
Subject: Image registration using Powell optimization
Posted by [mwirth](#) on Fri, 18 May 2001 20:56:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Does anybody have an example of how Powell's optimization procedure can be used in the context of image registration to find the minimum when trying to calculate the optimum match using a similarity measure such as mutual information?

Help would be greatly appreciated,

many thanks,

Michael

mwirth@snowwhite.cis.uoguelph.ca

Subject: Re: Image registration using Powell optimization
Posted by [nrh](#) on Mon, 21 May 2001 02:41:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

These papers may be of use to you...

Frederik Maes, Dirk Vandermeulen and Paul Suetens

Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information Medical image analysis, vol. 3, no. 4, pp. 373-386, December 1999

Capek,K.: Optimisation Strategies Applied to Global Similarity Based Image Registration Methods [Conference Paper] 7th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media'99. in co-operation with EUROGRAPHICS and IFIP WG 5.10. WSCG'99. Conference Proceedings. Univ. West Bohemia. Part vol.2, 1999, pp.369-74 vol.2. Plzen, Czech Republic.

Holmes DR III, Camp JJ, Robb RA. Evaluation of search strategies and cost functions in optimizing voxel-based image registration. [Conference Paper] SPIE-Int. Soc. Opt. Eng. Proceedings of Spie - the International Society for Optical Engineering, vol.3031, 1997, pp.554-62. USA.

There are a few image registration packages that use Powell's Method. One that comes to mind is the Computerized Brain Atlas-see Journal of Computer Assisted Tomography 21(1):136-144 (1997).

None that I know of that use IDL.

mwirth@snowwhite.cis.uoguelph.ca wrote:

> Hi,
>
> Does anybody have an example of how Powell's optimization procedure can be used
> in the
> context of image registration to find the minimum when trying to calculate the
> optimum match
> using a similarity measure such as mutual information?
>
> Help would be greatly appreciated,
>
> many thanks,
>
> Michael
>
> mwirth@snowwhite.cis.uoguelph.ca

--

Defunctus adhuc loquitur

Rochelle Hatton
Department of Nuclear Medicine and Ultrasound
Westmead Hospital

Subject: Re: Image registration
Posted by [thompson](#) on Wed, 19 Jun 2002 14:45:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

carsten@rad.uni-kiel.de (Carsten Liess) writes:

> Dear all,

> I am fairly new to IDL, so bear with me please.
> I have two images of the same object, only with different intensities.
> I want to create an image that displays the ratio of the two input
> images. I programmed that and it works.
> Problem is that the object I am imaging might have moved by up to 10
> pixels (at a 256*256 matrix) in between, so I need to match the
> position of these two images before I calculate their ratio. The
> movement might include translation as well as rotation.
> Is there a routine in IDL that does such an operation? Alternatively,
> does any among you have any clever suggestion how to go about this?
> Many thanks!

Carsten:

The way I've handled this problem in the past is to make a cross correlation matrix of one image against the other, and looking for where the correlation coefficient is highest. Throwing in rotation as well makes it more complicated, but in essence the problem is the same. The basic difference is that one is working with three dimensions instead of two.

In the classic problem with translation only, one shifts one image relative to the other by one pixel at a time, and then calculates the cross correlation coefficient between them (e.g. with CORRELATE). For example, if you think that the two images could be shifted relative to each other by a maximum of 10 pixels, you would build up a matrix of size (21,21) representing shifts of -10 to +10 pixels in each direction. If you display this matrix as an image, you should be able to see a clear peak representing the best match between the two images. On my data, I've been able to fit the area around the peak to get down to subpixel resolution--your mileage may vary.

If you add rotation into the mix, I would guess that the best place to proceed would be to build up an array of (N,N,M) of cross correlation coefficients, where N is the number of pixel shifts (21 in the above example), and M is the number of angular rotations you wish to explore. For each angle position (K=0,M-1), I would rotate one of the images (e.g. with ROT) and then calculate the NxN cross correlation matrix for that rotation. You would then have a three dimensional volume to examine for the peak correlation.

I hope this helps,

Bill Thompson

Subject: Re: Image registration

Posted by [Eric Vella](#) on Wed, 19 Jun 2002 20:33:18 GMT

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

Just a few additions to the previous good answer from Bill ...

If the X and Y shifts are fairly uncorrelated, and the images have enough distinctive features (like sharp horizontal and vertical brightness transitions), you may be able to determine each shift independently. It might also be possible to proceed iteratively -- determine best rotation, then best X shift, then best Y shift, then repeat the whole sequence until it converges. If the convergence is stable, this will be quicker than mapping out the whole 3D space. If your images are small enough that time is not a problem, then of course the 3D approach is safer and better.

At each step, you might be able to work with a 1D image projection instead of the full 2D image. For example, you might be able to determine the X shift from a projection of the image onto the X axis. For rotation, you

could do something similar, make a 1D vector where each element is a sum of pixels at a fixed angle, summed over radius.

The C_CORRELATE function lets you evaluate a set of shifts with a single function call.

"William Thompson" <thompson@orpheus.nascom.nasa.gov> wrote in message news:aeq5eo\$rn0\$1@skates.gsfc.nasa.gov...

> carsten@rad.uni-kiel.de (Carsten Liess) writes:

>

>> Dear all,

>

>> I am fairly new to IDL, so bear with me please.

>> I have two images of the same object, only with different intensities.

>> I want to create an image that displays the ratio of the two input images. I programmed that and it works.

>> Problem is that the object I am imaging might have moved by up to 10

>> pixels (at a 256*256 matrix) in between, so I need to match the

>> position of these two images before I calculate their ratio. The

>> movement might include translation as well as rotation.

>> Is there a routine in IDL that does such an operation? Alternatively,

>> does any among you have any clever suggestion how to go about this?

>> Many thanks!

>

> Carsten:

>

> The way I've handled this problem in the past is to make a cross correlation

> matrix of one image against the other, and looking for where the correlation

> coefficient is highest. Throwing in rotation as well makes it more

> complicated, but in essence the problem is the same. The basic difference is

> that one is working with three dimensions instead of two.

>

> In the classic problem with translation only, one shifts one image relative to

> the other by one pixel at a time, and then calculates the cross correlation

> coefficient between them (e.g. with CORRELATE). For example, if you think that

> the two images could be shifted relative to each other by a maximum of 10

> pixels, you would build up a matrix of size (21,21) representing shifts of -10

> to +10 pixels in each direction. If you display this matrix as an image, you

> should be able to see a clear peak representing the best match between the two

> images. On my data, I've been able to fit the area around the peak to get
down
> to subpixel resolution--your mileage may vary.
>
> If you add rotation into the mix, I would guess that the best place to
proceed
> would be to build up an array of (N,N,M) of cross correlation
coefficients,
> where N is the number of pixel shifts (21 in the above example), and M is
the
> number of angular rotations you wish to explore. For each angle position
> (K=0,M-1), I would rotate one of the images (e.g. with ROT) and then
calculate
> the NxN cross correlation matrix for that rotation. You would then have a
> three dimensional volume to examine for the peak correlation.
>
> I hope this helps,
>
> Bill Thompson
