RESEARCH ARTICLE OPEN ACCESS

FPGA Implementation of 2-D DCT & DWT Engines for Vision Based Tracking of Dynamic Obstacles

Remya Ramesan*, Pavana H**,Radhika G***
*(Department of ECE, Asst Professor, MVJCE, Bangalore-560067, Karnataka, India)
** (Department of ECE, Asst Professor, MVJCE, Bangalore-560067, Karnataka, India)
*** (Department of ECE, Asst Professor, MVJCE, Bangalore-560067, Karnataka, India)

ABSTRACT
Real time motion estimation for tracking is a challenging task. Several techniques can transform an image into frequency domain, such as DCT, DFT and wavelet transform. Direct implementation of 2-D DCT takes N^4 multiplications for an N x N image which is impractical. The proposed architecture for implementation of 2-D DCT uses look up tables. They are used to store pre-computed vector products that completely eliminate the multiplier. This makes the architecture highly time efficient, and the routing delay and power consumption is also reduced significantly. Another approach, 2-D discrete wavelet transform based motion estimation (DWT-ME) provides substantial improvements in quality and area. The proposed architecture uses Haar wavelet transform for motion estimation. In this paper, we present the comparison of the performance of discrete cosine transform, discrete wavelet transform for implementation in motion estimation.
***Keywords* -** Discrete Cosine Transform, Discrete Wavelet Transform, Look up tables, Haar wavelet transform

I. Introduction

The most commonly used vision based motion estimation (ME) algorithms are 2D-DCT Based Motion Estimation (DXT-ME), Full Search Block Matching algorithm (BKA-ME), Correlation based Approach (CLT-ME), Frame Differencing etc. The BKA-ME algorithm searches for the best candidate block among all the blocks in a search area of larger size in terms of either the mean-square error or the mean of absolute frame difference. But the computational complexity of this approach is very high i.e. O ($N^2.M^2$), for an N x N search blocks in an M x M visual region. The CLT-ME algorithm uses Complex Lapped Transforms to avoid the block effect but it still requires searching over a larger search area. Frame differencing algorithm, although computationally efficient, has limitations when the obstacle to be tracked is decelerating quickly.

Compared to the above mentioned ME algorithms, DXT-ME is found to be the most efficient algorithm in terms of computational complexity i.e. O (N^2) for visual region of size N x N and is also robust in noisy environments. It utilizes sinusoidal orthogonal principles to estimate the displacement of moving objects based on concept of pseudo phases. The DXT-ME algorithm provides for highly parallel and local operations. This property makes parallel implementation feasible which increases the system throughput. Hence, it is very useful for real time applications. The overall architecture when implemented on a general purpose processor failed severely in meeting the stringent deadlines of time; hence there is a need for hardware accelerators such as FPGAs. The advantage of using FPGAs is their flexibility in parallel implementation of large array of computational blocks. Also they have outstanding properties in metrics of size, power consumption, re-configurability, lower time to proto-type and lower overall turn-around time.

Look up Table (LUT) based architecture have been designed for fast 2-D DCT computation that can be used in the DXT-ME scheme. For computation of 2-D DCT, we have used row-column decomposition approach that makes use of one dimensional DCT (1-D DCT) operations twice and each 1-D DCT computation involves the use of Chen's algorithm. The Transformation coefficients of Chen's matrices, C(i) (i:0,1,…,7)are multiplied with all possible combinations of the inputs that range from 0-255 (e.g.: C(i)*0, C(i)*1, C(i)*2,…, C(i)*255) and these combinations are stored in LUTs. To obtain the result of multiplication of the input and the coefficient, the input is used as pointer to index the LUTs. This results in a purely combinational logic to compute 1-D DCT. Thus, for computation of 1-D DCT, it requires a latency of just 1 clock cycle. This makes the proposed architecture very time efficient.

Another method proposed is Haar Wavelet Based 2-D DWT-ME. Wavelet-based coding provides substantial improvements in picture quality at higher compression ratios. In DWT, the most prominent information in the signal appears in high amplitudes and the less prominent information appears in very low amplitudes.

II. Background

2.1 Motion estimation techniques

Motion estimation is the process of determining motion vector that describe the transformation from one 2D image to another.Figure.2.1 shows the different techniques which can be employed for motion estimation.

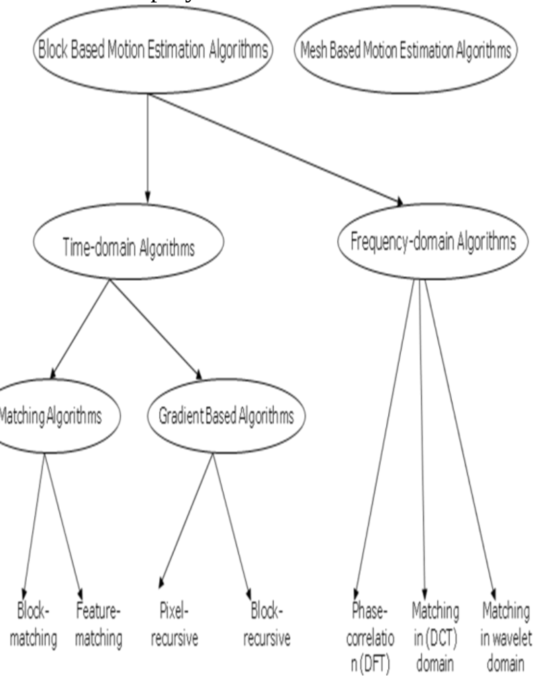


Fig 2.1.Motion estimation techniques

2.2 Discrete Cosine Transforms

The Discrete Cosine Transform (DCT) has been widely used in the field of image compression because of its excellent energy compaction properties. It is an orthogonal transform that is used to decorrelate the input image by exploiting the redundancies between adjacent pixels while still keeping the energy of the signal significant .There are other properties which can be used to compute DCT cost effectively. Before explaining the use of these properties, it is necessary to define 1-D DCT and 2-D DC. Also, since 1D-DCT is the basic building block of 2D-DCT for our proposed architecture, it is necessary for us to define 1D-DCT first.

The most common DCT definition of a 1-D sequence of length N is;

$$F(u)=\sqrt{\frac{2}{n}}\;C(u)\sum_{x=0}^{n-1} I(x)\cos\frac{(2x+1)u\pi}{2n}, u=0...n-1 \qquad ...(1)$$

Similarly, for an input signal x (m, n) {m, n: 0, 1, 2…N- 1}, the 2-D DCT is defined as;

$$F(u,v)=\frac{2}{\sqrt{nm}}C(u)C(v)\sum_{y=0}^{m-1}\sum_{x=0}^{n-1} I(x,y) * \cos\left(\frac{(2x+1)u\pi}{2n}\right) * \cos\left(\frac{(2y+1)v\pi}{2m}\right) \qquad ...(2)$$

$$where \quad C(u)=\begin{cases}\frac{1}{\sqrt{2}} & for \quad u=0,\\ 1 & otherwise\end{cases}$$

2.3 Discrete Wavelet Transform

Wavelet analysis can be used divided the information of an image into approximation and detailed sub signal. The approximation sub signal shows the general trend of pixel value, and three detailed sub signal show vertical, horizontal and diagonal details or changes in image. If these detail is very small than they can be set to zero without significantly changing the image. If the number of zeroes is greater than the compression ratio is also greater. There is two types of wavelet is used. First one is Continues wavelet transform and second one is discrete wavelet transform. Wavelet analysis is computed by filter bank. There is two type of filter:

1) High pass filter: high frequency information is kept, low frequency information is lost.

2) Low pass filter: law frequency information is kept, high frequency information is lost.

So signal is effectively decomposed into two parts, a detailed part(high frequency) and approximation part(law frequency). Level 1 detail is horizontal detail, level2 detail is vertical detail and level 3 detail is diagonal detail of the image signal.

III. Design And Implementation

3.1 2-D DCT Engine Architecture

The 2-D DCT engine is implemented using a systematic step by step process. The first step in the design is image acquisition process followed by designing of kalman filter, which estimates and reduces the size of image that has to undergo DCT operation. The main module to be designed is DCT module which involves a number of sub-modules. A two dimensional DCT has to be implemented as the input is an image. Inorder to perform 2-D DCT, firstly 1-D DCT is implemented using Look-up tables. Then a row transformation is done to obtain 2-D DCT.

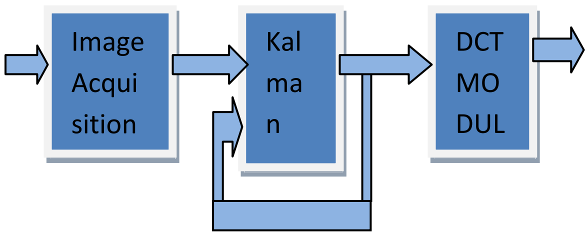


Figure 3.1 Block diagram of the DCT design

3.1. 1 Image acquisition

Image acquisition is done so that the input image of size 256x256 matrix is reduced to 3x3 matrix size. This operation is required as kalman filter needs an image of reduced size.

3.1.2 Kalman filter

A Kalman filter is an optimal estimator .It infers parameters of interest from indirect, inaccurate and uncertain observations. It is recursive so that new measurements can be processed as they arrive. The Kalman Filter is an important part of this project. The purpose of the Kalman Filter is to use measurements observed over time, which contains random variations of noise, and produce a value that is accurate to the true values of the measurements. It does this by predicting a value, estimating the uncertainty of the predicted value, and computing a weighted average of the predicted value and calculated value. The Kalman Filter first predicts the next value as well as the error covariance. When the next value comes into the filter, the Kalman gain is computed, the estimate is updated with the observed value, and the error covariance is updated. This helps to get rid of the noise within the signal.

3.1.3 FPGA implementation of proposed LUT based DCT architecture

Our approach uses separability property of DCT for computation of 2-D DCT. The following subsections give the implementation detail of proposed 1-D DCT and 2-D DCT architectures.

3.1.3.1 Implementation of 1-D DCT Architecture

The N-point DCT can be computed using Chen's Algorithm. Equations (1),(2) give the 8-point 1-D DCT matrices after applying Chen's algorithm.

$$\begin{bmatrix} X(0) \\ X(2) \\ X(4) \\ X(6) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} C(4) & C(4) & C(4) & C(4) \\ C(2) & C(6) & -C(6) & -C(2) \\ C(4) & -C(4) & -C(4) & C(4) \\ C(6) & -C(2) & C(2) & -C(1) \end{bmatrix} \begin{bmatrix} x(0)+x(7) \\ x(1)+x(6) \\ x(2)+x(5) \\ x(3)+x(4) \end{bmatrix} \quad \ldots(3)$$

$$\begin{bmatrix} X(1) \\ X(3) \\ X(5) \\ X(7) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} C(1) & C(5) & C(5) & C(7) \\ C(3) & C(7) & -C(1) & -C(3) \\ C(5) & C(1) & -C(7) & C(5) \\ C(7) & -C(3) & C(3) & -C(1) \end{bmatrix} \begin{bmatrix} x(0)-x(7) \\ x(1)-x(6) \\ x(2)-x(5) \\ x(3)-x(4) \end{bmatrix} \quad \ldots(4)$$

The top level scheme of our architecture for computation of 1-D DCT using Chen's algorithm consists of three blocks:

1. Adder/Subtractor block
2. LUT Structure with Input Vector Indexing.
3. Accumulator/Shifter block

3.1.3.1.1 Adder-Subtractor Block:

It is a block used to perform addition/subtraction operations of the input vectors, x (n) {n: 0, 1, 2… N-1} . The output of this block (zn) is fed to the LUT structure.

3.1.3.1.2 LUT Structure with Indexing:

The block uses Block RAMs in the form of LUT structure that stores the pre-computed products of all possible input values and the transformation coefficients. This replaces the use of multipliers. Since there are 7 constant coefficients in the Chen's transformation matrix, we use 7 Block ROM structures each having 256 precomputed results (for inputs that range from 0 to 255) as shown in Fig.3.2. The results are accessed by using z (n) as indexing pointers to the ROM/LUT. Fig. 3.2 shows the LUT/RAM structure which stores the pre-computed values and a converter block used to create vector for indexing the RA M/LUT.

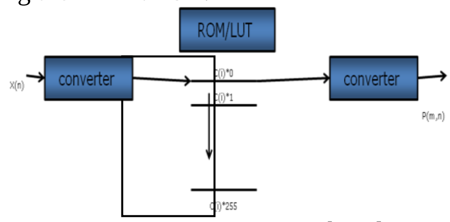


Figure 3.2 LUT Structure with Indexing

The converter sub-block before the RAM structure, is used to make the RAM reusable for inputs, z (n) that range from -256 to 0. This converter block takes the absolute value of the input, z(n). Once the pre-computed product is accessed from the RAM, the converter block after RAM structure is used to convert the result to positive or negative value, P(m,n) depending on the MSB of the input, z(n). The equations resulting from the LUT structure are shown in equation (5) and (6).

$$\begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(6) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} P(0,0) & P(0,1) & P(0,2) & P(0,3) \\ P(1,0) & P(1,1) & P(1,2) & P(1,3) \\ P(2,0) & P(2,1) & P(2,2) & P(2,3) \\ P(3,0) & P(3,1) & P(3,2) & P(3,3) \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \ldots(5)$$

$$\begin{bmatrix} x(1) \\ x(3) \\ x(5) \\ x(7) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} P(4,0) & P(4,1) & P(4,2) & P(4,3) \\ P(5,0) & P(5,1) & P(5,2) & P(5,3) \\ P(6,0) & P(6,1) & P(6,2) & P(6,3) \\ P(7,0) & P(7,1) & P(7,2) & P(7,3) \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \ldots(6)$$

3.1.3.1.3 Accumulator-Shifter Block

The accumulator-shifter block is used to compute the vector products of above equations. This requires a 2 stage 2's complement adder for additions involved in the vector product and a shifter that is used to multiply by ½ factors.

3.1.3.2 Implementation of 2-D DCT Architecture

For implementation of 2-D DCT, we use separability property of DCT. This property involves the row-column decomposition of 2D-DCT. The 2D-DCT given by (2) can be rewritten as;

$$X(l,k) = \frac{4}{N^2} C(k) C(l) \sum_{n=0}^{N-1} \left\{ \sum_{m=0}^{N-1} x(m,n) \cos\left[\frac{\pi(2m+1)k}{2N}\right] \right\} \cos\left[\frac{\pi(2n+1)l}{2N}\right] \quad \ldots(7)$$

Hence 2-D DCT can be computed by successive 1-D operations on rows and then on columns or vice versa of an image. The scheme used in our proposed architecture for computation of 2-D DCT which is shown in Fig. 3.3.

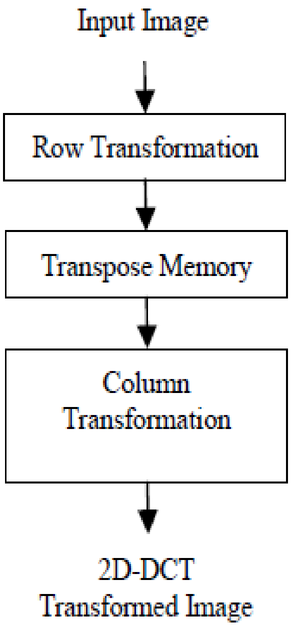


Figure 3.3 2-D DCT computation scheme

3.2 Implementation of dwt using haar wavelet transform

The implementation using DWT helps us to eliminate the use of memory area of LUT based structure. The simplest of wavelet transform is Haar wavelet transform, which is being utilised here.

The wavelet transform, as a multiresolution domain that hybrid the frequency and the spatial domain, has proved that it is a very appropriate and reliable domain for a powerful motion estimation and compensation. For this, we have been encouraged to study and exploit it, and more precisely the DWT, in our motion estimation system. The DWT consists on applying hierarchically low-pass (L) and high-pass (H) filters after decimation (sub-sampling the image on two parts). This procedure is repeated until reaching a prefixed level.

When the original image is decomposed into four-subband images, it has to deal with row and column directions separately. First, the high-pass filter G and the low-pass filter H are exploited for each row data, and then are down-sampled by 2 to get high- and low-frequency components of the row. Next, the highand the low-pass filters are applied again for each high- and low-frequency components of the column, and then are down-sampled by 2. By way of the above processing, the four subband images are generated: HH, HL, LH, and LL. Each subband image has its own feature, such as the low-frequency information is preserved in the LL-band and the high frequency information is almost preserved in the HH-, HL-, and LH-bands. The LL-subband image can be further decomposed in the same way for the second level subband image. By using 2-D DWT, an image can be decomposed into any level subband images, as shown in Fig. 3.4.

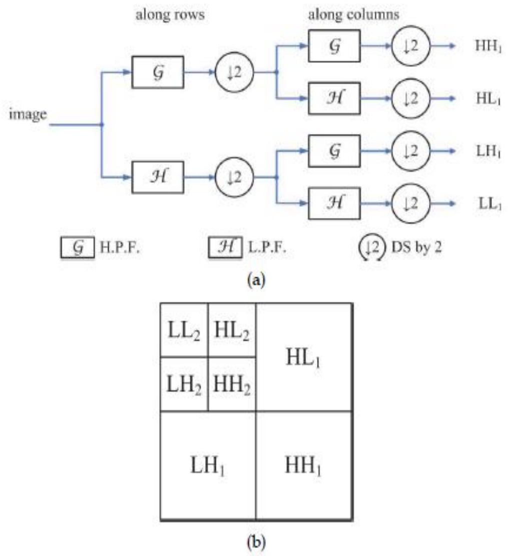


Fig. 3.4. Diagrams of DWT image decomposition: (a) the 1-L 2-D analysis DWT image decomposition process, (b) the 2-L 2-D analysis DWT subband.

3.2.1. Implementation of 1-D Haar Transform

To understand how wavelets work, let us start with a simple example. Assume we have a 1D image with a resolution of four pixels, having values [9 7 3 5]. Haar wavelet basis can be used to represent this image by computing a wavelet transform. To do this, first the average the pixels together, pairwise, is calculated to get the new lower resolution image with pixel values [8 4]. Clearly, some information is lost in this averaging process. We need to store some detail coefficients to recover the original four pixel values from the two averaged values. In our example, 1 is chosen for the first detail coefficient, since the average computed is 1 less than 9 and 1 more than 7. This single number is used to recover the first two pixels of our original four-pixel image. Similarly, the second detail coefficient is -1, since 4 + (-1) = 3 and 4 - (-1) =5. Thus, the original image is decomposed into a lower resolution (two-pixel) version and a pair of detail coefficients. Repeating this process recursively on the averages gives the full decomposition shown in Table 1:

Table 1.1-D Transform with average and detail coefficient

Resolution	Averages	Detail Coefficients
4	[9 7 3 5]	
2	[8 4]	[1 -1]
1	[6]	[2]

Thus, for the one-dimensional Haar basis, the wavelet transform of the original four-pixel image is given by [6 2 1 - 1]. We call the way used to compute the wavelet transform by recursively averaging and differencing coefficients, filter bank. We can reconstruct the image to any resolution by recursively adding and subtracting the detail coefficients from the lower resolution versions.

3.2.2 Implementation of 2-D Haar Transform

The transformation of the 2D image is a 2D generalization of the 1D wavelet transformed already discussed. It applies the 1D wavelet transform to each row of pixel values. This operation provides us an average value along with detail coefficients for each row. Next, these transformed rows are treated as if they were themselves an image and apply the 1D transform to each column. The resulting values are all detail coefficients except a single overall average co-efficient. In order to complete the transformation, this process is repeated recursively only on the quadrant containing averages.

3.2.3 Object Tracking

The 2-D DWT can be used for detecting and tracking moving objects and only the LL3-band image is used for detecting the moving object motion. Firstly, DWT of both the images are performed individually. Secondly, tracking of object is performed by comparing the DWT of images because noises are preserved in high frequency, it can reduce the computing cost for post processing by using the LL3-band image. This method can be used for coping with noise or fake motion effectively; however the conventional DWT scheme has the disadvantages of complicated calculation when original image is decomposed into the LL-band image. Moreover if it uses an LL3-band image to deal with the fake motion, it may cause incomplete moving object detecting regions.

IV. Result

4.1 DCT Module Simulation Result

This section evaluates the performance of the proposed DCT and DWT algorithm. The simulation results of the blocks used for the simulation are as shown.

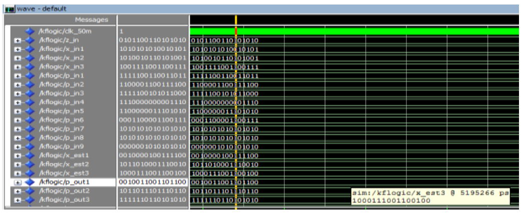


Fig.4.1.1.Simulation result of Kalman filter

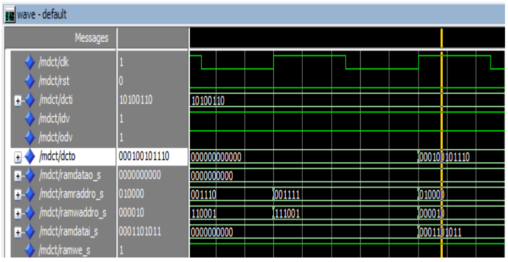
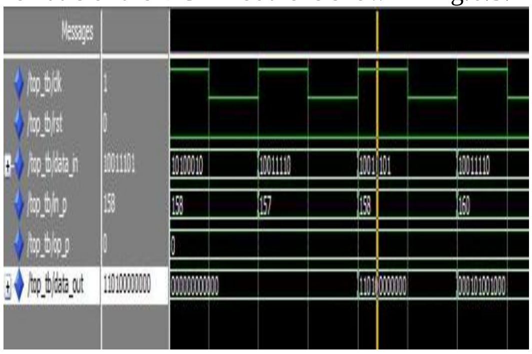


Fig.4.1.2.Simulation result of DCT Module

When the DCT input,ie data_in is 10100110.the output, data_out obtained is 000100101110.
For another input, data_in=10011110;
The corresponding outptut, data_out=000101001000.

DCT module is the main module designed and implemented in the project. It involves design of ROMs for storage of pre-computed vector products, adder/ subtractor, accumulator and shifter.The RTL schematic of the DCT module is shown in fig.6.3.



4.2. DWT Module Simulation Result

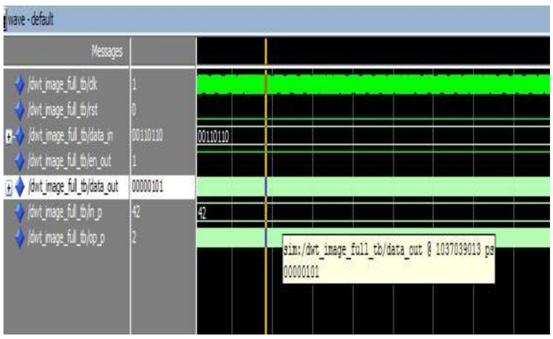


Fig.4.2.1.Simulation result of DWT of 1st image

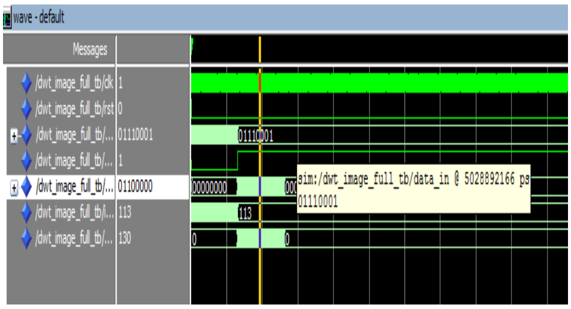


Fig.4.2.2.Simulation result of DWT of 2nd image

We have taken DWT of two images. Now the finally step of object tracking is done by comparing DWT transforms of the images.

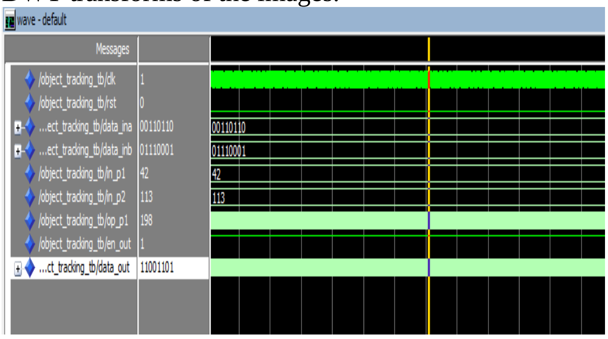


Fig.4.2.3.Simulation result of object tracking

V. Conclusions

The use look up table based architecture for 2-D DCT design completely eliminates the use of multipliers which require high power consumption. In addition to reduction to power consumption, it greatly reduces the routing delay. The main advantage of using Haar Wavelet transform for tracking dynamic obstacles is that it is highly memory efficient.

By doing these experiments we conclude that both techniques have its' own advantage and disadvantage. We can get quite reasonable compression ratio without loss of much important information. Though our experiments show that DWT technique is much efficient than DCT technique in quality and efficiency wise. But in performance time wise DCT is better than DCT. The DCT shows its best results in terms of energy compaction but MSE that is the error between original and recovered image is not acceptable. So to speed up the process and to improve the MSE, DWT based compression can be done.

The 2-D DCT architecture can be extended to the real-time video surveillance system applications, such as object classification and descriptive behaviors of objects. In the future, we will continue increasing the number of tracking object and extending the application fields of this method.

Furthermore regarding DWT-ME scheme, finding out the exact number of transformation level required in case of Haar wavelet transform for good tracking system can be studied as a future enhancement of the system.

A hybrid scheme combing the DWT and the DCT algorithms under high compression ratio constraint for image can be undertaken as a progression of the scheme to implement the tracking system.

REFERENCES

[1] Nathan Funk ,"*A Study of the Kalman filter applied to Visual Tracking*,", University of Alberta, Project for CMPUT 652,Probabilistic Methods in Artificial Intelligence, December 7, 2003

[2] Ut-Va Koc, Member, IEEE, and K. J. Ray Liu, Senior Member, IEEE, "*DCT-Based Motion Estimation," IEEE transactions on image processing, Vol. 7, No. 7, July 1998,pp.948-966.*

[3] Prabhakar Mishra, R. Apoorva, Bhavani B. Parvatikar, Lashmi Nair, *"Architectures for FPGA-Based Implementation of Motion Estimation of Dynamic Obstacles for Autonomous Robot Navigation," Proc. Third International Conference on Computational Intelligence, Communication Systems and Networks, (Cicsyn-2011), pp.292-297*

[4] Wen- Hsien Fang and Ming- Lu Wu, *"An Efficient Unified Systolic Architecture for the Computation of Discrete Trigonometric Transforms", Proc. IEEE International Symposium on Circuits and Systems, 9-12 June 1997, ISCAS '97, Vol 3, pp.2092-2095.*

[5] W. Chen, C.H. Smith and S. Fralick, *"A Fast Computational Algorithm for the Discrete Cosine Transform", IEEE Transactions on Communications, Vol. Com 25, Issue-9, Sep 1977, pp.1004- 1009.*

[6] K. J. Ray Liu, Ching- Te Chiu, *"Unified Parallel Lattice Structures for Time-Recursive Discrete Cosine/ Sine/ Hartley Transforms",IEEE Transactions on Signal Processing, Vol. 41, No. 3, March 1993.*