

Pixel Displays

by Dana W. Burman 04/13/2014

Greeting:

Thank you for purchasing this tutorial. I hope you learn some new skills and unlock greater potential for your own work. This is a project based tutorial, so I walk you through the process of setting up a few displays step-by-step, and I show off a few other scenes that are pre-made. My process for making tutorials tends to be to make a prototype scene and then record a tutorial showing you how I did it. Sometimes I have to go through it a few times. Generally, though, I don't put everything down in step-by-step writing beforehand, so I may end up exploring and trying new things while showing how this all works. The benefits to this approach are 1) you get to see how another artist works, 2) you get to see multiple approaches demonstrated, and 3) you get to see how one recovers when something goes wrong. Because the format isn't a strict, by the numbers approach, it may be best for you to watch a lesson in its entirety before trying to follow along. There are a few times in this course where I cut out or speed up the video and provide you with the final numbers used so you don't need to watch me trying to figure it out in real time.

There is a bonus video in this course that shows using the position of particles to put color splotches on a plane. The process used to do that is kind of buggy (as seen in the video), and not related directly to the subject at hand, but I thought someone might enjoy exploring that more.

Installation

This video is spread over two .zip files. You can uncompress/extract these to their own folders, or put all the contents into one folder (preferred).

Included Files

Scene files for all (or nearly all) of the scenes shown in this series are provided, including some illustrative example files that you may have fun trying to figure out.

Object files for each lesson are also included. The LED/LCD and block display objects are provided as untextured objects.

Some objects have multiple versions for different scenes. That's because LightWave was getting confused as to which object should control the surface settings when the same object was used in multiple scenes.

Chapters

- 01 Intro And Pixels.mp4 (1:33)
- 02 Object vs World Coordinates.mp4 (5:54)
- 03 Spot and the Replace Spot Node.mp4 (8:20)
- 04 Block Display – Instances.mp4 (12:55)
- 05 Texturing3colorLEDPixel.mp4 (10:03)
- 06 LCD_Pixel_Screen_Part1.mp4 (15:09)
- 07 LCD_Pixel_ScreenPart2_Eliminating the ScreenDoorEffect.mp4 (7:54)
- 08 TexturingLED_RED.mp4 (15:49)
- 09 LED_ScrollerDisplay.mp4 (14:47)
- 10 Block Display – Parts.mp4 (10:02)
- 11 Particle Scatter Display-SetUpInstances.mp4 (15:08)
- 12 Particle Scatter Display-Texturing.mp4 (11:02)
- 13 DEMO_8x8_LED.mp4 (15:09)
- 14 DEMO_IlluminatingPeople.mp4 (11:40)
- 15 Making the Displays Moveable.mp4 (11:41)
- 16. Bonus: Replace Spot Particle Fun (3:15)

Plugins

DP_Kit & DP_Filter (Part Move & Replace Spot) can be found at:

<http://dpont.pagesperso-orange.fr>

Terminology

Pixel

What is a pixel? There are couple of different ways the term “pixel” is used. When it comes to hardware, a pixel is “Picture Element”, a tiny dot, that either emits or detects light electronically. Your computer screen is made up of them as are the image sensors in digital cameras. In the early days of computer graphics, video screens were really low resolution and you could easily see the individual pixels. On most modern screens you need a magnifying glass to see the individual picture elements.

In terms of computer graphics software, a pixel is one of the individual blocks that make up raster images - such Jpegs or PNG files.

In this course, I use the term “pixel” a bit inconsistently,. Some lessons will focus on recreating pixels as seen in an LCD display, while other lessons will focus on using larger objects as individual pixels, and I’ll probably use pixel referring to blocks that make up the image we’re loading into LightWave.

Local Coordinates and World Coordinates

One important concept to this course is the difference between local and world coordinate.

The nodes we will be using have inputs and outputs for both world position and local position, so an understanding of local versus world coordinates will be helpful.

Textures, both images and procedurals, have coordinates of their own. Textures can be positioned, scaled and rotated, but where are they being positioned, scaled, and rotated from? Where is their origin, their center? That is determined by the position values you give them (by default that's 0,0,0) and also whether they are set to Local Coordinates or World Coordinates. If your object is in the center of the scene, sitting at the origin or zero on the x the y and the z, then you might not notice a difference between the different modes unless you're texturing objects that are cloned or used as instances and some of them are off center. By default, textures are in Local Coordinates mode. That means that the object they are attached to is the center of their universe. If you move the object, the texture moves with it. If your object is one meter to left, then your texture's origin is also one meter to the left. The texture can be positioned and rotated and scaled, but it's position is relative to the object. Again, when the object is moved, the texture moves with it.

If you check the "World Coordinates" check box in an image or procedural texture, then the textures origin will be Layout's origin at 0,0,0 meters. The texture can be positioned relative to it's origin, but when set to World Coordinates, the origin is fixed. When you move the object, the texture stays in place. It looks like the object is moving through the texture.

You CAN have an texture that is set to world coordinates move with an object and still be relative to the scene origin. You just need to link the texture's coordinates to the object position. The easiest way to do this is to set the Reference Object parameter in the texture to the object itself. This will connect the position, scale, and rotation to the object's position, scale, and rotation. You could also connect position, rotation, and scale up separately using expressions or the item info node for more control, but that is beyond the scope of this tutorial. There are times where you'll want to use a separate object, often a null object as the Reference Object so you can easily control the position, scale and rotation of the texture.

Another way of thinking about this is in terms of parenting. In local coordinate mode, textures are the child of the object. In world coordinate mode, textures are like separate objects.

Spot & Replace Spot Node

The main tool we will be using throughout this course is the Replace Spot Node from Denis Pontonnier's DP Filter plugin.

If we are replacing a spot, it might be helpful to define what a spot is. In surfacing terms, a spot is anywhere a ray hits a surface and is evaluated by LightWave. This is not a vertex or point or polygon, but rather a spot on the surface of the object. You could think of a spot as one of the pixels that makes up your final image. What angle is the surface in that pixel? What is the color of the surface in that pixel, what side of the surface is showing in that pixel? It's not entirely accurate to describe a spot as a pixel, but it may help to think of it in those terms.

The Replace Spot Node allows us to change or move the surface information that is being recorded by LightWave's camera, thus tricking it into seeing something different on the surface. By giving all the spots on an object (instance or part of an object) one single coordinate, we make all the spots return one set of values. That way we can make the entire surface of an object one solid color where it would normally be showing multiple colors.

If your image or texture is set to world coordinates, you need to replace the world coordinates. If it is set to object or local coordinates you need to replace the local coordinates.

Included Footage

The eyeball and marina (boats) footage was shot by Dana W. Burman and may not be redistributed or sold. I know it's not likely to happen, but it doesn't hurt to put this disclaimer in here.

Go Deeper:

Here are a couple of threads in Newtek's forums that helped in development.

<http://forums.newtek.com/archive/index.php/t-77420.html>

<http://forums.newtek.com/archive/index.php/t-124364.html>

Mikael "Cageman" Burman (no relation) has a few great tutorials showing uses of DP Filter including stuff like the bloom node shown in lesson 13.

<http://www.youtube.com/user/TemalCageman/videos>

Thanks

Thanks to Denis Pontonnier and Bryan Phillips and others on the Newtek forums for helping me learn and develop these techniques. Thanks to Liberty3D for providing an artist-friendly space to sell these tutorials.

Support

If you need support or help with this tutorial, you can contact me via my website, <http://dwburman.com> or through <http://Liberty3d.com>