

Assignment 4 – Color

This assignment emphasizes techniques used when working with color. It also requires the use of a tool (Bink & Smacker on a Windows machine, iMovie on a Macintosh, or some other suitable program) to convert a sequence of images into a video.

Color Processing

- Design and implement two algorithms that read a color image (24-bit) and slowly fade it to a gray scale image
 - The first algorithm should start from the RGB color space image. Read the image into RGB space and fade it to gray scale within the RGB color space. Note that you will have to convert the input RGB space image to YPbPr space and discard the Pb and Pr components to create your “target” image but your calculations for the actual fade must be in RGB space.
 - Save intermediate images then combine to create a video
 - The second algorithm should start from the YPbPr color space image. Read the image into RGB space, convert to YPbPr space, fade it to gray scale within the YPbPr space, convert it back to RGB space for saving to disk.
 - Convert intermediate images from YPbPr space to RGB space and save
 - Then combine to create a video

Floyd-Steinberg Dithering

- Design and implement an algorithm to convert an 8-bit gray scale image to a 1-bit dithered, binary image using the Floyd-Steinberg algorithm.
 - Read the image into RGB space, convert to YPbPr space, discard the Pb and Pr components, convert the Y image to binary using the Floyd-Steinberg dithering algorithm.

Deliverables

- Two video files (AVI or QuickTime)
 - Color to gray fade using RGB space
 - Color to gray fade using YPbPr space
- One image file
 - Dithered image
- All source code
- A description of the techniques used to perform the color to gray fades.

- All files should be archived into a .ZIP file. DO NOT ATTACH SOURCE CODE FILES DIRECTLY INTO BLACKBOARD. While they appear to attach properly, .java files will not attach successfully.

Notes

- Use the attached image file – rose.png to demonstrate all of your processes