
Subject: display gray image with NaN

Posted by [dashtabadi.m.r](#) on Fri, 03 Mar 2017 13:55:24 GMT

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

Hello everybody

i have a gray scale image (one band with NaN value) and display it with image() and widget_window().

i want to show specific values (maybe gt 1000 pixels) over image with red color.

like this. <http://pasteboard.co/EYXxQWAAW.jpg>

Thank you for your help.

Regards

Subject: Re: display gray image with NaN

Posted by [jpleahy2](#) on Sat, 04 Mar 2017 10:00:33 GMT

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

On Friday, March 3, 2017 at 1:55:26 PM UTC, dashtab...@gmail.com wrote:

> Hello everybody

>

> i have a gray scale image (one band with NaN value) and display it with image() and widget_window().

> i want to show specific values (maybe gt 1000 pixels) over image with red color.

> like this. <http://pasteboard.co/EYXxQWAAW.jpg>

>

> Thank you for your help.

> Regards

1) If required, define a colour scale to map your raw data values to pixel values, e.g. for a greyscale RGB should all be the same:

offset = MIN(raw_data)

scale = 255/(1000-offset)

data = BYTE(REBIN(scale*(raw_data - offset),N,M,3))

(doesn't have to be bytes with IMAGE(), but most display devices can only cope with 256 levels per RGB so you might as well).

2) Choose a colour well away from the main colour map to represent NaNs and saturated pixels.

nan_colour = [0,0,1] ; blue NaN

sat_colour = [1,0,0] ; red for values > 1000

nanpix = WHERE(FINITE(raw_data))

3) Flatten array temporarily so the indexing from WHERE will work:

data = REFORM(data, N*M,3)

data[nanpix,*] = nan_colour

satpix = WHERE(raw_data GT 1000)

```
data[satpix,*] = sat_colour
data = REFORM(data,N,M,3)
```

4) Display, with options as required:
graphic = IMAGE(data)

Subject: Re: display gray image with NaN
Posted by [Nikola](#) on Sat, 04 Mar 2017 10:02:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Friday, March 3, 2017 at 1:55:26 PM UTC, dashtab...@gmail.com wrote:

```
> Hello everybody
>
> i have a gray scale image (one band with NaN value) and display it with image() and
widget_window().
> i want to show specific values (maybe gt 1000 pixels ) over image with red color.
> like this. http://pasteboard.co/EYXxQWAAW.jpg
>
> Thank you for your help.
> Regards
```

It is not that trivial. What I normally do is to rescale all the non-NaN values in dataset to the range [0, 254] and assign 255 for the NaN's pixels. Then I make colorscale that has the grey tones between 0 and 254 and red in 255:

```
r = [FINDGEN(255), 255]
g = [FINDGEN(255), 0]
b = [FINDGEN(255), 0]
```

This should do the trick.

Subject: Re: display gray image with NaN
Posted by [Markus Schmassmann](#) on Mon, 06 Mar 2017 10:02:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

Am 04.03.2017 um 11:00 schrieb jpleahy2@gmail.com:

```
> On Friday, March 3, 2017 at 1:55:26 PM UTC, dashtab...@gmail.com wrote:
>> i have a gray scale image (one band with NaN value) and display it with image() and
widget_window().
>> i want to show specific values (maybe gt 1000 pixels ) over image with red color.
>> like this. http://pasteboard.co/EYXxQWAAW.jpg
>
> 1) If required, define a colour scale to map your raw data values to pixel values, e.g. for a
greyscale RGB should all be the same:
> offset = MIN(raw_data)
> scale = 255/(1000-offset)
```

```

> data = BYTE(REBIN(scale*(raw_data - offset),N,M,3))
>
> (doesn't have to be bytes with IMAGE(), but most display devices can only cope with 256 levels
per RGB so you might as well).
>
> 2) Choose a colour well away from the main colour map to represent NaNs and saturated
pixels.
> nan_colour = [0,0,1] ; blue NaN
> sat_colour = [1,0,0] ; red for values > 1000
> nanpix = WHERE(FINITE(raw_data))
>
> 3) Flatten array temporarily so the indexing from WHERE will work:
> data = REFORM(data, N*M,3)
> data[nanpix,*] = nan_colour
> satpix = WHERE(raw_data GT 1000)
> data[satpix,*] = sat_colour
> data = REFORM(data,N,M,3)
>
> 4) Display, with options as required:
> graphic = IMAGE(data)

```

this code fails, if raw_data[0] is a NaN. use

```
offset = MIN(raw_data,/nan)
```

for the first line to be safe.
