

sTeAm – A bald man in red, green and blue

A **Damian Perry** lesson on computers and colour

The idea of this article is to show how we can extend Art into the STEAM domain. I will use the example lesson from my Media class to give students a better understanding of how image manipulation software represents colours for digital portfolios. And then we will broaden that knowledge into Digital Technologies with an exploration of how computers store that data in bits and bytes.

Up until I started at St James, I used to participate in Shave for a Cure every year. Unfortunately, Shave for a Cure was always held just before school photos. So, I have many school photos with me bald as a cue ball. But they make good example photos for what we're doing here.

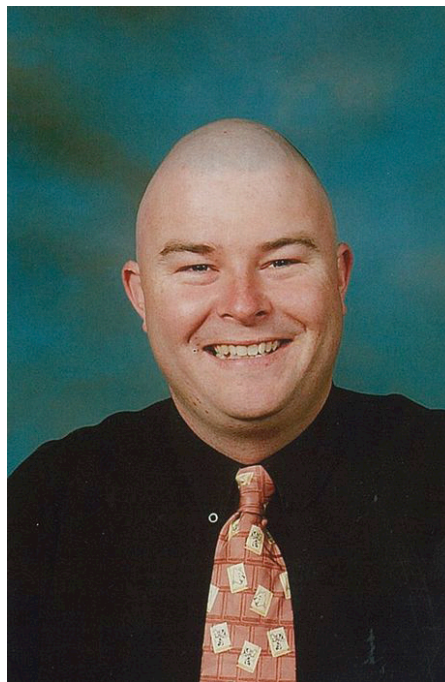
Recipe list

- Images – use your own or nab the ones from this article.
- Photo manipulation software – most schools have access to Photoshop these days.
- 10x10 grid printed onto paper for a class
- A piece of paper – half black, half white, folded in half.

If you want to get really tricky, you could print the red, green and blue images in red, green and blue onto overhead projector film for a practical demonstration.

Get right in – the Arts side

Teaching photography and image manipulation to my Year Nine Media class and film editing to my Year Tens, along with Web Design in DigiTech, requires all sorts of knowledge about how computers represent colour in software and on the web. The easy way to do this is to show them Hexadecimal codes and let them run with it. To achieve deeper learning and cross curricular exploration, let's take it a bit further.



Start with a picture. Explain the basics of how a picture works. There are so many pixels. The pixels are per-inch (dpi = dots per inch) to represent how they will look when printed. Each pixel (picture element) has a value – or rather a number of values. We're interested in the colours.

Discuss with your students about your audience. Is your product for screen or print?

Because the computer deals with images for the screen by starting with a *black* screen and then adding light. We start at zero red, zero green and zero blue and then light up the three elements until we get the right colour. All zero is black. All 255 is white.

Conversely, when printing, we start with a blank *white* page and add colours until we hit black. Zero Cyan (blue), zero Magenta (red), zero Yellow (...yellow) and zero black (to make CMYK).

I'm going to stick with RGB for my

demo because of the links to web design.

I have split my wonderful bald skull into the component colours in Figures 1 to 3.

Discuss the information shown in Photoshop.

- RGB is a colour value from 0 to 255
- The number next to the hashtag is Hexadecimal
- CMYK is a percentage (because of the print background).

From here I want to slip sideways into Digital Technologies. But first I will conclude on the reasons why this is important information for Media and Visual Art classes.

The first is that there are too few teachers that know the reasons behind a computer's reasons for doing things.

The second is that websites, tutorials, and the software packages themselves all refer to RGB and CMYK and hexadecimal values constantly, often without providing any background. Having the students able to visualise the different layers that make up each image and understanding what a pixel is and how it relates to printing and the screen means fewer 400x400 pixel pictures printed out in A4 in a final assessment.

But most importantly, when evaluating photographs, being able to judge and talk about the levels of different colour in the picture is important. In Photoshop you can hide different layers of colour to get different effects or hide all of the colour to explore how effective the picture is in its use of tone.

Get right in – the DigiTech side

Notice how RGB goes from 0 to 255? And how HTML colours are represented in Hexadecimal?

Or how students create 8-bit art because it's "retro"?

They will never know the pain of loading games into a computer with 64 bytes of memory (the Commodore 64) or the squeal of a 28.8k modem, where you get half way through downloading a website before your sister picks up the phone and you lose the lot.

Retro. Pshhhh.

Anyway, the point is, none of these numbers are random. They're all designed to work with a computer that thinks and works in *binary*.

I'd start by listing a heap of numbers on the screen – maybe using Excel.

2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096.

See how many of those numbers are familiar to them (32- and 64-bit architecture, 1024 x 768 screen resolutions, 2048 computer game, 4096 is a 4K tv's actual resolution).

Then present them with the same list next to a list of binary numbers

The computer takes every piece of information as either a zero or a one, called a bit (stands for **B**inary **d**igIT). Every other number has to be converted into binary to make sense to a computer.

So, for efficiency's sake, computer engineers have to set limitations on how much information can be transferred when translating text and colours and sounds etc.

The English language can squeeze into 4 bits of information (ASCII), and the Red, Green and Blue elements of any colour are represented as six *hexadecimal* numbers – 2 red, 2 green, 2 blue.

2 hex numbers represent a byte of data (0000 0000 to 1111 1111) which is quite a bit of information squished into two bits.

Before we move on consider this:

- Half a byte is a nibble! Computer programmers are funnier because they don't get out much.
- Is there a reason why The Simpsons number system isn't base 8?
- I was blown away when I realised that the computer power button symbol was a binary, 1 and 0 to represent on and off.

And a practical exercise to help it sink in

To link the theory and the practical, the DigiTech 1 and the Arts side of things, try this:

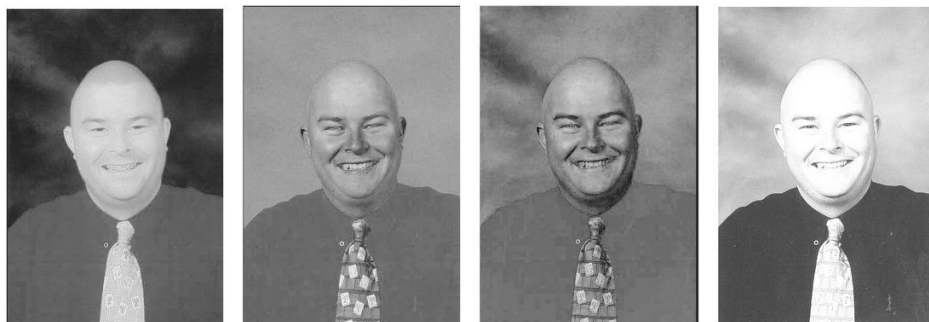


Figure 1: CMYK



Figure 2: RGB



Figure 3: Looking at colours in Photoshop

Print two class sets of 10x10 grids (I just print them from the Internet). And a bit of paper that is half white and half black, folded in half and printed enough times for half the class.

Have each of the students privately create a logo or symbol on their grid in black and white.

Break the class in half and have them pair up with someone opposite them in the room.

One of the pairs will transmit their image by holding up either the black or white side of the card over and over.

The other person will copy this information onto an empty grid.

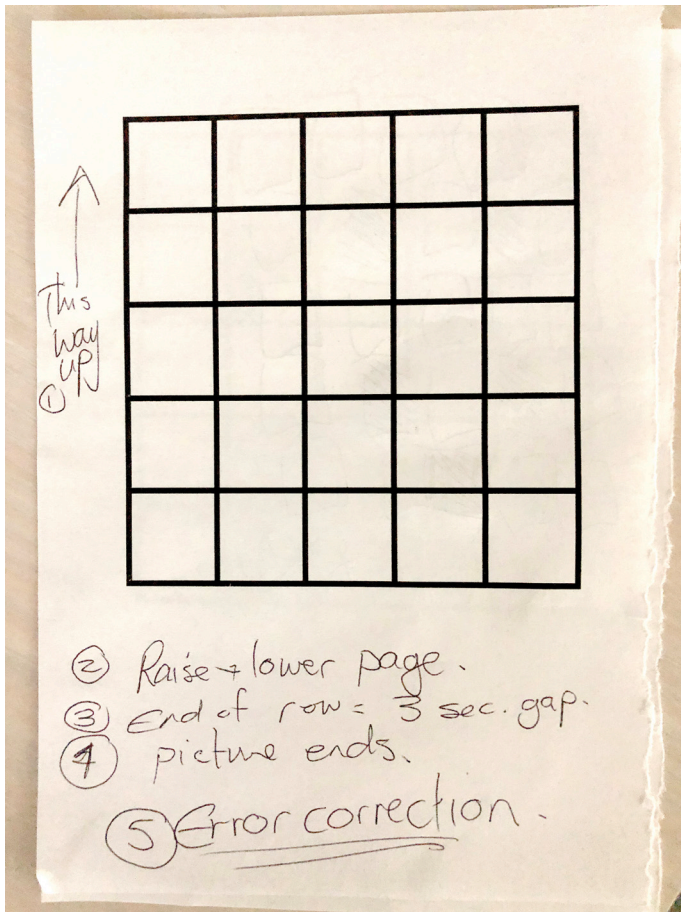


Figure 4: We agreed on some standards before we began

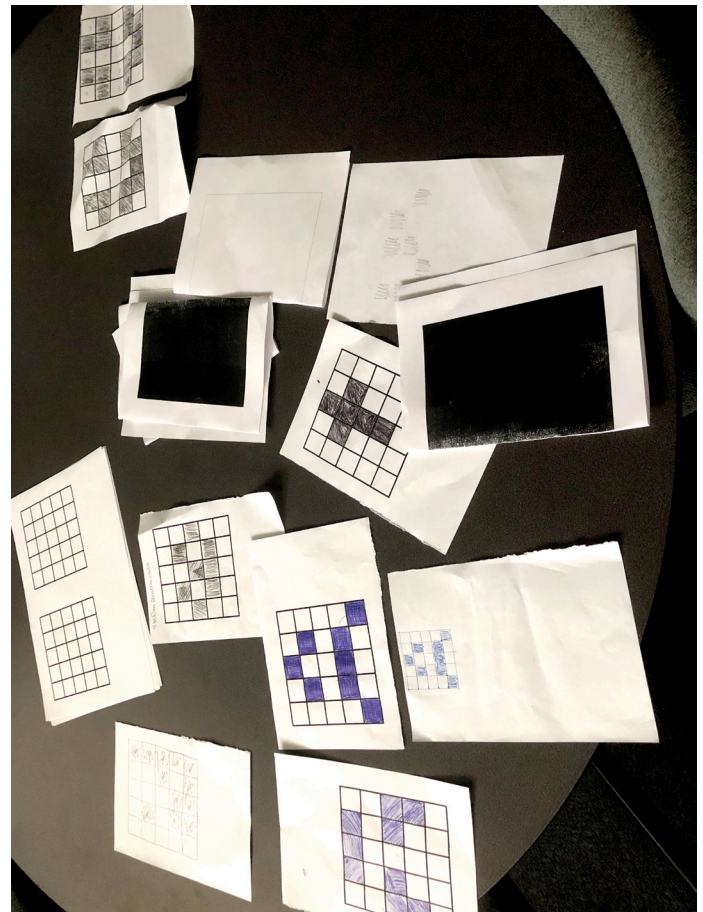


Figure 5: Some examples

At the end of the exercise, several issues pop up that can be turned into discussion or research into how the computer solves these problems:

- 1 How does the computer know when to head down to the next line in the grid? (discussion of standards and operating systems)
- 2 In a 10 x 10 grid, there's a good chance that they stuff things up. How does the computer handle error correction? (discussion or research into packets of data and synchronous/asynchronous communication)
- 3 And of course, the fact that this is just in black and white. Here is where you can discuss alternate forms of transmitting data and why Hexadecimal can be useful.

Conclusion

All of this can be simplified for earlier year levels. I use the binary activities with the Year 7s and then look at colours using background and text in HTML. The data transmission exercise can be simplified with smaller grids, or extended with a colour cube.

The important thing here is to know where we're coming from. Learn something new every day and then pass it onto your students.

And always check the calendar for photo day before agreeing to shave your head.

If you do any of these activities, I would love it if you let me know with images or video. Get onto me at @puterhead.

Curriculum standards

Vic Curric

VCAMAM043 – *Media Arts / Levels 9 and 10 / Media Arts Practices* Plan, structure and design media artworks for a range of purposes that challenge the expectations of specific audiences by particular use of media elements, technologies and production processes.

Visual Arts / Levels 9 and 10 / Visual Arts Practices Select and manipulate materials, techniques, and technologies and processes in a range of art forms to express ideas, concepts and themes

Aus Curric

ADAVAM126 – *Visual Arts / Years 9 and 10*

Manipulate materials, techniques, technologies and processes to develop and represent their own artistic intentions (specifically here, use of variations in tone and colour)

ACTDIK035 – *Digital Technologies / Years 9 and 10*

Analyse simple compression of data and how content data are separated from presentation

Resources

Use this website to quickly and easily introduce the concepts of binary and hexadecimal in class.

Look online for an explanation of Murray's shutter system codes – the basis for Terry Pratchett's Clacks (the Internet for the Discworld). GNU Terry Pratchett.