Subject: Complementary Color with XOBJVIEW
Posted by M R on Fri, 19 Aug 2011 20:25:28 GMT
View Forum Message <> Reply to Message

Hi

I am trying to display the output in white color instead of black as being currently displayed in black in XOBJVIEW window. The image is the maximum intensity pixel display in 3D (I can rotate the image, etc). But as white color should be 255, 255, 255 in the 3D array, why are the maximum pixels being displayed as black when I run the program? How do I display the maximum intensity pixels in white color? If its helps, I have posted the code below. Xinteranimate is displayed as white slices on black background (default).

```
fil = file_search('filepath*', COUNT = count)

image = read_dicom(fil[0])

s = size(image)

arm = make_array(s[1],s[2],count,/nozero)

xinteranimate, set = [2*s[1],2*s[2], count], /showload

    for i=0,count-1 do begin

      image = read_dicom(fil[i])

      arm[*,*,i] = image[*,*]

      xinteranimate, frame = I, image = arm[*,*,I]

     myvolume = OBJ_NEW('IDLgrVolume', bytscl(arm, min = 50, max =
200))

      myvolume1 = OBJ_NEW('IDLgrVolume', bytscl(arm))

    end

    xinteranimate, /keep_pixmaps

    XOBJVIEW, myvolume

    xOBJVIEW, myvolume1

     end
```

Subject: Re: Complementary Color with XOBJVIEW
Posted by Karl[1] on Fri, 19 Aug 2011 22:55:59 GMT
View Forum Message <> Reply to Message

On Aug 19, 3:01 pm, M R <manisha....@gmail.com> wrote:
> On Aug 19, 3:47 pm, David Fanning <n...@idlcoyote.com> wrote:
>
>
>
>> M R writes:
>>> I am trying to display the output in white color instead of black as
>>> being currently displayed in black in XOBJVIEW window. The image is
>>> the maximum intensity pixel display in 3D (I can rotate the image,
>>> etc). But as white color should be 255, 255, 255 in the 3D array, why
>>> are the maximum pixels being displayed as black when I run the
>>> program? How do I display the maximum intensity pixels in white color?
>>> If its helps, I have posted the code below. Xinteranimate is displayed
>>> as white slices on black background (default).
>
>> Sigh...
>
>> I know you have been working on this project for a long
>> time, but I think you are still doing nearly everything
>> wrong. :-)
>
>> The volume array you are building doesn't have
>> any "color" associated with it. But, maybe you
>> are now at least building the volume array
>> correctly. Are the images, in fact, slices from
>> a volume?
>
>> If so, maybe you could use the code found here:
>
>> http://www.idlcoyote.com/ip_tips/mip.html
>
>> Just load your volume array in the place where
>> it says "Load the data."
>
>> Cheers,
>
>> David
>
>> --
>> David Fanning, Ph.D.
>> Fanning Software Consulting, Inc.
>> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
>
> Thank you for remembering! But I can view the output, control the

> animation slice by slice and everything. Instead of a Black colored
> image in my XOBJVIEW window, I would like to convert it into white
> color. I just want to make a complementary image. Why would I need
> color information from the image for this? I have the pixel values
> where the maximum is 295. Instead of displaying this particular pixel
> as a black colored pixel, can I assign a color for it to be displayed
> in white color? Thank you for the help and tips!

It looks like you are working with single channel image data, and
that's greyscale if there is no palette attached. No color.

First, I would move

myvolume = OBJ_NEW('IDLgrVolume', bytscl(arm, min = 50, max =
200))

myvolume1 = OBJ_NEW('IDLgrVolume', bytscl(arm))

outside of the loop. You are creating count * 2 volumes and only
using 2 of them. This is a horrible memory/object leak. However,
fixing this doesn't change your results.

I would also suggest byte-scaling your input data for your slices the
same way you are doing so for the volume:

xinteranimate, frame = I, image = bytscl(arm[*,*,I])

I don't know what type 'arm' is, but you said the max is 295, so it
must be a data type bigger than a byte. I also don't know exactly
what xinteranimate does with a single channel image if the data type
and/or input range is greater than what a byte can store. But if you
bytscl it, you at least are giving it a reasonable single-channel
greyscale image as input.

I'm guessing that your max 295 values that you wanted to be white got
mapped into something else or dropped. Byte-scaling the input may
help that.

Actually, doing the bytscl one slice at a time might be bogus if the
min and max in each slice is not constant. You may end up with
different amounts of stretching/scaling in each slice. So, you might
want:

```
fil = file_search('filepath*', COUNT = count)
image = read_dicom(fil[0])
s = size(image)
arm = make_array(s[1],s[2],count,/nozero)
xinteranimate, set = [2*s[1],2*s[2], count], /showload
```

```
for i=0,count-1 do begin
   image = read_dicom(fil[i])
   arm[*,*,i] = image[*,*]
end
arm_scaled = bytscl(arm, min = 50, max = 200)
arm_scaled1 = bytscl(arm)
for i=0,count-1 do begin
   xinteranimate, frame = I, image = arm_scaled[*,*,I]
end
myvolume = OBJ_NEW('IDLgrVolume', arm_scaled)
myvolume1 = OBJ_NEW('IDLgrVolume', arm_scaled1)
xinteranimate, /keep_pixmaps
XOBJVIEW, myvolume
xOBJVIEW, myvolume1
end
```

Or something like that; I can't really tell what the actual intent is. I'm guessing that you just want xinteranimate to give you a slice-by-slice view of the volume. I think that something close to the above will do that.

Subject: Re: Complementary Color with XOBJVIEW
Posted by David Fanning on Fri, 19 Aug 2011 22:59:21 GMT
View Forum Message <> Reply to Message

M R writes:

> Thank you for remembering! But I can view the output, control the
> animation slice by slice and everything. Instead of a Black colored
> image in my XOBJVIEW window, I would like to convert it into white
> color. I just want to make a complementary image. Why would I need
> color information from the image for this? I have the pixel values
> where the maximum is 295. Instead of displaying this particular pixel
> as a black colored pixel, can I assign a color for it to be displayed
> in white color? Thank you for the help and tips!

Well, keep in mind that the graphics system used by XObjView is *completely and utterly* different from the graphics system used by XInteranimate (object graphics verses direct graphics). And you can't mix and match.

To get different colors in XObjView, you are going to have to investigate the RBG_TABLE* keywords for the IDLgrVolume object you are passing to XObjView.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Complementary Color with XOBJVIEW
Posted by M R on Mon, 22 Aug 2011 14:17:50 GMT
View Forum Message <> Reply to Message

On Aug 19, 5:55 pm, Karl <karl.w.schu...@gmail.com> wrote:
> On Aug 19, 3:01 pm, M R <manisha....@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>
>> On Aug 19, 3:47 pm, David Fanning <n...@idlcoyote.com> wrote:
>
>>> M R writes:
>>>> I am trying to display the output in white color instead of black as
>>>> being currently displayed in black in XOBJVIEW window. The image is
>>>> the maximum intensity pixel display in 3D (I can rotate the image,
>>>> etc). But as white color should be 255, 255, 255 in the 3D array, why
>>>> are the maximum pixels being displayed as black when I run the
>>>> program? How do I display the maximum intensity pixels in white color?
>>>> If its helps, I have posted the code below. Xinteranimate is displayed
>>>> as white slices on black background (default).
>
>>> Sigh...
>
>>> I know you have been working on this project for a long
>>> time, but I think you are still doing nearly everything
>>> wrong. :-)
>
>>> The volume array you are building doesn't have
>>> any "color" associated with it. But, maybe you
>>> are now at least building the volume array

>>> correctly. Are the images, in fact, slices from
>>> a volume?
>
>>> If so, maybe you could use the code found here:
>
>>> http://www.idlcoyote.com/ip_tips/mip.html
>
>>> Just load your volume array in the place where
>>> it says "Load the data."
>
>>> Cheers,
>
>>> David
>
>>> --
>>> David Fanning, Ph.D.
>>> Fanning Software Consulting, Inc.
>>> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
>>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
>
>> Thank you for remembering! But I can view the output, control the
>> animation slice by slice and everything. Instead of a Black colored
>> image in my XOBJVIEW window, I would like to convert it into white
>> color. I just want to make a complementary image. Why would I need
>> color information from the image for this? I have the pixel values
>> where the maximum is 295. Instead of displaying this particular pixel
>> as a black colored pixel, can I assign a color for it to be displayed
>> in white color? Thank you for the help and tips!
>
> It looks like you are working with single channel image data, and
> that's greyscale if there is no palette attached. No color.
>
> First, I would move
>
> myvolume = OBJ_NEW('IDLgrVolume', bytscl(arm, min = 50, max =
> 200))
>
> myvolume1 = OBJ_NEW('IDLgrVolume', bytscl(arm))
>
> outside of the loop. You are creating count * 2 volumes and only
> using 2 of them. This is a horrible memory/object leak. However,
> fixing this doesn't change your results.
>
> I would also suggest byte-scaling your input data for your slices the
> same way you are doing so for the volume:
>
> xinteranimate, frame = I, image = bytscl(arm[*,*,I])
>

> I don't know what type 'arm' is, but you said the max is 295, so it
> must be a data type bigger than a byte. I also don't know exactly
> what xinteranimate does with a single channel image if the data type
> and/or input range is greater than what a byte can store. But if you
> bytscl it, you at least are giving it a reasonable single-channel
> greyscale image as input.
>
> I'm guessing that your max 295 values that you wanted to be white got
> mapped into something else or dropped. Byte-scaling the input may
> help that.
>
> Actually, doing the bytscl one slice at a time might be bogus if the
> min and max in each slice is not constant. You may end up with
> different amounts of stretching/scaling in each slice. So, you might
> want:
>
> fil = file_search('filepath*', COUNT = count)
> image = read_dicom(fil[0])
> s = size(image)
> arm = make_array(s[1],s[2],count,/nozero)
> xinteranimate, set = [2*s[1],2*s[2], count], /showload
> for i=0,count-1 do begin
> image = read_dicom(fil[i])
> arm[*,*,i] = image[*,*]
> end
> arm_scaled = bytscl(arm, min = 50, max = 200)
> arm_scaled1 = bytscl(arm)
> for i=0,count-1 do begin
> xinteranimate, frame = I, image = arm_scaled[*,*,I]
> end
> myvolume = OBJ_NEW('IDLgrVolume', arm_scaled)
> myvolume1 = OBJ_NEW('IDLgrVolume', arm_scaled1)
> xinteranimate, /keep_pixmaps
> XOBJVIEW, myvolume
> xOBJVIEW, myvolume1
> end
>
> Or something like that; I can't really tell what the actual intent
> is. I'm guessing that you just want xinteranimate to give you a slice-
> by-slice view of the volume. I think that something close to the
> above will do that.

Hi

Thank you for the correction. The end goal of the program is to output a maximum intensity projection of the 3D array. They 3D array contains mri scan images. The output of the Xinteranimate window allow me to view the output slice by slice and I have no problems with that what

so ever. In the XOBJVIEW window, when I am supposed to pick the maximum intensity pixels only, the code seems to be doing so. But the selected pixels are black in color. Shouldn't the maximum intensity pixels be of (255,255,255) intensity indicating white color? Why are they being displayed as black colored pixels?

I have tried the above given code and the slices in Xinteranimate window are being displayed as required (white color pixels are showing up and the rest of the image is black). But this isn't happening, instead I see the maximum intensity pixels displayed in black color.

One solution for this problem would be to take the compliment of the black colored image and display it in white color. How do I do that with the XOBJVIEW window?
