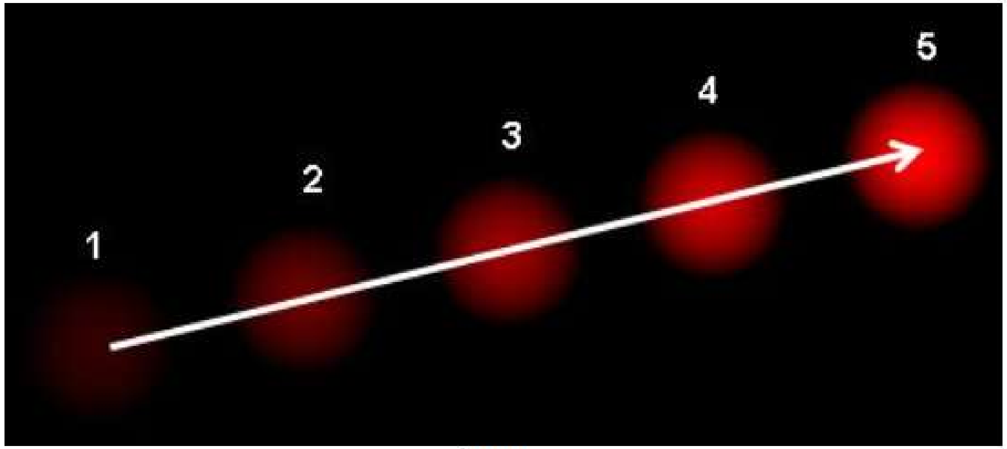
Optical Flow

Optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of object or camera. It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second. Consider the image below (Image Courtesy: Wikipedia article on Optical Flow).



image

It shows a ball moving in 5 consecutive frames. The arrow shows its displacement vector. Optical flow has many applications in areas like :

- Structure from Motion
- Video Compression
- Video Stabilization ...

Optical flow works on several assumptions:

1. The pixel intensities of an object do not change between consecutive frames.
2. Neighbouring pixels have similar motion.

Consider a pixel $I(x, y, t)$ in first frame (Check a new dimension, time, is added here. Earlier we were working with images only, so no need of time). It moves by distance (dx, dy) in next frame taken after dt time. So since those pixels are the same and intensity does not change, we can say,

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

Then take taylor series approximation of right-hand side, remove common terms and divide by dt to get the following equation:

$$f_x u + f_y v + f_t = 0$$

where:

$$f_x = \frac{\partial f}{\partial x} \; ; \; f_y = \frac{\partial f}{\partial y}$$

$$u = \frac{dx}{dt} \; ; \; v = \frac{dy}{dt}$$

Above equation is called Optical Flow equation. In it, we can find f_x and f_y, they are image gradients. Similarly f_t is the gradient along time. But (u, v) is unknown. We cannot solve this one equation with two unknown variables. So several methods are provided to solve this problem and one of them is Lucas-Kanade.

Lucas-Kanade method

We have seen an assumption before, that all the neighbouring pixels will have similar motion. Lucas-Kanade method takes a 3x3 patch around the point. So all the 9 points have the same motion. We can find (f_x, f_y, f_t) for these 9 points. So now our problem becomes solving 9 equations with two unknown variables which is over-determined. A better solution is obtained with least square fit method. Below is the final solution which is two equation-two unknown problem and solve to get the solution.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum_i f_{x_i}{}^2 & \sum_i f_{x_i} f_{y_i} \\ \sum_i f_{x_i} f_{y_i} & \sum_i f_{y_i}{}^2 \end{bmatrix}^{-1} \begin{bmatrix} - \sum_i f_{x_i} f_{t_i} \\ - \sum_i f_{y_i} f_{t_i} \end{bmatrix}$$

(Check similarity of inverse matrix with Harris corner detector. It denotes that corners are better points to be tracked.)

So from the user point of view, the idea is simple, we give some points to track, we receive the optical flow vectors of those points. But again there are some problems. Until now, we were dealing with small motions, so it fails when there is a large motion. To deal with this we use pyramids. When we go up in the pyramid, small motions are removed and large motions become small motions. So by applying Lucas-Kanade there, we get optical flow along with the scale.

TWO DIMENSIONAL DISCRETE WAVELET TRANSFORM (2D-DWT)

The scaled and translated basis elements of the 2D wavelet transform are given by [3].

$$LL = \phi(x,y) = \phi(x)\,\phi(y)$$

$$LH = \psi^{H}(x,y) = \psi(x)\phi(y)$$

$$HL = \psi^{V}(x,y) = \phi(x)\,\psi(y)$$

$$HH = \psi^{D}(x,y) = \psi(x)\,\psi(y)$$

where the super scripts H, V, and D refer to the decomposition direction of the wavelet. Two-dimensional wavelets are utilized as a part of image manipulation. The multiresolution representation of scaling and wavelet functions for the 2-D given below:

$$\phi_{j,m,n}(x,y) = 2^{j/2}\phi(2^{j}x - m, 2^{j}y - n)$$

$$\psi^{i}_{j,m,n}(x,y) = 2^{j/2}\psi^{i}(2^{j}x - m, 2^{j}y - n)$$

where I = { H,V,D}

The discrete wavelet transform function f(x, y) of size M x N is given below:

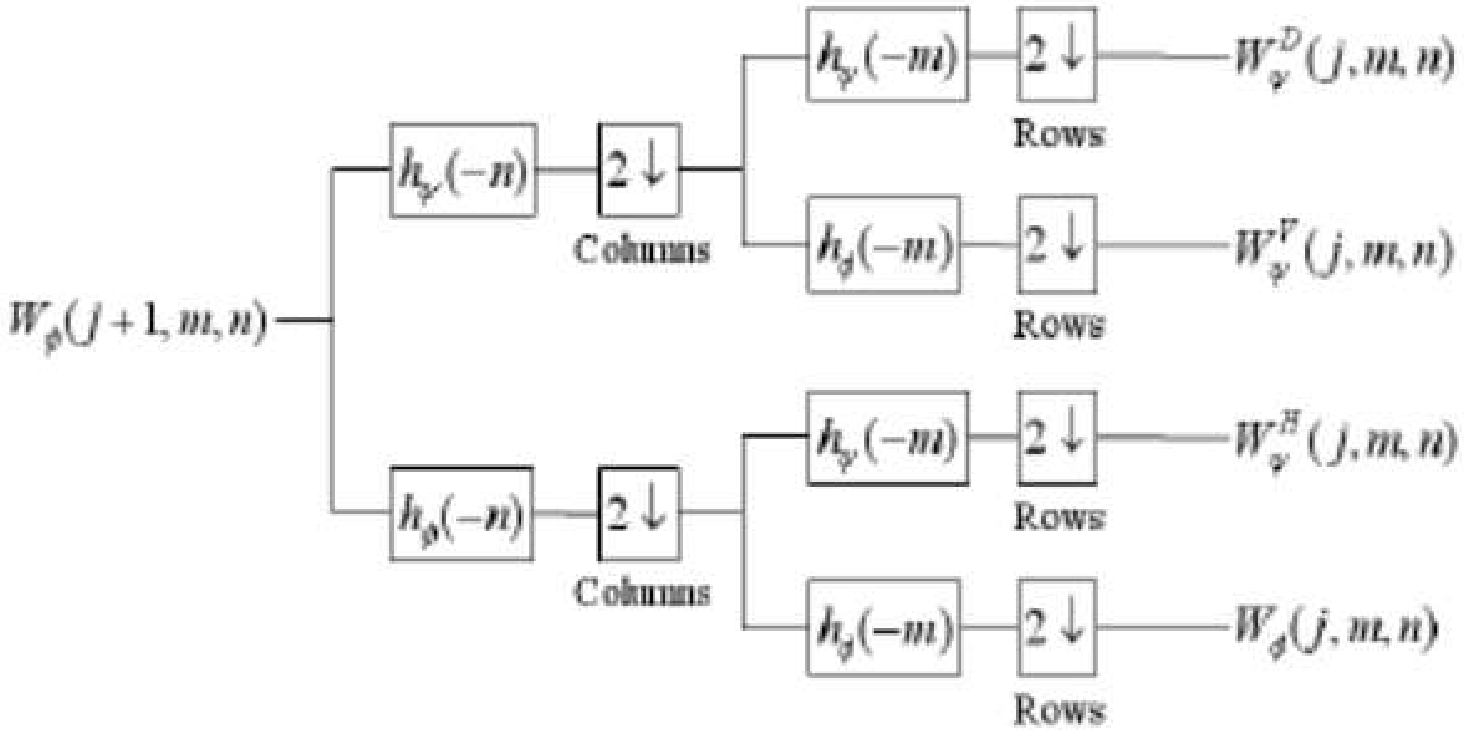


Fig. Analysis Filter bank of 2D

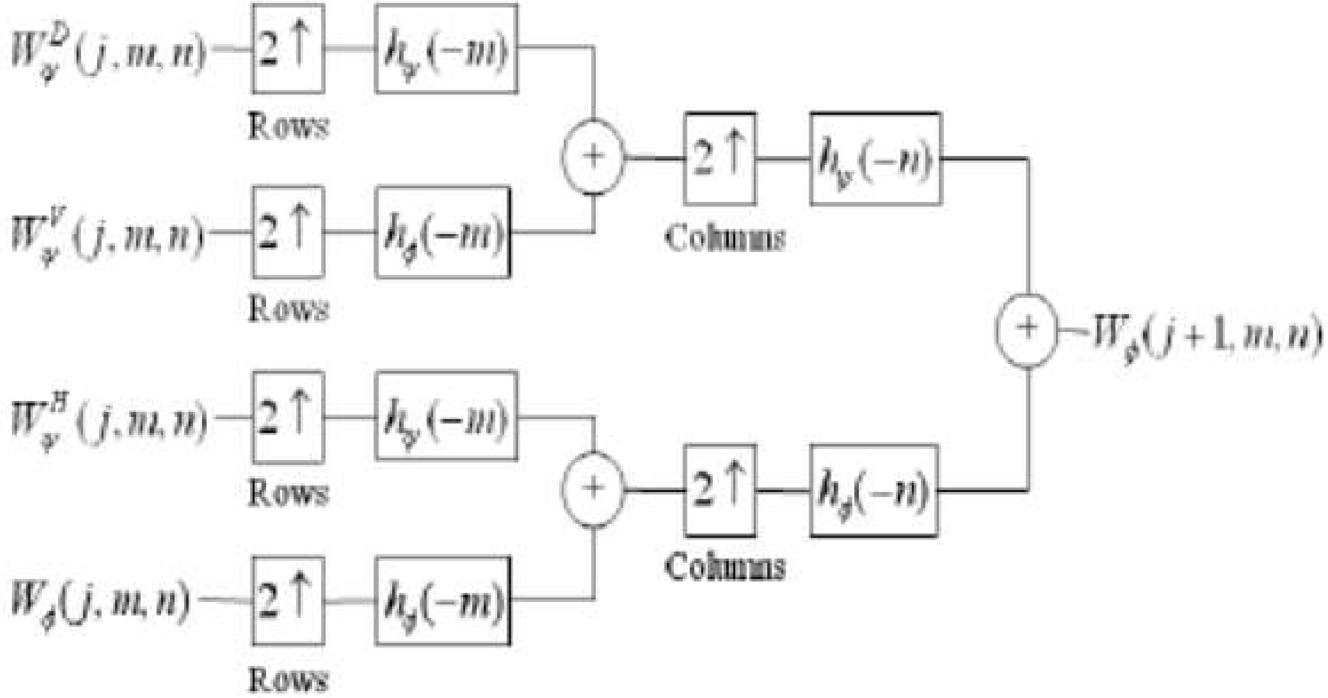


Fig . Synthesis Filter bank of 2D

1. Scaling function of the wavelet representation is shown as

$$W_\phi(j_{o,}m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, \phi_{j_{o,m,n}}(x,y)$$

The corresponding wavelet function of the horizontal, vertical and diagonal representation of the images are as follows:

2. Horizontal subband representation

$$W_{\psi}^{H}(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, \psi_{j,m,n}^{H}(x,y)$$

3. Vertical subband representation

$$W_{\psi}^{V}(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, \psi_{j,m,n}^{V}(x,y)$$

4. Diagonal subband representation

$$W_{\psi}^{D}(j,m,n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, \psi_{j,m,n}^{D}(x,y)$$

From the above mentioned scaling and wavelet functions, namely, W_{ϕ} and W_{ψ} , one can easily acquired through the inverse discrete wavelet transform for the image signal as

$$f(x,y)=\frac{1}{\sqrt{MN}}\sum_{m}\sum_{n}W_{\phi}\,(j_0,m,n)\,\phi_{j_o,m,n}(x,y)$$
$$+\frac{1}{\sqrt{MN}}\sum_{i=H,V,D}\sum_{j=j_0}^{\infty}\sum_{m}\sum_{n}W_{\psi}^{i}\,(j,m,n)\,\psi_{j,m,n}^{i}(x,y)$$