Cody Dowell

Film Project Notes

“A Theory Based on Conversion of RGB image to Gray image”

<https://www.researchgate.net/profile/Karun_Verma/publication/46286639_A_Theory_Based_on_Conversion_of_RGB_image_to_Gray_image/links/5704a3b008ae44d70ee0662c.pdf>

NOTES:

* Color in image processing is motivated by two factors
  + Color is a powerful descriptor that simplifies object identification and extraction from a scene
  + Humans can discern thousands of color shades/intensities
* In RGB each color appears in its components of red, green and blue
* RGB based of off Cartesian coordinate system
* Images represented in RGB consist of three component images, one for each primary, which then combine on screen to make composite colored image
* Pixel depth is number of bits used to represent each pixel
  + If 8 bit image, then each RGB color pixel has depth of 24
* True color images are image where each pixel is specified by 3 values, one for each primary
  + These values define the intensity of each primary for that pixel
  + Intensity is normally stored using 8 bits, from 0-255
  + This means that a pixel in a color image needs 24 bits to store

CONCLUSIONS:

* If each primary value has 256 possible values, that means 16,777,216 possible values can be displayed in the normal RGB system. If we expand this to a 6P system, ~280 trillion unique colors could technically be displayed. If hardware that can actually display this wider range of colors it utilized, then the amount of color options available will increase dramatically.
* Right now, the RGB system is based on the Cartesian coordinate system, and most likely in 3 dimensions, due to RGB being composed of 3 values. This does not seem possible for the 6P color system as this would require a 6-dimensional coordinate system, which simply does not exist, as far as I know. I see two different solutions to this problem:
  + Using the 3D coordinate system, making one point for the RGB values, another for the CMY values, and then overlapping and/or combining them in some way.
  + Moving away from the Cartesian coordinate system entirely to something else.
* In the RGB system, 24 bits are needed to store a single pixel. So it is safe to assume that in the 6P system, 48 bits are going to be needed to store a single pixel. Because of this, most images created or displayed in the 6P color system will need to be double the size of a normal RGB image, in terms of memory. The computational power needed to render an image in a 6P format will also probably rise, but its too early to tell if that will turn out to be a problem or not.

“Computational Color”

<http://printingcode.runemadsen.com/lecture-color/>

This contains basic examples for rune.js