

---

Subject: Re: skip the corrupted hdf file and continue the for loop for uncorrupted  
hdfs

Posted by [Brent Fallarcuna](#) on Wed, 29 Jan 2014 03:14:22 GMT

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

---

On Tuesday, January 28, 2014 1:47:22 PM UTC+8, David Fanning wrote:

> David Fanning writes:

>

>

>

>> I would write something along these lines:

>

>>

>

>> filelist = file\_search('\*.h29v07.005.\*.hdf', Count=count)

>

>> array = StrArr(count)

>

>> FOR j=0,count-1 DO BEGIN

>

>>

>

>>     Catch, theError

>

>>     IF theError NE 0 THEN BEGIN

>

>>         void = cgErrorMsg()

>

>>         Print, 'Bad File: ', filename[j]

>

>>         Message, /Reset

>

>>         Continue

>

>>     ENDIF

>

>>

>

>>     thisFile = filelist[j]

>

>>     hdf\_id = hdf\_sd\_start(thisFile)

>

>>     ...

>

>>

>

>> ENDFOR

>

```
>
>
> Whoops! I just violated the cardinal rule of error handling: Don't
>
> introduce errors into your error handling code! Kill your IDL session
>
> with your mouse when you get into the infinite loop I put you into, then
>
> make this change to your code. Sorry! :-(
>
>
>
>
> filelist = file_search('*.h29v07.005.*.hdf', Count=count)
>
> array = StrArr(count)
>
> FOR j=0,count-1 DO BEGIN
>
>
>
>   Catch, theError
>
>   IF theError NE 0 THEN BEGIN
>
>     void = cgErrorMsg()
>
>     Print, 'Bad File: ', thisFile
>
>     Message, /Reset
>
>     Continue
>
>   ENDIF
>
>
>
>   thisFile = filelist[j]
>
>   hdf_id = hdf_sd_start(thisFile)
>
>   ...
>
>
>
> ENDFOR
>
>
>
```

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

Thank you Sir Fanning, I've successfully ran my code and able to see corrupted hdf files for redownload.

On my code, my input x,y pixel location for my area of interest was based from your coyote page: [http://www.idlcoyote.com/map\\_tips/latlon2pixel.php](http://www.idlcoyote.com/map_tips/latlon2pixel.php).

I've read it again and it says, "Note that we are not checking that the latitude and longitude is actually inside our image. We should be, since we will get incorrect results if this is the case." How am I able to check if my input pixel location falls within my desired area (based field based lat long reading)?

Cheers,  
Brent

---