CMSI 371-01

Computer Graphics

Spring 2016

Assignment 0308 Feedback

Outcomes that eventually cover both 2D and 3D continue to max out at | for now because this assignment remains in 2D.

Joseph Barbosa

jbarbosa1 / josephb.eshs@gmail.com

*Notes while running (high-priority notes are marked with ***):*

- Filters of both kinds run OK.
- Circle gradient takes quite a while but eventually looks OK; we'll see how the code looks…

Code review (refer to http://lmucs.github.io/hacking-guidelines/ for code-review abbreviations):

1. No issues with the various filter algorithms; nice variety, genuine use of neighboring pixels for the demonstrated filters (noting that `threshold` and `invert` do *not* actually use the neighborhood, but fortunately for you, those aren't the filters you chose to demonstrate). (*+2c*, *+3c*)
2. Code presentation-wise, though, you have incorrect indentation and an overly long single line of code in *nanoshop-neighborhood.js*. (*4c*)
3. The circle gradient code calls the top-level functions OK, except the order of the colors does not intuitively map to the way the resulting gradient travels. We'll need to dig in deeper then… (*4c*—readability because the order of the colors in the code is not suggestive of the colors onscreen)
4. *** Here is the performance flaw with the circle gradients: *the circle is filled for every vertex provided to* `plotCirclePoints`*!* This is easy to see: if you take out the loop in the top-level circle functions, you'll *still* get the full circle. That is the reason for the big performance downgrade. Of course, the idea here is to genuinely take advantage of the octant that `plotCirclePoints` is being given, one vertex at a time, and make sure to fill the portion of the circle "covered" solely by that vertex. (*1a*, *2d*, *4a*)
5. Gradient color direction is actually not as severe of an issue; you have a miscoordination between the colors you choose as the start and end points and the nested structure of your *whole-circle* loop, which goes top-to-bottom then left-to-right, vs. the other way around (i.e., left-to-right first, then top-to-bottom). (*2d*)

1a — | …A miss for losing the benefit of having `plotCirclePoints` as a helper function.

2c (max |) — |

2d — / …Repeated full circles + unclear gradient direction.

3c — +

4a — + …Everything works, but…

4b — | …not optimally.

4c — |

4d — |

4e — Two commits seen, though effectively just one because the first was the initial sample code. And that one commit can have a better message, pertaining to the specific code work that was accomplished. I think it's fair to expect better habits at this point. (/)

4f — Nanoshop work not submitted on time. (|)