Twice wavelet transform
(2DWT)

Uday Mewada(CS171089)
[udaymewada1@gmail.com]

Karina Rathore(CS183D08)
[karinarathore0705@gmail.com]

December 13, 2020

Contents

Chapter 1
Introduction

1.1 What is Wavelet transform

Now-a-days wavelet transforms become very popular when it comes for doing analysis,de-noise, and compression of signals and images. In this the image is actually decomposed into four sub-bands and critically sub-sampled by applying DWT as described in fig(1).
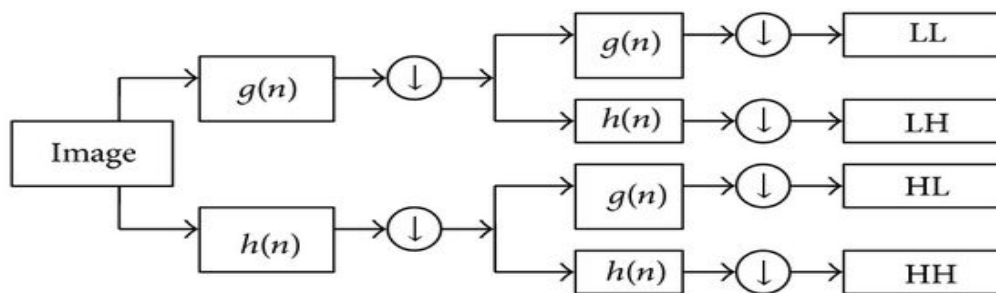

Fig(1)

These sub-bands labeled LH1, HL1 and HH1 represent the finest scale wavelet coefficients i.e., detail images while the sub-bands LL1 corresponds to coarse level coefficients i.e., approximation image.
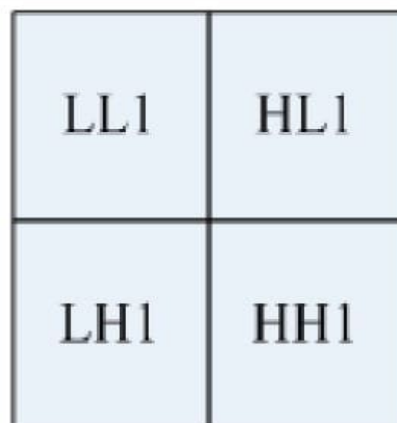
1.2 What is Twice wavelet transform

To obtain the next coarse level of wavelet coefficients, the sub-band LL1 alone is further decomposed and critically sampled. This results in a two-level wavelet decomposition as shown below. Similarly, to obtain further decomposition, LL2 will be used. This process continues until some final scale is reached.
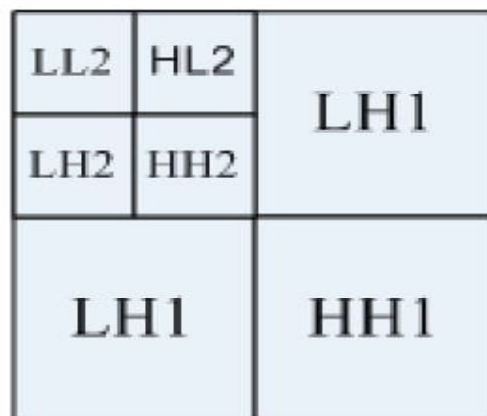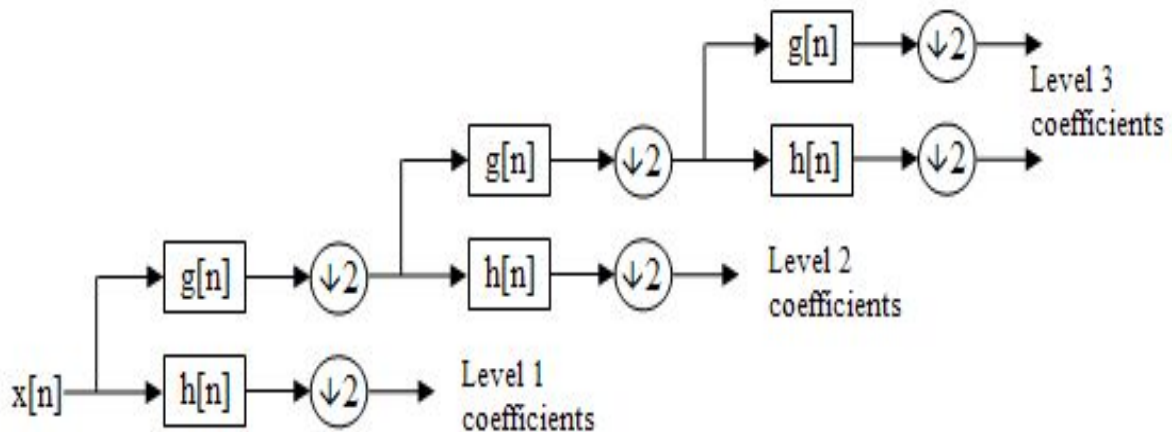
The values or transformed coefficients in approximation and detail images (sub-band images) are the essential features,which are shown here as useful for texture analysis and discrimination. As micro-textures or macro-textures have non-uniform gray level vari-ations, they are statistically characterized by the features in approximation and detail images. In other words, the values in the sub-band images or their combinations or the derived features from these bands uniquely characterize a texture. The features obtained from these wavelet transformed images are shown to be used for texture analysis

Chapter 2
Mathematical Formulation

    The basic idea of wavelet transform is to represent any arbitrary function f as a superposition of wavelets. Any such superposition decomposes f into different scale levels, where each level is then further decomposed with a resolution adapted to the level. One way to achieve such a decomposition writes f as an integral over a and b with appropriate weighting coefficients.

The wavelet transform extracts directional details that capture horizontal (cH), vertical (cV) and the diagonal (cD) activity.

$$\psi^{(a,b)} * t = mod\ (a)^{-0.5} * \psi \frac{(t-b)}{a}$$

where,

a=scale (1/frequency)

b=time shift.

cA2	cH2	cH1	cH
cV2	cD2		
cV1	cD1		
cV		cD	

Chapter 3
Algorithm

In this section, we give an overview of the Single level discrete wavelet transform algorithm and Multilevel decomposition :-

3.1 Single level discrete wavelet transform

Input: Image

Output: Here we get a decomposed image in which we get four images which are horizontal,vertical, diagonals image and approximation image.

pywt.**dwt**(data, wavelet, mode='symmetric', axis=-1)

Single level Discrete Wavelet Transform.

Parameters

data [array_like] Input signal

wavelet [Wavelet object or name] Wavelet to use

mode [str, optional] Signal extension mode, see

Modes.

axis: int, optional Axis over which to compute the DWT. If not given, the last axis is used.

Returns

(cA, cD) [tuple] Approximation and detail coefficients

Notes

Length of coefficient arrays depends on the selected mode. For all modes except periodization:

len(cA) == len(cD) == floor((len(data) + wavelet.dec_len - 1) / 2)

For periodization mode ("per"):

len(cA) == len(cD) == ceil(len(data) / 2)

3.2 Multilevel decomposition using wavedec

Input: Image

Output: Here we get a decomposed image in which we get four images which are horizontal,vertical, and diagonal image and in LL we get one more decomposed image which also contains four images.

pywt.**wavedec**(data, wavelet, mode='symmetric', level=None, axis=-1)

Multilevel 1D Discrete Wavelet Transform of data.

> **Parameters**
>> **data: array_like** Input data
>>
>> **wavelet** [Wavelet object or name string] Wavelet to use
>>
>> **mode** [str, optional] Signal extension mode, see Modes.
>>
>> **level** [int, optional] Decomposition level (must be >= 0). If level is None (default) then it will be calculated using the dwt_max_level function
>>
>> **axis:** int, optional Axis over which to compute the DWT. If not given, the last axis is used.
>
> **Returns**
>> **[cA_n, cD_n, cD_n-1, . . . , cD2, cD1]** [list] Ordered list of coefficients arrays where n denotes the level of decomposition. The first element (cA_n) of the result is an approximation coefficient array and the following elements (cD_n - cD_1) are detailed coefficient arrays.

Chapter 4
Documentation of API

4.1   Package organization

pywt
pywt.data
matplotlib.pyplot
Matplotlib.image

4.2   Methods

mpimg.imread()- The **imread() function** in pyplot module of matplotlib library is used to read an image from a file into an array.

pywt.dwt2()- The dwt2() function can be used to compute the maximum *useful* level of decomposition for the given *input data length* and *wavelet filter length*.

fig.add_subplot()- The add_subplot() method figure module of matplotlib library is used to add an Axes to the figure as part of a subplot arrangement.
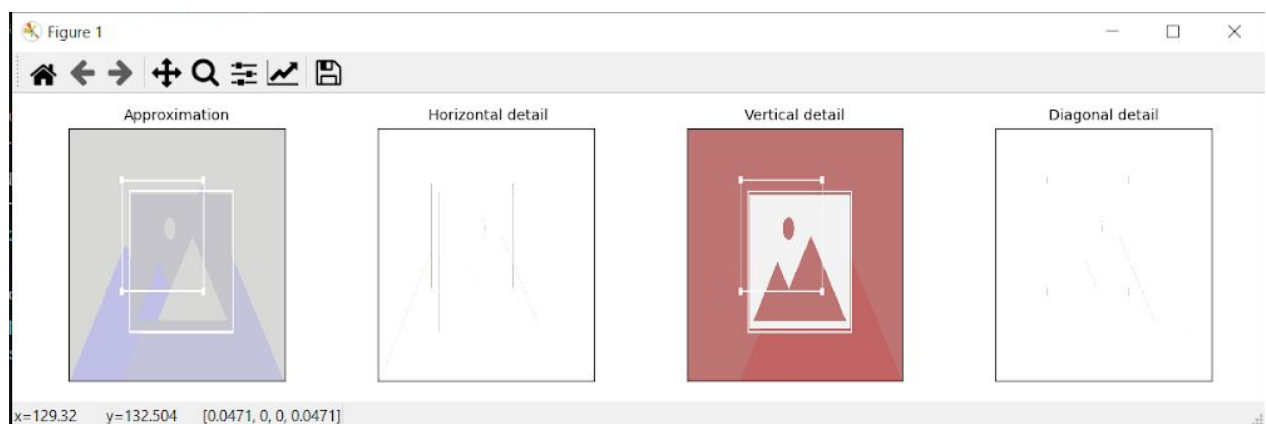
Chapter 5
Examples

```python
import matplotlib.pyplot as plt
import pywt
import pywt.data
import matplotlib.image as mpimg


img=mpimg.imread('C:/ProgramData/imgg.jpeg')


titles = ['Approximation', ' Horizontal detail',
          'Vertical detail', 'Diagonal detail']
coeffs2 = pywt.dwt2(img, 'bior1.3')
LL, (LH, HL, HH) = coeffs2
fig = plt.figure(figsize=(12, 3))
for i, a in enumerate([LL, LH, HL, HH]):
    ax = fig.add_subplot(1, 4, i + 1)
    ax.imshow(a, interpolation="nearest", cmap=plt.cm.gray)
    ax.set_title(titles[i], fontsize=10)
    ax.set_xticks([])
    ax.set_yticks([])

fig.tight_layout()
plt.show()
```

Output

Chapter 6
Learning Outcome

➔ Understand wavelet transformation
➔ Decomposition of image at level 1
➔ Can do decomposition of image at 2 or 3 level

Chapter 7
References

[1] Wavelet tutorial by Robi Polikar .
[2] http://www.ijiset.com/v1s6/IJISET_V1_I6_64.pdf