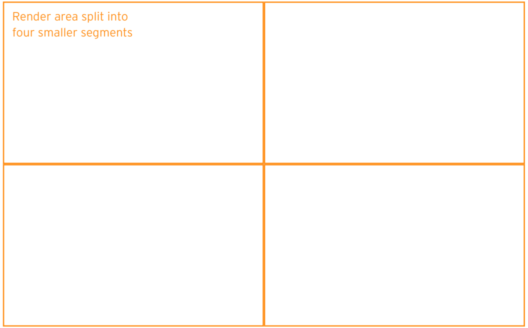
mangoldphotography

ADVANCED CAMERA RENDERING

INTRODUCTION



One of the major drawbacks in Lightwave are the problems which occur when you try to render at print resolution. So far you could try to lower the segment memory or use limited render areas to avoid any error messages and finally get you render started.

But even then, depending on the image size, the poly count and RAM your image maps take, Lightwave might refuse to do the required task It seems that even a small limited render area, needs the buffers for the whole image size. This way it's impossible to go for any decent high res image.

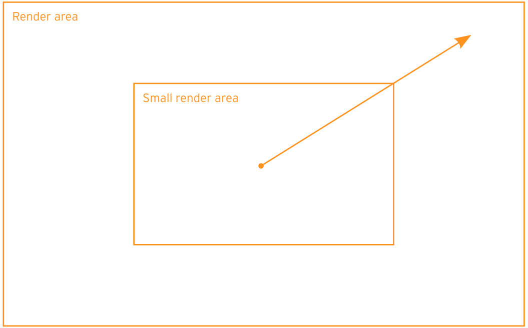
There has been a trick to use several rotated cameras with smaller resolutions to get your final image, which then either needed postproduction in Photoshop or in an image stitching software. This method is rather inconvenient, but so far has been the only life saviour. The introduction of the advanced camera finally gave users a new way which is rather easy to set up, and lets you "render" each resolution you might want to go for.

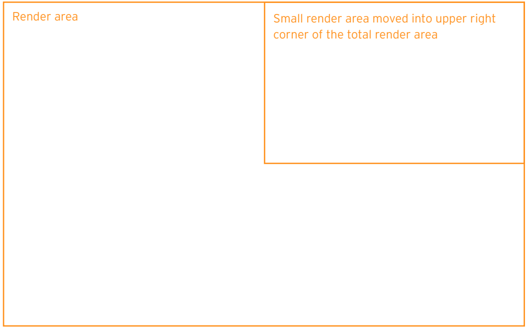
Except from Photoshop and a calculator there's nothing you will need except from Lightwave 9. William Vaughan's video on the advanced camera may be a good and brief introduction to what it is all about. Although everybody with a photographic background will understand this method pretty easily.

The concept is to copy the behaviour of a view camera to render several smaller images which then are stitched together in Photoshop. It's more or less like the older method with the big advantage that there won't be any kind of distortion and the images will fit together without any seams or overlapping areas.

If you ever needed the quality of 8x10" film in regular photography, but only owned a 4x5" film back, then you could have shot 4 exposures, changing the position of the back standard for each shot. If you worked precisely you then would have exposed the same film area as for a 8x10" film. Afterwards the scans could be recombined to show the whole view without the borders of the film.

THEORY





Work like you always do. Render your tests with your master camera at low-res and do your lighting, posing, texturing like usually. If you feel satisfied with everything, then divide your total image area into several smaller chunks (in your brain). If you need a final resolution of 10.000 by 4.000 pixels, then 10 render areas of 2.000 pixel in width and height should work fine (Five in an upper row, five in a lower one). With 9.000 by 6.000 you could try 6 areas with 3.000 pixels in width and height.

It's not necessary to render squares. So an image of 6.000 by 2.000 pixels could be split up into two pieces of 3.000 by 2.000 pixel. The only point is that Lightwave is able to render the smaller areas without complaining.

Just take care that you can divide your overall resolution you want to go for by the number of segments in the vertical and horizontal direction and still have an even number. Example:

Final resolution:
9.400 x 6.700 pixel

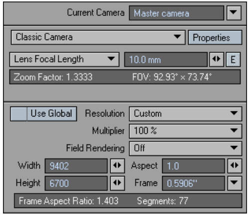
Possible amount of small render areas:
3 x 2

Resolution of small render area:
9.400 pixels / 3 = 3.133,3333 pixels (bad)
6.700 pixels / 2 = 3.350 pixels (good)

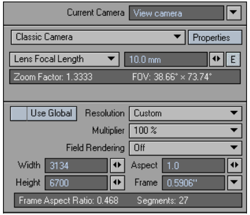
In order to get proper results multiply 3.134 pixels by 3 and use the result as your new width for the master camera. In this case 9.402 x 6.700 pixels will give you a good start.

As the next step clone your master camera and name the clone view camera. The resolution is then set to 3.134 x 3.350 with some minor adjustments to the field of view. The miracle now is that we won't need to rotate the camera to look in all areas which our master camera covers, a null object in form of a grid will do the trick for us without any movement of the camera itself. The only movement is turning the camera button from classic to advanced.

PRACTICE



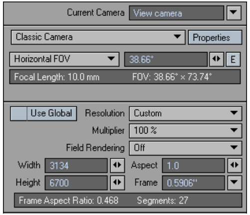
1.
Set-up the *Master camera's* resolution as explained on the previous page.



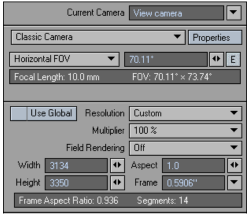
2.
Clone your *Master camera*.

Rename the new camera, e.g. *View camera*.

Change the **Width** of the *View camera* to the fractional amount as explained on the previous page, e.g. 3.134 pixels.



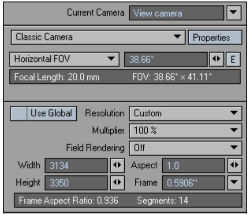
3.
Change the **Lens focal length** or **Zoom factor** button to **Horizon-tal FOV**. Memorise the displayed angle.



4.
Change the **Height** of the *View camera* to the fractional amount as explained on the previous page, e.g. 3.350 pixels.

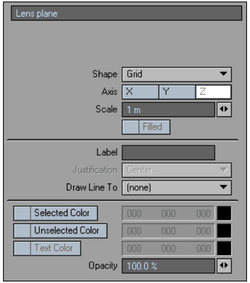
mangoldphotography

PRACTICE

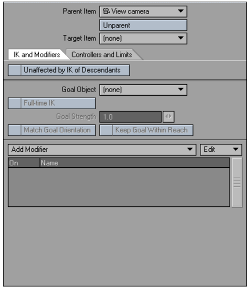


5.
Replace the angle in the **Horizontal FOV** entry field with the memorised angle.

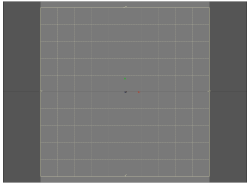
6.
Do a quick test render, either by disabling all raytracing options, choosing a lower resolution (**Multiplier** at 50% or 25%) or both.



7.
Create a **Null** object. Use the edit button to get all the options as seen above. Rename the **Null** to *Lens plane*, set **Shape** to **Grid**, **Axis** to **Z** and leave **Scale** at 1 m.

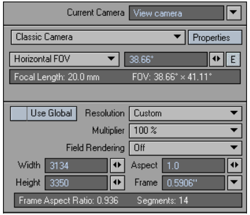


8.
Parent the *Lens plane* to the *View camera*. Reset all **Position** and **Rotation** values of the *Lens plane* to 0. Disable **X**, **Y** for **Position** and **H**, **P** and **B** for **Rotation** to avoid any unwanted changes.



9.
Move the *Lens plane* on the Z-axis as long as it fills the images width.

Next step is to use trigonometry to calculate the exact position on the Z-axis.



10.
Memorise the angle from the **Horizontal FOV** entry field and divide it by 2.

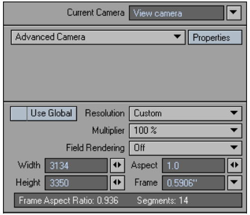
Get your calculator.

Get the tangent of this angle, e.g. 19,33° (38,66° / 2). In this case the result is 0,350782938.

Get the reciprocal value of this number (x^{-1}). In this case the result is 2.850765787.

Divide this number by 2. Result in this case: 1,425382894.

This is the exact position of the *Lens plane* on the Z-axis.



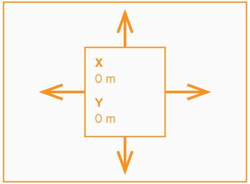
11.
Z value from manual positioning the *Lens plane* and calculated value should be more or less the same.

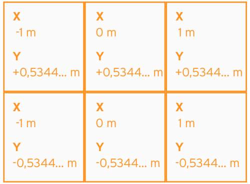
Make sure your calculator is running in the DEG mode if there are any problems.

Use the calculated value in the **Z** entry field. Use as many digits as your calculator shows. Lightwave will then round the number.

Switch camera type to **Advanced camera**.

12.
Select the *View camera* as the **Ray start Item**. Select the *Lens plane* as the **Ray direction Item**. For **Ray direction** choose **Through item XY**.





13.
Make another test render. Both rendered images should be identical.

If they don't, repeat the previous steps.

14.
So far everything has been done to set up your rendering camera.

The big difference compared to a real view camera is the centre of the camera. While the centre of the real camera is somewhere in the centre of the lens, Lightwave's view camera centre is in the film plane/camera position. Although we called the Null object *Lens plane*, it works more like a mask in front of a camera.

For our purposes it doesn't matter, but keep this in mind for the future.

15.
The only step left is to render each segment. In our initial example there are 6 single areas.

To render the left column set the X value of the *Lens plane* to -1 m, for the centre column leave it at 0 m and for the right column set it to 1 m.

For the upper row it's a bit more difficult. Divide the **Height** of the *View camera* by its **Width** and then divide it by 2.

In our case 3.350 / 3.134 / 2 = 0,534460 ...

16.
All numbers in this tutorial are only examples.

It's necessary that you understand the principal of how to subdivide the full image area and how to move the *Lens plane* to all positions you need to create this puzzle which then can be put together in any retouching program.

Hopefully not too many problems will turn up by following my steps.

Good luck!

SUMMARY

In general the way I described is pretty much the same as using **Limited render area in** Lightwave. The crucial difference is the amount of memory each render will take. By rendering several smaller segments, the render buffers will need much less memory. And in the end the render buffers decide whether they won't to work or not. And in general they behave rather touchy and unreliable. As soon as you cross a certain pixel/poly count, you enter insecure render territory. You'll just find out in the very moment of trying to do your final image if Lightwave will cooperate or not.

By following the way I described it should be you who's finally smiling, and not Lightwave's render buffer.

Cheers

Thomas

mangoldphotography

CONTACT

PHOTOGRAPHER
THOMAS MANGOLD

IN CASE YOU WANT TO CONTACT ME, PLEASE SEND A MAIL TO @ THE ADDRESS THOMAS IN THE MANGOLD. ORANGE COM TYPE.

YOU ARE ALSO WELCOME TO VISIT THE WEBSITE IN THE WORLD WIDE WEB. THE NAME IS THOMAS AND MANGOLD. THE DOMAIN ENDS WITH A COM.

SORRY FOR THE INCONVENIENCE, BUT I GET FAR TOO MUCH JUNK MAIL THESE DAYS FROM SPIDERS AND BOTS WHICH PICKED UP MY ADDRESS SOMEWHERE.

COPYRIGHT