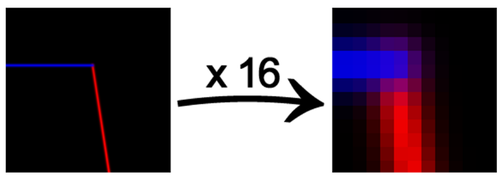
Graphics: A Sample of Our Work

by Alexandr Pshenichkin
in collaboration with Michael Stone

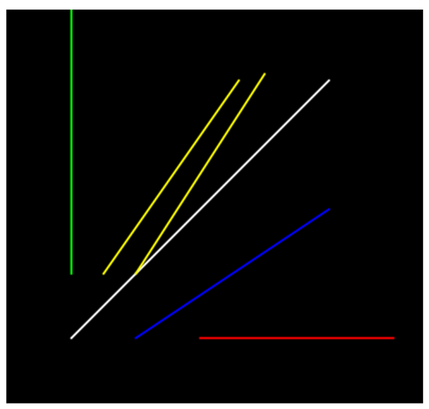
Lines

$$h_P(r) = 2\int_r^\infty h(\rho)\cos^{-1}\left(\frac{r}{\rho}\right)\rho\, d\rho.$$

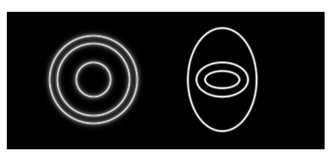
Our line algorithm relies on numerical integration for anti-aliasing. It calculates a distance from each pixel to the line and colors the pixel appropriately.



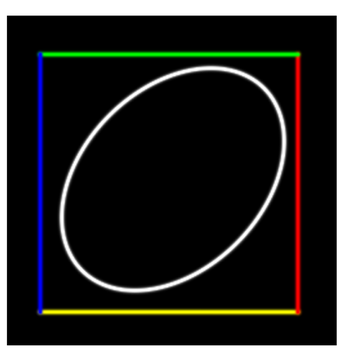
It produces smooth, nicely jaggy-free lines at any angle:



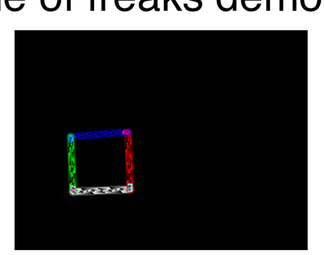
Any arbitrary shape can be approximated as long as we have the right distance equations. For example, the ellipse:



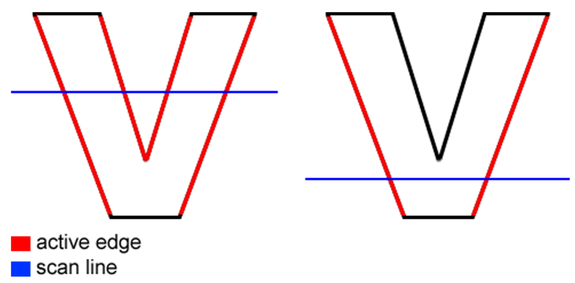
... arbitrarily oriented:



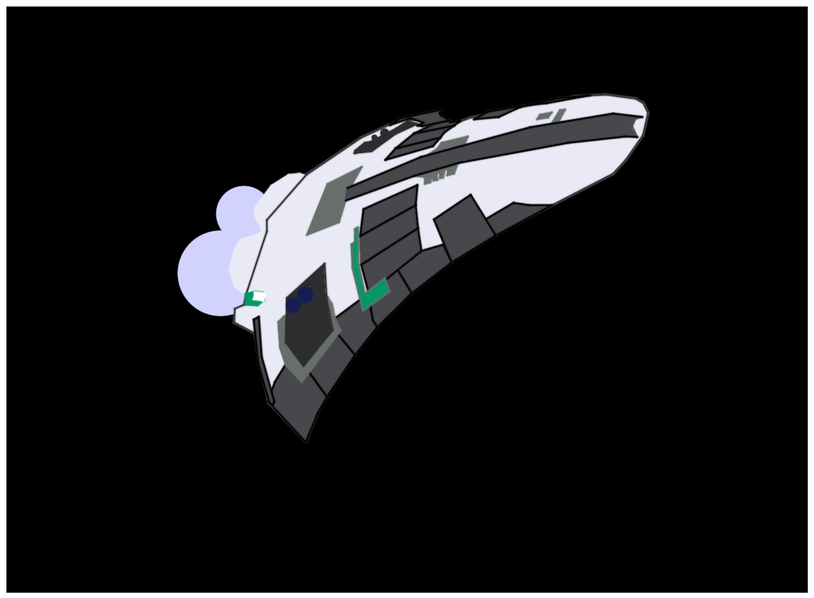
The process by which we arrived at all this was, of course, arduous, as our menagerie of freaks demonstrates:



Scanline Fill



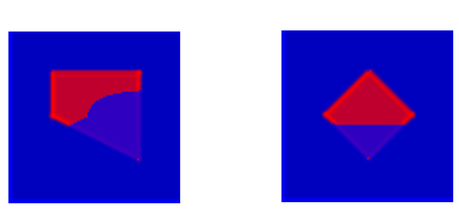
We use a fairly standard scanline fill algorithm, so it is omitted here. Some of the images that result, however, are worth noticing:



A faux-3D Starbridge (from the game *EV Nova*). This was created by hand by picking boundary points from an image file. Note the slight irregularities between our line algorithm (used to anti-alias the edges) and scanline fill.

The A Buffer

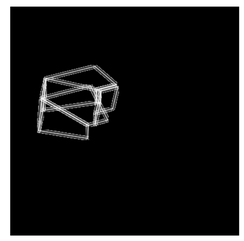
The scanline fill algorithm's main purpose is to run our depth buffer, which is based on the A Buffer algorithm. It is nearly identical to the standard Z Buffer in nearly all ways except one: instead of overwriting pixels with nearer ones, our program stores all color values for a pixel in an ordered list, and blends them when it's time to render.



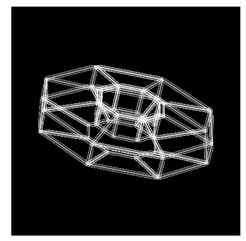
Example semitransparent A-buffer images.

Heirarchical Models

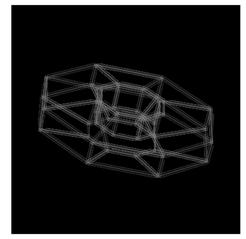
Our system incorporates a parser for model files. This parser not only reads in points from text files, but also supports structured models. Here is an example of the how we can make a cartoon-ish looking UFO:



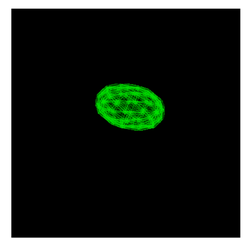
Here we see a single section of our alien spacecraft. This is just a few polygons clumped together.



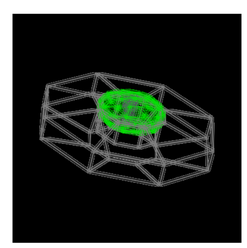
We clone and rotate this section to create the basic body of the alien spacecraft.



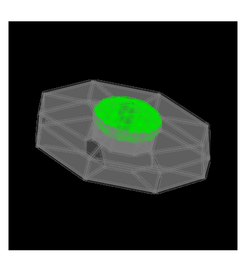
Add a few more bits (namely a floor in the middle) and we have ourselves a space hull.



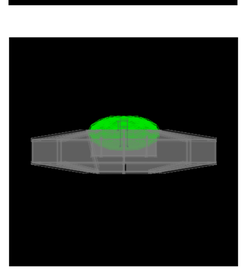
This space hull, is, of course, incomplete without a big green semi-transparent alien bubble for our alien to sit in.

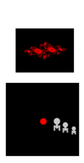


The bubble is actually a compressed sphere. We created triangularizations of cylinders and spheres at various levels of detail using a simple generating program, and then inserting them into our model file.



Since our lighting system is still rather weak, we turn to A-buffer magic to show the individual faces of our spacecraft. Note that any artifacting comes from errors in our scanline algorithm, not the line drawer.





See website for:
Julia/Mandelbrot sets
Groups of models