

Class 11: Transforming Colors

Held: Wednesday, February 10, 2010

Summary: Today we explore common ways in which to transform colors, pixels, and images.

Related Pages:

- EBoard.
- Lab: Transforming RGB Colors.
- Reading: Transforming RGB Colors.
- Due: Assignment 3: Drawing Generally and Concisely.

Notes:

- Reading for Friday: Transforming Images. Note that the reading is on the schedule for Monday, but I'd prefer you to do it for Friday in case Mr. Kluber is unable to make it to class.
- Exam 1 distributed.

Overview:

- Review: Color basics.
- Computing new colors from old.
- An example.

Review: RGB Colors

- A standard way to represent transmissive colors.
- Think of it as a combination of three colored lights of varying intensities
 - Red
 - Green
 - Blue
- Why these colors? It just works that way.
- How do we measure intensities?
 - Some people like a 0-1 scale
 - Computers like a 0-255 scale (powers of two rock!)
- Important procedures:
 - `(rgb-new red green blue)` - create a new color
 - `(image-set-pixel! image col row color)` - set the color of a pixel
 - `(image-get-pixel image col row)` - get the color of a pixel
 - `(rgb-red color)` - get the red component
 - `(rgb-green color)` - get the green component
 - `(rgb-blue color)` - get the blue component

- `(color->rgb-list color)` - get a list of the three components
- `(color->rgb color)` - convert some representations to the RGB representation.

Transforming Colors

- Today we are exploring ways to transform colors.
- Why?
 - Deepens our understanding of colors.
 - We can apply a color transformation to a pixel in the image.
 - We can apply a color transformation to each pixel in an image. (More on that tomorrow.)

An Example

- MediaScript comes with a variety of built-in color transformations.
- However, we can also define our own (yay!).
- The normal framework for a color transformation is

```
(define color-transform
  (lambda (color)
    (rgb-new ___ ; computation of red component
             ___ ; computation of green component
             ___ ; computation of blue component
            )))
```

- Let's consider a simple one: We'll keep only the blue component of an image.
- How would we write this?

```
(define only-blue
  (lambda (color)
    (rgb-new ___
             ___
             ___)))
```

- Some tests
- While we'll spend today using this idea with pixels, for this example, we'll explore the effect on a sample image.

Copyright © 2007-10 Janet Davis, Matthew Kluber, Samuel A. Rebelsky, and Jerod Weinman. (Selected materials copyright by John David Stone and Henry Walker and used by permission.) This material is based upon work partially supported by the National Science Foundation under Grant No. CCLI-0633090. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.