Subject: 2D DWT Multiresolution decomposition
Posted by Amara.Graps on Sun, 22 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

I've been away, and I just saw this on the IDL group. Even if tardy,
perhaps my answer will still be useful.

From raudipaul@my-deja.com Thu, 05 Aug 1999 18:11:56 GMT:

> I was wondering if there is any code for doing a 2D DWT decompositions
> with octave filter banks out there. I notice that Amara's Wavelet
> Workbench has such a program for the 1-D case, but not the 2D case. Is
> there any such software out there?
>
> Thanks,
> Paul

Wavelet Workbench (WWB) _does_ have the capability to do this, but
some of the documentation may be lacking, so I will explain how to
do a 2D wavelet transform using 2D wavelets.

First of all, WWB already has a 2D forward and inverse discrete
wavelet transforms called: WFWT2PO and WIWT2PO. (They are a bit slow
on my old computer, and I've done nothing to optimize the routines,
but maybe that's not an issue for the faster computers). The
following I pulled from the header information of those subroutines.

```
 ========================================================== ======
; NAME:
;  WFWT2PO
;
; PURPOSE:
;  This function performs a 2-dimensional wavelet transform
;   (periodized, orthonormal).
;
; CATEGORY:
;  Wavelets
;
; CALLING SEQUENCE:
;    wc = WFWT2PO(x,L,qmf)
;
; INPUTS:
;  x: 2-d image (n by n array, n dyadic)
;  L: coarse level
;  qmf:  quadrature mirror filter
```

```
;
; OUTPUTS:
;  wc:   2-d wavelet transform


;+
; NAME:
;  WIWT2PO
;
; PURPOSE:
;  This function performs an inverse 2-dimensional wavelet transform
;   (periodized, orthonormal).
;
; CATEGORY:
;  Wavelets
;
; CALLING SEQUENCE:
;    x = WIWT2PO(wc,L,qmf)
;
; INPUTS:
;     wc:   2-d wavelet transform [n by n array, n dyadic]
;     L: coarse level
;     qmf:  quadrature mirror filter
;
; OUTPUTS:
;    x:  2-d signal reconstructed from wc.
```

========== ======

Now a little WWB IDL code showing how to use a forward
discrete (periodized, orthonormal) wavelet transform.

========== ======

```
 ;------------------------------------------------------------ ----
;Set up Wavelet Transform Parameters: wavelet, scale level, len
 ;------------------------------------------------------------ ----

;Generate a Quadrature Mirror Filter (QMF)
;for example, a 4-parameter Daubechies wavelet
WType = 'Daubechies'
Par = 4
QMF = WMKOFILT(WType,Par)

;Plot the QMF (wavelet) to see if it looks OK
;Select the Mother Wavelet at the lowest scale to look at
wave = WMKWVLET(1,3,Family=WType,Par=Par,Gender='Mother',n=512)
title = strlowcase(WType)+'_'+strtrim(string(Par),2)
```

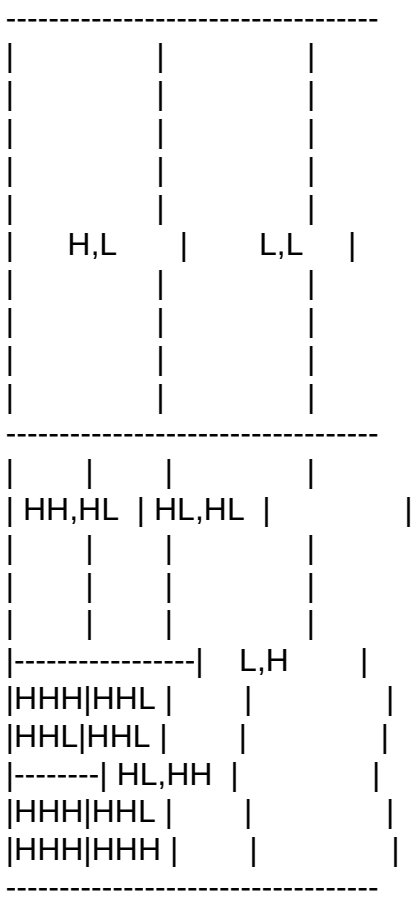
```
WPLOTIT,title, 2, wave

;Set LD, the lowest resolution (scale) level
LD = 2

;Set len, the image size (2^n, square)
len = 256

;---------------------------------------------------
;Read in a sample image from the folder of 2D data,
;for example the image of Ingrid Daubechies by Donoho
;---------------------------------------------------

;Set the working directory (for this example, change this for you)
wd = 'Mac HD:Wavelet Workbench:WWBv4:'

st = 'Daubech'
WREADIMG, 1, st, len, wd, sig

;-------------------------------
;Do Forward Wavelet Transform
;-------------------------------

wc = WFWT2PO(sig,LD,QMF)

;-----------
;Plot it
;-----------

wttitle = 'WT of ' + st
WPLOTIT, wttitle, 10, ABS(wc)
```

That's It.

The DWT operates in the following way. For each scale, it performs a high-pass downsampling (throw away every other data point) and a low-pass smoothing operation of the other half of the data. The result is a square with squares-within-squares of low-pass (L) operations and high-pass operations (H). The filter operations always work in pairs (Quadrature Mirror Filters), the filters being the wavelets you've chosen (say the Daubechies wavelet).

Next is an ASCII sketch of some output filter operations. The sketch shows three scale levels of smoothing, down-sampled operations. The square labelled all LL will be the most bland, smoothed result. The square with all HH will be the tiniest representation of your original image (just downsampled, with every other point thrown away

several times). The others will show you different things. Some combinations of LH will show the fine detail in your image. Other combinations will show the sharp contrast portions of your image (those are the combinations that make wavelets good edge detectors.



Your output coefficient matrix may have the squares flipped to have the smallest scale be at the top left corner, but this explanation should still apply.

Hope that this helps,

Amara

--

 ** *****
Amara Graps | Max-Planck-Institut fuer Kernphysik
Interplanetary Dust Group | Saupfercheckweg 1
+49-6221-516-543 | 69117 Heidelberg, GERMANY
* http://galileo.mpi-hd.mpg.de/~graps
 ** *****

"Never fight an inanimate object." - P. J. O'Rourke