Subject: Re: Zooming in place
Posted by JD Smith on Mon, 31 Oct 2005 22:05:30 GMT
View Forum Message <> Reply to Message

On Mon, 31 Oct 2005 10:15:29 -0700, David Fanning wrote:

> Pravish writes:
>
>> Zooming in place is one feature that does not seem to be popular with
>> IDL users. It is really strange that we are satisfied with the zoomed
>> image being in another window. That does not allow us to work with the
>> zoomed image. We can just view at the zoom image. I would like to zoom
>> on an image, in place and not in a separate window, and then work on
>> this image and if need be, zoom back. But, I am using a draw widget to
>> work with the image. That might seem to make things more complex.
>>
>> I have seen David's Zimage program and I think I could play around with
>> it and with some modifications, I will be able to use it.
>>
>> Any thoughts on this one?
>
> Zooming in place is one of those things that sound like a good idea until
> you start to implement it. Then you find it burdened with one constraint
> or another that make it more difficult to implement than you initially
> thought it was going to be.
>
> For example, if you allow the user to select the zoomed area, then you
> have problems of selection aspect ratio verses window aspect ratio. If you
> don't allow selection, then you have to determine how to allow the user to
> indicate selection, how much to zoom, etc.
>
> The biggest challenge, though, is going from a device coordinate system to
> the "data" coordinate system of the image in the zoomed window. It it not
> trivial to figure out where exactly you are in the image when you are in a
> display window.
>
> Once you have figured all this out once or twice, it is not something you
> really want to do again. So I would advise writing an image object that
> can do it all for you. Your image will need the ability to set up both a
> "normal" and "zoomed" coordinate system. You will need, probably, both the
> original image and a "display" image. (You might be able to create this on
> the fly.) You will need a method that can convert a point in window space
> to an image coordinate (the image "data" space) and visa versa, which
> means you will need an image that knows where it is in the display window.

Just my $0.02 on the "best" algorithm for in-place zooming. My viewer
(which might be released one of these days) lets you select a zoom region
with a rubber-band, or just click to double the zoom. The final selected

region is zoomed up in integer multiples to maximally fill the display window, and the rest of the window defaults to some neutral background color.  Single right clicks zoom you out one step, and double right-clicks zoom you out "all the way".  Works fairly well, in practice.

JD

---