Fast 2-D convolution

by David Young
08 Jan 2009 (Updated 20 Mar 2014)

Speeds up many 2-D convolutions using the SVD; also finds a fast approximation in other cases.

Watch this File

5.0
5.0 | 5 ratings
Rate this file

49 Downloads (last 30 days)
File Size: 5.77 KB
File ID: #22619

Download Zip

Code covered by the BSD License

Highlights from Fast 2-D convolution

convolve2(x, m, shape, tol)
CONVOLVE2 Two dimensional convolution.

exindex(arr, varargin)
EXINDEX extended array indexing

View all files

File Information

Description CONVOLVE2 can be used wherever CONV2 is used, taking the same arguments and returning the same results to within a small tolerance. The computation is speeded up by using the singular value decomposition of the mask to express it as a sum of outer products. Each of these can be computed efficiently as convolution with a row and a column vector. CONV2 is used to carry out the individual convolutions.

Separable masks are a particular case and are handled by CONVOLVE2 much as FILTER2 does. Many other masks which are not separable have low rank (e.g. Gabor function masks) and are handled more efficiently by CONVOLVE2.

The function will also compute a reduced-rank approximation to a given mask if required and will use this if it will speed up the computation. An extra argument allows control over the error.

Additional shape options allow: (a) 2-D "circular" convolution - that is, the input array is taken to be periodic rather than surrounded by zeros; (b) a "reflection" boundary condition - that is, the input array is taken to be surrounded by reflected copies of itself.

See 'Computer and Robot Vision' Vol I, by R.M. Haralick and L.G. Shapiro (Addison-Wesley 1992), pp. 298-299.

Acknowledgements *This file* inspired Image Correspondences Using Cross Correlation.

Required Products MATLAB

MATLAB release MATLAB 8.2 (R2013b)

Tags for This File

Please login to tag files.

- 2d
- convolution
- filtering
- image processing
- svd

Save Cancel

Please login to add a comment or rating.

Comments and Ratings (7)

27 Mar 2013 Evgeny Pr

17 Oct 2010 Rossella Blatt

Thank you David! I use this a lot and it always improves dramatically the speed of my algorithms. Great job!

31 Mar 2010 David Young □ Comment only

Rob, you are right that the tolerance parameter makes no difference if the mask is ones(n). The reason is that the mask can be decomposed into 1D masks without any loss of accuracy, so this is done regardless of the tolerance. The same is true of many simple masks, e.g. those produced by the 'gauss' and 'log' arguments to fspecial, and Gabor function masks.

The tolerance becomes useful if the mask is complex - for example, if it is obtained by snipping out part of an image. Then, to get full accuracy, a full 2D convolution is needed and there will be no gain over conv2 (indeed, a slight loss of time); however, if the tolerance is large enough an approximation to the mask will be used, allowing a speed-up at a cost of accuracy.

30 Mar 2010 Rob Campbell □ Comment only

I get the following speed improvements. I didn't see an effect of changing the tolerance value. It would be nice to have usage examples where this parameter makes a difference to computation speed.

```
>> tic,for i=1:100;C=conv2(randn(1000),ones(10),'same');end,toc
>> Elapsed time is 17.121093 seconds.
>> tic,for i=1:100;C=convolve2(randn(1000),ones(10),'same',1);end,toc
>> Elapsed time is 10.800977 seconds.
>> tic,for i=1:100;C=convolve2(randn(1000),ones(10),'same',0);end,toc
```

		>> Elapsed time is 10.419711 seconds.
30 Mar 2010	Rob Campbell	
20 Nov 2009	James	I'm fairly sure this used to have loads of 5* comments? Anyway, this script is great. The increase in speed over conv2 for me was 10 times. Really easy to use - outstanding!
24 Feb 2009	Joey Hyde	I use this a lot. It is very fast and well documented. A great replacement for conv2(). In some cases, a fft based convolution is faster. I've seen this be the case when the filter and image are the same size. Does anybody know a c/c++ code which is as useful as this one? Multi-purpose, fast, reliable, 2d convolution.

Updates

30 Mar 2010	Now uses exindex to simplify the boundary condition computation.
11 Apr 2011	Minor correction to help comments; updated exindex to current version.
20 Mar 2014	Alternative names for shape options for consistency with other functions.

Contact us

Site Help | Patents | Trademarks | Privacy Policy | Preventing Piracy | Terms of Use

Featured MathWorks.com Topics: New Products | Support | Documentation | Training | Webinars | Newsletters | MATLAB Trials | Careers