
Subject: Inverting the image

Posted by [idlfreak](#) on Wed, 12 Dec 2001 19:23:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

HI,

I have an image of size(512 x 512 x 195). Its a CT image of chest and the image appears with the backbone in the upward direction and the sternum in the downward direction. I want to get it inverted.

For example:

```
filename = Filepath(SubDirectory = ['examples', 'data'],  
'worldelv.dat')  
image= BytArr(360, 360)  
openR, lun, filename  
ReadU, lun, image  
Free_lun, lun
```

Tvscl, image

I want the image reversed. The image should be rotated by 180 degrees.
I want the original data to be inverted and not the final display.

Can anybody help me with this please?

Any help is appreciated.

Regards,
Akhila.

Subject: Re: Inverting the image

Posted by [John-David T. Smith](#) on Wed, 19 Dec 2001 22:33:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Akhila wrote:

>

> HI,

> I have an image of size(512 x 512 x 195). Its a CT image of chest and
> the image appears with the backbone in the upward direction and the
> sternum in the downward direction. I want to get it inverted.

> For example:

>

> filename = Filepath(SubDirectory = ['examples', 'data'],

> 'worldelv.dat')

> image= BytArr(360, 360)

> openR, lun, filename

> ReadU, lun, image

> Free_lun, lun

>
> Tvscl, image

How about "1/image". Pretty inverted there.

But seriously, you don't really say what you mean by "inverted", which is a concept which applies quantitatively only to colormaps, where it means "reverse the order of the colormap entries". Maybe you want something like:

image=max(image)-image

JD

Subject: Re: Inverting the image
Posted by [David Fanning](#) on Wed, 19 Dec 2001 22:53:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith (jdsmith@astro.cornell.edu) writes:

> Akhila wrote:
>>
>> HI,
>> I have an image of size(512 x 512 x 195). Its a CT image of chest and
>> the image appears with the backbone in the upward direction and the
>> sternum in the downward direction. I want to get it inverted.
>> For example:
>>
>> filename = Filepath(SubDirectory = ['examples', 'data'],
>> 'worldelv.dat')
>> image= BytArr(360, 360)
>> openR, lun, filename
>> ReadU, lun, image
>> Free_lun, lun
>>
>> Tvscl, image
>
> How about "1/image". Pretty inverted there.
>
> But seriously, you don't really say what you mean by "inverted", which
> is a concept which applies quantitatively only to colormaps, where it
> means "reverse the order of the colormap entries". Maybe you want
> something like:
>
> image=max(image)-image

I think more likely:

TVSCL, image, Order=1

Or,

TVSCL, Reverse(image, 2)

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438, E-mail: david@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Inverting the image

Posted by [John-David T. Smith](#) on Wed, 19 Dec 2001 23:20:22 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

>

> JD Smith (jdsmith@astro.cornell.edu) writes:

>

>> Akhila wrote:

>>>

>>> HI,

>>> I have an image of size(512 x 512 x 195). Its a CT image of chest and
>>> the image appears with the backbone in the upward direction and the
>>> sternum in the downward direction. I want to get it inverted.

>>> For example:

>>>

>>> filename = Filepath(SubDirectory = ['examples', 'data'],

>>> 'worldelv.dat')

>>> image= BytArr(360, 360)

>>> openR, lun, filename

>>> ReadU, lun, image

>>> Free_lun, lun

>>>

>>> Tvscl, image

>>

>> How about "1/image". Pretty inverted there.

>>

>> But seriously, you don't really say what you mean by "inverted", which
>> is a concept which applies quantitatively only to colormaps, where it
>> means "reverse the order of the colormap entries". Maybe you want

```
>> something like:
>>
>> image=max(image)-image
>
> I think more likely:
>
>   TVSCL, image, Order=1
>
> Or,
>
>   TVSCL, Reverse(image, 2)
```

Ahah... that's what I get for not reading ahead. Apologies.

JD

Subject: Re: Inverting the image
Posted by [James Kuyper Jr.](#) on Thu, 20 Dec 2001 00:12:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith wrote:

```
> Akhila wrote:
>
>> HI,
>> I have an image of size(512 x 512 x 195). Its a CT image of chest and
>> the image appears with the backbone in the upward direction and the
>> sternum in the downward direction. I want to get it inverted.
```

...

```
> But seriously, you don't really say what you mean by "inverted", which
> is a concept which applies quantitatively only to colormaps, where it
```

There are many different possible meanings of "inverted"; the term doesn't just apply to colormaps. To the geometrically inclined, the meaning that seems most likely to apply in this case is the transformation that replaces a function $\text{image}(x,y,z)$ with $\text{image}(-x,-y,-z)$. Assuming that $x=0,y=0,z=0$ corresponds to the center of his array, then in IDL terms that becomes:

```
invert = reverse(reverse(reverse(image, 1),2),3)
```

Subject: Re: Inverting the image
Posted by [idlfreak](#) on Fri, 28 Dec 2001 07:17:05 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

i think my question was not clear. The problem i have is not in inverting the display. The code that is attached performs the segmentation of image and saves the segmented image and then reads them too. The problem is, when i mark the region of interest in the image, the segmented image is not the region of inverted display but of the original image and so i don't get the region that i want. I've inverted the world map image and displayed it. use the mouse to mark the region of interest. you can see that the segmented image is not what was marked but what was there in the original data.

```
*****
```

```
PRO Select_Button_Events, event
```

```
Widget_Control, event.top, Get_UValue=state
```

```
possibleEvents = ['DOWN', 'UP', 'MOTION', 'SCROLL', 'EXPOSE']
```

```
possibleButtons = ['NONE', 'LEFT', 'MIDDLE', 'NONE', 'RIGHT']
```

```
thisEvent = possibleEvents(event.type)
```

```
thisButton = possibleButtons(event.press)
```

```
;kl = 110
```

```
CASE thisEvent OF
```

```
    'EXPOSE': BEGIN
```

```
        WIDGET_CONTROL, event.top, GET_UVALUE=state
```

```
        state.oWindow->Draw, state.oView
```

```
    ENDCASE
```

```
    'DOWN': BEGIN
```

```
        ;print, 'im in mouse press'
```

```
        CASE thisButton OF
```

```
            'LEFT': BEGIN
```

```
                oROI = state.oCurrROI
```

```
                if (OBJ_VALID(oROI) eq 0) then begin
```

```
                    oOldSelROI = state.oSelROI
```

```
                    if (OBJ_VALID(oOldSelROI) ne 0) then
```

```
oOldSelROI->SetProperty, COLOR=[0,0,255]
```

```
                    oROI = OBJ_NEW('IDLgrROI', COLOR=[255,0,0], STYLE=1 )
```

```
                    state.oCurrROI = oROI
```

```
                    state.oModel->Add, oROI
```

```
                endif
```

```
                oROI->AppendData, [event.x, event.y, 0]
```

```
state.oWindow->Draw, state.oView
state.bButtonDown = 1b
```

END

'RIGHT': BEGIN

```
if (event.modifiers eq 1) then begin
state.oModel -> SetProperty, HIDE = 1
;kl = kl - 5
state.k = state.k - 5
state.image = state.v2(*,*,state.k-5)
state.oImage = Obj_New('IDLgrImage', state.image)
state.oModel = Obj_New('IDLgrModel', HIDE = 0)
```

```
state.oModel -> add, state.oImage
state.oView -> add, state.oModel
state.oWindow -> Draw, state.oView
```

```
endif else begin
state.oModel -> Scale, 0.95, 0.95, 1.00
state.oWindow -> Draw, state.oView
endelse
```

END

'MIDDLE': BEGIN

```
state.oModel -> Scale, 1.05,1.05,1.00
state.oWindow -> Draw, state.oView
```

END

ELSE: BEGIN

END

ENDCASE

ENDCASE

'UP': BEGIN

```
;print, 'im in mouse release'
WIDGET_CONTROL, event.top, GET_UVALUE=state
oROI = state.oCurrROI
if (OBJ_VALID(oROI) EQ 0) then return
if (state.bButtonDown eq 1) then begin
; End ROI
oROI->GetProperty, DATA=roiData
if ((N_ELEMENTS(roiData)/3) ge 3) then begin
oROI->SetProperty, STYLE=2
```

```
state.oWindow->Draw, state.oView
state.oModel->Remove, oROI
state.oCurrROI = OBJ_NEW()
```

```
mask = oROI->ComputeMask(DIMENSIONS = [state.xdim, state.ydim],
MASK_RULE=2)
```

```
final = (mask) and (state.image)
help, final
region = search2d(final, 10,10,0,90)
help, region
final[region] = 0B
state.oModel -> SetProperty, HIDE = 1
state.oImage = Obj_New('IDLgrImage', final)
state.oModel = Obj_new('IDLgrModel', HIDE = 0)
state.oModel -> Add, state.oImage
state.oView -> Add, state.oModel
state.oWindow -> Draw, state.oView
state.oWindow-> Getproperty, IMAGE_DATA = myimage
res = DIALOG_WRITE_IMAGE(final, filename = 'segim1.tif')
```

```
result = DIALOG_READ_IMAGE(IMAGE = Segim)
state.oImage = Obj_New('IDLgrImage', Segim)
state.oModel = Obj_New('IDLgrModel')
state.oModel ->Add, state.oImage
state.oView -> Add, state.oModel
state.oWindow -> Draw, state.oView
;help, myimage
;s = replicate(1,3,3)
;res = ERODE(myimage,s)
;help, res
;state.oImage = Obj_New('IDLgrImage', res)
;state.oModel = Obj_New('IDLgrModel')
;state.oModel -> Add, state.oImage
;state.oView -> Add, state.oModel
;state.oWindow -> Draw, state.oView
```

```
;tv, res
;write_tiff, 'myfile1.tiff', myimage
;WRITE_IMAGE, 'mytrial.dcm', 'DICOM', myimage
```

```
; Reset button down state.
state.bButtonDown = 0b
```

```
endif else begin
; Fewer than 3 vertices; delete.
state.oModel->Remove, oROI
OBJ_DESTROY, oROI
```

```

state.oCurrROI = OBJ_NEW()

; Reset color of formerly selected ROI.
oOldSelROI = state.oSelROI
if (OBJ_VALID(oOldSelROI) ne 0) then oOldSelROI->SetProperty,
COLOR=[0,0,255]
state.oWindow->Draw, state.oView

; Reset button down state.
state.bButtonDown = 0b

endelse
endif

ENDCASE

'MOTION': BEGIN

WIDGET_CONTROL, event.top, GET_UVALUE=state

state.oImage->GetProperty, DIMENSIONS=dimensions
oROI = state.oCurrROI
if (OBJ_VALID(oROI) EQ 0) then return

; If button down, append a vertex.
if (state.bButtonDown NE 0) then begin
oROI->AppendData, [event.x, event.y]
state.oWindow->Draw, state.oView
;state.bTempSegment = 1b
endif

;ENDIF
ENDCASE

ELSE: BEGIN
ENDCASE

ENDCASE

Widget_Control, event.top, Set_UValue=state
END

;-----

PRO trial6_Resize, event

Widget_Control, event.top, Get_UValue = state

```

```
Widget_Control, state.drawID, Draw_XSize = event.x, Draw_YSize =  
event.y
```

```
state.xsize = event.x  
state.ysize = event.y  
state.oWindow -> Draw, state.oScene
```

```
Widget_Control, event.top, Set_UValue = state
```

```
END
```

```
;-----
```

```
PRO trial6
```

```
filename = FilePath(SubDirectory = ['examples', 'data'],  
'worldelv.dat')  
image = BytArr(360,360)  
OpenR, lun, filename, /Get_Lun  
ReadU, lun, image  
Free_Lun, lun
```

```
xdim = 360  
ydim = 360
```

```
;imagefin = invert(image)  
;imagefin = bytarr(imagefin)  
;help, imagefin
```

```
;restore, filename = 'trial'  
;v2 = bytscl(v1)  
;k = 110  
;image = v2(*,*,k)
```

```
tlb = Widget_Base(/COLUMN)  
drawID = Widget_Draw(tlb, /BUTTON_EVENTS, /MOTION_EVENTS,  
/EXPOSE_EVENTS, Retain = 0, GRAPHICS_LEVEL = 2, $  
XSize = xdim, YSize = ydim, EVENT_PRO ='Select_Button_Events')  
Widget_Control, tlb, /REALIZE  
Widget_Control, drawID, Get_Value = oWindow
```

```
oView = Obj_New('IDLgrView', VIEWPLANE_RECT = [0,0,xdim,ydim], COLOR =  
[0,0,0])  
oImage = Obj_New('IDLgrImage', image, /ORDER)  
oModel = Obj_New('IDLgrModel', Select_Target = 1)  
oModel -> Add, oImage  
oView -> Add, oModel
```

```
oROIModel = OBJ_NEW('IDLgrModel')
oROIGroup = OBJ_NEW('IDLanROIGroup')
oModel -> Add, oROIModel
help, oROIModel
oWindow -> Draw, oView
```

```
state = {oWindow:oWindow, $
  oModel:oModel, $
  oView:oView, $
  image:image, $
  oImage:oImage, $
  oCurrROI:OBJ_NEW(), $
  oSelROI:OBJ_NEW(), $
  bButtonDown:0B, $
  oROIModel:OBJ_NEW(), $
  oROIGroup:OBJ_NEW(), $
  xsize:0, $
  ysize:0, $
  xdim:xdim, $
  ydim:ydim}
;v2:v2, $
;k:k}
```

```
Widget_Control, tlb, Set_UValue = state
XManager, 'trial6', tlb, /No_Block, Event_Handler = 'trial6_Resize'
END
,*****
```

I think i've clarified my question. Incase i'm still not clear please let me know. Any help is appreciated.

-Regards,
Akhila

"James Kuyper Jr." <James.R.Kuyper.1@gsfc.nasa.gov> wrote in message news:<3C212CFF.3060602@gsfc.nasa.gov>...

> JD Smith wrote:

>

>> Akhila wrote:

>>

>>> HI,

>>> I have an image of size(512 x 512 x 195). Its a CT image of chest and
>>> the image appears with the backbone in the upward direction and the
>>> sternum in the downward direction. I want to get it inverted.

> ...

>

>> But seriously, you don't really say what you mean by "inverted", which

>> is a concept which applies quantitatively only to colormaps, where it
>
> There are many different possible meanings of "inverted"; the term
> doesn't just apply to colormaps. To the geometrically inclined, the
> meaning that seems most likely to apply in this case is the
> transformation that replaces a function image(x,y,z) with
> image(-x,-y,-z). Assuming that x=0,y=0,z=0 corresponds to the center of
> his array, then in IDL terms that becomes:
>
> invert = reverse(reverse(reverse(image, 1),2),3)

Subject: Re: Inverting the image
Posted by [David Fanning](#) on Fri, 28 Dec 2001 18:03:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Akhila (idlfreak@yahoo.com) writes:

> i think my question was not clear. The problem i have is not in
> inverting the display.

Right, don't invert the display, invert the image. In place
of this code:

```
olImage = Obj_New('IDLgrImage', image, /ORDER)
```

Substitute this:

```
image = Reverse(image, 2)  
olImage = Obj_New('IDLgrImage', image)
```

Cheers,

David

--

David W. Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155
