

PRAKTIKUM 8

Wavelet

Materi:

- Transformasi wavelet

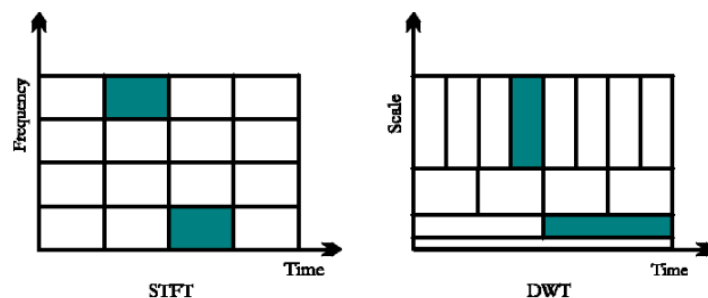
Tujuan Praktikum:

- Mahasiswa dapat melakukan transformasi pada citra menggunakan wavelet transform

A. PENYAJIAN

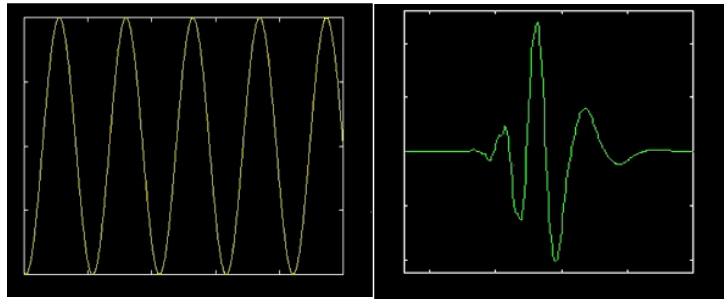
Transformasi Wavelet

Transformasi wavelet merupakan metode ekstraksi fitur sekaligus mereduksi dimensi input citra berukuran besar untuk mempercepat waktu komputasi pada saat melakukan proses pengenalan citra. Ekstraksi fitur merupakan proses mendapatkan penciri atau fitur dari suatu citra. Transformasi wavelet yang biasa digunakan pada citra adalah *discrete wavelet transform* (DWT). Prinsip dasar transformasi wavelet menyerupai fourier transform, dimana terdapat fungsi transformasi yang didalamnya terdiri dari **fungsi basis** dan **faktor skala**. Berbeda dengan analisis fourier yang membagi sinyal kedalam gelombang sinus-cosinus dengan ukuran frekuensi yang sama, wavelet membagi sinyal kedalam skala dan translasi yang berbeda. Ilustrasi dapat dilihat pada Gambar 1.



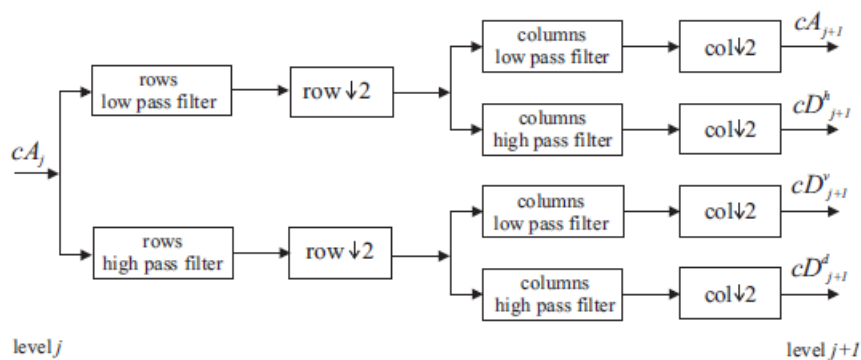
Gambar 1 Sampling window pada *Fourier* dan *Wavelet*

Perbedaan juga terdapat pada fungsi basis yang digunakan. Fourier hanya menggunakan sinus sebagai fungsi basis, sedangkan wavelet dapat menggunakan fungsi basis wavelet seperti Haar, Daubechies, Symlet, Coiflet, Morlet, dan lain-lain. Fungsi basis *wavelet* dapat dilihat pada Gambar 2.

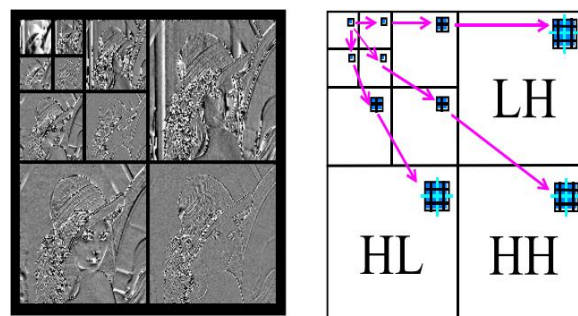


Gambar 3 Kernel sinus dan Kernel Morlet

DWT dilakukan proses dekomposisi dengan memisahkan bagian *low-frequency* dan *high-frequency* berdasarkan kombinasi baris dan kolom pada citra. *Low-frequency* menggunakan skala besar (filter smoothing), sementara *high-frequency* menggunakan skala yang kecil (filter untuk *edge detection*). Koefisien yang dihasilkan yaitu koefisien aproksimasi cA dan *detail* cD , dimana untuk koefisien detail ada di 3 orientasi yaitu horizontal, vertikal, dan diagonal. Skema dekomposisi dapat dilihat pada Gambar 3.

Gambar 4 Skema dekomposisi *wavelet*

Adapun contoh hasil dekomposisi *wavelet* pada citra Lenna yang dapat dilihat pada Gambar 4.

Gambar 4 Hasil dekomposisi *haar wavelet* pada citra Lenna.jpg

Contoh Penggunaan Wavelet :

1. Kompresi citra (format JPEG 2000)
2. Analisis tekstur

3. Analisis ciri
4. Ekstraksi ciri
5. Penghilangan noise
6. Grafika komputer
7. Kompresi video
8. Dll

B. LATIHAN

Membuat program transformasi *wavelet* menggunakan fungsi basis haar

wavelet-haar.cpp

```
#include <iostream>
#include <cv.h>
#include <highgui.h>

using namespace std;
using namespace cv;

void haarwavelet(Mat src, Mat dst, int iter)
{
    float LL, LH, HL, HH;
    assert(src.type() == CV_32FC1);
    assert(src.type() == CV_32FC1);
    int width = src.cols;
    int height = src.rows;
    for(int k=0; k<iter; k++)
    {
        for(int y = 0; y<(height>>(k+1)); y++)
        {
            for(int x=0; x<(width>>(k+1)); x++)
            {
                LL=(src.at<float>(2*y,2*x)+src.at<float>(2*y,2*x+1)+src.
                    at<float>(2*y+1,2*x)+src.at<float>(2*y+1,2*x+1))*0.5;
                dst.at<float>(y,x)=LL;

                LH=(src.at<float>(2*y,2*x)+src.at<float>(2*y+1,2*x)-
                    src.at<float>(2*y,2*x+1)-
                    src.at<float>(2*y+1,2*x+1))*0.5;
                dst.at<float>(y,x+(width>>(k+1)))=LH;

                HL=(src.at<float>(2*y,2*x)+src.at<float>(2*y,2*x+1)-
                    src.at<float>(2*y+1,2*x)-
                    src.at<float>(2*y+1,2*x+1))*0.5;
                dst.at<float>(y+(height>>(k+1)),x)=HL;

                HH=(src.at<float>(2*y,2*x)-src.at<float>(2*y,2*x+1)-
                    src.at<float>(2*y+1,2*x)+src.at<float>(2*y+1,2*x+1))*0.5
                ;
            }
        }
    }
}
```

```

        dst.at<float>(y+(height>>(k+1)),x+(width>>(k+1)))=HH;
    }
}
dst.copyTo(src);
}
}

int main()
{
    Mat img = imread("cameraman.jpg");
    double M=0,m=0;
    Mat gray(img.rows,img.cols,CV_8UC1,Scalar(0,0,0));
    cvtColor(img,gray,COLOR_BGR2GRAY);
    Mat src=Mat(img.rows, img.cols, CV_32FC1);
    Mat dst=Mat(img.rows, img.cols, CV_32FC1);
    gray.convertTo(src,CV_32FC1);

    haarwavelet(src,dst,2);
    minMaxLoc(src,&m,&M);
    if((M-m)>0) {src=src*(1.0/(M-m))-m/(M-m);}
    imshow("Hasil transformasi wavelet",src);
    waitKey(0);
    return 0;
}

```

Output wavelet-haar.cpp



Nama :

NRP :

Nama Dosen :

Nama Asisten :

C. Lembar kerja praktikum

Lakukanlah analisis terhadap tekstur menggunakan *discrete wavelet transform* (DWT) pada semua citra yang telah disediakan di LMS. Simpan kode program (bagian main saja) beserta screenshot citra hasil DWT. **File disimpan dengan format LKP8_NIM_Kelas dalam file .pdf.**

- 1 Lakukan DWT level 3 pada semua citra
- 2 Jelaskan pengaruh level DWT pada hasil transformasi
- 3 Jelaskan hasil transformasi pada bagian LL
- 4 Jelaskan hasil transformasi pada bagian LH
- 5 Jelaskan hasil transformasi pada bagian HL
- 6 Jelaskan hasil transformasi pada bagian HH