Directional Blur (like Photoshop's motion blur)
Posted by quality - 19 Jul 2012 04:14

So I really needed a directional blur for the project I'm working on and couldn't find one in Lightworks already (maybe I missed it?) Anyway, I tried my hand at writing my own and have something working although it is still clunky. I've never programmed in HLSL before and have mostly used python so it's pretty foreign to me. I don't know how to normalize my sliders so the values make sense. I really just need a slider with more decimal places for higher precision. I also tried to add a samples parameter but I'm guessing that it didn't like that inside the loop because I couldn't get it to compile.

Anyway, I thought I would throw this out there anyhow. if anyone has tips or feedback I'd gladly receive them. I'm hoping to polish this up and make it more useful.

And to give credit where it is due, this is hacked together from looking at many of the fx files posted here as well as the motion blur effect explained here:

windowsclient.net/wpf/wpf35/wpf-35sp1-more-effects.aspx

I called it motionblur because that is what I'm using the effect for, to give a feeling of a fast pan between cuts. (Think of the cuts you see on Food Network's "Chopped")

============================================================================
Re: Directional Blur (like Photoshop's motion blur)
Posted by quality - 19 Jul 2012 04:18

oops I tried to attach the .fx file and it didn't like that. I'll attach as .txt

============================================================================
Re: Directional Blur (like Photoshop's motion blur)
Posted by khaver - 25 Jul 2012 15:55

For the angle slider you can use 0.0 to 360.0 and in your main function use:

sincos(radians(Angle), offset.y, offset.x);

For the strength slider you could use 0.0 to 1.0 and in the main function use:
offset *= (Strength*0.005);

Since you're compiling to Pixel Shader 3.0, you can also increase the number of "Samples" to get smoother blur at higher strengths (60 works good), or you could do multiple technique passes.

Another thing you could do is add an if statement to bypass the processing if Strength = 0.0. This way the effect will allow better performance while the strength is at 0.0.

[code]if (Strength > 0.0)
{
    sincos(radians(Angle), offset.y, offset.x);
    offset *= (Strength*0.005);
    for(int i=0; i < 0.5;

    //-------------------------------------------------------------------------------

    if (Strength > 0.0) {
        sincos(radians(Angle), offset.y, offset.x);
        offset *= (Strength*0.005);
        for(int i=0; i

    =============================================================================

Re: Directional Blur (like Photoshop's motion blur)
Posted by quality - 26 Jul 2012 03:30

Wow, really great suggestions Khaver! I love the idea of adding the linear interp. I will put these in tonight and try them out. I haven't done a lot of shader programming since college (and that was renderman shaders) so I'm getting out of touch I'm afraid. I'll have to do some reading to get back up to speed.

============================================================================
Re: Directional Blur (like Photoshop's motion blur)
Posted by brdloush - 06 Feb 2013 20:08

I'm adding file containing original quality's motionblur effect with khaver's enhancements.

Rename to .fx and enjoy.

[edit] FIXED the motionblur.txt file

Re: Directional Blur (like Photoshop's motion blur)
Posted by quality - 06 Feb 2013 21:09

Sorry I meant to put up the finished result! Thanks for doing that brdloush.

Re: Directional Blur (like Photoshop's motion blur)
Posted by brdloush - 11 Feb 2013 00:43

Hi,

I like the directional (motion) blur effect, but it has one real downside. When you for example apply strength 15% strength and angle 0°, you will get smudged horizontal lines, the left part of the image will consist of ugly horizontal lines instead of blurry image.

Can we do something with that? I understand that there's no image information for the real blur to happen.. so maybe can the effect be rewritten in such way, that these areas will be "just blurred"?

Any other idea regarding how to compensate this glitch?

If this was fixed somehow, one could create quite nice "whip-wipe" (left-right) transitions..