Coordinating unit: 270 - FIB - Barcelona School of Informatics
Teaching unit: 723 - CS - Department of Computer Science
Academic year: 2019
Degree: MASTER'S DEGREE IN INNOVATION AND RESEARCH IN INFORMATICS (Syllabus 2012). (Teaching unit Optional)
ECTS credits: 6
Teaching languages: English

Prior skills

Students that take this course need to know OpenGL and have taken some computer graphics course previously.

Degree competences to which the subject contributes

Basic:
CB8. Capability to communicate their conclusions, and the knowledge and rationale underpinning these, to both skilled and unskilled public in a clear and unambiguous way.
CB9. Possession of the learning skills that enable the students to continue studying in a way that will be mainly self-directed or autonomous.

Specific:
CEE1.1. Capability to understand and know how to apply current and future technologies for the design and evaluation of interactive graphic applications in three dimensions, either when prioritizing image quality or when prioritizing interactivity and speed, and to understand the associated commitments and the reasons that cause them.

General:
CG3. Capacity for mathematical modeling, calculation and experimental designing in technology and companies engineering centers, particularly in research and innovation in all areas of Computer Science.

Transversal:
CTR5. APPROPRIATE ATTITUDE TOWARDS WORK: Capability to be motivated by professional achievement and to face new challenges, to have a broad vision of the possibilities of a career in the field of informatics engineering. Capability to be motivated by quality and continuous improvement, and to act strictly on professional development. Capability to adapt to technological or organizational changes. Capacity for working in absence of information and/or with time and/or resources constraints.
CTR6. REASONING: Capacity for critical, logical and mathematical reasoning. Capability to solve problems in their area of study. Capacity for abstraction: the capability to create and use models that reflect real situations. Capability to design and implement simple experiments, and analyze and interpret their results. Capacity for analysis, synthesis and evaluation.

Teaching methodology

This course is structured in three session types:

* T sessions (theory): presentation by the corresponding professor. The professor will ask the students to do some short exercises on the subjects covered in these sessions.

* D sessions (discussion): sessions conducted by the professor, in which some students will solve exercises or present previously distributed papers. Each student has to prepare the corresponding presentation and a supporting document, which have to sent to the course coordinator before his D session.

* L sessions (lab): in these sessions students will have to solve practical problems programmings some of the algorithms presented in the theory sessions. L sessions will start with a short lecture section.
Learning objectives of the subject

1. Using Hierarchical Geometric Models for the display of very large models.
2. Simplification algorithms for triangle meshes.
3. Visibility computation algorithms.
4. Interactive navigation in complex environments.

Study load

<table>
<thead>
<tr>
<th>Total learning time: 150h</th>
<th>Theory classes: 24h 16.00%</th>
<th>Practical classes: 12h 8.00%</th>
<th>Laboratory classes: 12h 8.00%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Guided activities: 6h 4.00%</td>
<td>Self study: 96h 64.00%</td>
<td></td>
</tr>
</tbody>
</table>
## Content

### Hierarchical geometric models

**Degree competences to which the content contributes:**

**Description:**
Algorithms for space subdivision (regular grids, octrees, BSP trees, Kd-trees), scene subdivision (BVHs) and external memory-based data structures.

### Mesh representation data structures

**Degree competences to which the content contributes:**

**Description:**
Triangle and polygonal mesh representation: Independent face set, Indexed face set, Adjacency lists, Winged edge, Half edge, Corner table.

### Simplification of triangle meshes

**Degree competences to which the content contributes:**

**Description:**
Introduction to the basic concepts, operators and error metrics used in geometry and topology-based simplification. Its application to appearance-preserving simplification and out-of-core gigantic model simplification.

### Level of detail

**Degree competences to which the content contributes:**

**Description:**
Introduction to object level of detail (LOD) and its application to complex scenes (time critical rendering). Strategies for LOD: Discrete, Continuous, or View-Dependent. Popping effect prevention.

### Visibility computation

**Degree competences to which the content contributes:**

**Description:**
Introduction to the basic concepts and algorithms for visibility computation, including visibility preprocessing, point and region visibility, and visibility computation using the GPU. PVS compression.

### Interactive navigation in complex environments

**Degree competences to which the content contributes:**
Description:
### Planning of activities

| Hierarchical Geometric Models | Hours: 45h  
Theory classes: 12h  
Practical classes: 0h  
Laboratory classes: 6h  
Guided activities: 0h  
Self study: 27h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific objectives: 1</td>
<td></td>
</tr>
</tbody>
</table>

| Simplification algorithms for triangle meshes | Hours: 30h  
Theory classes: 8h  
Practical classes: 0h  
Laboratory classes: 4h  
Guided activities: 0h  
Self study: 18h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific objectives: 2</td>
<td></td>
</tr>
</tbody>
</table>

| Visibility computation algorithms | Hours: 30h  
Theory classes: 8h  
Practical classes: 0h  
Laboratory classes: 4h  
Guided activities: 0h  
Self study: 18h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific objectives: 3</td>
<td></td>
</tr>
</tbody>
</table>

| Interactive navigation in complex environments | Hours: 30h  
Theory classes: 8h  
Practical classes: 0h  
Laboratory classes: 4h  
Guided activities: 0h  
Self study: 18h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific objectives: 4</td>
<td></td>
</tr>
</tbody>
</table>

| Paper presentation | Hours: 9h  
Guided activities: 3h  
Self study: 6h |
|--------------------|-------------------|

Hours: 45h

Theory classes: 12h

Practical classes: 0h

Laboratory classes: 6h

Guided activities: 0h

Self study: 27h

Hours: 30h

Theory classes: 8h

Practical classes: 0h

Laboratory classes: 4h

Guided activities: 0h

Self study: 18h

Hours: 30h

Theory classes: 8h

Practical classes: 0h

Laboratory classes: 4h

Guided activities: 0h

Self study: 18h

Hours: 30h

Theory classes: 8h

Practical classes: 0h

Laboratory classes: 4h

Guided activities: 0h

Self study: 18h

Hours: 30h

Theory classes: 8h

Practical classes: 0h

Laboratory classes: 4h

Guided activities: 0h

Self study: 18h

Hours: 9h

Guided activities: 3h

Self study: 6h
270674 - SRGGE - Scalable Rendering for Graphics and Game Engines

Description:
Each student has to prepare the corresponding presentation and a supporting document, which have to sent to the course coordinator before the session.

Specific objectives:
1, 2, 3, 4

Exercises

Hours: 9h
- Theory classes: 0h
- Practical classes: 0h
- Laboratory classes: 0h
- Guided activities: 0h
- Self study: 9h

Description:
Set of exercises raised during the course to assess knowledge acquisition by students during the course.

Qualification system

The final qualification is computed as:

\[ \text{FinalQualification} = 0.25 \times \text{ShortExercises} + 0.25 \times \text{DPresentation} + 0.5 \times \text{LabQualification} \]

where:

* ShortExercises represents the short problems the instructor will ask during T sessions.
* DPresentation is the presentation the students will do on a paper selected from a list.
* LabQualification will be the qualification obtained by the students in the L sessions.

Bibliography

Basic:

Complementary: