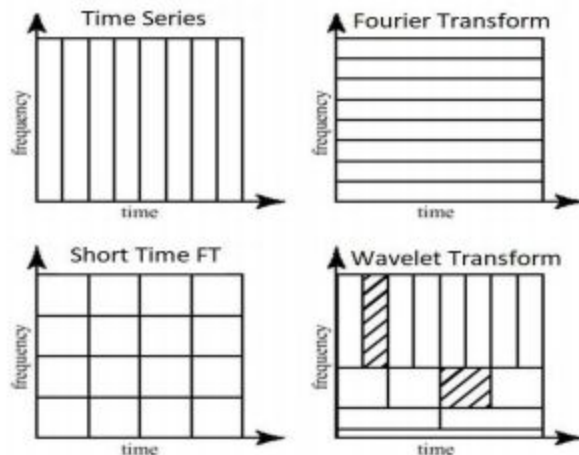


### 3. TRANSFORMASI FOURIER DAN WAVELET

Kode Dosen: YHY

Gambar dibawah ini menjelaskan perbedaan antara transformasi Fourier dan Wavelet.



a. (10) Jelaskan perbedaan Fourier dan Wavelet berdasarkan gambar tersebut?

Pada Gambar tersebut dapat dilihat ketepatan waktu dan frekuensi dari berbagai transformasi. Ukuran blok pada grafik menunjukkan seberapa tepat ketepatan yang dapat kita bedakan dalam domain waktu dan frekuensi. Baik Fourier Transform maupun Wavelet Transform keduanya melakukan perubahan domain pada citra dari domain spasial ke domain frekuensi.

Grafik *Time Series* memiliki ketepatan tinggi dalam domain waktu dan ketepatan nol dalam domain frekuensi. Ini berarti bahwa kita dapat membedakan fitur yang sangat kecil dalam domain waktu dan tidak ada fitur dalam domain frekuensi.

Kebalikannya, pada grafik Transformasi Fourier, yang memiliki ketepatan tinggi adalah domain frekuensi dan domain waktu memiliki ketepatan nol.

Grafik *Short Time Fourier Transform* memiliki ketepatan yang seimbang di domain frekuensi dan waktu.

Sedangkan grafik Transformasi Wavelet ketepatannya dibedakan berdasarkan frekuensi, kemudian berdasarkan waktu:

- Jika nilai frekuensinya kecil, maka akan memiliki ketepatan

tinggi dalam domain frekuensi, ketepatan rendah dalam domain waktu

- Jika nilai frekuensinya besar, maka akan memiliki ketepatan rendah dalam domain frekuensi, ketepatan tinggi dalam domain waktu.

Dari gambar dapat dilihat bahwa jika Transformasi Wavelet dibandingkan dengan Fourier Transform, wavelet dapat menjawab masalah yang tidak dapat diselesaikan oleh Fourier Transform. Pada Fourier Transform, frekuensi pada gambar diproses tanpa melihat frekuensi tersebut merupakan frekuensi tinggi (H) atau frekuensi rendah (L). Sedangkan Transformasi Wavelet dapat memberikan informasi waktu terjadinya sebuah sinyal pada domain frekuensi dan pada proses pengolahan citra, dan dapat memproses frekuensi pada citra, berdasarkan tinggi atau rendah frekuensi tersebut.

- b. (15) Tulis kode program yang sederhana menggunakan python untuk menjelaskan perbedaan tersebut.

- c. (25) Buat program dan bandingkan kinerja kedua metode transformasi tersebut pada kasus menghilangkan noise pada gambar berikut (file image terlampir). Mahasiswa dapat menggunakan fungsi-fungsi pada library opencv.

```

# 3C
#baca gambar dan simpan di variable img
img = cv2.imread('image-soalno-3.jpg',0)

#transformasi wavelet
coeffs2 = pywt.dwt2(img, 'bior1.3')
LL, (LH, HL, HH) = coeffs2
coeffs2 = pywt.dwt2(LL, 'bior1.3')
LL2, (LH, HL, HH) = coeffs2

#Plot gambar awal dan hasil wavelet
plt.subplot(121),plt.imshow(img, interpolation="nearest", cmap=cm.gray)
plt.title('Gambar awal'), plt.xticks([], plt.yticks([]))
plt.subplot(122),plt.imshow(LL2, cmap=cm.gray)
plt.title('Hasil Wavelet'), plt.xticks([], plt.yticks([]))
plt.show()

#Fourier Transform DFT
dft = cv2.dft(np.float32(img),flags=cv2.DFT_COMPLEX_OUTPUT)
dft_shift = np.fft.fftshift(dft)

#Load parameters
rows, cols = img.shape
crow,ccol = rows//2, cols//2

# create a mask first, center square is 1, remaining all zeros
mask = np.zeros((rows,cols),np.uint8)
mask[crow-30:crow+30, ccol-30:ccol+30] = 1

# apply mask and inverse DFT
fshift = dft_shift*mask
f_ishift = np.fft.ifftshift(fshift)
lpf = cv2.idft(f_ishift)
lpf = cv2.magnitude(lpf[:,0],lpf[:,1])

# plot untuk menunjukkan perbedaan gambar awal dan hasil fourier transform
plt.subplot(121),plt.imshow(img, cmap=cm.gray)
plt.title('Gambar Awal'), plt.xticks([], plt.yticks([]))
plt.subplot(122),plt.imshow(lpf, cmap=cm.gray)
plt.title('Hasil Fourier Transform'), plt.xticks([], plt.yticks([]))
plt.show()

```

Kumpulkan jawaban anda dalam bentuk file pdf untuk jawaban masing-masing soal tersebut dan kode program python. Kemudian simpan dalam satu file dan bernama NIM-SOALNO-3.ZIP

**Referensi:**

<http://ataspinar.com/2018/12/21/a-guide-for-using-the-wavelet-transform-in-machine-learning/>

[https://opencv-python-tutroals.readthedocs.io/en/latest/py\\_tutorials/py\\_imgproc/py\\_transforms/py\\_fourier\\_transform/py\\_fourier\\_transform.html](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_transforms/py_fourier_transform/py_fourier_transform.html)