**From:** Michael Yang, David Chang (CS61BL Sec 105)

**Memorandum for:** Courtney Wang

**Subject: Discussion of run-time for blur feature in new Picture manipulation software product**

**Summary:** The blur function in the Picture software takes significantly longer to run than other functions because the nature of its algorithm requires that, for each pixel, an area of surrounding pixels is analyzed. The size of this area is decided by the user, and every increase in this size causes a significantly longer run-time.

**Introduction:** Blur is a function in the Picture package that allows the client to blur the picture. Essentially, it averages the red, green, blue, and alpha values of an area of pixels surrounding the selected pixel, and sets that pixel to those average RGBA values. A method called averagePatch does this. We use averagePatch for all pixels in the picture. And by inputting a parameter called threshold to blur, the user determines the size of this area, which is a square with sides (2\*threshold+1).

The run-time of blur tends to be relatively long because it must check each pixel of the area, which is (2n+1)x(2n+1) pixels large, and must do this for each pixel in the picture. Therefore, the number of pixels to analyze would be **((2n+1)2 x total pixels)**. This means that as n increases, the run-time would increase quadratically.

**Methods:** Blur’s run-time was analyzed by varying the two factors that affect its run-time: total pixels and threshold.

Our first test varied the number of total pixels: we tested the run-time of differently sized “Creek” pictures, from 50x50 pixels to 450x450 pixels, and incremented by 50x50 pixels each time. The area parameter was held constant at n = 2 (so each pixel checked the surrounding 5x5 square of pixels). We analyzed the average run-time of each these differently sized pictures and checked if it was in accordance to the run-time growth model we discussed in the Introduction.

The second test varied the area parameter n and used the same picture.

**Evidence:**

**Conclusion:**