

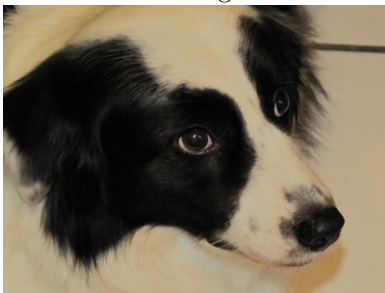
CMSC 117: Introduction to Computing: Interactive Systems

Software Mirror

Due by Class April 21, 2016

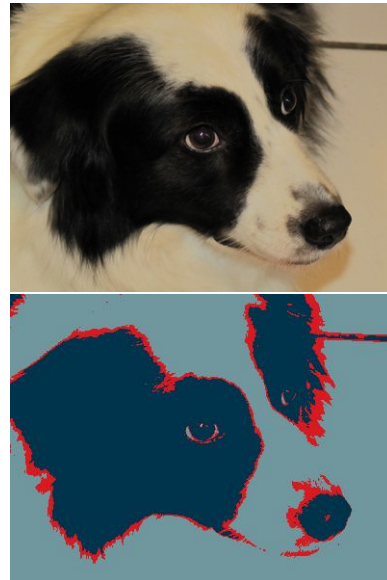
The next assignment asks you to create a series of effects for images, similar to Apple's Photobooth application. This "software mirror" sketch should load an image from the web cam (see last page), and then based on the user's input, the image should be transformed by at different effects. You should start with the code we used in class (see last page) and trigger your effects when the user presses the specified key. Each effect (including those provided) should be fully explained in comments. All the effects should operate at the pixel level and not use `filter`. Some examples:

1. **brighten** - 'b' - adds a constant amount to each color channel, or scales each color by some factor greater than 1.0.
2. **dim** - 'd' - subtract a constant amount to each color channel, or scales each color by some factor less than 1.0.
3. **swap** - 'w' - swaps the red, green, and blue channels.
4. **gray** - 'g' - turn the image into a gray scale by average the red, green, and blue channels.
5. **threshold** - 't' - turn the image into a binary, black-and-white image using a conditional statement.
6. **sepia** - 's' - applies a sepia effect, which raises the red and green channels, and lowers the blue. In particular it adds $2 * \text{sepiaAmount}$ to the red, adds `sepiaAmount` to green, and subtracts `sepiaAmount` from blue. 20 is a good value for `sepiaAmount`.



7. **fairey** – 'f' – applies an effect similar to Shepard Fairey's iconic Obama "HOPE" poster. Each pixel is colored one of four colors depending upon the sum of RGB values. It assigns roughly equal intervals for each of the four colors.

rgb sum	Color
0 – 181	darkBlue (0, 51, 76)
182 – 363	red (217, 26, 33)
364 – 545	lightBlue (112, 150, 158)
545 – 765	yellow (252, 227, 166)



8. **your effect** – 'y' – an image effect of your choice.

Learning Objectives

- Use for loops.
- Analyze images at the pixel level.
- Work with image coordinate systems.

Deliverable

Submit a link to your sketch via moodle. Include a comment at the top including your name, date, assignment, collaboration statement, and description of the custom effect your created.

<https://jsbin.com/ranira/>

```

var video;

// find pixels array index based on (x,y)
function findIndex(x, y){
// for retina displays: http://p5js.org/reference/#/p5/pixels[]
  return 4 * (y * width + x);
}

function setup() {
  createCanvas(640, 480);
  video = createCapture();
  video.hide();
}

function draw() {
  image(video, 0, 0, width, height);
  loadPixels();
  for (var x = 0; x < width; x = x + 1) {
    for (var y = 0; y < height; y = y + 1) {

      var i = findIndex(x,y);

      var r = pixels[i + 0]; // get red channel of pixel
      var g = pixels[i + 1]; // get green channel of pixel
      var b = pixels[i + 2]; // get blue channel of pixel

      if (key === "M") {
        var ni = findIndex(width - 1 - x, y);
        r = pixels[ni + 0];
        g = pixels[ni + 1];
        b = pixels[ni + 2];
      } else if (key === "I") {
        r = 255 - r;
        g = 255 - g;
        b = 255 - b;
      } else if (key === "R") {
        r = 255;
      }

      pixels[i + 0] = r; // set red
      pixels[i + 1] = g; // set green
      pixels[i + 2] = b; // set blue
    }
  }
  updatePixels();
}

```