

---

Subject: Re: adding a 4th dimension to 3D array during concatenation

Posted by [Guilherme Gualda](#) on Thu, 17 Mar 2016 05:52:14 GMT

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

---

Hi,

It may not be the most efficient, but it is easy to do what you want if you use reform. In your example, if each array x you create has dimensions [sx, sy, sz], then you can use:

```
array = [array, reform(x, 1, sx, sy, sz)]
```

Hope this helps!

Best,

Guil

On Tuesday, March 15, 2016 at 5:37:18 PM UTC-5, Wayana Dolan wrote:

> So outside a loop I start out with an empty array (ex. array=[ ] ). Then each time through the loop, I make an array with 3 dimensions (for example, array x has dimensions[91, 41, 33] ), and then concatenate it to the previous array. (ex. array=[array, x]).

>

> Lets say we run through the loop 16 times. What I'd like as a result is something that has dimensions like this [16, 91, 41, 33].

>

> I'm not sure how to do this... I've looked at IDL coyote's concatenation tutorial, and still am having trouble.

>

> I'm pretty new to coding period, so this is a challenge. Any ideas?

---