Subject: Re: IDL batch indexing
Posted by R.Bauer on Tue, 11 Mar 2008 17:39:24 GMT
View Forum Message <> Reply to Message

Beverly LaMarr schrieb:
> In article <b0bd3b46-ab67-4216-b3ff-dfbd88535fcb@s37g2000prg.googlegroups.com>,
> Raghu <raghuram.narasimhan@gmail.com> writes:
>> Hi all,
>>
>> I am writing a batch processing routine in order to do some querying.
>>
>> Steps:
>>
>> 1) Read in 30 images (could be 60 or more..hence using batch)
>>
>> Since i am using a file_search method to read in all the files, i want
>> each file to be given a unique name or ID because i need to compare
>> them with each other. So, instead of reading in 30 files separately
>> and assigning variable names to them, i want to do it automatically
>> using a counter in a loop or something like that.
>>
>> Eg: if i=0 and numfiles=30,
>> i want to name the first file thats read b1, second file b2, third b3,
>> 30th as b30.
>> Once these are named, i want to be able to compare b1 to b2 or b3 to
>> b5 etc.
>> So, its not just about naming these files since that can be done with
>> a strmid function.
>>
>> my idea was something like
>>
>> names=file_search(*.img,count=numfiles)
>> while i lt numfiles
>> b[i]=names(i)
>>
>> I have tried it and it says b is not defined. I am guessing i cannot
>> name variables dynamically.
>>
>> I don't know if i am very clear about my problem.
>> Basically, i want to be able to assign unique variable names to each
>> file thats read so that i can use those unique IDs to query and
>> compare.
>>
>> Any ideas ? i can give more information or try to be clearer about my
>> problem if necessary.
>>
>> Raghu
>

>
>
> Hi,
>
> I think I know what you want - and it's something that I have to do too.
>
> You could try using execute:
>
> in your loop, read each file in as "infile" then
> res=execute(b[i]+'=infile')
>
>
> In the end you'll have a variable for each value of b[i].

>
>
> The other way to go, if all of the files have the same format, would be to use
> a structure like:
> alldata=replicate({name:a,data:fltarr(100,100)},numfiles)
> where you just replace the "fltarr(100,100)" bit with whatever your data looks
> like - you could even have it check the first file to figure the format on it's
> own.
> Then in your loop you just read into alldata[i].data
>
>
> Bev
>

Ah

if that is the case you may want to use a structure.
You can do something like

```
for i=0,5 do begin
  tag = 'A'+strtrim(i,2)
if i eq  0 then struct=create_Struct(tag,i) else $
   struct=create_Struct(struct, tag,i)
endfor

help,struct,/str
end
```

```
IDL> example
** Structure <78bff8>, 6 tags, length=12, data length=12, refs=1:
  A0             INT             0
```

```
   A1          INT          1
   A2          INT          2
   A3          INT          3
   A4          INT          4
   A5          INT          5
```

You can access each tag with it's position index e.g.

```
IDL> print, struct.(0) - struct.(1)
      -1
```

You may want to avoid to use execute otherwise that will block you from using your code with the virtual machine of idl.

cheers
Reimar