

Lecture 12

Adaptive Games

• Applying action prediction and reinforcement learning to games
• Dynamic difficulty adjustment
• Adaptive pacing
• Adaptive content
Lecture 16

Monte Carlo Tree Search

• State-of-the-art game playing agents based on Monte Carlo methods

• Capable of beating professional Go players

Part Three

Players

• To build software for play we need to understand players

• What do they experience?

• What motivates them?

• How do we design for them? (And measure success?)
Lecture 13
Interactive Play

- What is play?
- Player experience of video games
- Theories of emotion & pleasure in games
- Altered states: immersion & flow

Lecture 14
Designing for Play

- How is designing a game different from designing any other interactive system?
- Game design theory
  - Design concepts and methods
- Evaluation methods
  - Measures of playful experience
Lecture 15
Social Networks

- Social networks & games
- Network measures and properties
- Degree distributions
- Applications: friendship, trust and recommendation networks; online communities; social gaming; etc.
- Models of network growth
- Diffusion of innovations

Lecture Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Thu 4pm</th>
<th>Fri 2pm</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1. Introduction</td>
<td>2. CC</td>
</tr>
<tr>
<td>3</td>
<td>3. Evolutionary Design 1</td>
<td>4. Evolutionary Design 2</td>
</tr>
<tr>
<td>4</td>
<td>5. Shape Grammars</td>
<td>6. L-Systems</td>
</tr>
<tr>
<td>5</td>
<td>7. Image Filters</td>
<td>8. NPR Rendering</td>
</tr>
<tr>
<td>8</td>
<td>13. Interactive Play</td>
<td>14. Designing for Play</td>
</tr>
<tr>
<td>9</td>
<td>15. Social Networks</td>
<td>16. Monte Carlo Tree Search</td>
</tr>
<tr>
<td>10</td>
<td>17. Revision lecture</td>
<td></td>
</tr>
</tbody>
</table>

Subject to change!
Logistics

- Guest lecturer: Cameron Browne
  - Computational Creativity Group (ccg.doc.ic.ac.uk)

- Lectures: Thursday at 4pm in room 311 and Friday at 2pm in room 145
- Tutorial immediately after the 4pm lecture on Thursdays
  - No tutorial today
  - No lecture tomorrow
- Slides on web (at start of each week). No handouts

Logistics

- No background required: AI, graphics and/or HCI would help
- Individual coursework
  - Two weeks in end February
  - Based on Ludic Agents material
- Exam
Student Feedback

The module ran for the first time in 2009-10

- “Love it!”
- “Very enjoyable”
- “Best course of this year”
- “Too much material”
- “Lacked a bit of focus”
- “Too fluffy”

Additional Reading

Context for the Module