INFOGR – Computer Graphics


Welcome!
Today's Agenda:

- Topic Introduction
- Course Introduction
- Team
- Practical Details
- Assignments
- Field Study
- State of the Art
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction
Introduction

Computer Graphics 2015:

Looking for realism (in several wrong places):

1. Rasterization
   - Geometry
   - Textures, shaders
   - Clipping, culling
   - Post processing
   - ...

2. Ray tracing
   - Ray/triangle intersections
   - Bounding volume hierarchy
   - Snell, Fresnel, Beer
   - Whitted, Cook, Kajiya
   - ...

3. Mathematics
   - Vectors
   - Matrices
   - Transformations

...
Language: English, because of reasons.

Prerequisites: C#.

Literature: Fundamentals of Computer Graphics (3rd edition), by Peter Shirley and Steve Marschner (or 2nd, or 1st).

13 lectures (due to Liberation Day, Ascension Day and retakes).

Supporting practica in all lecture weeks:

- On Tuesdays,
- In BBG-112, -175, -106, -109, -103
Introduction

Supporting tutorials in all lecture weeks:

- On Thursdays
- In BBG-083, -169, -165 and -079.

Exams:

- Mid-term: May 21st.
- End of term: June 23rd.
- Retake: July 9th.

Attendance:

*You are not required to attend any of the lectures / tutorials / practica (i.e., if you are here, it’s because you want to*).

*Obviously, attendance is highly recommended.*
Introduction

http://www.cs.uu.nl/docs/vakken/gr
Course characteristics:

This is a very intensive course. Be sure to keep up, i.e. don’t miss lectures.

Be aware that this course will be attended by a diverse student population:

- Math-savvy students;
- Programming gurus;
- Game people;
- Informatics guys.

Regardless of your skill level and interests, make use of this course to improve.
Today's Agenda:

- Topic Introduction
- Course Introduction
- Team
- Practical Details
- Assignments
- Field Study
- State of the Art
Team

Lecturer:

Jacco Bikker

bikker.j@gmail.com / j.bikker@uu.nl
Office: BBL 425

Background:

Gamedev:
- Lost Boys
- Davilex
- Green Dino
- Overloaded
- Vanguard

Academia:
- IGAD

Education:
- HBO
- Doctoral (Delft; Ray Tracing in Games, 2012)
Teaching Assistants:

1. Forough Madehkhaksar
2. Coert van Gemeren
3. Anna Aljanaki
INFOGR – Lecture 1 – “Introduction”

Team

Student Assistants:

1. Tigran Gasparian
2. Jordi Vermeulen
3. Casper Schouls
4. Sander Vanheste
5. Jan Posthoorn
Today’s Agenda:

- Topic Introduction
- Course Introduction
- Team
- Practical Details
- Assignments
- Field Study
- State of the Art
INFOGR – Lecture 1 – “Introduction”

Practical Details

Assignment Overview:

i. P1: Tutorial;

ii. P2: Basic shader programming;

iii. P3a: Advanced shader programming, or:


Final practicum grade is $0.2 \times P1 + 0.4 \times P2 + 0.4 \times \max(P3a, P3b)$.

Exam overview:

i. T1: Mid-term exam;

ii. T2: Final exam.

Final exam grade is $0.5 \times T1 + 0.5 \times T2$.

Final grade: $(2T + P) / 3$

Passing criteria:

Final Grade $\geq 6.0$ (after rounding); both $T$ and $P$ $\geq 5.0$ (after rounding).
INFOGR – Lecture 1 – “Introduction”

Practical Details

How to hand in assignments:

- [http://www.cs.uu.nl/docs/submit](http://www.cs.uu.nl/docs/submit)

Retake:

- You must have submitted all programming assignments
- You must have participated in both exams
- Your total grade must be at least a 4.0 (after rounding)
- Retake covers whole course, and replaces min( T1, T2 ).
Today's Agenda:

- Topic Introduction
- Course Introduction
- Team
- Practical Details
- Assignments
- Field Study
- State of the Art
INFOGR – Lecture 1 – “Introduction”

Assignments

PART 1: Mathematics

Tutorial 1 will be available on Thursday, April 23\textsuperscript{th}.
TA assistance is available on April 30\textsuperscript{th} in rooms
BBG-083, -169, -165 and -079.

PART 2: Programming assignment

P1 (XNA tutorial) is now available from the website.
Assistance is available on Tuesday, April 28\textsuperscript{th} in rooms
BBG-112, -175, -106, -109 and -103.
Today’s Agenda:

- Topic Introduction
- Course Introduction
- Team
- Practical Details
- Assignments
- Field Study
- State of the Art
INFOGR – Lecture 1 – “Introduction”

Field Study

Field Study
INFOGR – Lecture 1 – “Introduction”

Field Study
Field Study

Early graphics:

2D, with limitations

- Tiles
- Few colors
- Sprites
Field Study
INFOGR – Lecture 2 – “Field Study”

History of Graphics
INFOGR – Lecture 2 – "Field Study"
History of Graphics

Lecture 2

“Field Study”
History of Graphics

INFOGR

–

Lecture 2

–

"Field Study"
Field Study

Game production:

Code

Art

Crysis:

> 1M lines of code; 85k shaders

Unreal 3 engine:

2M lines of code

Frostbite:

“10x Unreal 3”

Minecraft:

< 200k lines of code.
History of graphics in games, digest

Initially fast progression:

- from 2D to 3D,
- from monochrome to true-color,
- from wireframe to shaded,
- from sparse to highly detailed.

But also:

- from reasonably efficient to produce to extremely labor-intensive.
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- **Lights**
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- **Reflections**
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- **Ambient occlusion**
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- **Light shafts**
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- **Indirect lighting cache**
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- Bump mapping
State of the Art

Industry example: Unreal Engine 4

- Lights
- Shadows
- Reflections
- Ambient occlusion
- Light shafts
- Indirect lighting cache
- Ray traced soft shadows
- **Bump mapping**
State of the Art

Modern rendering in games:

Stacking algorithms that solve part of the problem:

- Shadows
- Reflections
- Participating media
- Indirect light

Designed to ‘look good’, not to be (necessarily) correct

Each partial solution comes with parameters and limitations

But: well-suited for today’s hardware.
Next week:

Foundation
INFOGR – Computer Graphics


END of “Introduction”

next lecture: “Graphics Fundamentals”