Subject: Re: Composition structure
Posted by James Kuyper on Fri, 01 Oct 2004 18:49:44 GMT
View Forum Message <> Reply to Message

Jeferson E. wrote:
> Hello there!
>
> Assuming I have 5 images and I want to get the maximum value
> composition. I can use something like that:
>
> Composite = Image1 > Image2 > Image3 > Image4 > Image5
>
> However, in the case I don't know the number of images, I can't use
> this structure. What could be done?

Composite = Image1

Then, for each additional image:

Composite = Composite > ImageN

Subject: Re: Composition structure
Posted by btt on Fri, 01 Oct 2004 19:07:02 GMT
View Forum Message <> Reply to Message

Jeferson E. wrote:
> Hello there!
>
> Assuming I have 5 images and I want to get the maximum value
> composition. I can use something like that:
>
> Composite = Image1 > Image2 > Image3 > Image4 > Image5
>
> However, in the case I don't know the number of images, I can't use
> this structure. What could be done?
>

For any old situation you can loop through the values, for example...

```
x = 5 & for i = 0, 10 do begin  & x >= i & print, i, x  & endfor
      0       5
      1       5
      2       5
      3       5
      4       5
      5       5
      6       6
```

```
 7      7
 8      8
 9      9
10      10
```

In your case you would do something like the following...

composite = image1
for i = 0, nImages-1 do composite = composite > imagei (or composite >= imagei)

Ben

Subject: Re: Composition structure
Posted by Paul Van Delst[1] on Fri, 01 Oct 2004 19:12:11 GMT
View Forum Message <> Reply to Message

Jeferson E. wrote:
> Hello there!
>
> Assuming I have 5 images and I want to get the maximum value
> composition. I can use something like that:
>
> Composite = Image1 > Image2 > Image3 > Image4 > Image5
>
> However, in the case I don't know the number of images, I can't use
> this structure. What could be done?

Depends how you call your application, no?

Let's say you have 10 images (say, 800x640) - you'd need to stick them in some sort of array to allow you code to process an arbitrary number of images,

```
  n = 20
  ImageArray = bytarr(800,640,n)
  .... here fill you image array....
  GetImageMax, ImageArray, ImageMax
```

where the procedure is something like below:

```
PRO GetImageMax, ImageArray, $  ; Input
                 ImageMax       ; Output

  ; -- Get some array info
  Info = SIZE( ImageArray, /STRUCTURE )

  ; -- Is the image array 3-d?
```

```
  IF ( Info.N_DIMENSIONS NE 3 ) THEN $
    MESSAGE, 'Must pass in a 3-D array!'

  ; -- Determine the number of images
  n_Images = Info.DIMENSIONS[2]

  ; -- Find the maximum
  ImageMax = ImageArray[*,*,0]
  FOR i = 1, n_Images-1 DO BEGIN
    ImageMax = ImageMax > ImageArray[*,*,i]
  ENDFOR

END
```

I'm sure other folks will post cleverer solutions that don't use loops.

paulv

Subject: Re: Composition structure
Posted by jcesq on Sun, 03 Oct 2004 21:57:05 GMT
View Forum Message <> Reply to Message

Thanks for all suggestions! I'm going to use Kuyper procedure (the simplest), once my image have always the same size:

Composite = Image1

Then, for each additional image:

Composite = Composite > ImageN

Best Regards for all...
Jeferson

Paul Van Delst <paul.vandelst@noaa.gov> wrote in message news:<cjkboj$ls7$1@news.nems.noaa.gov>...
> Jeferson E. wrote:
>> Hello there!
>>
>> Assuming I have 5 images and I want to get the maximum value
>> composition. I can use something like that:
>>
>> Composite = Image1 > Image2 > Image3 > Image4 > Image5
>>
>> However, in the case I don't know the number of images, I can't use
>> this structure. What could be done?

```
>
>  Depends how you call your application, no?
>
>  Let's say you have 10 images (say, 800x640) - you'd need to stick them in some sort of
>  array to allow you code to process an arbitrary number of images,
>
>    n = 20
>    ImageArray = bytarr(800,640,n)
>    .... here fill you image array....
>    GetImageMax, ImageArray, ImageMax
>
>  where the procedure is something like below:
>
>  PRO GetImageMax, ImageArray, $  ; Input
>                   ImageMax       ; Output
>
>    ; -- Get some array info
>    Info = SIZE( ImageArray, /STRUCTURE )
>
>    ; -- Is the image array 3-d?
>    IF ( Info.N_DIMENSIONS NE 3 ) THEN $
>      MESSAGE, 'Must pass in a 3-D array!'
>
>    ; -- Determine the number of images
>    n_Images = Info.DIMENSIONS[2]
>
>    ; -- Find the maximum
>    ImageMax = ImageArray[*,*,0]
>    FOR i = 1, n_Images-1 DO BEGIN
>      ImageMax = ImageMax > ImageArray[*,*,i]
>    ENDFOR
>
>  END
>
>  I'm sure other folks will post cleverer solutions that don't use loops.
>
>  paulv
```