
Subject: Re: How do I morph an image.

Posted by [David Fanning](#) on Sat, 07 Aug 2004 17:04:37 GMT

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

Marc Reinig writes:

- > I need to compensate images for optical distortion (barrel, pincushion, ...)
- > in my system.
- >
- > I have a reference image (a grid of identifiable features) and I can tell
- > how the system is distorting it by comparing the reference image to the
- > captured, distorted image.
- >
- > Currently I have a table of points of the center of the original features
- > and a table of the center of these features in the distorted image. A
- > Matlab program is used to correct the images of interest based on these
- > tables.
- >
- > I need to integrate this process into my IDL code.
- >
- > Any pointers on how to morph an arbitrary image using these tables or
- > another technique would be appreciated.

There are at least two methods you can use to do this.

Suppose your input control points are in the vectors
xi and yi and your reference control points are in
the vectors xo and yo. You can do something like
this with the WARP_TRI command:

```
s = Size(refImage, /Dimensions)
warpedImg = Warp_Tri(xo, yo, xi, yi, inputImage, $
  OUTPUT_SIZE=s, /QUINTIC)
```

(There may be other keywords to set depending upon the
location and number of your control points, etc. I
assume here four of your control points are in the
corners of the image.)

Another method involves the POLYWARP and POLY_2D
commands. First, use POLYWARP to get the coefficients
required by POLY_2D for the warping:

```
POLYWARP, xi, yi, xo, yo, 1, p, q
```

Then use the p and q arrays to warp the image:

```
warpedImg = POLY_2D(inputImage, p, q, 1, s[0], s[1])
```

You can try both and see which is better for your application.

Cheers,

David

P.S. If you would like to e-mail the images to me, I would be happy to write an article about this and show the before and after results.

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: How do I morph an image.

Posted by [Marc Reinig](#) on Sat, 07 Aug 2004 18:13:46 GMT

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

David,

Thanks, for the response. I'll try your suggestions. Also, I'll send the images.

Marco

"David Fanning" <davidf@dfanning.com> wrote in message news:MPG.1b7eb98e4d7ba51a989815@news.frii.com...

> Marc Reinig writes:

>

>> I need to compensate images for optical distortion (barrel, pincushion, ...)

>> in my system.

>>

>> I have a reference image (a grid of identifiable features) and I can tell

>> how the system is distorting it by comparing the reference image to the captured, distorted image.

>>

>> Currently I have a table of points of the center of the original features

>> and a table of the center of these features in the distorted image. A

>> Matlab program is used to correct the images of interest based on these tables.

>>

>> I need to integrate this process into my IDL code.

```
>>
>> Any pointers on how to morph an arbitrary image using these tables or
>> another technique would be appreciated.
>
> There are at least two methods you can use to do this.
> Suppose your input control points are in the vectors
> xi and yi and your reference control points are in
> the vectors xo and yo. You can do something like
> this with the WARP_TRI command:
>
>   s = Size(refImage, /Dimensions)
>   warpedImg = Warp_Tri(xo, yo, xi, yi, inputImage, $
>     OUTPUT_SIZE=s, /QUINTIC)
>
> (There may be other keywords to set depending upon the
> location and number of your control points, etc. I
> assume here four of your control points are in the
> corners of the image.)
>
> Another method involves the POLYWARP and POLY_2D
> commands. First, use POLYWARP to get the coefficients
> required by POLY_2D for the warping:
>
>   POLYWARP, xi, yi, xo, yo, 1, p, q
>
> Then use the p and q arrays to warp the image:
>
>   warpedImg = POLY_2D(inputImage, p, q, 1, s[0], s[1])
>
> You can try both and see which is better for your
> application.
>
> Cheers,
>
> David
>
> P.S. If you would like to e-mail the images to me,
> I would be happy to write an article about this
> and show the before and after results.
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
```
