Subject: Image fusion algorithm...
Posted by Gonzalo Rojas on Mon, 21 Oct 2002 14:22:43 GMT
View Forum Message <> Reply to Message

Hi:

Do you know what algorithm uses the image blend functionality of IDL 5.5?...

Best regards, and thanks in advance,

Gonzalo Rojas

Subject: Re: Image fusion algorithm...
Posted by Gonzalo Rojas on Thu, 24 Oct 2002 01:36:33 GMT
View Forum Message <> Reply to Message

"Karl Schultz" <kschultz@devnull.researchsystems.com> wrote in news:ap6iuf$4sk$1@news.rsinc.com:

>
> "Gonzalo Rojas" <rojas_gonzalo_news@hotmail.com> wrote in message
> news:3db5f86e$1_2@nova.entelchile.net...
>> "Dick Jackson" <dick@d-jackson.com> wrote in
>> news:1cWs9.546844$v53.23181338@news3.calgary.shaw.ca:
>>
>>> "Gonzalo Rojas" <rojas_gonzalo_news@hotmail.com> wrote in message
>>> news:3db40db3$1_2@nova.entelchile.net...
>>>> Hi:
>>>>
>>>> Do you know what algorithm uses the image blend functionality of
>>>> IDL
>>>> 5.5?...
>>>
>>> If you're asking about the Object Graphics class IDLgrImage and its
>>> blending function options, you'll find it in Online Help under
>>> "IDLgrImage:class Init method", "Blend_Function" keyword. There are
>>> several algorithms available.
>>>
>>> For a coherent explanation of this issue and much else, see Ronn
>>> Kling's excellent new book "Power Graphics with IDL".
>>> http://www.kilvarock.com/books/powergraphics.htm
>>>
>>> Cheers,
>>> --
>>> -Dick

>>>
>>> Dick Jackson / dick@d-jackson.com
>>> D-Jackson Software Consulting / http://www.d-jackson.com
>>> Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392
>>>
>>>
>>
>> Hi Dick:
>>
>> Yes... I am asking about the blending function of Object Graphics
>> class
>> IDLgrImage...
>>
>> I read the online help, but I need to know details such as: how the
>> algorithm does the blending between two true color images ?... Could
>> the algorithm do such blending ?...
>
> It really doesn't blend between two images. The image in question
> (the source) is blended with what is already on the screen (the
> destination). So, if you wanted to blend two images, you'd draw one to
> the screen with no blending and then draw the other image with your
> blend factor (alpha).
>
> Is your question about color components? Each component is blended
> independently according to the discussion in the docs for
> BLEND_FUNCTION.
>
> For example, a common choice for the blend function is (3,4) which
> produces traditional "alpha blending". This gives:
>
> C(dest)' = (ImageAlpha * C(image)) + ((1-ImageAlpha) * C(dest))
>
> This equation is applied to *each* of the R, G, and B components
> independently. "C" in the above equation means one of the color
> components. The docs call "C" a pixel, but it really means each color
> component or channel of the pixel.
>
> Note that you can't do blending in indexed mode.
>
> As far as the actual implementation goes, IDL doesn't actually do the
> blending. It tells OpenGL to perform the blending in the requested
> manner and then IDL tells OpenGL to write the image to the screen.
> OpenGL then modifies the pixels on the screen according to the
> specified blend function, the input (image) pixels and what's already
> on the screen.
>
> I still may not understand your question, but I hope that these
> explanations help.

>
> Karl
>
>
>
>

Dear Karl:

Yes... this is my question, and your explanation answered it...

Best regards,

Gonzalo Rojas
