CoE4TN3
Image Processing

Wavelet and Multiresolution Processing

McMaster University

Introduction

- Unlike Fourier transform, whose basis functions are sinusoids, wavelet transforms are based on small waves, called wavelets, of limited duration.
- Fourier transform provides only frequency information, but wavelet transform provides time-frequency information.
- Wavelets lead to a multiresolution analysis of signals.
- Multiresolution analysis: representation of a signal (e.g., an images) in more than one resolution/scale.
- Features that might go undetected at one resolution may be easy to spot in another.

Multiresolution

FIGURE 7.1 A natural image and its local histogram variations.

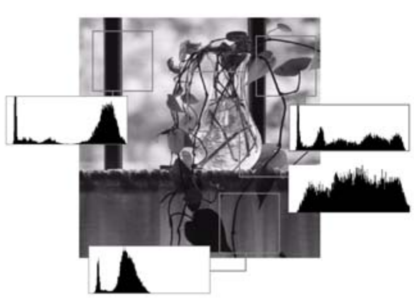
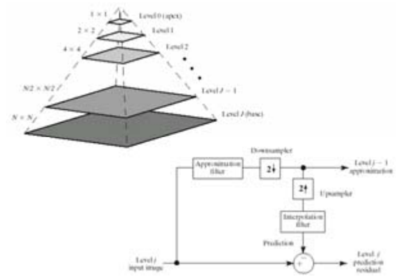


Image Pyramids



a
b
FIGURE 7.2 (a) A pyramidal image structure and (b) system block diagram for creating it.

Image pyramids

- At each level we have an approximation image and a residual image.
- The original image (which is at the base of pyramid) and its P approximation form the approximation pyramid.
- The residual outputs form the residual pyramid.
- Approximation and residual pyramids are computed in an iterative fashion.
- A P+1 level pyramid is build by executing the operations in the block diagram P times.

Image pyramids

- During the first iteration, the original 2^Jx2^J image is applied as the input image.
- This produces the level J-1 approximate and level J prediction residual results
- For iterations j=J-1, J-2, …, J-p+1, the previous iteration's level j-1 approximation output is used as the input.

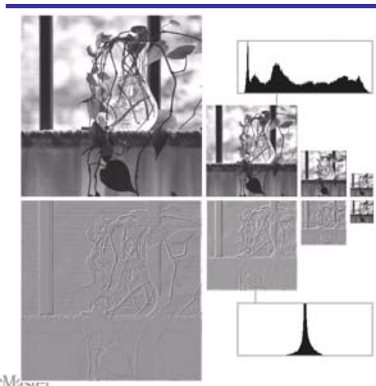
Image pyramids

- Each iteration is composed of three sequential steps:

1. Compute a reduced resolution approximation of the input image. This is done by filtering the input and downsampling (subsampling) the filtered result by a factor of 2.
   - Filter: neighborhood averaging, Gaussian filtering
   - The quality of the generated approximation is a function of the filter selected

Image pyramids

2. Upsample output of the previous step by a factor of 2 and filter the result. This creates a prediction image with the same resolution as the input.
   - By interpolating intensities between the pixels of step 1, the interpolation filter determines how accurately the prediction approximates the input to step 1.
3. Compute the difference between the prediction of step 2 and the input to step 1. This difference can be later used to reconstruct progressively the original image

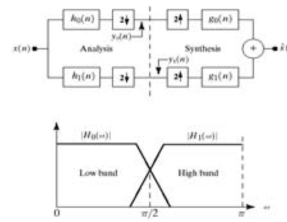


a
b
FIGURE 7.3 Two image pyramids and their statistics: (a) a Gaussian (approximation) pyramid and (b) a Laplacian (prediction residual) pyramid.

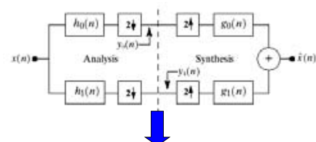
Subband coding

- In subband coding, an image is decomposed into a set of bandlimited components, called subbands.
- Since the bandwidth of the resulting subbands is smaller than that of the original image, the subbands can be downsampled without loss of information.

a
b
FIGURE 7.4 (a) A two-band filter bank for one-dimensional subband coding and decoding, and (b) its spectrum splitting properties.



Perfect Reconstruction Filter



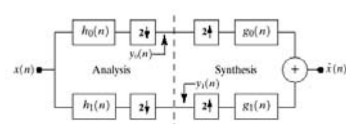
Z transform:

$$\hat{X}(z) = \frac{1}{2}\left[H_0(z)G_0(z) + H_1(z)G_1(z)\right]X(z)$$
$$+\frac{1}{2}\left[H_0(-z)G_0(z) + H_1(-z)G_1(z)\right]X(-z)$$

Goal: find H_0, H_1, G_0 and G_1 so that

$$x(n) = \hat{x}(n) \quad \left(i.e.\ X(z) = \hat{X}(z)\right)$$

Perfect Reconstruction Filter: Conditions



If

$$\begin{cases} H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0 \\ H_0(z)G_0(z) + H_1(z)G_1(z) = 2 \end{cases}$$

Then

$$X(z) = \hat{X}(z)$$

Perfect Reconstruction Filter Families

Filter	QMF	CQF	Orthonormal
$H_0(z)$	$H_0^2(z) - H_0^2(-z) = 2$	$H_0(z)H_0(z^{-1}) + H_0^2(-z)H_0(-z^{-1}) = 2$	$G_0(z^{-1})$
$H_1(z)$	$H_0(-z)$	$z^{-1}H_0(-z^{-1})$	$G_1(z^{-1})$
$G_0(z)$	$H_0(z)$	$H_0(z^{-1})$	$G_0(z)G_0(z^{-1}) + G_0(-z)G_0(-z^{-1}) = 2$
$G_1(z)$	$-H_0(-z)$	$zH_0(-z)$	$-z^{-2K+1}G_0(-z^{-1})$

TABLE 7.1 Perfect reconstruction filter families.

QMF: quadrature mirror filters

CQF: conjugate mirror filters

2-D

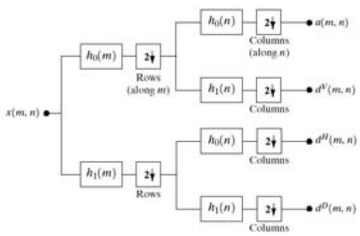
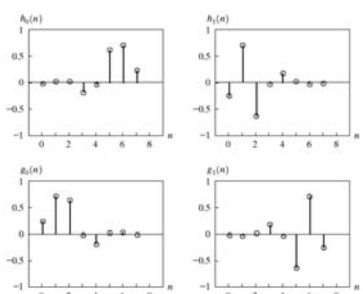


FIGURE 7.5 A two-dimensional, four-band filter bank for subband image coding.

Example of Filters

FIGURE 7.6 The impulse responses of four 8-tap Daubechies orthonormal filters.



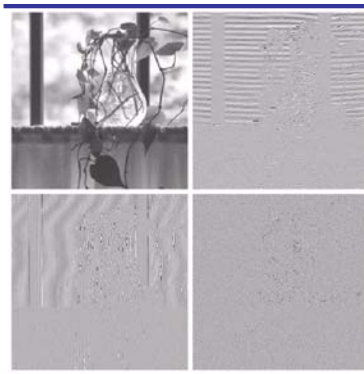
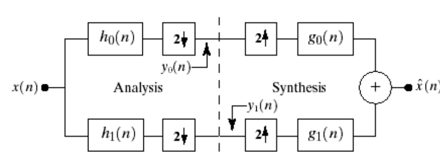


FIGURE 7.7 A four-band split of the vase in Fig. 7.1 using the subband coding system of Fig. 7.5.

The Haar Transform

- Haar proposed the Haar Transform in 1910, more than 70 years before the wavelet theory was born.
- Actually, Haar Transform employs the Haar wavelet filters but is expressed in a matrix form.
- Haar wavelet is the oldest and simplest wavelet basis.
- Haar wavelet is the only one wavelet basis, which holds the properties of orthogonal, (anti-)symmetric and compactly supported.

The Haar Wavelet Filters



$$h_0 = \left\{\sqrt{2}/2, \sqrt{2}/2\right\} \quad h_1 = \left\{-\sqrt{2}/2, \sqrt{2}/2\right\}$$

$$g_0 = \left\{\sqrt{2}/2, \sqrt{2}/2\right\} \quad g_0 = \left\{\sqrt{2}/2, -\sqrt{2}/2\right\}$$

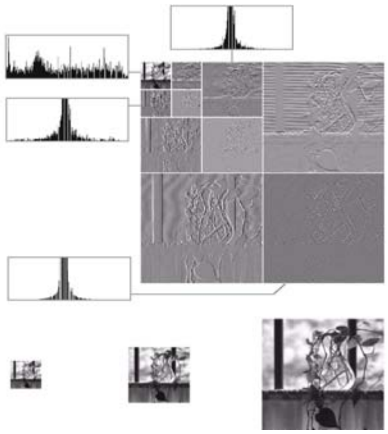


FIGURE 7.8 (a) A discrete wavelet transform using Haar basis functions. Its local histogram variations are also shown; (b)–(d) Several different approximations (64 × 64, 128 × 128, and 256 × 256) that can be obtained from (a).

Multiresolution Expansions

- Series Expansions
  - ➢A function can be expressed as

$$f(x)=\sum_k \alpha_k \varphi_k(x)$$

  - ➢where

$$\alpha_k=\langle \tilde{\varphi}_k(x), f(x)\rangle=\int \tilde{\varphi}_k^*(x) f(x)dx$$

$\tilde{\varphi}_k(x)$ → **Dual function of** $\varphi_k(x)$

$*$ → **Complex conjugate operation**

McMaster University

Multiresolution Expansions

- Series Expansions
  - ➢Orthonormal basis

$$\varphi_k(x)=\tilde{\varphi}_k(x)$$

$$\langle \varphi_j(x), \varphi_k(x)\rangle=\begin{cases}0 & j\neq k\\ 1 & j=k\end{cases}$$

  - ➢biorthogonal

$$\langle \varphi_j(x), \varphi_k(x)\rangle=0 \quad j\neq k$$

$$\langle \varphi_j(x), \tilde{\varphi}_k(x)\rangle=\begin{cases}0 & j\neq k\\ 1 & j=k\end{cases}$$

McMaster University

Multiresolution Expansions

- Scaling functions
  - ➢Integer translations and dyadic scalings of a scaling function $\varphi(x)$

$$\varphi_{j,k}(x)=2^{j/2}\varphi(2^j x-k)$$

  - ➢Express $f(x)$ as the combination of $\varphi_{j_0,k}(x)$

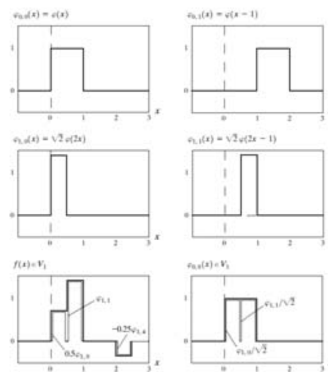
$$f(x)=\sum_k \alpha_k \varphi_{j_0,k}(x)$$

McMaster University

$$\varphi(x)=\begin{cases}0 & 0\leq x<1\\ 1 & otherwise\end{cases}$$

$$\varphi_{0,k}(x)=\frac{1}{\sqrt{2}}\varphi_{1,2k}(x)+\frac{1}{\sqrt{2}}\varphi_{1,2k+1}(x)$$

$$f(x)=0.5\varphi_{1,0}(x)+\varphi_{1,1}(x)-0.25\varphi_{1,4}(x)$$



McMaster University

Multiresolution Expansions

- Scaling functions
  - ➢Dilation equation for scaling function $\varphi(x)$

$$\varphi(x)=\sum_n h_\varphi(n)\sqrt{2}\varphi(2x-n)$$

  - ➢$h_\varphi(n)$ are called scaling function coefficients
  - ➢Example: Haar wavelet, $h_\varphi(0)=h_\varphi(1)=1/\sqrt{2}$

$$\varphi(x)=\frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x)\right]+\frac{1}{\sqrt{2}}\left[\sqrt{2}\varphi(2x-1)\right]$$

McMaster University

Multiresolution Expansions

- Wavelet functions

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\varphi(2x-n)$$

➢ $h_\psi(n)$ are called wavelet function coefficients

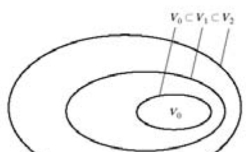
➢ Translation and scaling of $\psi(x)$

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$$

➢ condition for orthogonal wavelets

$$h_\psi(n) = (-1)^n h_\varphi(1-n)$$

FIGURE 7.10 The nested function spaces spanned by a scaling function.



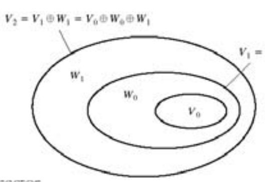
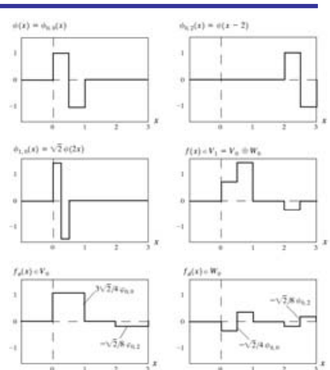


FIGURE 7.11 The relationship between scaling and wavelet function spaces.

Haar Wavelet

$$\varphi(x) = \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1 \\ 0 & elsewhere \end{cases}$$



Wavelet Transform: 1-D

- Wavelet series expansion

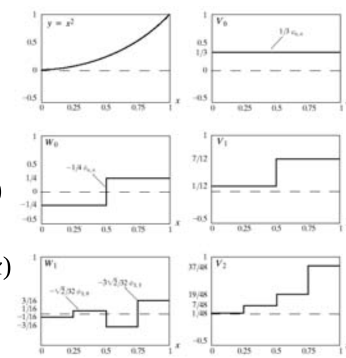
$$f(x) = \sum_k c_{j_0}\varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty}\sum_k d_j(k)\psi_{j,k}(x)$$

➢ where

$$c_{j_0}(k) = \langle f(x), \varphi_{j_0,k}(x)\rangle = \int f(x)\varphi_{j_0,k}(x)dx$$

$$d_j(k) = \langle f(x), \psi_{j,k}(x)\rangle = \int f(x)\psi_{j,k}(x)dx$$

$$y(x) = \frac{1}{3}\varphi_{0,0}(x)$$
$$-\frac{1}{4}\psi_{0,0}(x)$$
$$-\frac{\sqrt{2}}{32}\psi_{1,0}(x)$$
$$-\frac{3\sqrt{2}}{32}\psi_{1,1}(x)$$
$$+\ldots\ldots$$



Wavelet Transform: 1-D

- Discrete Wavelet Transform

$$f(x) = \frac{1}{\sqrt{M}}\sum_k W_\varphi(j_0,k)\varphi_{j_0,k}(x)$$
$$+\frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_k W_\psi(j,k)\psi_{j,k}(x)$$

➢ where

Approximation coefficients $$W_\varphi(j_0,k) = \frac{1}{\sqrt{M}}\sum_x f(x)\varphi_{j_0,k}(x)$$

Detail coefficients $$W_\psi(j,k) = \frac{1}{\sqrt{M}}\sum_x f(x)\psi_{j,k}(x)$$

Fast Wavelet Transform: Decomposition

$$W_\psi(j,k) = h_\psi(-n) * W_\varphi(j+1,n)\big|_{n=2k,k\ge 0}$$

$$W_\varphi(j,k) = h_\varphi(-n) * W_\varphi(j+1,n)\big|_{n=2k,k\ge 0}$$

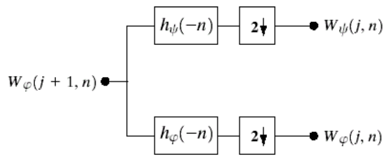


FIGURE 7.15 An FWT analysis bank.

Fast Wavelet Transform: Decomposition

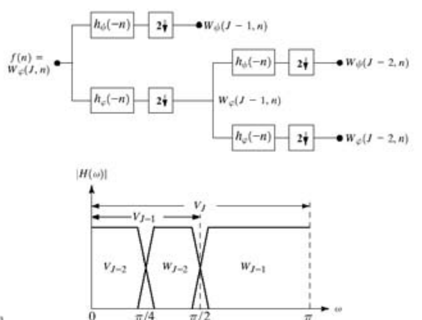


FIGURE 7.16 (a) A two-stage or two-scale FWT analysis bank and (b) its frequency splitting characteristics.

Example: Haar Wavelet

$$h_\varphi(n) = \begin{cases} 1/\sqrt{2} & n = 0,1 \\ 0 & otherwise \end{cases} \qquad h_\psi(n) = \begin{cases} 1/\sqrt{2} & n = 0 \\ -1/\sqrt{2} & n = 1 \\ 0 & otherwise \end{cases}$$

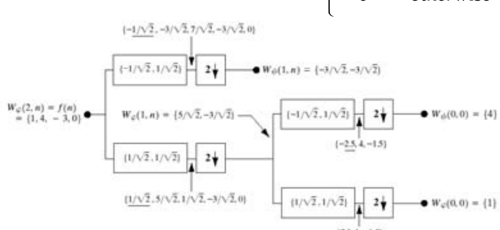


FIGURE 7.17 Computing a two-scale fast wavelet transform of sequence {1, 4, −3, 0} using Haar scaling and wavelet vectors.

Fast Wavelet Transform: Reconstruction

$$W_\varphi(j+1,k) = h_\varphi(k) * W_\varphi^{up}(j,k) + h_\psi(k) * W_\psi^{up}(j,k)\big|_{k\ge 0}$$

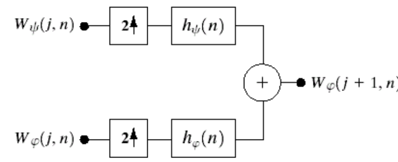


FIGURE 7.18 The FWT^{-1} synthesis filter bank.

Fast Wavelet Transform: Reconstruction

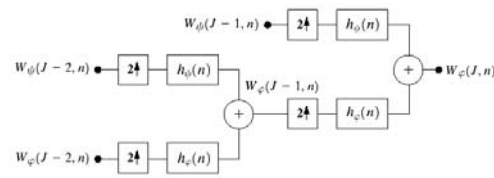


FIGURE 7.19 A two-stage or two-scale FWT^{-1} synthesis bank.

Fast Wavelet Transform: Reconstruction

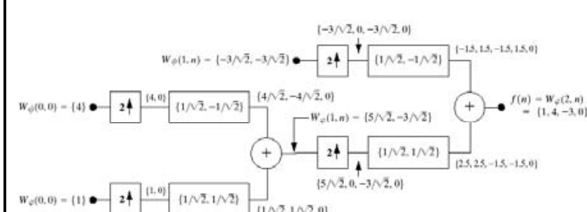
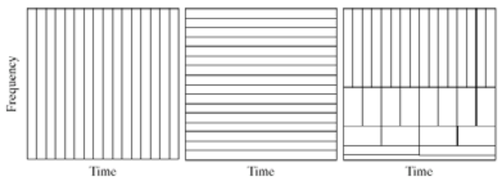


FIGURE 7.20 Computing a two-scale inverse fast wavelet transform of sequence $\{1, 4, -1.5\sqrt{2}, -1.5\sqrt{2}\}$ with Haar scaling and wavelet vectors.

Wavelet Transform vs. Fourier Transform



a b c

FIGURE 7.21 Time-frequency tilings for (a) sampled data, (b) FFT, and (c) FWT basis functions.

Wavelet Transform: 2-D

Scaling function:

$$\varphi(x,y) = \varphi(x)\varphi(y)$$

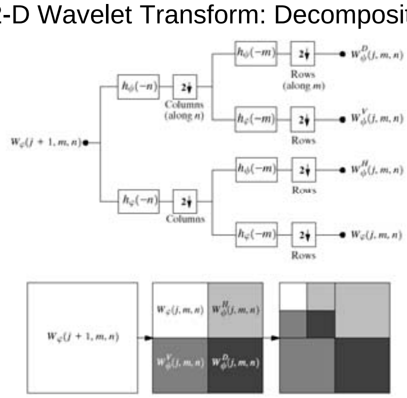
Wavelet functions:

$$\psi^H(x,y) = \psi(x)\varphi(y)$$ **Horizontal direction**

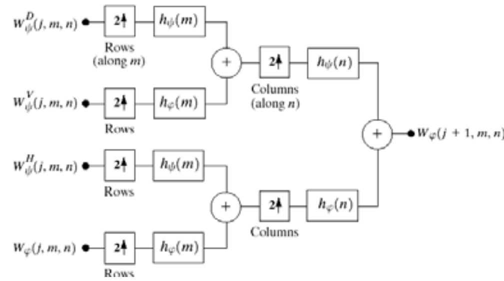
$$\psi^V(x,y) = \varphi(x)\psi(y)$$ **Vertical direction**

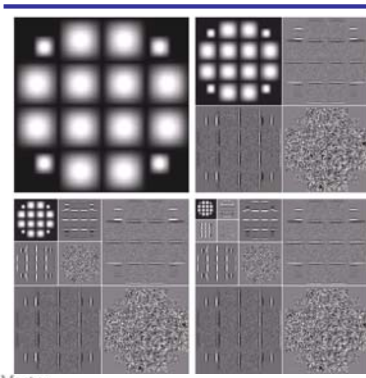
$$\psi^D(x,y) = \psi(x)\psi(y)$$ **Diagonal direction**

2-D Wavelet Transform: Decomposition



2-D Wavelet Transform: Reconstruction



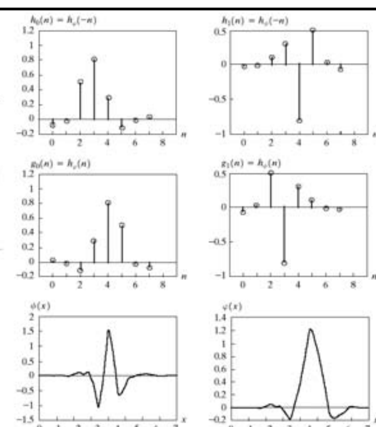


a b
c d

FIGURE 7.23 A three-scale FWT.

a b
c d
e f
g

FIGURE 7.24 Fourth-order symlets: (a)–(b) decomposition filters; (c)–(d) reconstruction filters; (e) the one-dimensional wavelet; (f) the one-dimensional scaling function; and (g) one of three two-dimensional wavelets, $\psi^H(x,y)$.



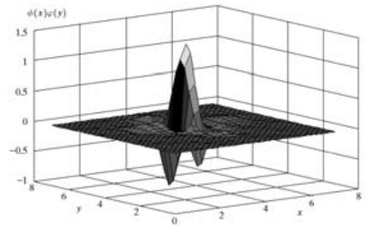


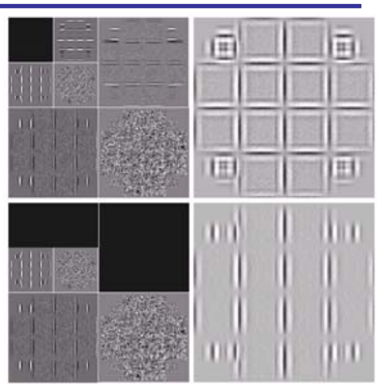
Fig. 7.24 (g) and (g) one of three two-dimensional wavelets, $\psi^{H}(x, y)$.

Image Processing by Wavelet Transform

- Three Steps:
  - ➢Decompose the image into wavelet domain
  - ➢Alter the wavelet coefficients, according to your applications such as denoising, compression, edge enhancement, etc.
  - ➢Reconstruct the image with the altered wavelet coefficients.

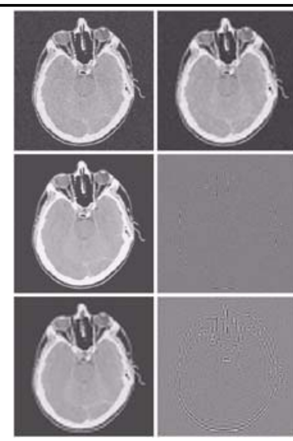
a b
c d

FIGURE 7.25 Modifying a DWT for edge detection: (a) and (c) two-scale decompositions with selected coefficients deleted; (b) and (d) the corresponding reconstructions.



Wavelet Transform based Denoising

- Three Steps:
  - ➢Decompose the image into several scales.
  - ➢For each wavelet coefficient y:
    - ❖Hard thresholding: $y = \begin{cases} y & |y| \ge t \\ 0 & |y| < t \end{cases}$
    - ❖Soft thresholding: $y = \begin{cases} sign(y) \bullet (|y| - t) & |y| \ge t \\ 0 & |y| < t \end{cases}$
  - ➢Reconstruct the image with the altered wavelet coefficients.



a b
c d
e f

FIGURE 7.26 Modifying a DWT for noise removal: (a) a noisy MRI of a human head; (b), (c) and (e) various reconstructions after thresholding the detail coefficients; (d) and (f) the information removed during the reconstruction of (c) and (e). (Original image courtesy Vanderbilt University Medical Center.)

Assignment

- Get familiar with the Matlab Wavelet Toolbox.
- By using the Wavelet Toolbox functions, write a program to realize the soft-thresholding denoising on a noisy MRI image.

End of the lecture

McMaster University