Changing The View

HyperCuber draws an n-dimensional object (a hypercube by default) and lets you look at it in different ways. Though it may seem difficult to believe at times, when the picture on your screen is undergoing strange and terrible transformations, the original n-dimensional object never changes. It is a rigid and stationary n-dimensional object. All that changes is your viewpoint. Any changes on the screen are results of looking at the same object in different ways.

You control your viewpoint by manipulating scroll bars in floating windows. They let you change your viewing angles for different dimensions, and they let you change the amount of perspective you want to use. In order to understand what the different angles are, it is necessary to understand a little of how HyperCuber constructs the image you see.

When you're looking at a hypercube, there are actually two viewpoints involved— a four-dimensional viewpoint and a three-dimensional viewpoint. HyperCuber first constructs a three-dimensional projection of the four-dimensional object. Then, treating the three-dimensional projection as a three-dimensional object (which it is), it creates a two-dimensional projection of it. The angles [4:1] through [4:3] control your four-dimensional view of the object; the angles [3:1] and [3:2] control your three-dimensional view of the three-dimensional projection of the object. [4:P] controls the amount of four-dimensional perspective, and [3:P] controls the amount of three-dimensional perspective. See Projections in the Viewpoints and Projections chapter for more information.

The results of changing higher-dimensional angles like [4:3] or [5:2] are virtually impossible to describe in English. It's best you play with them yourself and try to get an idea of what they do. However, [3:2] and [3:1] have simple effects. If you imagine yourself viewing the Earth from far away, [3:2] and [3:1] correspond to lattitude and longitude, respectively. Thus, increasing [3:2] "raises your viewpoint north of the equator," and increasing [3:1] "moves your viewpoint east." A few clicks on these scroll bars will make their effects very clear.

The [3:P] and [4:P] scroll bars (and the [n:P] scroll bars, if you're looking at higher-dimensional objects) control the amount of perspective. There are several ways of thinking of this. One is to think of it as "changing lenses." High values of [3:1] correspond to the use of a wide-angle lens; nearby things become very large, and distant things very small. Low values of [3:1] correspond to the use of a telephoto lens; relative distance isn't as important in determining size when things are farther away. Another

(similar) way to think of it is as a sort of “distance from the object” control. Decreasing the [3:1] bar moves you farther away from the object, and perspective decreases accordingly. You have to imagine also that the object is being magnified as you get farther away, so the size of the object on the screen doesn’t change; only the perspective does. Higher-dimensional perspectives again correspond to the three-dimensional perspective; object which are “farther away” in the n-dimensional sense are smaller than object which are “closer.” In the case of a hypercube and [4:P], this results in one of the “faces” (cubes) getting larger while one of them gets smaller. Again, the easiest way to understand these controls is to play with them. For a little more discussion of perspective, see Projections in the Viewpoints and Projections chapter.

The method I used to do perspective for HyperCuber, while mathematically sound, has caused me a lot of problems. In early versions of HyperCuber, use of a large amount of perspective could cause some very strange-looking results from some viewpoints. I “fixed” that somewhat, though my fix still yields some strange results for large perspectives, and occasionally results in a “skip,” where one point of the object will fling itself off to infinity for one viewpoint, and return to normal for all others. This is admittedly a problem, and some would even call it a bug. I may fix it later, but for now, there’s a work-around: don’t use a lot of high perspective settings. Setting all the perspective parameters to 10 is bound to cause problems. If you set your perspective low enough that the object is almost completely in the window all the time, you won’t have any problems.

Using Scroll Bars

The most obvious way to change your viewpoint of the object is to use the scroll bars. HyperCuber’s scroll bars work almost like other scroll bars you’ve seen, but not quite. You can change an angle by one degree by clicking once quickly on a scroll bar arrow. You can change by ten degrees at a time by clicking once quickly in the “page region” (the part that’s gray). You can move directly to any value by dragging the “thumb” (the little square). You can repeatedly move any direction by clicking in the appropriate place and holding down the mouse button. There are two major differences between these scroll bars and normal scroll bars. First, they have a “live thumb,” which means that the object is continuously updated as you drag the thumb. Second, angle scroll bars wrap around, which means that if the thumb reaches one end of the scroll bar, it will jump to the other end and keep going, rather than just stopping there. This means you can rotate an object forever with a single click, if you wish. Perspective scroll bars do not wrap around.

Using Keyboard Controls

At some point while you’re using HyperCuber you will attempt to use the scroll bars while viewing in two-image mode. The problem with this is that

in two-image mode, you eyes have to focus in a very strange way, and if you unfocus to look at the scroll bars, you lose the image. The solution to this problem is keyboard controls. You can define an key to be a control. When you define a control, you choose a key, an angle, an increment, and a direction. When you press that key, it will change the angle you chose by the amount you chose in the direction you chose. So if you have defined a control so that the K key will “Increase [3:2] by 3” then whenever you hit the K key, HyperCuber will add 3 to the value of [3:2], as though you had clicked the scroll bar three times. HyperCuber comes with a complete set of key controls for three and four dimensions, but you can easily change them or add new ones if you like.

Using Mouse Controls

For those of you who don’t like the keyboard, there are also mouse controls. A mouse control maps the horizontal movements of the mouse cursor to one angle and maps the vertical movements of the mouse cursor to another angle. The best way to figure this out is to play with it. For an example, start up HyperCuber (so that it shows a colored hypercube); then click and drag in the HyperCube window. Isn’t that neat?! HyperCuber comes with a complete set of mouse controls, but you can easily change the or add new ones.