LECTURE 1
WHAT IS INTERACTIVE AND TANGIBLE MEDIA?

EL542 Creativity in interactive and tangible media
Dr Jim Ang

https://www.youtube.com/watch?v=30yGOuJJ2PQ
Quiz

- Please select all the following examples which can be considered as “interactive media”
  a. Video games
  b. 3D animated films
  c. Augmented reality
  d. Pop-up books
  e. Social media

Interactive media

- Interactive media is the integration of digital media including combinations of electronic **text**, **graphics**, **moving images**, and **sound**, into a structured digital computerised environment that allows people to **interact with the data** for appropriate **purposes**. ([ATSF White Paper—Interactive Media UK 2011](#))
- It is a multidisciplinary field, involving concepts from new media, interaction design, digital culture, ambient intelligence, virtual reality, social media, etc.
- Interactive media does not have to be digital. For instance, board games, pop-up books, flipbooks are examples of printed interactive media

A pop-up book

A flipbook
Examples of interactive media

- The Treachery Sanctuary – an art project that turns you into birds
- 3 display panels: i) spread your arms that turn into wings, ii) birds pecking at your body, iii) your body disintegrating into birds
- It makes use of the Kinect sensor

Tangible media

- Extending beyond the limitations of the computer mouse and keyboard, tangible user interfaces allow users to interact with digital information through grasping and manipulating physical objects, and through gestures.
- By allowing users to draw on their natural skills for interacting with digital information, tangible user interface could offer an intuitive and creative interface to support activities such as learning, problem solving, design and entertainment.
- For simplicity, we define tangible media as digital media that makes use of novel human-computer interaction style that interlinks the physical and virtual worlds.
Quiz

• Apart from using the keyboard and the mouse, name at least three different ways how we can interact with digital contents?

- Touch
- Gesture
- Speech
- Brainwave
- Eye gaze
- Skeletal Muscle (EMG)

Examples of tangible media

• inFORM: created by MIT Media Lab, demonstrating the a display made of physical blocks, rather than virtual pixels
• It also shows how the virtual can be linked more closely with the physical

http://www.youtube.com/watch?v=lvtfD_r2HIE
Examples of tangible media

- Electronic Popables: created by a MIT student, it is an interactive paper-based interfaces powered through a (lilypad) arduino
- It makes use of tangible interface through various sensors such as pressure sensor, skin conductance sensors.

http://www.youtube.com/watch?v=AI-6wMlaVF0

Arduino

- Arduino is an open source microcontroller, intended to make the application of interactive objects or environments easier.
- The hardware is relatively small and cheap (about £20)
- It can be used to develop interactive objects, and can also be connected to software such as Flash and Processing
- Most importantly, it is developed specifically for artists and designers! It is based on an easy-to-use development environment
Sensors, actuators and others...

- We can connect a range of sensors and actuators to Arduino, to create our interactive project. Let’s see how many components you can name:

![Sensors and actuators diagram]

About this module

- Art + science = creative computation
- We will explore computation as a fundamental of creative medium
- Computer code for digital artists/designers is the same as paint for a traditional painters
- We will emphasise on aesthetics and personal expression with fundamental principles in programming
- Note that the core of this module is NOT programming per se, but how we can produce creative media using programming as a tool
- We assume absolutely no background knowledge in microcontroller programming
What will you learn?

• You will create beautiful visualisation and realistic motions
• You will also design interactive electronic objects for your creative project
• You will develop cool interface techniques using motion sensing, speech, and other sensors
• Some projects you will do:
  • Interactive pointillist art
  • Speech-controlled light
  • Home-made gamepad
  • Super ultrasonic flappy bird!
  • And many more!

Processing

• In 1996, the Aesthetics + Computation Group at MIT developed a new programming environment named “Design by Numbers,” a simplified Java language that enables fast prototyping of graphics patterns, art and design.
• This idea was then further developed by two MIT students (Ben Fry and Casey Reas) into a new tool called Processing (2001).
• Processing has an easy-to-use IDE (integrated development environment) and a strong focus on graphics and multimedia
• Like Arduino, the goal is to create a zero-entry approach to coding for artists and designers
Processing + Hardware

Physical world ➔ Hardware/sensors ➔ Processing ➔ Virtual world

Processing Programming environment

Editor: this is where you type your code
Run and stop the program
New, open, save, export app.
Debug messages
Your first Processing code – drawing with numbers

• Just type in the editor the following and hit the “run” button:

1. void setup() {
2.   size(480, 120);
3.   ellipse(120, 50, 100, 80);
4. }

The setup() function is called once when the program starts.

Set the size of the window
Draw an ellipse

Sketch 140130a
(0,0) 120 pixels
100 pixels
50 pixels
80 pixels
120 pixels
480 pixels

Quiz

• What is the likely outcome of the following code:

1. void setup() {
2.   size(120, 120);
3.   ellipse(120/2, 120/2, 80, 80);
4.   ellipse(120/2, 120/2, 60, 30);
5.   ellipse(120/2, 120/2, 30, 60);
6. }

• Please rearrange line 3-5 to produce graphics b, c and d.
Drawing with numbers

- Let’s now take a look at the void draw() function.

```cpp
1. void setup(){
2. size(500,300);
3. }
4. int x = 0;
5. void draw(){
6. ellipse(x,150, 30, 30);
7. x = x + 30;
8. }
```

- Create a variable called `x`.
- the draw() function continuously executes the lines of code contained inside its block until the program is stopped.
- Draw an ellipse at the location (x, 150).
- Add 30 to `x`.

Draw an ellipse at the location (x, 150)

If logic

- In the previous code, the program will keep drawing circles as long as the program is running, even the circles are drawn outside the window. How can we address this?

```cpp
1. void setup(){
2. size(500,300);
3. }
4. int x = 0;
5. void draw(){
6. if (x < 250){
7. ellipse(x,150, 30, 30);
8. x = x + 30;
9. }
10. }
```

- The “if statement” check if `x` is still smaller than 250 (half the window width). If it’s true (x<250), keep drawing. If it is false, stop drawing.

- Now, how do we make sure the circles fill up till the end of the screen and stop there?

~250 pixels
Ellipse interactive media

- Even with only ellipses, we can create amazing interactive media project

http://www.youtube.com/watch?v=raWfLyellHs

Other shapes

angle_in_radians = angle_in_degrees * PI/180;
angle_in_radians = radians(angle_in_degrees);
Most importantly…. Creativity!

- I couldn’t have stressed enough: the module is NOT about programming per se!
- It is about expressing yourself aesthetically, and creatively; it is about manipulating colours, lines, and shapes with numbers to create nice visualisation; it is also about linking the digital world with the physical world to create innovative interactive projects
- And we do this with programming as a tool!

Module structure

- Week 25, day 1: Interactive and tangible media
  - Week 25, day 2: Algorithmic graphics
- Week 26, day 1: Mixed reality
  - Week 26, day 2: Motions
- Week 27, day 1: Tangible user interfaces
- Week 27, day 2: Revision lecture + class quiz
- Week 28, day 1: Environment sensing
- Week 28, day 2: Complexity
- Week 29, day 1: Advanced topics
- Week 29, day 2: Project week (no lecture)
How would you do it?

- If you have only ONE drawing function at hand, pixel(x,y) which draws one single point (pixel) on the screen at (x,y), how do you draw a circle?
- Without using ellipse()!

![A pixel](image1.png)  ![A circle](image2.png)

Workshop

- In this workshop, you will learn a number of basic Processing topics:
  - Drawing shapes
  - Using the mouse to create interactivity
  - Using the if statement
  - Understanding some advanced techniques in drawing such as using filter() and translate()
- Note that each workshop carries 5% of your overall mark.
- Enjoy drawing with numbers!
HOME EXERCISES

You are highly encouraged to do all these exercises at home (even if you just spend 5-10 min thinking about each exercise if you have no time!)

Questions

- Give 3 examples of tangible interfaces
- List down 2 application areas where tangible interfaces can be used to improve user experience
- List down at least 3 sources of information one can draw from the physical world to create interactive media
Design

• Please refer to slide 21, “Ellipse interactive media.”
• Can you re-design this work so that users can interact with the ellipses using interaction mechanism other than the mouse?
  • Can you use an tangible interface?
  • What other forms of visualisations would you include apart from ellipses to create a more engaging experience?
• For this exercise, don’t worry about the technicality of implementation. Let you imagination run wild with design! Design your work on a piece of paper.

Design

• Conceptually, try to think of how you can design the following work with code. What techniques would you use? Can you randomly generate rectangles of various sizes at random locations on screen to achieve this?

Composition with Gray and Light Brown by Piet Mondrian 1918
Composition in Red, Yellow, and Blue (1926)