

COMS30020 - Computer Graphics

Introduction Briefing

Dr Simon Lock

Welcome to Computer Graphics

I'm Dr Simon Lock

You might remember me
from units such as SPE !



We also have our team of TAs !

Aim of unit: Introduce "Computer Graphics" !

This term means different things to different people

Our perspective on the area will encompass:

- Low-level "individual pixel" manipulation
- Drawing primitives (lines and triangles)
- 2D and 3D geometry (numerical position data)
- *Approximating* behaviour of light
- Camera views, movement and navigation

As you might have sensed, it's all pretty low-level...
This is definitely NOT a "Graphic Design" unit !

A Word of Warning !

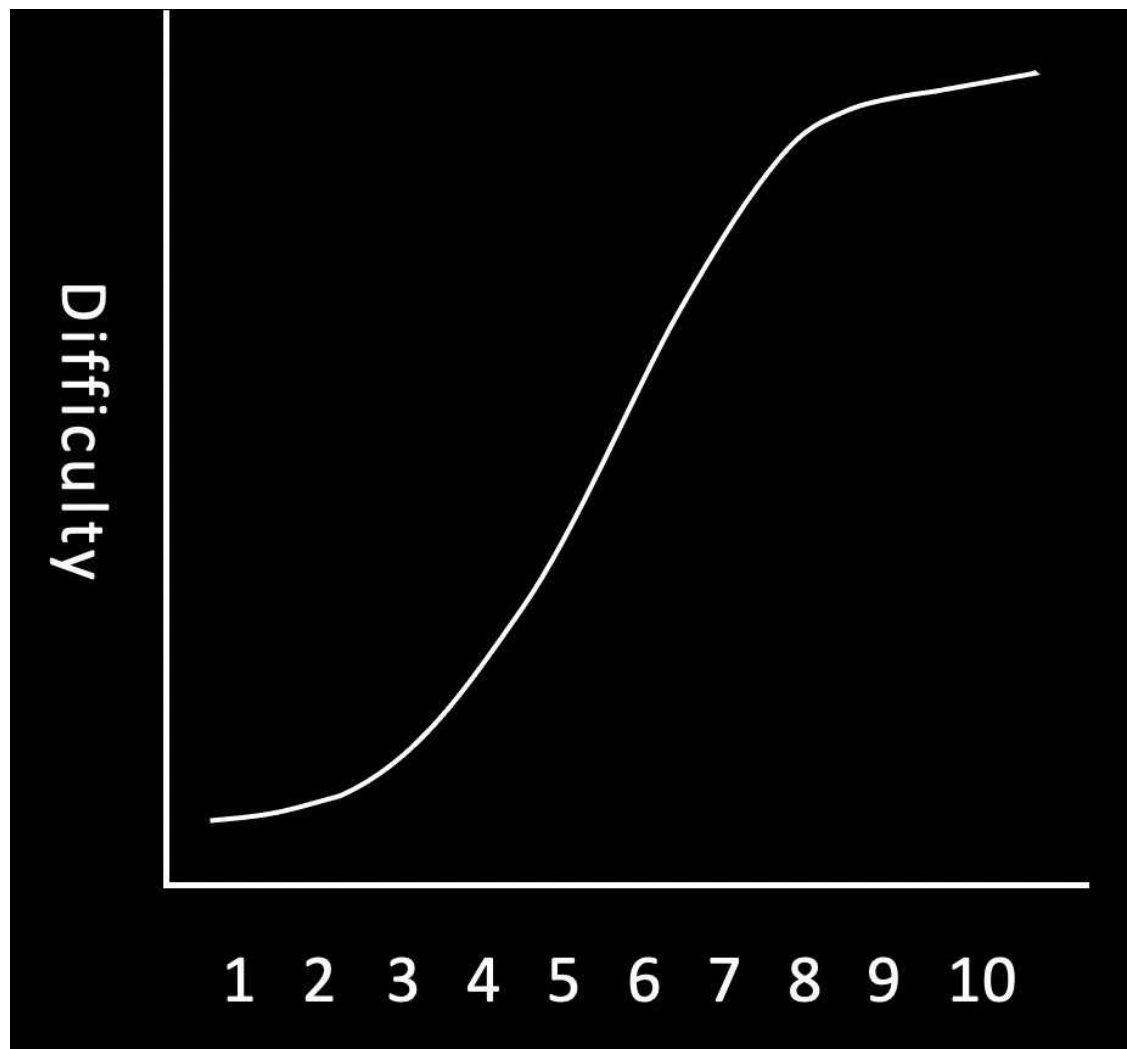
This unit starts off slowly and very gently :o)
(to make sure everyone gets off to a good start)

But it will speed up later on (week 3 onwards)

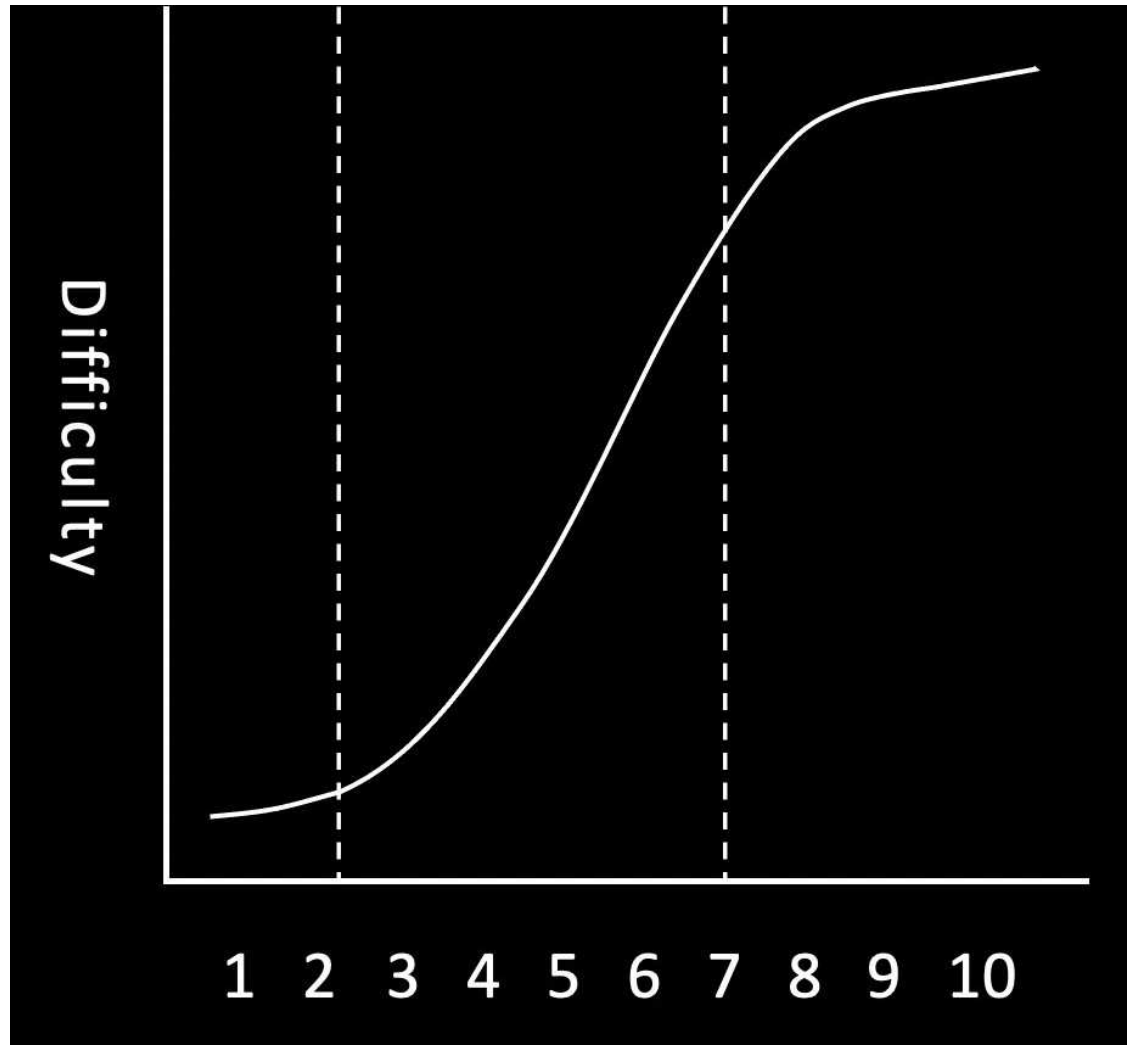
Watch out you don't get left behind !
Don't take your eye off the ball !!

It is not particularly hard to pass the unit
But there are a bunch of fails/resits every year !!!

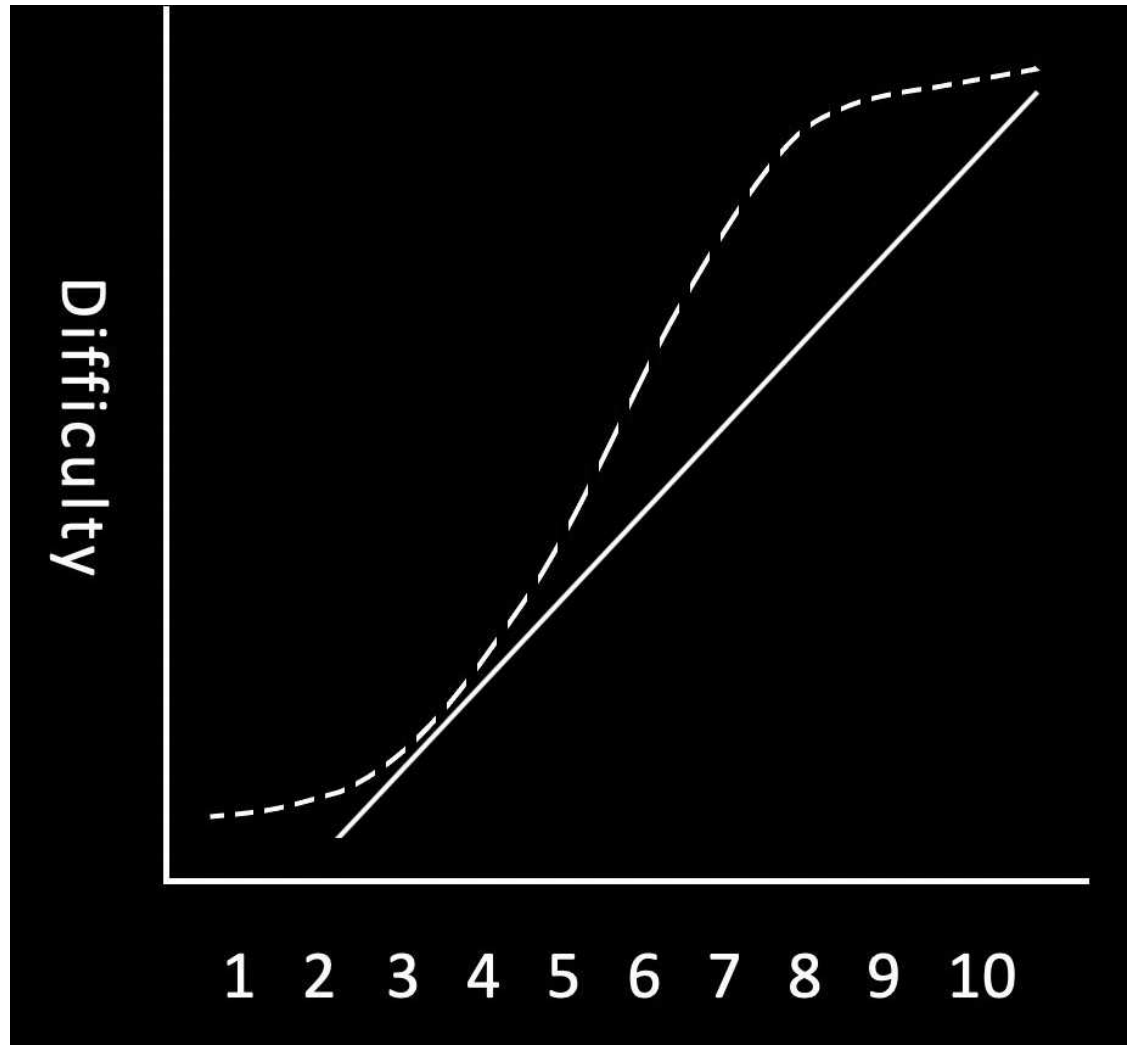
Learning Curve



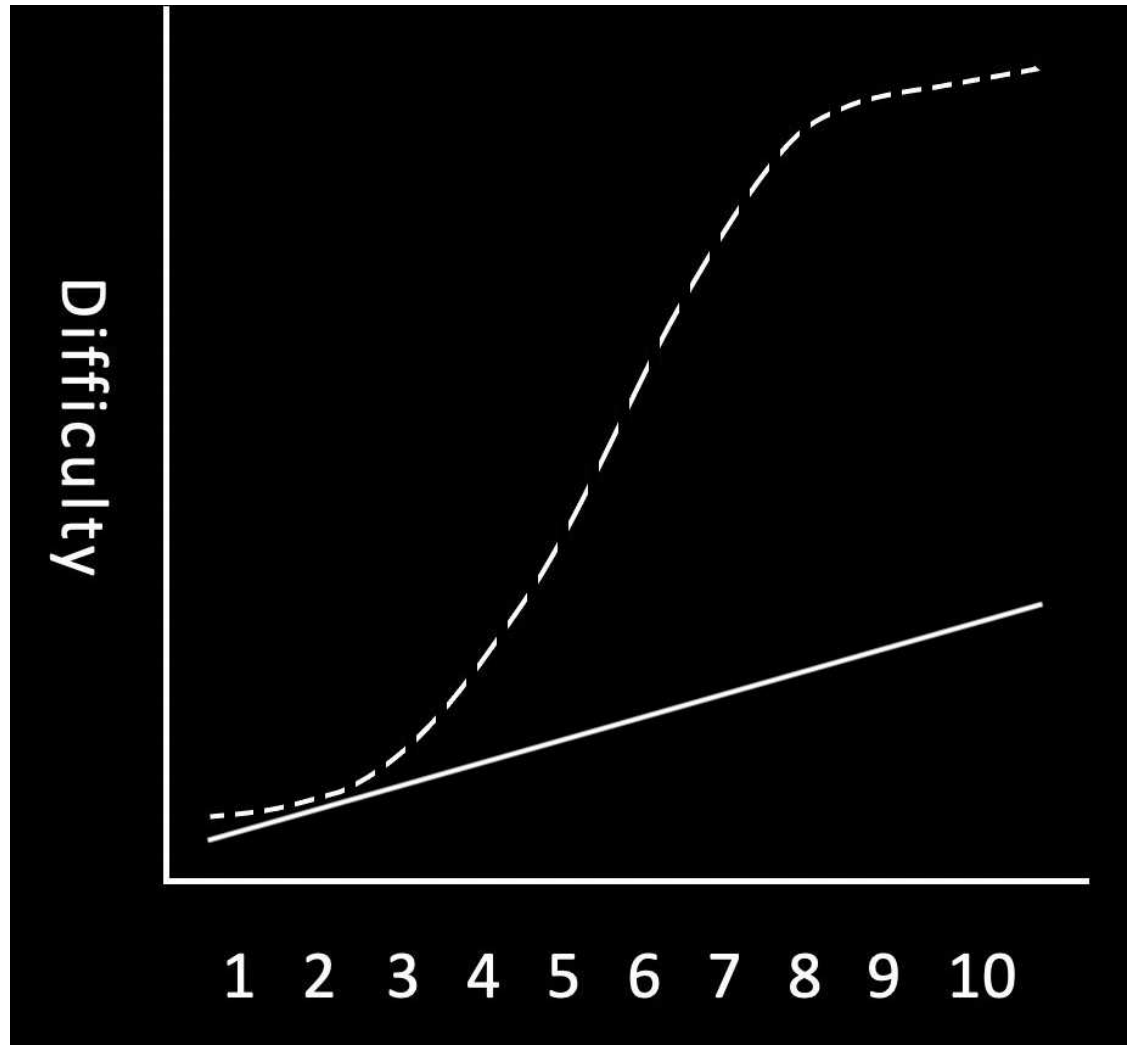
Key Weeks



What people usually do



What NOT to do



Weekly Activities

- Each week we'll release a "workbook" of activities
More on this in some later slides !
- Practical session to support completing workbook
Monday 1-3pm in MVB 2.11 (this session !)
- Lecture briefing to introduce following week's topics
Friday at 3pm in Queens Building 1.15 (SLT)
- Q&A opportunity (at the end of the briefing)
- Asynchronous support via Teams discussion forum

Weekly Workbooks

Each workbook contains a set of tasks to complete
Lead you step-by-step towards a practical end-goal

Key concepts introduced with rich-media materials:

- Written slides and descriptions
- Recorded audio narration
- 2D and 3D images and diagrams
- Graphical animations (using 3D to teach 3D !)

Workbooks are made available via GitHub
(For ease of upload and download !)

Importance of Weekly Workbooks

It is ESSENTIAL that you keep up with practical tasks
Even if you are taking the "exam only" variant of unit
Workbooks are backbone of the teaching materials
They are an integrated bundle of content & activities

Blackboard

We will use Blackboard as a set of bookmarks
Pointing to other platforms (GitHub, Teams etc.)

Note that there are 3 pages for this unit !
("Teaching" page, Coursework Page, Exam Page)
They generally have identical information on them
(apart from the "Assessment" pages !)

Your own GitHub repositories

In the workbooks you will be asked to use GitHub
Ensure your code is managed throughout the unit

This allows us to monitor everyones progress
Spot any problems or anomalies along the way

Your repo will feed into 3 week coursework period
Will be used as part of the final marking process

You won't need to use all the features of GitHub
Just be able to create repositories, push and pull
Ask us in the lab if you need help !

Access to your repositories

In order to stop other people stealing your code...
You must be sure to keep your repository "private" !

In order for markers to be able to access your code
You need to invite this user to your repository:

sl17668

You'll also need to register your GitHub username:
<https://forms.office.com/r/mydejZ3X9z>



More detail on unit content ?

Shared Language ?

Various types of "language" are at our disposal:

- Natural Language (written & spoken)
- Mathematical Notation
- Algorithmic Expression
- Graphical Representation

We'll be using ALL of these to discuss key concepts
We need them - some things are tricky to explain !

Implementation

We will be using C++ for implementation
(Pretty much the standard for low-level graphics)

For drawing to the screen, we will use "SDL2"
Platform independent graphics library, used to:

- Create windows to show on the screen
- Manipulate individual screen pixels
- Allow user interaction via keys and mouse

Everything else YOU will build on top of this !
(Although we'll use some Maths functions from GLM)

Suggested Textbooks

Basic introduction to topics covered in unit:

Computer Graphics from Scratch

by Gabriel Gambetta

More detailed coverage and advanced rendering:

<https://www.scratchapixel.com/>

You might need to do some additional reading
(especially for coursework extended topics)

Questions ?

Why not teach an existing framework ?

One difficulty is choosing which one to teach !

DirectX, OpenGL, Vulkan or proprietary framework ?

Better to teach the fundamental concepts
(Which make it easy to pick up any framework)

Besides, "fundamentals" are much more "BSc/MEng"
We don't like to focus a unit around a single API

Can I use <insert_language>

Soz, no !

C++ is a well-established standard in the area

All templates/examples are written in C++

Teaching assistants are all skilled in C++

It's hard to mark an unfamiliar language

Need to maintain a level playing field !

Aim of the first workbook

The aim of the first practical session is simple:

To compile and run the "RedNoise" project
(the base template for all practical exercises)

The deeper purpose is more serious:

Find a "workable" way to compile and run SDL code
(native OS / virt machine / lab machine / remote login)

Let's take look at that first workbook...

<https://github.com/COMS30020/CG2023>

Extra Slides

Nature of Coursework

Your task will be to create a short 3D animation

This is NOT like the CGI unit

(Where it is all about the modelling and design)

This assignment will be much more low-level...

You will NOT be using an existing application

You will build your OWN rendering engine !

Aim of animation: show off features of YOUR engine

More about the assignment when we get to week 8

Nature of Exam

The exam will be fairly mathematical in nature
Applying principles of Computer Graphics "on paper"

Essential to complete workbooks and practical task
(to gain deep understanding of all the techniques)

There won't be a big TA resource during revision
So don't just leave the workbooks until the end !