# Summary

The program was designed to take a single frame image file and apply a series of filters or effects on the image then write the rendered result to an output file. It supports multiple formats including: .gif, .png, .jpg, .pbm, .pgm, .ppm. In addition, it can convert from one format to another.

# Filters

* Scale: the scale filter will resize the image depending on the parameters provided. When scaling down it uses the median color from a window to create each pixel in the new image (the size of the window is determined by the scale factor). When scaling up, the new pixel’s position is quantized to a parent pixel in the original image and the color value is copied. Afterwards, a median filter is applied to the image to make the image appear smoother (the window size for the median filter is determined by the scale factor).
* Threshold: the threshold filter uses color binalization to convert the image into a series of black or white pixels. It does this by calculating the brightness of each pixel. The formula for the brightness is:  
  The input value is a number between 0 and 1. The resulting pixel is determined by:
* Grayscale: the grayscale filter converts the image into gray colors only. It does this by finding the brightness of the color like it does with the threshold filter. It then uses that value to set all the color channels for the new pixel.
* Median: the median filter blurs the image based on the filter parameters. The window parameter is used to look at the neighbor pixels and the algorithm uses the neighbor pixels to find the average color. The x parameter means the filter is only applied in the x direction and the y parameter means the filter is only applied in the y direction. You can provide both the x and y parameters to apply the effect in both directions.

# Performance

The threshold and grayscale filter algorithms perform at . The median filter is at most where is the window size. The complexity is because the filter is applied in one direction and uses only the pixels in the same line as the center pixel then is applied again in the other direction. This has the same effect of getting the median color of the entire window. Below shows the original image, method1, and method 2. Method 1 uses the median color of the entire window and method 2 uses only the pixels in the same line as the center pixel. Notice that both images look the same:



The scale filter is at most because it may or may not use the median filter to smooth out the image.