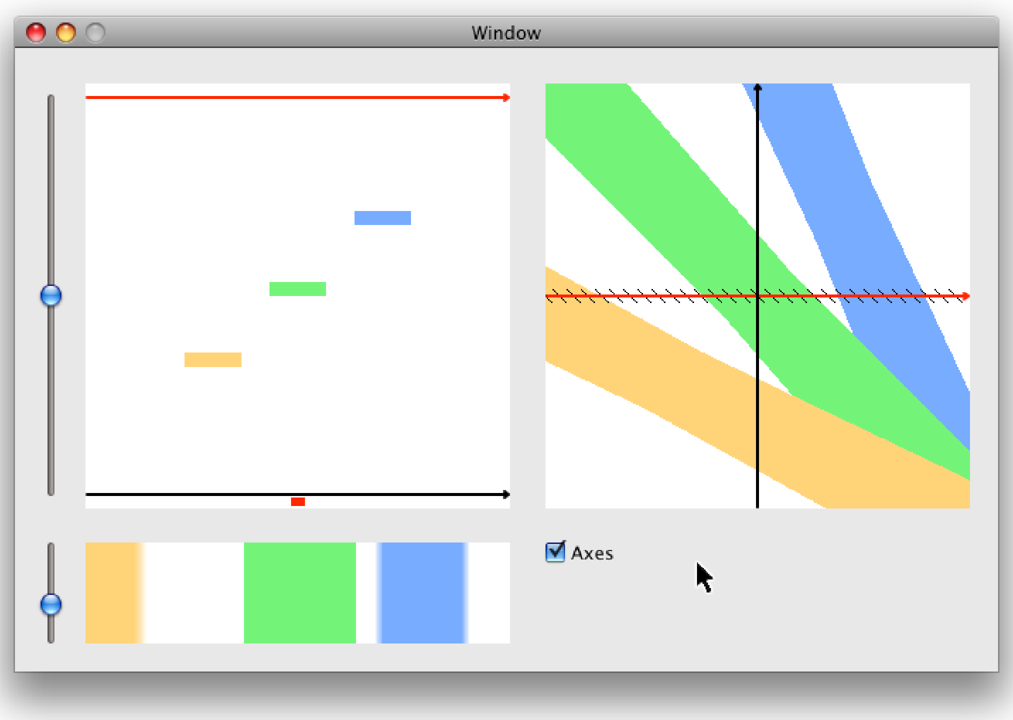
2D Lightfield Simulator

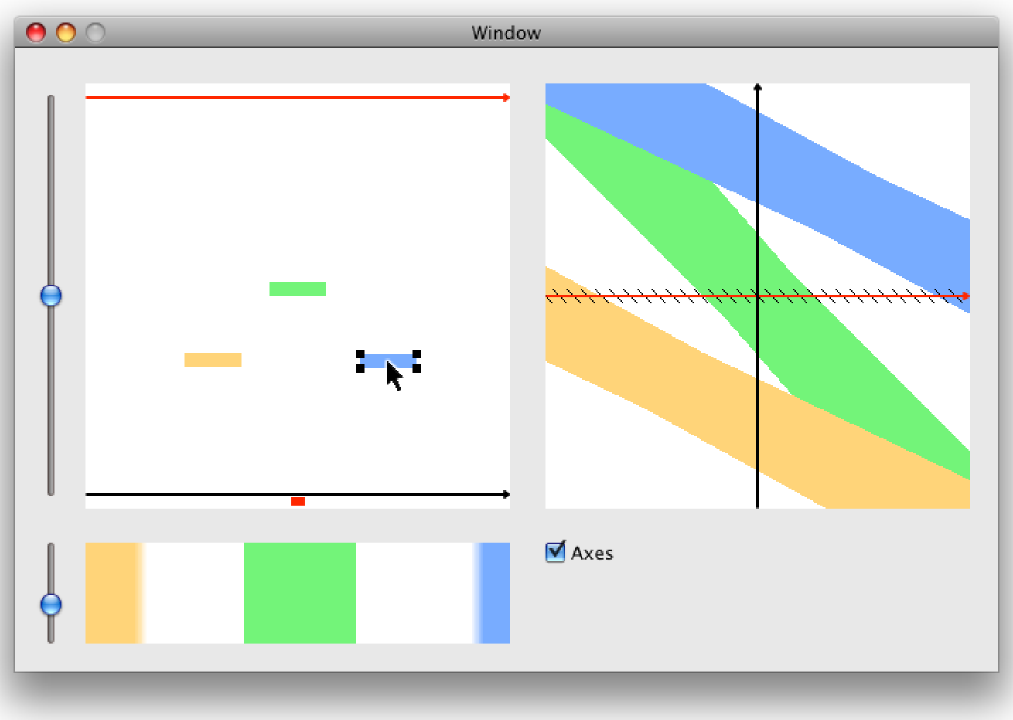
Ian Spiro

Computational Photography
Spring 2008, Prof. Rob Fergus

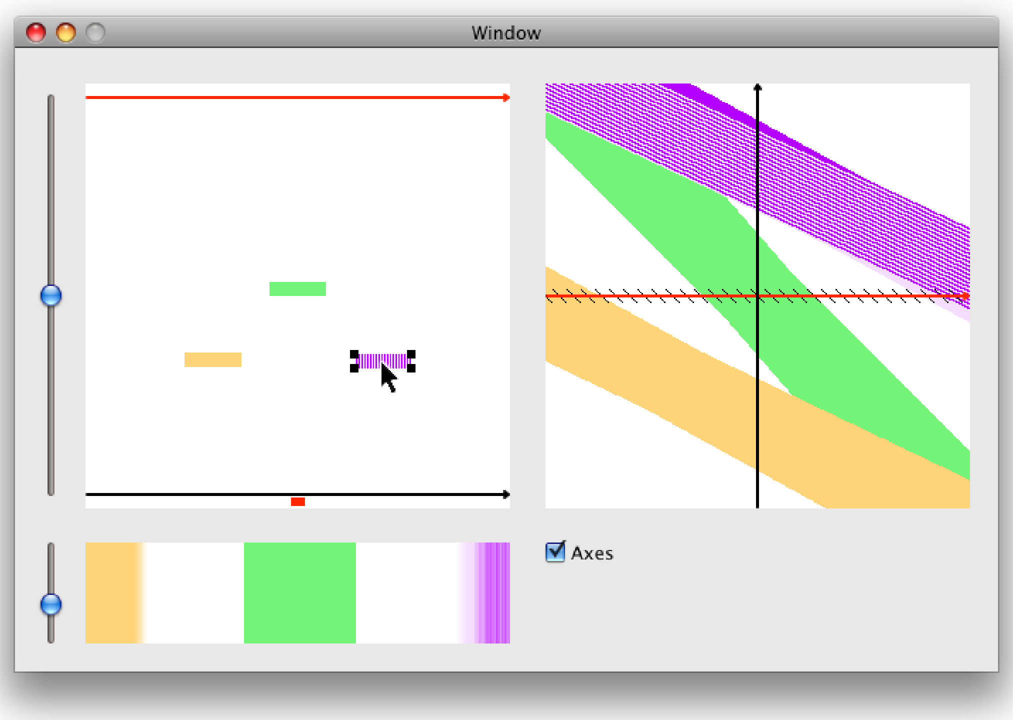
Plenovision is a 2D lightfield simulator written in Objective C / C++ with Cocoa frameworks. The user creates a simple scene by dragging around several rectangular objects. The application computes the resulting 2D lightfield by shooting rays from all pixel positions that lie along the black axis, to all pixel positions along the red axis. The user can specify a camera's focal distance and depth of field to generate a 1D photo of the scene. The photo is computed by integrating across the lightfield.



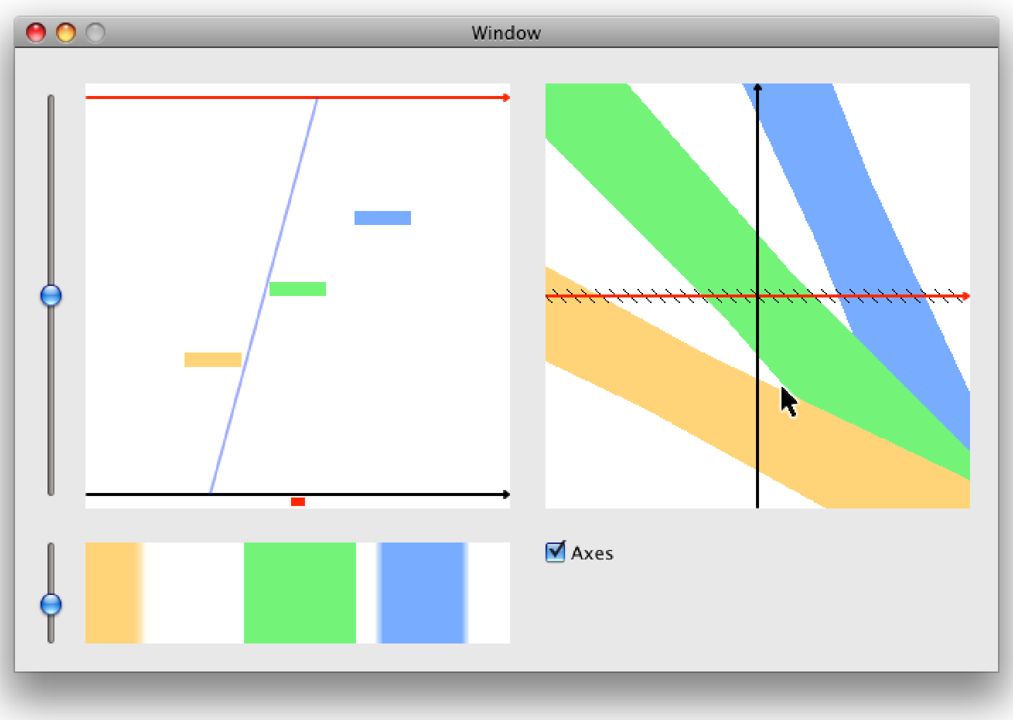
This is how the application looks when you open it. The left panel shows a 2D scene, comprised of a white background which represents open space, and three colored billboards. On the right, we see the approximated lightfield for this scene. Below the scene, we see a 1D photo taken by the red camera positioned at the bottom of the scene. Imagine you are standing at the small red box, looking out perpendicularly to the black axis. You would see the gold, green, and blue billboards, which will appear as bands. Though the photo appears to be in 2D, it is actually representing a 1D image which has been stretched over a second dimension to make it easier to view.



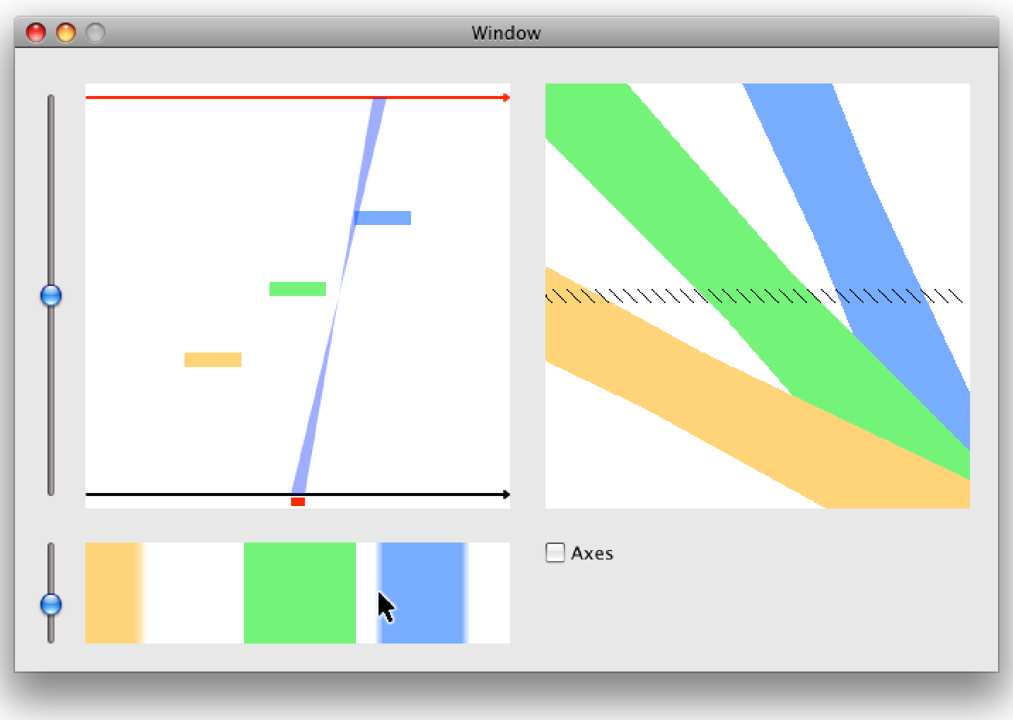
You can click the billboards and drag them around. When a billboard is selected, you can also use the arrow keys to move pixel by pixel. Hold shift to move ten pixels at a time. In this example, we move the blue billboard and the lightfield is updated accordingly. Since the blue billboard was moved closer to the camera, its slope in the lightfield gets closer to horizontal.



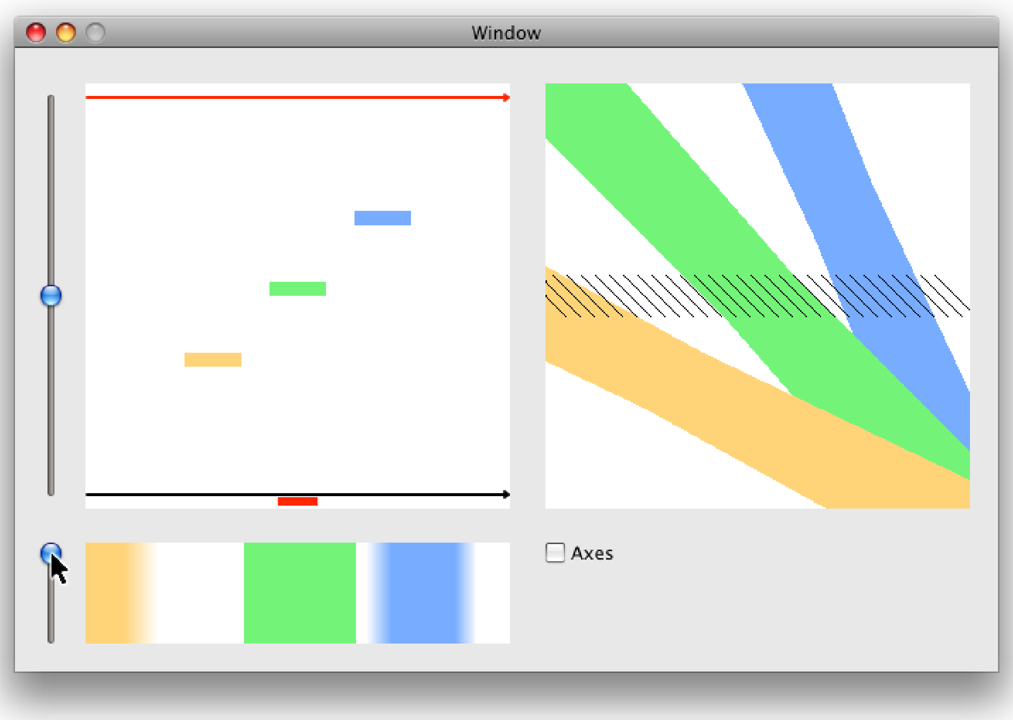
You can apply graphics to the billboards. Select a billboard and press the spacebar. Locate a .png file and the image will be tiled across the billboard. This will affect the lightfield. The pattern will appear smeared across at the appropriate slope.



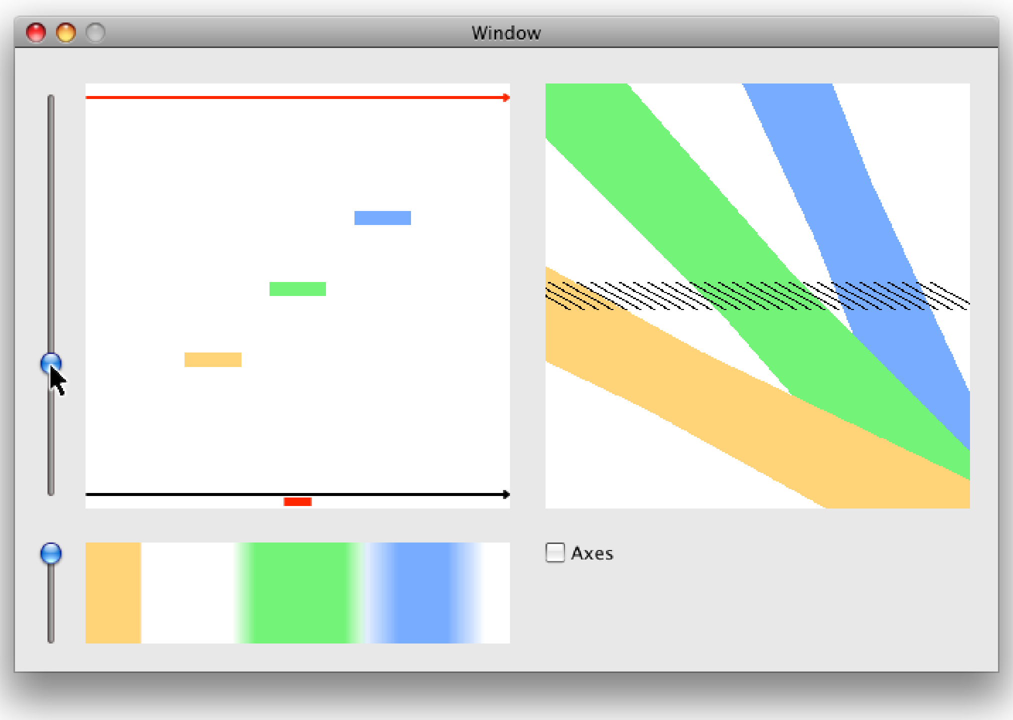
By mousing over the lightfield, we can see how each pixel in the lightfield was determined. In this example, the mouse is over a blank region and we are shown a ray shooting across the scene at about 80 degrees. Note that this ray doesn't hit a colored pixel and therefore the lightfield is left blank. If the mouse were moved right, the ray would project to a point along the red axis further to the right. Where the ray hits the green billboard, the lightfield will be colored green.



You can uncheck the "Axes" control to hide the axes in the lightfield. Now it is easier to see a series of black lines across the middle of the lightfield. These lines represent the integration that is performed over the lightfield to obtain the 1D photo pictured on the bottom left. This integration is a consequence of using a spherical lens. By mousing over the 1D photo, as in this picture, we are shown all of the rays that were used to obtain the current photo pixel's color.



This slider changes the depth of field of the camera. In a typical physical camera, this corresponds to opening or closing the aperture. Sliding all the way up, as in this figure, is the widest setting of the aperture. This corresponds to decreasing the f-number and decreasing the depth of field. Note how in the photo, things that are not in focus grow blurrier. This is a result of the increased integration length which is shown in the lightfield. By integrating over a longer distance, areas around objects that are not in focus become blurrier.



By sliding the control on the left of the scene, we change the focal distance of the camera. In this example. we slide the focus to line up with the gold billboard. Note in the lightfield that the angle of the integration lines has changed. This angle now aligns with the slope of the gold billboard. In the 1D photo, the gold object is now sharp, the green object is no longer sharp, and the blue billboard is even blurrier.