Subject: Re: Opening and read .dat file double format
Posted by alghafisuct on Wed, 23 Dec 2015 08:49:24 GMT
View Forum Message <> Reply to Message

On Monday, December 21, 2015 at 3:17:33 AM UTC-8, Helder wrote:
> On Monday, December 21, 2015 at 11:04:11 AM UTC, algha...@gmail.com wrote:
>> On Monday, December 21, 2015 at 1:42:19 AM UTC-8, Helder wrote:
>>> On Monday, December 21, 2015 at 8:14:22 AM UTC, algha...@gmail.com wrote:
>>>> On Sunday, December 20, 2015 at 9:52:23 AM UTC-8, Helder wrote:
>>>> > On Sunday, December 20, 2015 at 4:42:19 PM UTC, algha...@gmail.com wrote:
>>>> > > On Friday, December 18, 2015 at 7:22:10 AM UTC-8, Craig Markwardt wrote:
>>>> > > > On Friday, December 18, 2015 at 7:26:07 AM UTC-5, algha...@gmail.com wrote:
>>>> > > > > Hi everyone
>>>> > > > >
>>>> > > > > I have a .dat file double and is 5000 by 8039. I'm using the file in Matlab and I want to try to open it in IDL. So I know it is very basic question but I'm new in IDL and I want to start using it. After I open the file, I want to read like the value of (1,1)
>>>> > > >
>>>> > > > What format is the data in? ASCII? Binary? Matlab format? It makes a difference.
>>>> > >
>>>> > > Thanks Craig
>>>> > >
>>>> > > My data is Binary double format and I used the following because I want to extact the image to see as I'm doing in Matlab
>>>> > >
>>>> > > template = BINARY_TEMPLATE('C:\Users\asus\Desktop\Metasensing\IDL_PROJECTS\20151021115853_11_SAR_CPLX_0_pres_8.dat')
>>>> > > Result = READ_BINARY('C:\Users\asus\Desktop\Metasensing\IDL_PROJECTS\20151021115853_11_SAR_CPLX_0_pres_8.dat', DATA_DIMS = [4999, 8038])
>>>> > > im = IMAGE(Result)
>>>> > > end
>>>> > >
>>>> > > But I got an image meaning nothing to me only white and black image
>>>> > >
>>>> > > Am I doing the right way or there are other ways to see the image
>>>> > >
>>>> > > Thanks
>>>> >
>>>> > Difficult to tell.
>>>> > My suggestion would be to try:
>>>> > template = BINARY_TEMPLATE('C:\Users\asus\Desktop\Metasensing\IDL_PROJECTS\20151021115853_11_SAR_CPLX_0_pres_8.dat')
>>>> > Result = READ_BINARY('C:\Users\asus\Desktop\Metasensing\IDL_PROJECTS\20151021115853_11_SAR_CPLX_0_pres_8.dat', DATA_DIMS = [4999, 8038])
>>>> > mn = min(Result, max=mx)
>>>> > print, mn, mx
>>>> > im = IMAGE(Result)
>>>> > im.min_value = mn

>>>> > im.max_value = mx
>>>> >
>>>> > Any better? If not, what does "print, mn, mx" show? If you're getting the same value, then there must a problem with the import (read_binary()). (*)
>>>> >
>>>> > Cheers,
>>>> > Helder
>>>> >
>>>> > (*) - or the min/max values are not evenly distributed. Meaning that if one pixel is 0, one is 1000 and the rest of your data is distributed between 500 and 501... then you won't see much unless you set min_value and max_value at 500 and 501. You could also try to have a look at the histogram distribution.
>>>>
>>>> Hi
>>>>
>>>> I got the same thing and the min vale is 0 and the max value is 255
>>>>
>>>> any suggestions ?
>>>
>>> I just noticed that you import using BINARY_TEMPLATE() and READ_BINARY(). I never used these functions because I normally know how my template looks like and use readu directely. I'll try to do some guess work: it seems to me as if you're missing something. With binary_template you generate a structure for importing data that you call "template". This template should be used in READ_BINARY() to import the data, but you don't use it.
>>> Why don't you try:
>>> Result = READ_BINARY('C:\Users\asus\Desktop\Metasensing\IDL_PROJECTS\20151021115853_11_SAR_CPLX_0_pres_8.dat', template=template, DATA_DIMS = [4999, 8038])
>>>
>>> and see what happens.
>>>
>>> If that doesn't work out. Do you know how the data is stored? If it's a simple array of doubles, you could use something like:
>>>
>>> array = make_array(4999, 8038, /double, /nozero)
>>> filename = 'C:\....' ;your file name
>>> OPENR, importUnit, filename, /GET_LUN
>>> READU, importUnit, array
>>> CLOSE, importUnit
>>> FREE_LUN, importUnit
>>> im = IMAGE(array)
>>>
>>> cheers,
>>> Helder
>>
>> Thank you very much it's good know. I tried the OPENR way it's double.
>>
>> On my MATLAB code I seprated real and imaginary such as the following

>>
>> array = array (1:2:end, :) + j * array (2:2:end, :);
>>
>> how can that be done in IDL so I can have imagenary and real values
>>
>> Thanks
>
> I don't know Matlab. But I guess (yes, it's a guess...) that those Matlab commands put every odd position value in the real part and even position values in the imaginary part. If so, then I would try with:
> array = make_array(4999, 8038, /dcomplex, /nozero)
> This is also a guess... but I suppose that imaginary data is written this way.
> Try this and then display the image as:
>
> im_r = IMAGE(real_part(array))
> im_i = IMAGE(imaginary(array))
>
> and you should see real and imaginary part displayed. Then you can play around and use other useful commands for imaginary data: abs() and atan(). See the help for details:
> http://www.exelisvis.com/docs/ABS.html
> http://www.exelisvis.com/docs/ATAN.html with the phase keyword.
>
> Cheers,
> Helder

Thanks Helder for the help

I tried print,ABS(image(array)) and im_r = IMAGE(real_part(array)) and im_i = IMAGE(imaginary(array)) to see images but I'm getting the following :

% IMAGE: Unable to allocate memory: to make array.

if the problem is the memory is not enough, how can I get a small subset of the image and save it as a new file then I can see if it is working

Thanks
