HW2 LMS Filter and DWT Filter design

VLSI DSP HW2

HSUAN-YU LIN, NCHU Lab716

INDEX

I. Adaptive FIR Low pass filter

Problem

Q1. LMS filter design

For a least mean square (LMS) adaptive filter, assume the filter is of the form finite impulse response (FIR) and 15-tap long (i.e., with 15 coefficients b_0~b_{14} for x(n)~x(n-14)). Given an input signal consisting of 2 frequency components

$s(n) = \sin(2\pi*n/12)+\cos(2\pi*n/4)$

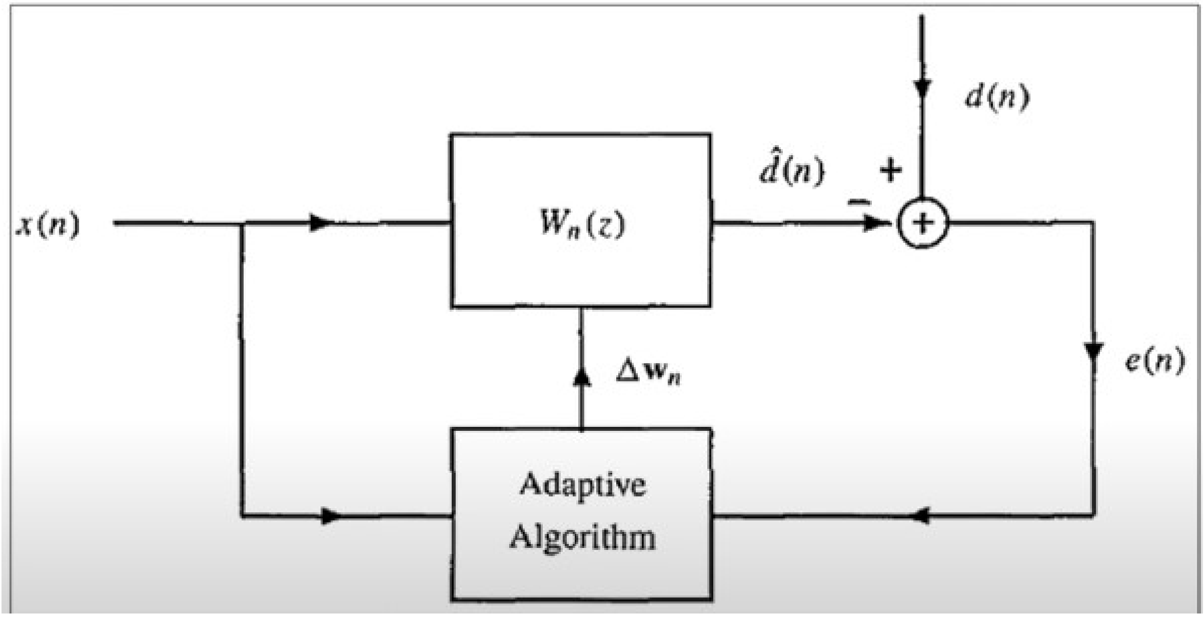
develop an adaptive low pass filter design

Set the target as a low pass filter to remove the high frequency component $\cos(2\pi*n/4)$ and use $\sin(2\pi*n/12)$ as the desired (or training) signal for LMS adaptation. Assume the step size μ is 10^{-2}.

- write a Matlab code to simulate the LMS based adaptive filtering. Calculate the RMS (root mean square) value of the latest 16 prediction errors (i.e., $r = \mathrm{sqrt}((e^2(n) + e^2(n-1) + \ldots + e^2(n-15))/16)$ and the adaptation is considered being converged if this value is less than 10% of RMS (root mean square) value of the desired signal, which equals 0.1/sqrt(2).
- Show the plot of "r" versus "n" and indicate when the filter converges, i.e. how many training samples are required
- Show the plot of filter coefficients $b_i(n)$, for i = 0~14, versus "n" and see if the values of filter coefficients remain mostly unchanged after convergence
- Apply a 64-point FFT to the impulse response of the converged filter and verify the filter is indeed a low pass one. Note that the input vector to the 64-point FFT is (b_0, b_1, ..., b_{14}, 0,0,.....,0) with 49 trailing zeros.
- Change the step size μ to 10^{-4} and see how the behavior of the adaptive filter changes.
- Conduct simulation with a sufficiently large number of samples to see how small the value of "r" can be (the convergence bias)

Derivation steps

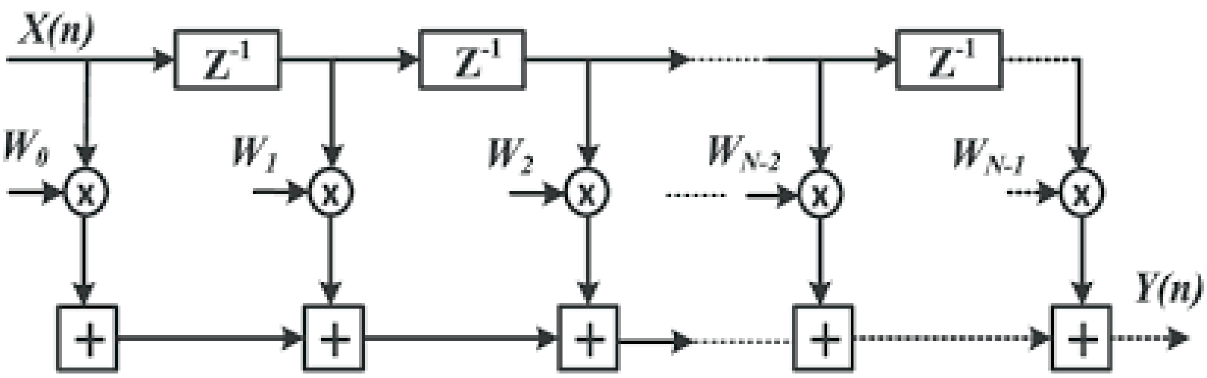
Adaptive Filter specification



1. x(n) is the input signal, wn(z) is the adaptive filter block with coefficients of wn.
2. d_hat(n) is the generated system response and d(n) is the desired signal.
3. e(n) is the error between d_hat(n) and d(n)
4. The adaptive algorithm block determines which kind of policy we should use to find the suitable filter coefficients. In this HW, LMS algorithm is chosen.

The adaptive FIR filter

$$\hat{d}(n) = \sum_{k=0}^{p} \omega_n(k)x(n-k) = \mathbf{w}_n^T \boldsymbol{X}(n)$$



- The desired output is genereated through the p-tap FIR filter design, where wn is the coefficients that gets updated on the fly.

Error function

$$
\begin{aligned}
e(n) &= d(n) - \hat{d}(n) \\
&= d(n) - w_n^T \boldsymbol{X}(n)
\end{aligned}
$$

$$E\{e(n)x^*(n-k)\} = 0 \ ; \ k = 0,1,\dots,p$$

- Error function simply is the difference between the desired signal and the generated system response.
- Ultimate goal is to minimize the autocorrelation between error vector and input signal.

LMS algorithm

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu e(n)\boldsymbol{X}^*(n)$$

$$\omega_{n+1} = \omega_n(k) + \mu e(n)\boldsymbol{X}^*(n-k)$$

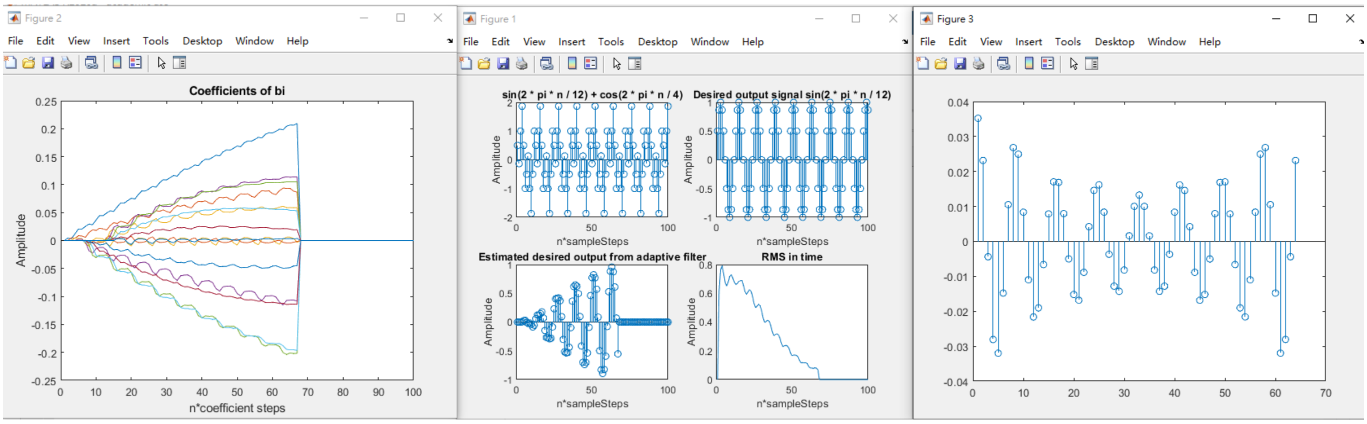
- mu is the step sizes for the algorithm, which governs the variability of the coefficients in each iteration.
- e(n)X*(n) is the factor of auto-correlation between the input signal and the error function.

RMS(Root mean square)

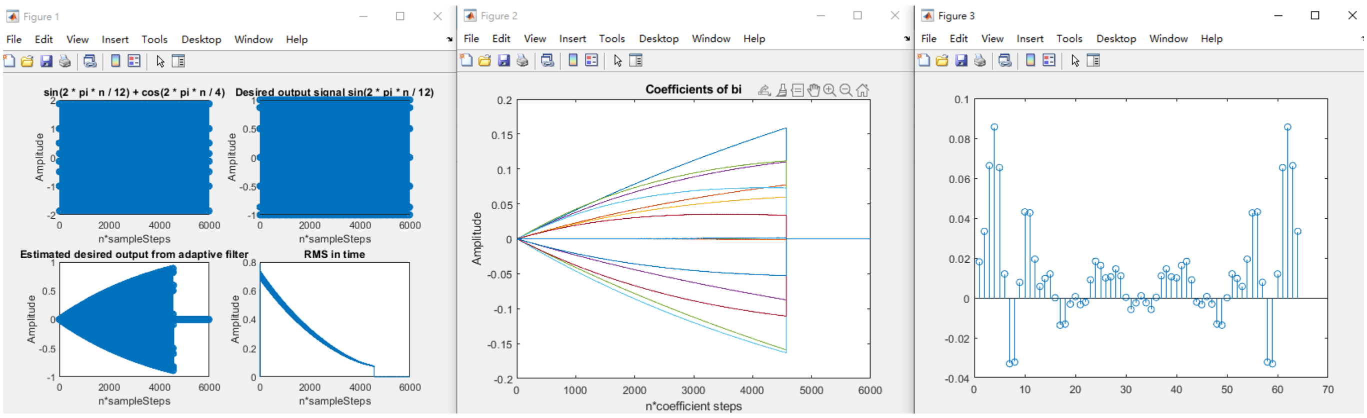
$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

- Root mean square used to find the norm of the error vector, we hope that this value be as small as possible s.t. the system is converged.

Q1 Result




```
Sample_size = 100
mu = 0.01
Total Steps needed to reach 10% of RMS 67
RMS value 0.0673
```




```
Sample_size = 6000
mu = 0.0001
Total Steps needed to reach 10% of RMS 4570
RMS value 0.0707
```

II. 2D-DWT

Problem

Q2. Discrete Wavelet Transform

For a discrete wavelet transform (DWT) adopting (9/7) filters, i.e. the low pass filter $h(i)$ is 9-taped and the high pass filter $g(i)$ is 7-taped. Both filters are liner phased and have symmetric coefficients. The filter coefficients are given in Table 1. For a corresponding inverse discrete wavelet transform, the low pass filter $q(i)$ is 7-taped and the high pass filter $p(i)$ is 9-taped. The filter coefficients are given in Table 2.

Analysis Filter Coefficients		
i	Lowpass Filter h_i	Highpass Filter g_i
0	0.852698679009	-0.788485616406
±1	0.377402855613	0.418092273222
±2	-0.110624404418	0.040689417609
±3	-0.023849465020	-0.064538882629
±4	0.037828455507	

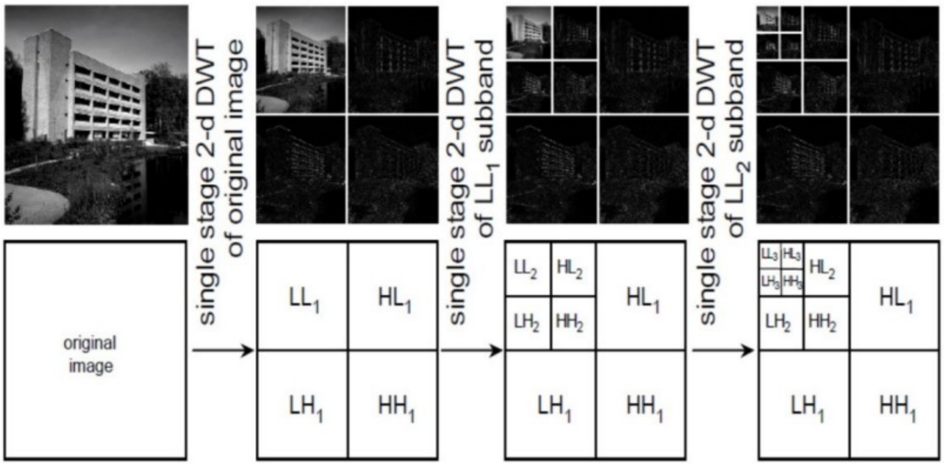
Note: the high pass and low pass filter notations here are opposite to those in the lecture note

Table 2. Synthesis filter coefficients for the floating point 9/7 filter

Synthesis Filter Coefficients		
i	Low pass Filter q_i	High pass Filter p_i
0	0.788485616406	-0.852698679009
±1	0.418092273222	0.377402855613
±2	-0.040689417609	0.110624404418
±3	-0.064538882629	-0.023849465020
±4		-0.037828455507

a) For a 512×512 gray scale image (will be provided along with the homework assignment), please conduct a 2-D 3-level DWT transform (as shown in Figure 1) and show the transformed result. Then conduct a 2-D 3-level IDWT to convert it back. Please compare if the reconstructed image (after IDWT) is same as the original image by calculating its PSNR value.

b) By setting all three level 1 sub-bands HL1, LH1 and HH1 coefficients to zeros and perform IDWT. See how the reconstructed image is different from the original one and calculate its PSNR value.



symmetric_extension

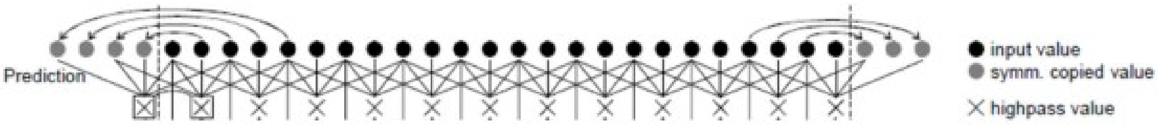
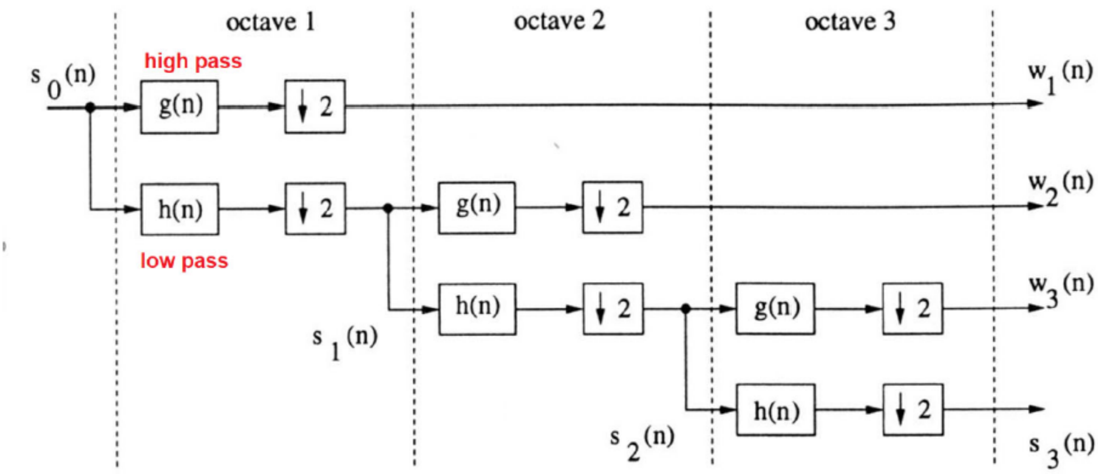


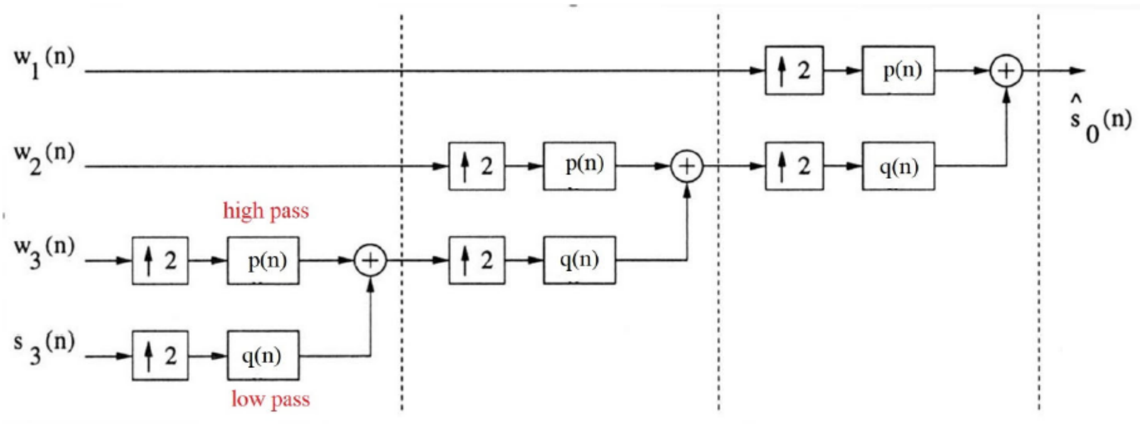
Figure 3. Symmetric extension scheme for boundary pixels

1D 3-level_DWT



(a) One-dimensional 3-level DWT transform

1D 3-level_IDWT

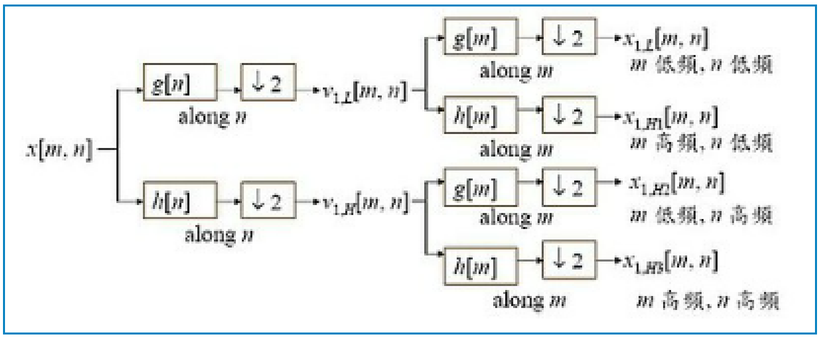


(b) One-dimensional 3-level IDWT transform

Figure 2. 1-D 3-level DWT versus IDWT

2D-DWT

2-D Discrete Wavelet Transform



此时的输入信号变成 $x[m,n]$，而转换过程变得更复杂，说明如下：

首先对n方向作高通、低通以及降频的处理

$$v_{1,L}[m,n]=\sum_{k=0}^{K-1}x[m,2n-k]g[k]$$

$$v_{1,H}[m,n]=\sum_{k=0}^{K-1}x[m,2n-k]h[k]$$

接着对 $v_{1,L}[m,n]$与 $v_{1,H}[m,n]$延著m方向作高低通及降频动作

$$x_{1,LL}[m,n]=\sum_{k=0}^{K-1}v_{1,L}[2m-k,n]g[k]$$

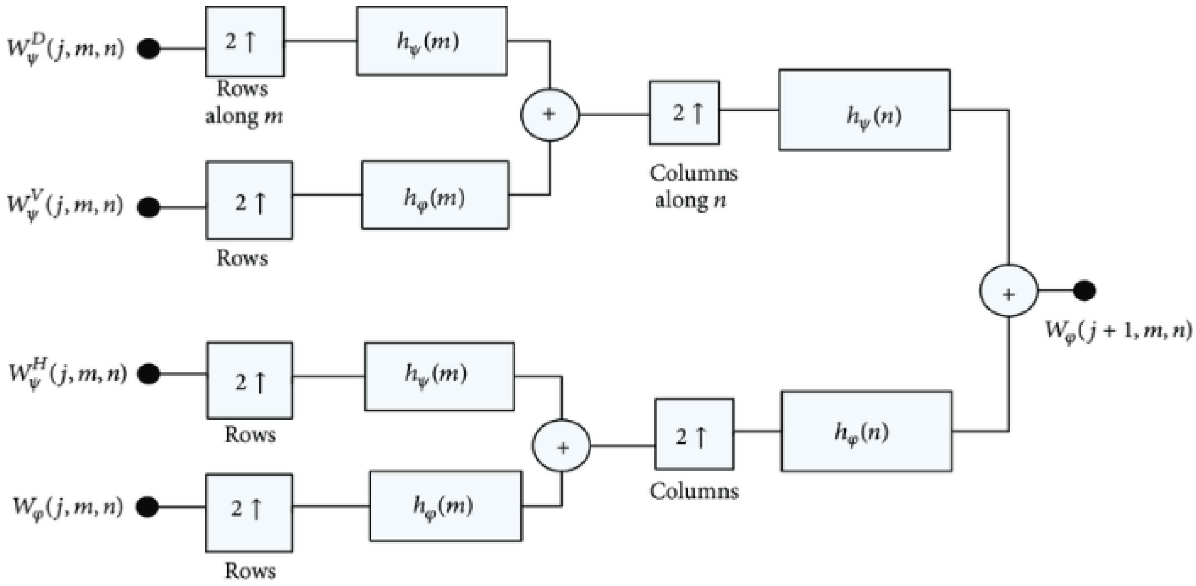
$$x_{1,HL}[m,n]=\sum_{k=0}^{K-1}v_{1,L}[2m-k,n]h[k]$$

$$x_{1,LH}[m,n]=\sum_{k=0}^{K-1}v_{1,H}[2m-k,n]g[k]$$

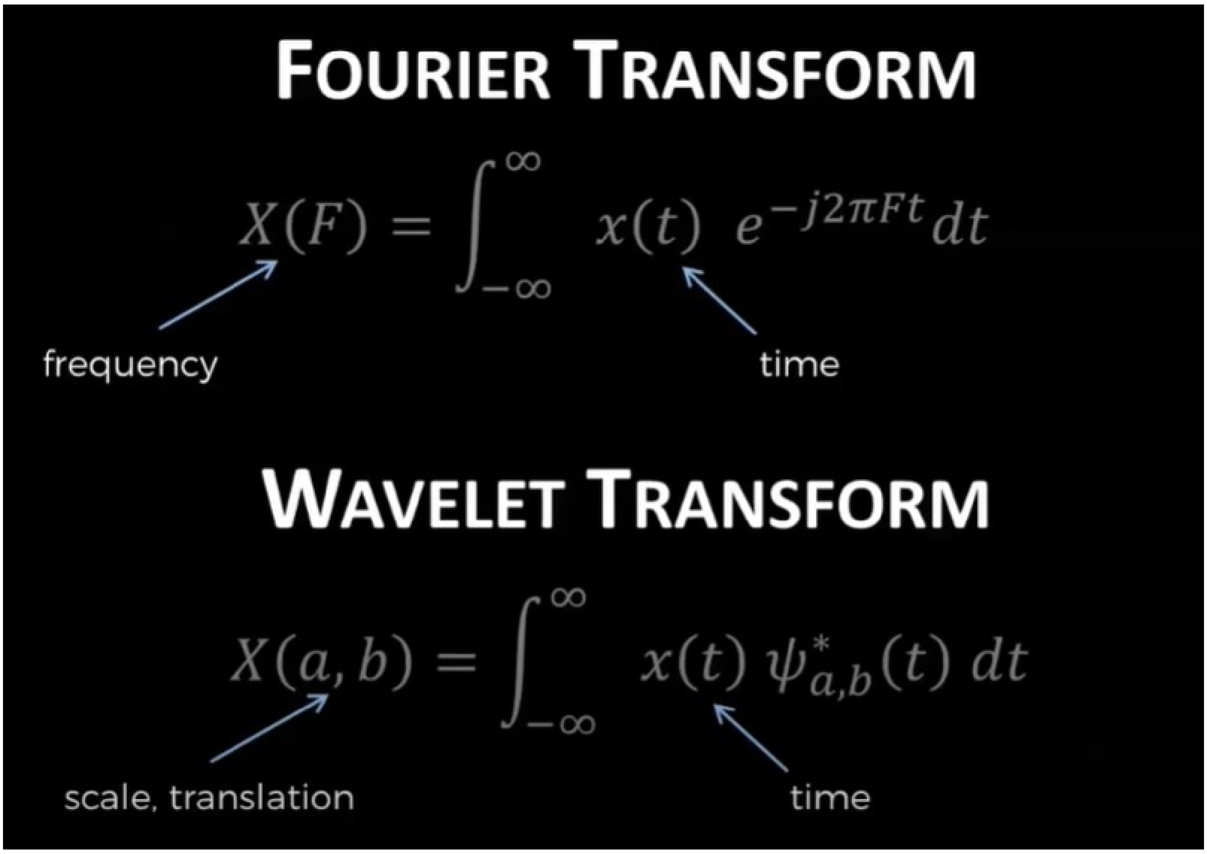
$$x_{1,HH}[m,n]=\sum_{k=0}^{K-1}v_{1,H}[2m-k,n]h[k]$$

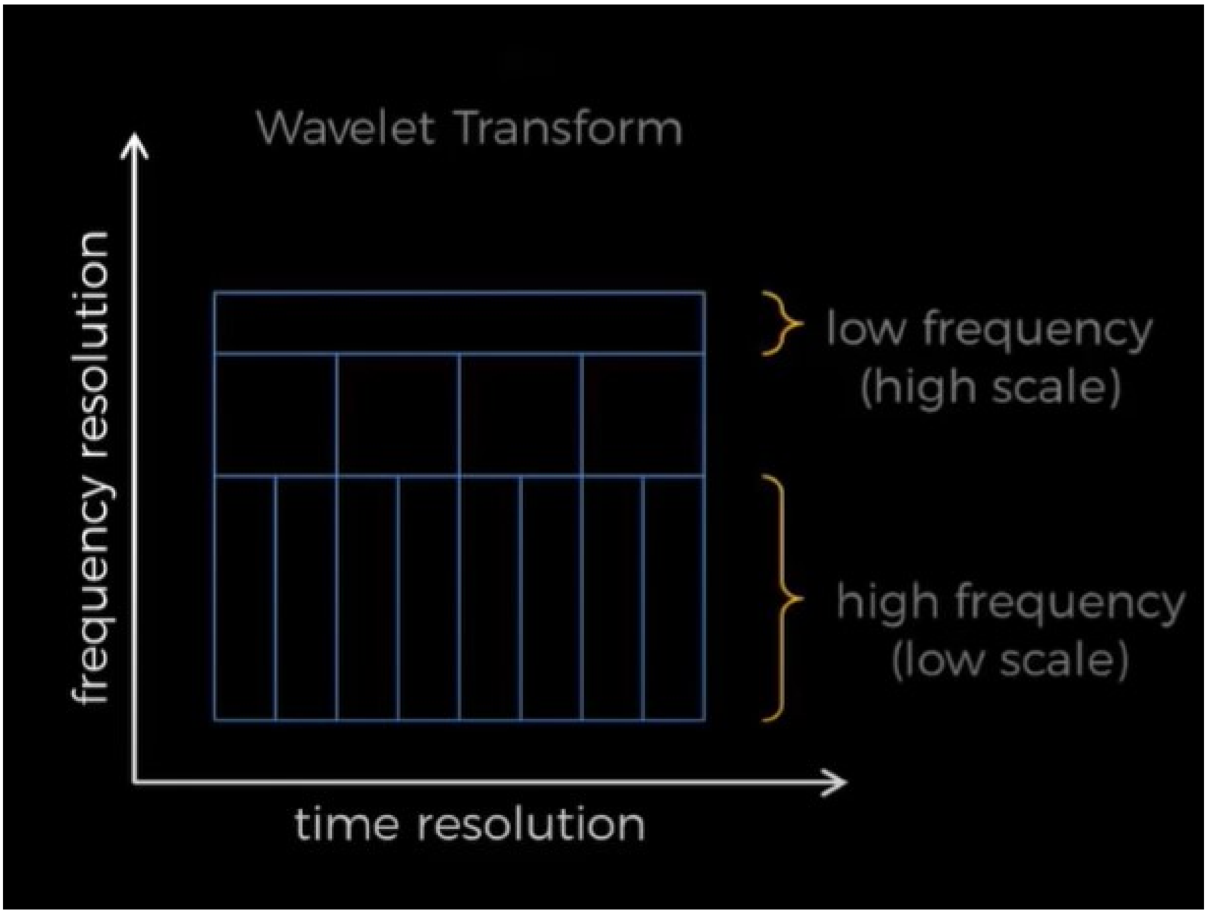
经过(1)(2)两个步骤才算完成2-D DWT的一个stage。

2D-IDWT


$W_\psi^D(j,m,n)$
2 ↑
Rows along m
$h_\psi(m)$
+
2 ↑
Columns along n
$h_\psi(n)$
$W_\psi^V(j,m,n)$
2 ↑
Rows
$h_\varphi(m)$
+
$W_\varphi(j+1,m,n)$
$W_\psi^H(j,m,n)$
2 ↑
Rows
$h_\psi(m)$
+
2 ↑
Columns
$h_\varphi(n)$
$W_\varphi(j,m,n)$
2 ↑
Rows
$h_\varphi(m)$

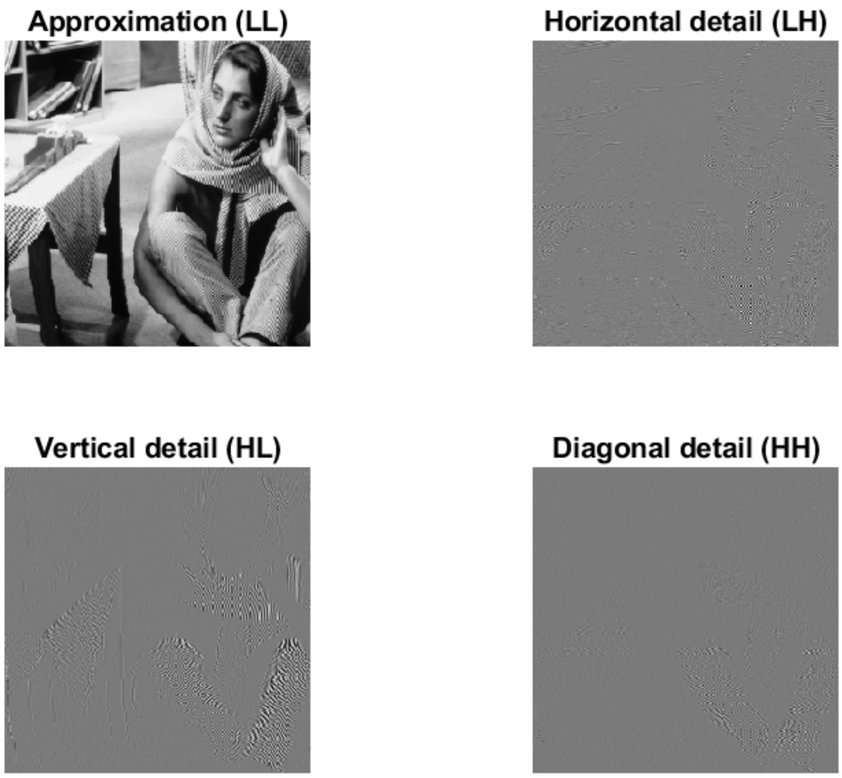
wavelet


FOURIER TRANSFORM
$X(F) = \int_{-\infty}^{\infty} x(t)\ e^{-j2\pi Ft} dt$
frequency
time
WAVELET TRANSFORM
$X(a,b) = \int_{-\infty}^{\infty} x(t)\ \psi_{a,b}^{*}(t)\ dt$
scale, translation
time

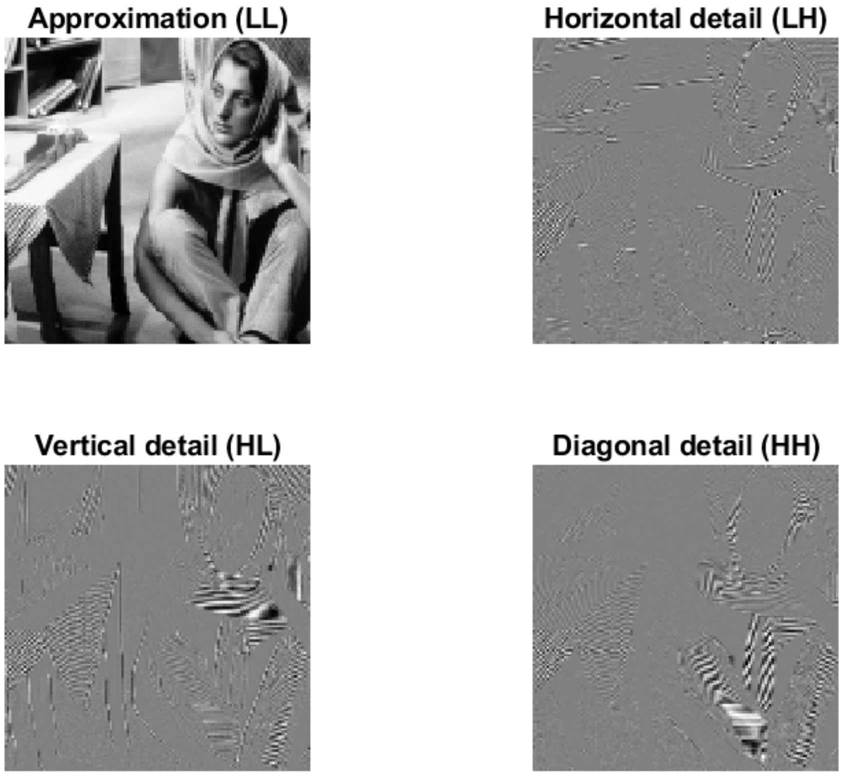


Q2 Result

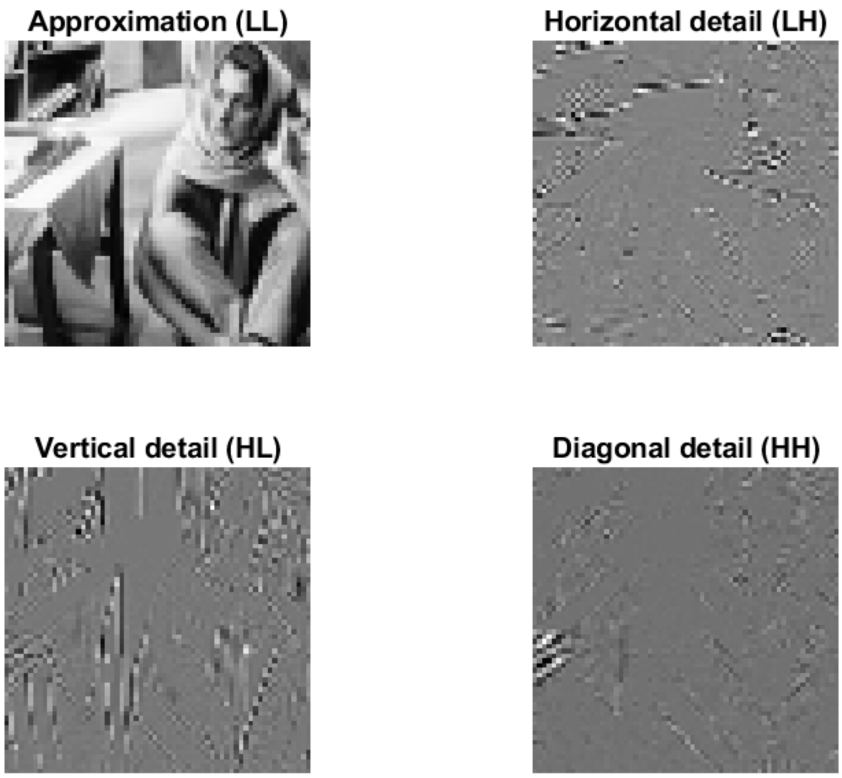
octave1_dwt



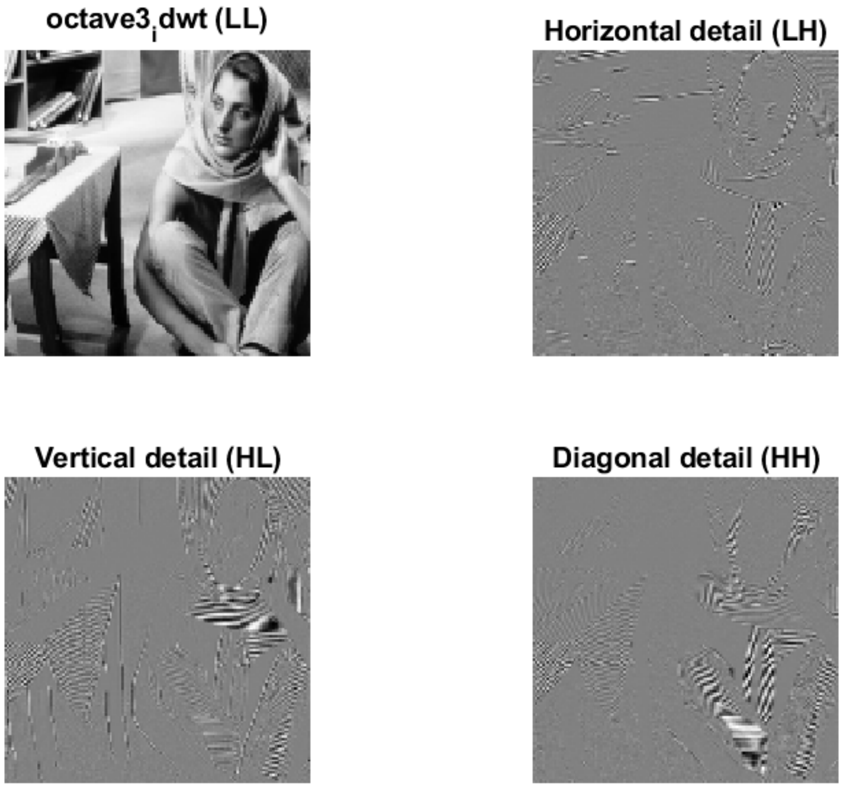
octave2_dwt



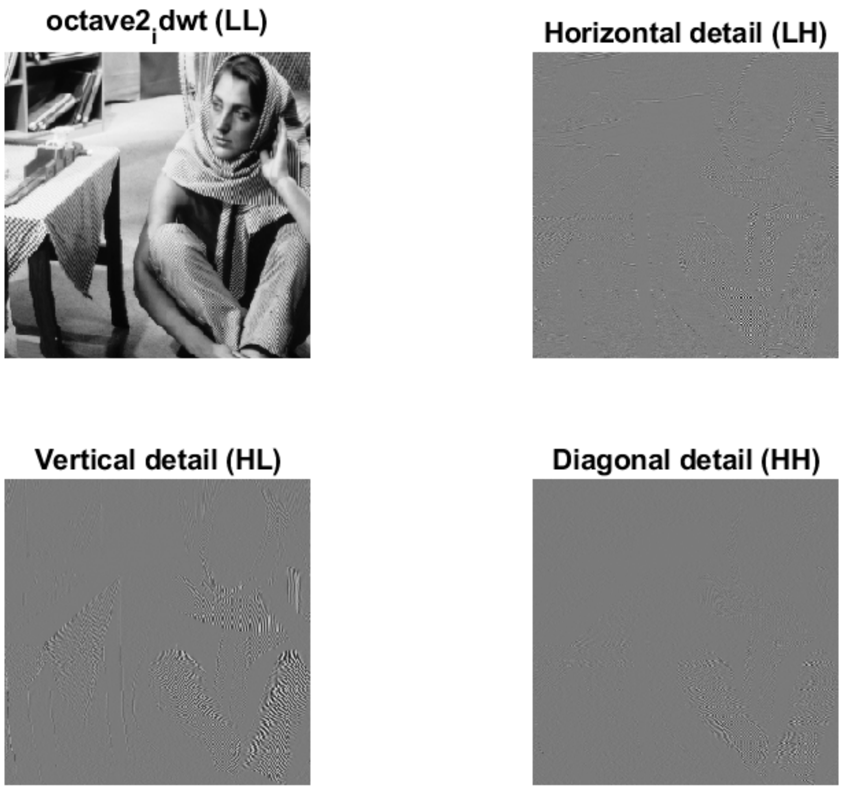
octave3_dwt



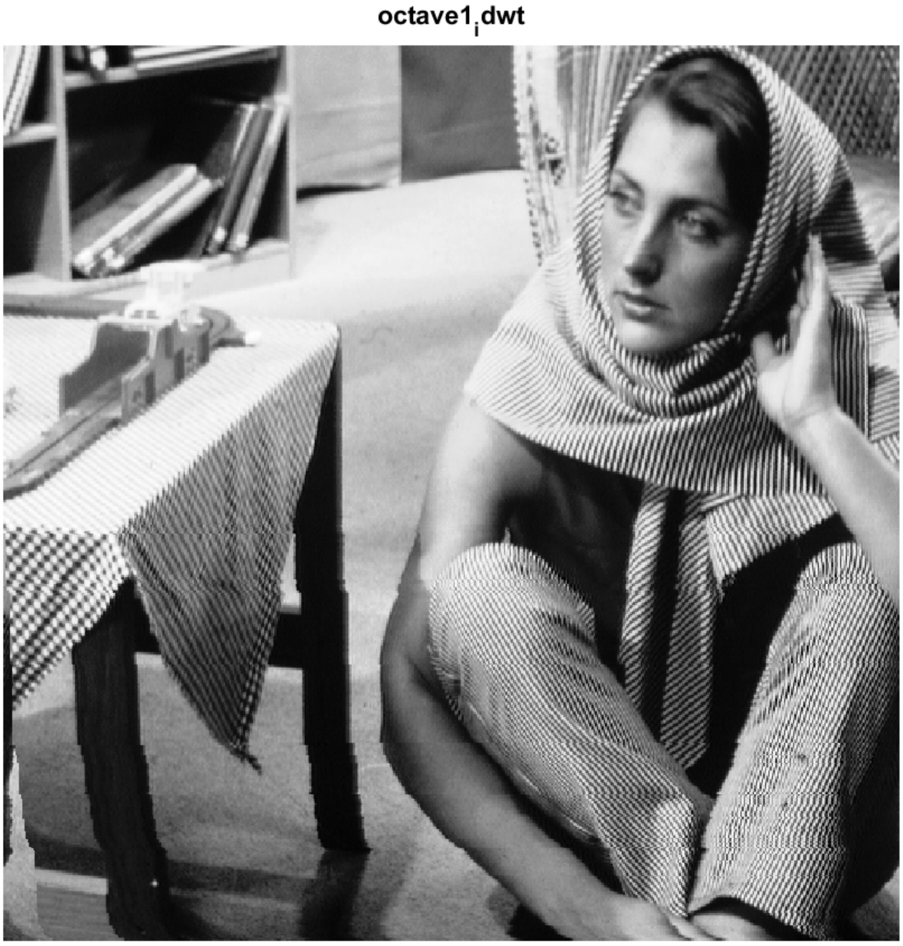
octave3_idwt



octave2_idwt



octave1_idwt



psnr

$$MSE = \frac{1}{M \cdot N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(I(i,j) - \hat{I}(i,j) \right)^2$$

$$PSNR = 10 \log_{10} \left(\frac{MAXI^2}{MSE} \right)$$

```
psnr = 234.2033
```

III. References

[1] Advanced Digital Signal Processing, Adaptive Filters by Prof.Vaibhav Pandit

[2] Advanced Digital Signal Processing, LMS Algorithm by Prof.Vaibhav Pandit

[3] MIT RES.6-008 Digital Signal Processing,Lec 17, 1975 by Alan Oppenheim

[4] EE123 Digital Signal Processing, SP'16 L12 - Discrete Wavelet Transform

[5] Easy Introduction to Wavelets, by Simon Xu

[6] VLSI Digital Signal processing systems Design and Implementation, p25~28 by Parhi

[7] Image Denoising Based on Improved Wavelet Threshold Function for Wireless Camera Networks and Transmissions,Sep 2015, Reserach Gate,Xiaoyu Wang Xiaoxu Ou Bo-Wei Chen Mucheol Kim

[8] 离散小波变换(Discrete Wavelet Transform)

.