

Proposal

Summary:

My project would focus on mipmaps. Mipmaps is incredibly useful as it increases rendering speed and reduce aliasing artifacts.

Sometimes, a large image would cause low sampling frequency. For example, the image size is 500x500 and the screen size is 100x100, then a pixel on the screen will display a bunch of pixels of the image. So, we can precalculate the mipmaps and display a level from mipmaps based on the size of pixmap.

The program would build mipmaps from an image and store them in a file and display the mipmap using OIIO and OpenGL. The program would provide the ability to have the user interactively scale the pixmap via the mousewheel. The window will split into two parts, the left displays the mipmap and the right displays the original image. Users would see the aliasing artifacts on the left view less than the right because of mipmaps. Users can scale down or up the image to observe the effect of mipmaps.

Objectives and Expected Outputs:

1. build mipmaps and store it in a file
2. display both the original and mipmap within a single window
3. scale down or up the pixmap via the mousewheel
4. choose a level from mipmaps to display on the screen based on size of pixmap

References:

<https://sbcode.net/threejs/mipmaps/>

<https://www.cs.cornell.edu/courses/cs5625/2016sp/docs/pa4.html>