More About Stereo

It is difficult, under normal circumstances, to understand a wire-frame two-dimensional drawing of a three-dimensional object. Lacking the ability to tell the “near” wires from the “far” wires, the brain is easily fooled into seeing objects totally wrong. HyperCuber creates some truly wild three-dimensional wire-frame images, images which seem to be particularly easy to misinterpret (especially during higher-dimensional rotations). To alleviate this problem, HyperCuber supports stereoscopic viewing.

Stereoscopic viewing is based on the observation that we have two eyes, and each eye sees a slightly different image. The brain automatically processes these images and uses them to determine where things are with respect to each other. Using both images, it is possible to determine whether one object is behind or in front of another, a feat which is often impossible using only one image. Stereoscopic viewing provides each eye with a separate image, and fools the brain into thinking that they are really the same thing seen from the different perspectives of the two eyes.

HyperCuber can do this in two ways. It can generate two images, each in a different color, and overlay them, one on top of the other (Figure 1). To a normal person looking at the screen, it looks like garbage, but to someone wearing special glasses which let in only one color for each eye, it takes on a three-dimensional look. These glasses can be made from normal glasses (don’t use sunglasses unless they have only very light shading) by placing something transparent and red over the left eye and something transparent and green over the right. Transparent colored plastic sheets can be purchased at art supplies stores. You can also find small pieces of colored plastic at stationary stores, sold as covers for school reports or as tabs for filing cabinets. Be creative...you can find them. I actually made my glass frames from a piece of posterboard (since there were no glasses in the house) with scissors and a hobby knife (they look pretty wild too— I got to try my hand at artistic glass frame design...).

wo-color stereoscopic viewing is usually satisfactory, but it’s hard to find plastics and matching colors which will block out one color and let the other in (though HyperCuber lets you choose the colors to match your glasses; see The Colors… Command in the How To Use HyperCuber chapter). Consequently, the image quality is not usually very good. I therefore prefer two-image stereo.

Two-image stereo (Figure 2) requires no special glasses or apparatus, but you do have to train yourself to do it. HyperCuber draws the two images not on top of each other as in the two-color viewing, but side-by-side. The image on the left is meant to be viewed by the right eye, and vice versa. This seemingly impossible feat can be accomplished by looking at the images cross-eyed. Your left eye will then be looking to the right of your right eye, as desired. Basically, you need to look at the images cross-eyed, and then focus. When you first look cross-eyed, you should see four out-of-focus images, two from your left eye and two from your right. Change your cross-eyedness until the two inner images of the four merge. It may be useful to focus on a finger in front of your face, and move it closer and farther until the images merge. Now the hard part: focus on the middle image. You need to focus on the middle image as though it were really a normal image, rather than a bizarre combination of two images. If you succeed (and you can, if you try long enough), it will look three-dimensional! You will be able to see which wires are in front, and which are behind. You will most likely jump out of your seat and shout “Eureka!” At least, I did.

wo-image stereo has to fit two images on the screen, so the images are

smaller than those generated by two-color stereo. However, two-image stereo images are in color (if the original object uses different colors, like the default hypercube), which can be nice. Finally, two-image stereo is slightly faster than two-color stereo.

I implemented stereo in a rather stupid but simple way. The eyes are always assumed to be a few degrees apart. This works fine for most purposes but fails miserably if you get too close to vertical (that is, if the eye point gets too close to the z axis). For best stereo viewing, don’t go too high— keep [3:2] between 120 and 240 degrees, say. If people actually complain about this, I may improve the algorithm, but it works for me at the moment.