

Chapter 4 : A FRAMEWORK FOR DYNAMIC BACKGROUND MODELING AND SHADOW SUPPRESSION

Moving object segmentation is a challenging task due to the various real time practical problems, like small movements of non-static objects, tree branches and bushes blowing in the wind, waving trees, water surface, boat wakes, weather issues (such as bright sun, fog, heavy rain), and moving object in rain fall. To deal with these issues, the following two different frameworks for dynamic background modelling and shadow suppression for moving object segmentation in complex wavelet domain are proposed and further discussed in sections 4.3.1 and 4.3.2:

- Dynamic background modelling and shadow suppression using improved Gaussian mixture modeling and HSV (Hue, Saturation and Value) color model in case of normal and noisy video
- Dynamic background modelling and shadow suppression in case of normal video, noisy video but also in case of maritime video having highly dynamic background and poor contrast

A comparative analysis of the proposed methods is presented both qualitatively and quantitatively with other standard methods available in the literature for publicly available datasets of videos in different maritime scenarios, with varying light and weather conditions. Experimental results indicate that the proposed method is performing better in comparison to other standard methods for all the test cases.

4.1. Introduction

The objective of moving object segmentation is to decompose a video into background and the moving foreground objects. Moving object segmentation is a basic step for many computer vision applications and it is very useful in robotics, video surveillance, video

indexing, traffic monitoring and the many other applications [23-24]. In the recent years, various traditional and recent approaches for moving object segmentation have been proposed [135]. Sobral and Vacavant [97] presented the experimental analysis of different traditional and recent moving object segmentation methods along with their practical performance evaluation in terms of memory requirements. Several problems arise while segmenting the moving object because of gradual variations of the lighting conditions in the scene; noises; due to poor quality image source; sudden changes in the light conditions; waves on the water surface; weather issues (such as bright sun, fog, heavy rain) which contribute to generate a highly dynamic background; gradual and sudden illumination changes; camera jitter, and shadows. To address the above mentioned problems, in this chapter two new methods for dynamic background modeling and shadow suppression in Daubechies complex wavelet transform have been proposed. First method handles, the small movements of non-static objects such as tree branches and bushes blowing in the wind, waving trees, shadow regions that are projected by foreground objects and detected as moving objects. In the first proposed method, we have improved the Gaussian mixture model and use mode value to find the variance of K-Gaussian for dynamic background modeling. For shadow detection and removal, we have used saturation component from HSV model and Grey level model and ratio of standard deviation and mean in complex wavelet domain. Second method deals with highly dynamic background such as moving object in water surface, boat wakes, and weather issues (such as bright sun, fog, heavy rain), moving object in rain fall, and maritime object detection in night. In the second proposed method, we have used frame difference, background registration, background difference, and background difference mask for dynamic background modeling. For shadow detection and removal, we exploit the high frequency sub-band in the complex wavelet domain.

The main contributions are as follows:

(1) Two new methods for dynamic background modeling and shadow suppression have been proposed namely:

- a) Dynamic background modelling and shadow suppression using improved Gaussian mixture modeling and HSV (Hue, saturation and value) color model in case of normal and noisy video.
- b) Dynamic background modelling and shadow suppression in case of normal video, noisy video but also in case of maritime video having highly dynamic background and poor contrast.

(2) Comparative study of the proposed method with other state-of -the-art algorithms on a set of challenging video sequences.

(3) Analysis of the computational complexity and memory consumption of the proposed algorithm.

The rest of the chapter is organized as follows: Section 4.2 presents the related works. Methods and models are given in Section 4.3. Experimental results and analysis are given in Sections 4.3.1.1 & 4.3.2.1. Finally conclusion of this work is given in Section 4.4.

4.2. Related Works

Detection and segmentation of moving objects are challenging scenario due to the practical problem of the observed scene such as waves on the water surface, weather issues (such as bright sun, fog, heavy rain) which contribute to generate a highly dynamic background, gradual and sudden illumination changes, camera jitter, and shadows. To take into account these problems, in the recent years, many approaches have been proposed [19, 95] and these approaches can be classified in two categories [92] namely non-recursive and recursive.

A non-recursive technique uses a sliding-window approach for background estimation. It stores a buffer of the previous L video frames, and estimates the background image based on the temporal variation of each pixel within the buffer. Kim *et al.* [33] proposed moving object segmentation method based on codebook where a codebook method is formed to represent significant states in the background using quantization and clustering [33]. It solves some of the above mentioned problems, such as sudden changes in illumination, but does not consider the problems of ghost regions or shadow detection. A more refined application of this algorithm was proposed by Kushwaha *et al.* [26] which is based on construction of basic background model where in the variance and covariance of pixels are computed to construct the model for scene background which is adaptive to the dynamically changing background. But when we used [26] in maritime video sequence, the segmented object is distorted. Bloisi *et al.* [136] proposed a method for maritime video surveillance which aims to imitate a highly dynamic background such as a water background, where the light and weather conditions change drastically. In this paper, the authors achieved the modeling of a highly dynamic background by creating a ‘discretization’ of an unknown distribution. The major disadvantage of this method is that in case of sun reflection condition in water surface the segmented object is distorted and suffers the problem of shadow regions and the presence of ghosts objects. Non-recursive techniques need to buffer the frames and its computational complexity is high due to their non-recursive nature.

The major feature of recursive techniques is that they do not maintain a buffer for background estimation. They recursively update either a single or multiple background model(s) based on each input frame. McFarlane and Schofield [27] have proposed an approximation median filter method for segmentation of multiple video objects. This technique has also been used in background modeling for urban traffic monitoring [104].

This method needs many frames to learn the new background region revealed by an object that moves away after being stationary for a long time [95] but it is computationally cost effective. Stauffer *et al.* [29] have proposed a tracking method wherein motion segmentation was done using a mixture of Gaussians and on-line approximation to update the model. This model was considered promising since it showed good foreground object segmentation results for many outdoor sequences. However, weaker results were reported [29] for video sequences containing non-periodical background changes. This is the case for most of the marine sequences, which exhibit frequent background changes due to waves and water surface illumination, cloud shadows, and similar phenomena. To deal with the issue mentioned in [29], Sanderson [137] proposed a method suitable for object detection in maritime scenes based on anisotropic diffusion. This approach performs well for horizontal and vertical edges, but fails for object detection in other directions and also the time complexity of the method is high. Zivkovic and Heijden [32] proposed a moving object segmentation technique which is combination of temporal and spatial features. This approach automatically adapts the number of Gaussians being used to model a given pixel. However, when we applied the algorithm to marine sequences, the object boundaries were not particularly accurate, and the segmented frames contained too many noise-related and scattered pixels. Ivanov *et al.* [106] have proposed an improvement over the background subtraction method, which is faster than that proposed by [29] and is invariant to runtime change illuminations. But this method is adaptive to only the small and gradual changes in the background and in case of sudden changes it distorts.

To address the above mentioned problems some wavelet domain analysis techniques have been developed recently using change detection for video surveillance [138-141, 121]. Gao *et al.* [138] proposed an algorithm for moving object segmentation based on wavelet. In this paper, the authors proposed to create a model which keeps a sample of intensity

values for each pixel in the image and uses this sample to estimate the Marr wavelet probability density function of the pixel intensity. But it suffers from the problem of distortion of moving segmented objects due to change in their speed. Jalal and Singh [139] refined the work of Gao *et al.* [138] using background modeling in the complex wavelet domain. In this method, a temporal median filter is used to generate an initial background model in a training stage and then foreground pixels are obtained by applying background subtraction scheme in the subsequent frames. The limitation of this method is that, it suffers from the problem of shadows and ghost objects. Mendizabal and Salgado [140] proposed a moving object segmentation method based on region-level in a wavelet multi-resolution framework. In this approach, characterization is made for each region independently as a mixture of K -Gaussian modes and approximation coefficients to model the regions which allow handling efficiently illumination changes. This method is adaptive to only the small and gradual changes in the background and in case of sudden changes segmented object distorts. Khare *et al.* [121] refined the work of Mendizabal and Salgado [140] using single change detection in the complex wavelet domain. In this approach, the author use change detection to obtain frame difference in the complex wavelet domain. In this paper, author suppresses the shadow problem, but in case of sudden light changes segmented object distorts. To concern these issues, Hsia *et al.* [120] proposed a Modified Directional Lifting-based 9/7 Discrete Wavelet Transform (MDLDWT) based approach, which is based on the coefficient of Lifting-based 9/7 Discrete Wavelet Transform (LDWT). Its advantages are that they have a low critical path, fast computational speed and the LL3-band of the MDLDWT is employed solely to reduce the image transform computing cost and remove noise but it cannot handle large dynamic background changes. Here, LL3 band of the video frame belongs to approximation coefficients matrix obtained after MDLDWT up to the third level of

decomposition. To deal with the issues mentioned in [138-140], Khare *et al.* [141] proposed a moving object segmentation technique which is based on double change detection in the complex wavelet domain. The method proposed by Khare *et al.* [141] reduces the noise disturbance and speed change somehow. However, when we applied the algorithm [141] to marine sequences, the object boundaries were not particularly accurate, and the segmented frames contained too many noise and scattered pixels.

All the above discussed methods suffer from the problem of either slow speed of moving object or abrupt lighting variation changes leading to shadow and ghost like appearances. To solve the problem of shadow detection many approaches are proposed in wavelet domain in the literatures [88, 142-143]. Shadows are mainly of two types: self-shadow; and cast shadow [88]. Guan [142] proposed a shadow detection and removal algorithm, by using HSV color model in multi-scale wavelet domain and the real valued wavelet transform, which suffers from the problem of shift sensitivity and poor edge detection. Khare *et al.*[143] proposed shadow detection and removal method, which is based on complex wavelet transform with standard deviation of wavelet coefficients as a threshold but in dynamic background it does not suppress the shadow problem more accurately.

There are many general approaches to background modeling in dynamic environment but only few of them have been tested on water background. A water background is more difficult than other kinds of dynamic background since waves on the water surface do not belong to the foreground even though they involve motion. Also, sun reflections do not have the same behavior of a reflective surface. Therefore, after analyzing these approaches, we are able to conclude that there is a slim chance of finding an already available solution, which is complete, effective and real time simultaneously. Motivated by these facts, in this chapter, two new methods for dynamic background modeling and

shadow suppression in the complex wavelet domain are proposed. First method handles the small movements of non-static objects such as tree branches and bushes blowing in the wind, waving trees, shadow regions that are projected by foreground objects and are detected as moving objects. Second method deals with highly dynamic background such as moving object in water surface, boat wakes, and weather issues (such as bright sun, fog, heavy rain), moving object in rain fall, and maritime object detection in night.

4.3. Method and Models

This section presents two different frameworks for dynamic background modelling and shadow suppression under rapidly changing illumination conditions for moving object segmentation in complex wavelet domain are proposed and further discussed in sections 4.3.1 and 4.3.2 respectively. First, we proposed a method based on improved Gaussian mixture modeling and HSV (Hue, saturation and value) color model in case of normal and noisy video and then extended the method based on frame difference, background registration, background difference, and background difference mask for maritime video having highly dynamic background and poor contrast. The two proposed methods are as follows

- Dynamic background modelling and shadow suppression using improved Gaussian mixture modeling and HSV (Hue, saturation and value) color model in case of normal and noisy video.
- Dynamic background modelling and shadow suppression in case of normal video, noisy video but also in case of maritime video having highly dynamic background and poor contrast.

4.3.1. Dynamic Background Modelling and Shadow Suppression using Improved Gaussian Mixture Modeling and HSV (Hue, saturation and value) Color Model

The proposed dynamic background modelling and shadow suppression technique will have many advantages like elimination of shadows and ghosts, reduction of noise, high execution speed, adaptability to the dynamic background changes and lighting conditions. In the proposed approach, the above mentioned issues as indicated in [120, 121, 138-143] are addressed using dynamic background modeling step in complex wavelet domain. In proposed approach, eight major steps are applied on the given video frames which include: wavelet de-composition of frame using complex wavelet transform; use of change detection on detail coefficients (LH, HL, HH); use of improved Gaussian mixture based dynamic background modeling on approximate co-efficient (LL sub-band); cast shadow suppression; use of soft thresholding for noise removal; strong edge detection; inverse wavelet transformation for reconstruction; and finally using closing morphology operator. For dynamic background modeling, we have improved the Gaussian mixture model and used mode value to find the variance of K-Gaussian. For shadow detection and removal, we have computed difference between grey level components to detect moving object with shadow and adopt logical AND of wavelet coefficients of saturation component with grey level components to remove shadow from detected moving object with shadow. All these steps are iteratively applied until the result does not surpass the set threshold value for object segmentation. Applying the change detection in Daubechies complex wavelet domain have following advantages (a) it is shift invariant and have a better directional selectivity as compared to real valued wavelet transforms [4] (b) it has perfect reconstruction property (c) it provides true phase information [4], while other complex wavelet transform does not provide true phase

information [4] (d) Daubechies complex wavelet transform has no redundancy [4]. The working of the proposed framework is given as follows and illustrated in Fig. 4.1 & 4.2:

Step 1: Wavelet Decomposition of frames

In the proposed approach, a 2-D Daubechies complex wavelet transform is applied on current frame and previous frame to get wavelet coefficients in four sub-bands: LL, LH, HL and HH. The generating Daubechies complex wavelet transform is described as follows:

The basic equation of Multiresolution theory is the scaling equation [4]

$$\phi(u) = 2 \sum_i a_i \phi(2u - i) \quad (4.1)$$

where a_i 's are coefficients, and $\phi(u)$ is the scaling function. The a_i 's can be real as well as complex valued. Daubechies's wavelet bases $\{\psi_{j,k}(t)\}$ in one-dimension is defined using the above mentioned scaling function $\phi(u)$ and multi-resolution analysis of $L_2(\mathfrak{R})$ [4]. The generating wavelet $\psi(t)$ is defined as:

$$\psi(t) = 2 \sum (-1)^n \overline{a_{1-n}} \phi(2t - n) \quad (4.2)$$

where $\psi(t)$ share same compact support $[-L, L+1]$.

Any function $f(t)$ can be decomposed into complex scaling function and mother wavelet as:

$$f(t) = \sum_k C_k^{j_0} \phi_{j_0,k}(t) + \sum_{j=j_0}^{j_{\max}-1} d_k^j \psi_{j,k}(t) \quad (4.3)$$

where, j_0 is a given low resolution level, $\{C_k^{j_0}\}$ is called approximation coefficient and $\{d_k^j\}$ is known as detail coefficient.

Step 2: Application of change detection method on wavelet co-efficients

In step 2, a simple change detection based method is applied on detail wavelet coefficients i.e. on sub-bands: LH, HL, and HH. Let $Wf_{n,d}(i, j)$ ($d = \{LH, HL, HH\}$) and

$Wf_{n-1,d}(i, j)$ ($d = \{LH, HL, HH\}$) are the wavelet coefficients at location (i, j) of the current and previous frames respectively. Instead of a prior assignment of a fixed threshold $V_{th,d}$ to each frame difference, this method uses the fast Euler number computation technique [126] to automatically determine $V_{th,d}$ from the video frame. The stable Euler number technique is one of the most effective algorithms for determining thresholds for change differences. However, its high computational complexity has always precluded its employment in real-time applications. A fast Euler number computation method was proposed [126] to overcome the high computational complexity of the stable Euler number method.

The fast Euler numbers algorithm calculates the Euler number for every possible threshold with a single raster of the frame difference image using following equation:

$$E(i) = \frac{1}{4} [(q_1(i) - q_3(i) - 2q_d(i))] \quad (4.4)$$

where q_1 , q_3 , and q_d is the quads (quad is a 2*2 masks of bit cells) contained in the given image.

The output of the algorithm is an array of Euler numbers: one of each threshold value.

The Zero crossings find out the optimal threshold. Detailed algorithms for the fast Euler number computation method can be found in [126].

The wavelet domain frame difference $WD_{n,d}(i, j)$ for respective sub-bands for every pixel location (i, j) which belongs to the co-ordinate of frame is computed as follows:

$$WD_{n,d}(i, j) = \begin{cases} 1 & \text{if } \left| Wf_{n,d}(i, j) - Wf_{n-1,d}(i, j) \right| > V_{th,d} \\ 0 & \text{otherwise} \end{cases} \quad (4.5)$$

Step 3: Background Model Creation using LL sub-band

In the dynamic background modeling, the most prevalent background is the continuous variations of lighting condition. To model such phenomenon, Gaussian Mixture Models (GMM) is useful due to their inherent properties [29] and this model may perform better if parameters associated with it are computed carefully. Therefore, in this method an improved GMM based model is used to model the pixel distributions of background. To improve the performance and robustness of the original GMM model here we propose to use the statistical mode value in place of mean value to find the probability distribution function (PDF) of K - Gaussian distribution.

This small change of way of computing mean value is more robust towards noise, gradual and sudden illumination change in complex wavelet domain using LL sub-band. This PDF is used to update the background pixel by considering the likelihood of observing a particular intensity for a pixel. If the pixels are normally distributed, then only the mode and variance are needed to maintain the background model. When the PDF is multimodal, then mode, variance and weights of K -Gaussian distribution are needed to maintain the background model.

The working of improved mixture of Gaussian is given as follows:

In improved Gaussian mixture model each background pixel is modeled independently by PDF of K - Gaussian distribution where each Gaussian represents the intensity distribution of one the different environment e.g. gradual and sudden illumination changes, camera jitter, shadows, and observed by the pixels to construct a dynamic background. Let the k^{th} Gaussian mixture of PDF is represented by its mean μ_k , median Δ_k , variance ξ_k^2 , standard deviation ξ_k and its weight in the mixture be denoted by θ_k .

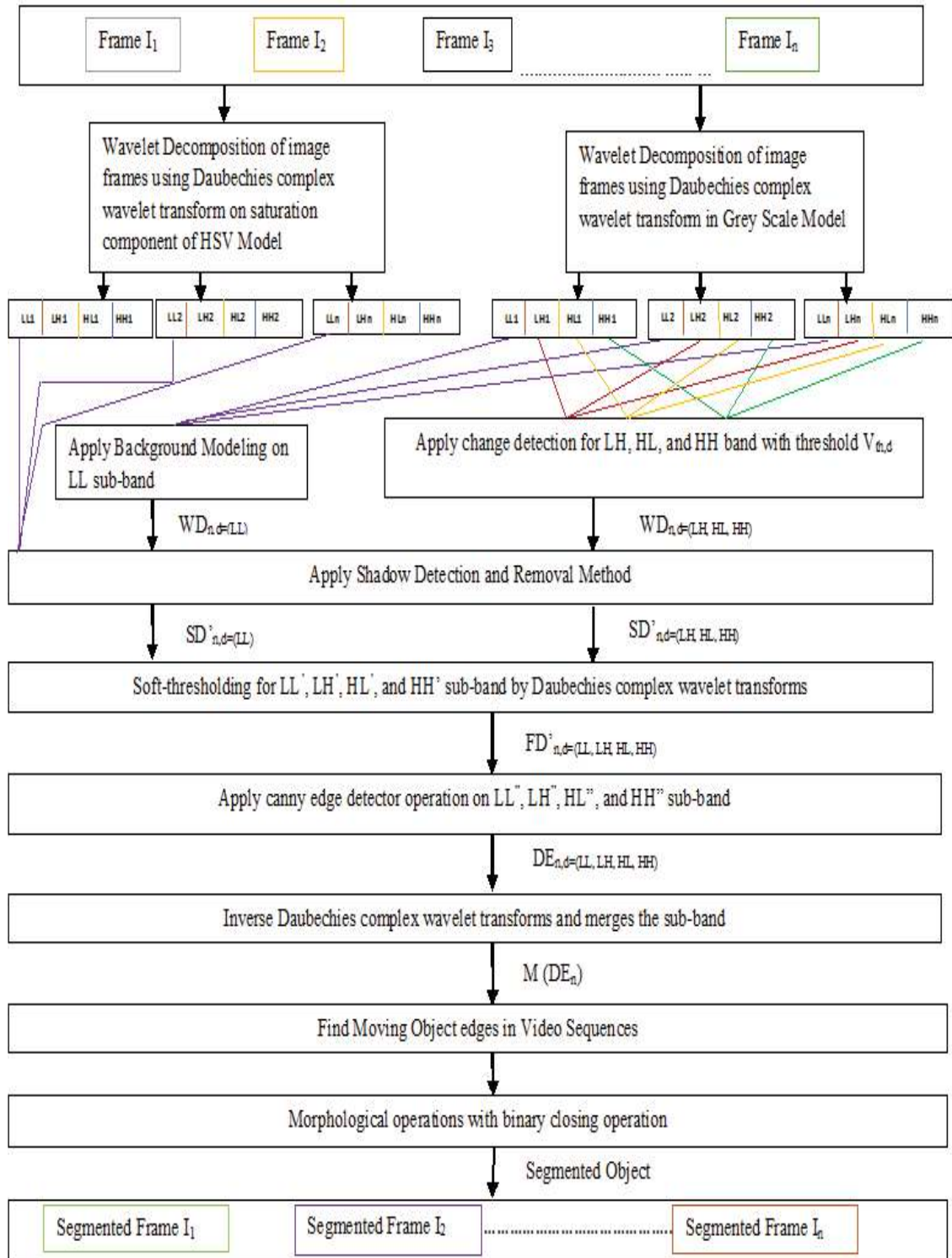


Figure 4.1: Block Diagram of the Proposed Method

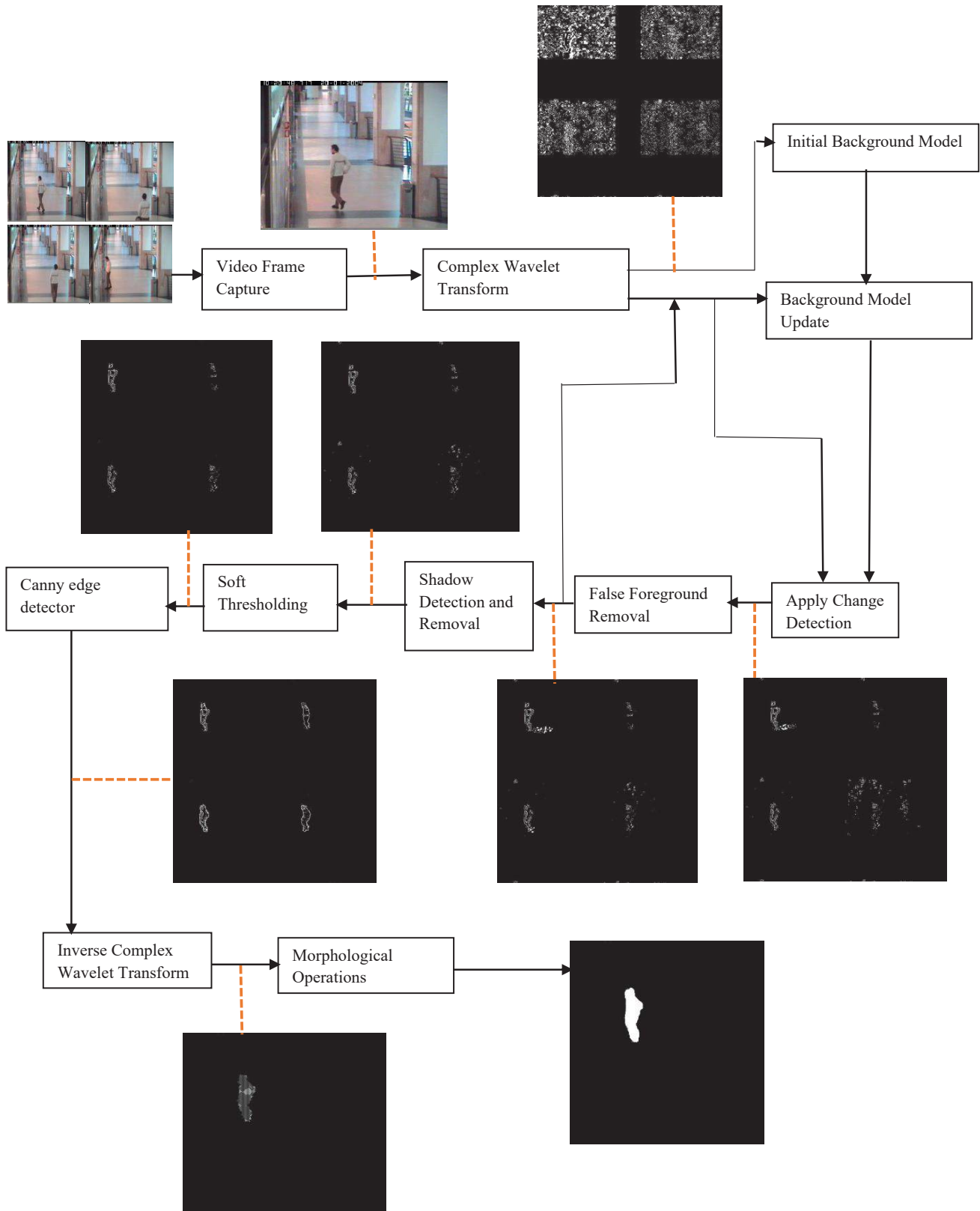


Figure 4.2: Sub-block Diagram of the proposed approach

In the first step of dynamic background modeling, we denote an entire image sequence up to the n^{th} frame by $\Omega_n = (Wf_{n,LL1}, Wf_{n,LL2}, \dots, Wf_{n,LLn})$, where $f_{k,LL_k} \in G^{\{0,1,2,\dots,p-1\} * \{0,1,2,\dots,q-1\}}$, and $G \subseteq \mathbb{N}$; with p and q representing the number of rows and columns of f_{k,LL_k} respectively. Then, we find frame difference mask $WD_{n,LL3}(i, j)$ of the LL image which is obtained by taking the difference between coefficients in two LL sub-bands as follows:

$$|WD_{n,LL3}(i, j)| = |Wf_{n,LL2}(i, j) - Wf_{n-1,LL1}(i, j)| \quad (4.6)$$

In the second step of dynamic background modeling, the system starts with an empty set of models and for every new observation $|WD_{n,LL3}(i, j)|$, we check the condition such that $|WD_{n,LL3}(i, j)| < 2\xi(i, j)$, where ξ is the standard deviation of K-Gaussian distribution. If such a condition is true then its associated parameters of PDF are updated as follows:

$$\beta(i, j) = \frac{\alpha}{\Theta(i, j)} \quad (4.7)$$

where α is the learning rate. Here, 10 different sequences are considered over 800 frames for each of the sequences is recorded. This resulted in 50 samples of size 800 each. The test statistic was calculated for each of the samples and the value of α is set to 0.01 and $\Theta(i, j)$ is the weight of K-Gaussian distributions. The mean of K-Gaussians is computed as follows:

$$\Phi(i, j) = (1 - \beta) * \Phi(i, j) + \beta * Wf_{n,LL2}(i, j) \quad (4.8)$$

where $\Phi(i, j)$ is the mean of K-Gaussian distribution and $Wf_{n,LL2}(i, j)$ is the sub-band of complex wavelet transform. After finding out the mean of K^{th} distribution, the mode value is calculated instead of mean using following equation [144]:

$$\delta(i, j) = 3 * \Delta(Wf_{n,LL2}(i, j)) - 2 * \Phi(i, j) \quad (4.9)$$

where $\delta(i, j)$ is mode of K-Gaussian distribution and Δ is the median value.

In the next step, the variance of K-Gaussian distribution using mode value is calculated as follows:

$$\xi^2(i, j) = (1 - \beta) * \xi^2(i, j) - \beta(i, j) * (Wf_{n,LL2}(i, j) - \delta(i, j))^2 \quad (4.10)$$

where $\xi^2(i, j)$ is the variance of K-Gaussian distribution and $Wf_{n,LL2}(i, j)$ is the sub-band of complex wavelet.

The weight of the Gaussians is computed as follows:

$$\Theta(i, j) = (1 - \alpha) * \Theta(i, j) + \alpha \quad (4.11)$$

If given condition is not true i.e. $|WD_{n,LL3}(i, j)| < 2 \xi(i, j)$ then the weights of the remaining Gaussians of PDF are updated as

$$\Theta(i, j) = (1 - \alpha) * \Theta(i, j) \quad (4.12)$$

In the last step of dynamic background modeling, the background frames are updated using weight of the K- Gaussian distributed (which is obtained by Eq. 4.12) and mode value (which is obtained by Eq. 4.9). The updated pixels in the background frame is obtained as follows.

$$Wf_{n,LL1}(i, j) = Wf_{n,LL1}(i, j) + \delta(i, j) * \Theta(i, j) \quad (4.13)$$

where $Wf_{n,LL1}(i, j)$ is the updated background frame.

Step 4: Shadow detection and removal

For detection & removal of shadow, in the proposed work we have used HSV color model for detection of pixels which have been changed due to moving objects and moving shadow. Since Hue and value component of HSV changes quite dramatically depending on the strength of shadow, therefore we have used only saturation component of each

frame for processing (See Fig. 4.1). In the proposed approach, we have computed the difference between grey level components of LL sub-band frame to detect moving object with shadow and adopt logical AND of wavelet coefficients of saturation component with grey level components to remove shadow from detected moving object with shadow. Let

$|WD_{n,LL3}(i, j)|_{\Delta E}$ is difference between grey level components ΔE of LL sub-band frame and

$|WD_{n,LL3}(i, j)|_{\Delta H}$ is difference between saturation components ΔH of LL sub-band frame.

$|WD_{n,LL3}(i, j)|_{\Delta E}$ and $|WD_{n,LL3}(i, j)|_{\Delta H}$ are calculated as follows:

$$|WD_{n,LL3}(i, j)|_{\Delta E} = |Wf_{n,LL2}(i, j) - Wf_{n-1,LL1}(i, j)|_{\Delta E} \quad (4.14)$$

$$|WD_{n,LL3}(i, j)|_{\Delta H} = |Wf_{n,LL2}(i, j) - Wf_{n-1,LL1}(i, j)|_{\Delta H} \quad (4.15)$$

The proposed condition for detection of foreground object with shadow at each pixel (i, j) is given as follows:

$$N = \left\{ \begin{array}{ll} 1 & \text{if } |WD_{n,LL3}(i, j)|_{\Delta E} \geq \left(\frac{\xi}{\Delta} \right) |WD_{n,LL3}(i, j)|_{\Delta E} \\ 0 & \text{otherwise} \end{array} \right\} \quad (4.16)$$

where Δ is the mean of absolute wavelet coefficients which represents average luminance of the object in wavelet domain and ξ is the standard deviation of the luminance of the object plus its surround in wavelet domain and $\left(\frac{\xi}{\Delta} \right) |WD_{n,LL3}(i, j)|_{\Delta E}$ is the relative deviation and is defined as ratio of standard deviation (ξ) to the mean (Δ) in wavelet domain.

The proposed condition for detection of foreground object without shadow at each pixel (i, j) is given as follows:

$$N = \left\{ \begin{array}{l} 1 \quad \text{if } \left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta E} \geq \left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta H} \wedge \left| WD_{n,LL3(i,j)} \right|_{\Delta H} \geq \left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta E} \\ 0 \quad \text{otherwise} \end{array} \right\} \quad (4.17)$$

where $\left| WD_{n,LL3(i,j)} \right|_{\Delta E}$ is difference between grey level components of LL sub-band frame

(which is given in Eq. 4.14), $\left| WD_{n,LL3(i,j)} \right|_{\Delta H}$ is difference between saturation

components of LL sub-band frame (which is given in Eq. 4.15) and $\left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta E}$,

$\left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta H}$ is the relative standard deviation of $\left| WD_{n,LL3(i,j)} \right|_{\Delta E}$ and $\left| WD_{n,LL3(i,j)} \right|_{\Delta H}$

respectively.

The proposed algorithm for shadow detection and removal are as follows:

1. Perform complex wavelet de-composition transform on saturation component of HSV model and Grey level model of reference frame and current frame.
2. Take absolute difference of LL wavelet co-efficient of both saturation component from HSV model and Grey level model (which is taken from in Eq. 4.14 & 4.15) in complex wavelet domain.
3. Compute relative standard deviation of $\left| WD_{n,LL3(i,j)} \right|_{\Delta E}$ and $\left| WD_{n,LL3(i,j)} \right|_{\Delta H}$ i.e.

$$\left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta E} \quad \text{and} \quad \left(\frac{\xi}{\Delta} \right) \left| WD_{n,LL3(i,j)} \right|_{\Delta H} .$$

4. Check condition for shadow detection (detection of foreground object with shadow) using Eq. (4.16).

5. Check condition for shadow removal (detection of foreground object without shadow) using Eq. (4.17).

The detected moving object after applying shadow removal condition has some noise regions because of irregular object motion and noise. Also, the boundary region may not be very smooth.

Step 5: Noise removal using wavelet based soft thresholding

After applying change detection based method and background modeling, the obtained result may have noise. This step deals with the noise reduction from the data obtained in step 4. In presence of noise, the equation is expressed as:

$$WD_{n,d=(LL,LH,HL,HH)}(i,j) = WD_{n,d=(LL,LH,HL,HH)}^*(i,j) + \eta \quad (4.18)$$

where $WD_{n,d=(LL,LH,HL,HH)}^*(i,j)$ is frame difference without noise, $WD_{n,d=(LL,LH,HL,HH)}(i,j)$ is the original frame difference with noise, and η is the additive noise. The wavelet domain soft thresholding T is applied on wavelet coefficients for noise reduction. The value of soft thresholding parameter T for de-noising is computed as [127]

$$T = \frac{1}{2^{j-1}} \left(\frac{\psi}{\xi} \right) \omega \quad (4.19)$$

where j is wavelet decomposition level and ψ , ξ and ω are standard deviation, absolute mean and absolute median of wavelet coefficients of a sub-band.

Step 6: Strong edges detection in wavelet domain

Canny edge detection method is one of the most useful and popular edge detection methods, because of its low error rate well localized edge points and single edge detection response [128]. Here, the canny edge detection operator is applied on $WD_{n,d=(LL,LH,HL,HH)}^*(i,j)$ to detect the edges of significant difference pixels in all sub-bands as follows:

$$DE_{n,d=(LL,LH,HL,HH)}(i,j) = \text{canny}(WD_{n,d=(LL,LH,HL,HH)}^*(i,j)) \quad (4.20)$$

where $DE_{n,d=(LL,LH,HL,HH)}(i,j)$ is an edge map of $WD_{n,d=(LL,LH,HL,HH)}^*(i,j)$

Step 7: Application of inverse Daubechies complex wavelet transform

After finding edge map $DE_{n,d=(LL,LH,HL,HH)}(i,j)$ in wavelet domain from step 6, inverse wavelet transform is applied to get moving object edges in spatial domain i.e. E_n .

Step 8: Application of closing morphological operation to sub-band

As a result of step 7, the obtained segmented object may include a number of disconnected edges due to non-ideal segmentation of moving object edges. Extractions of object using these disconnected edges may lead to inaccurate object segmentation. Therefore, some morphological operation is needed for post-processing of object edge map to generate connected edges. Here, a binary closing morphological operation is used [128] which gives $M(E_n)$ i.e. the set of connected edge. In this step, the final segmented output is obtained.

4.3.1.1. Experiments and Result

Dataset Description

For experimentation purpose, we have taken four datasets namely Pets Dataset [129], Visor datasets [130-132], Caviar Dataset [133], CVCR Dataset [134] as discussed in chapter 3 (in section 3.4.1).

Qualitative Analysis

The proposed method for segmentation of moving object has been applied on a number of video clips datasets as discussed in chapter 3 (in section 3.4.1) [129-134]. For the segmentation of the video object by various methods, the numbers of frames taken into consideration at a time include 125, 150, 175 and 200. Here, in this section, the

qualitative & quantitative analysis of the proposed method & other methods in consideration are presented.

(A) Experiment-1

Here in this experiment, we have taken one step video sequence from the Pets data sets [129]. From Fig. 4.3, one can conclude that the segmentation result obtained by the proposed method has better segments in full and partial occlusions between human beings in outdoor environments (see frame nos. 125-200). From the obtained result of other methods are suffering the problem of shadow but the proposed method is able to suppress the shadow problem (see frame 125 - 200). The proposed method handles all these critical condition and segments the result properly (see frame 125 - 200).

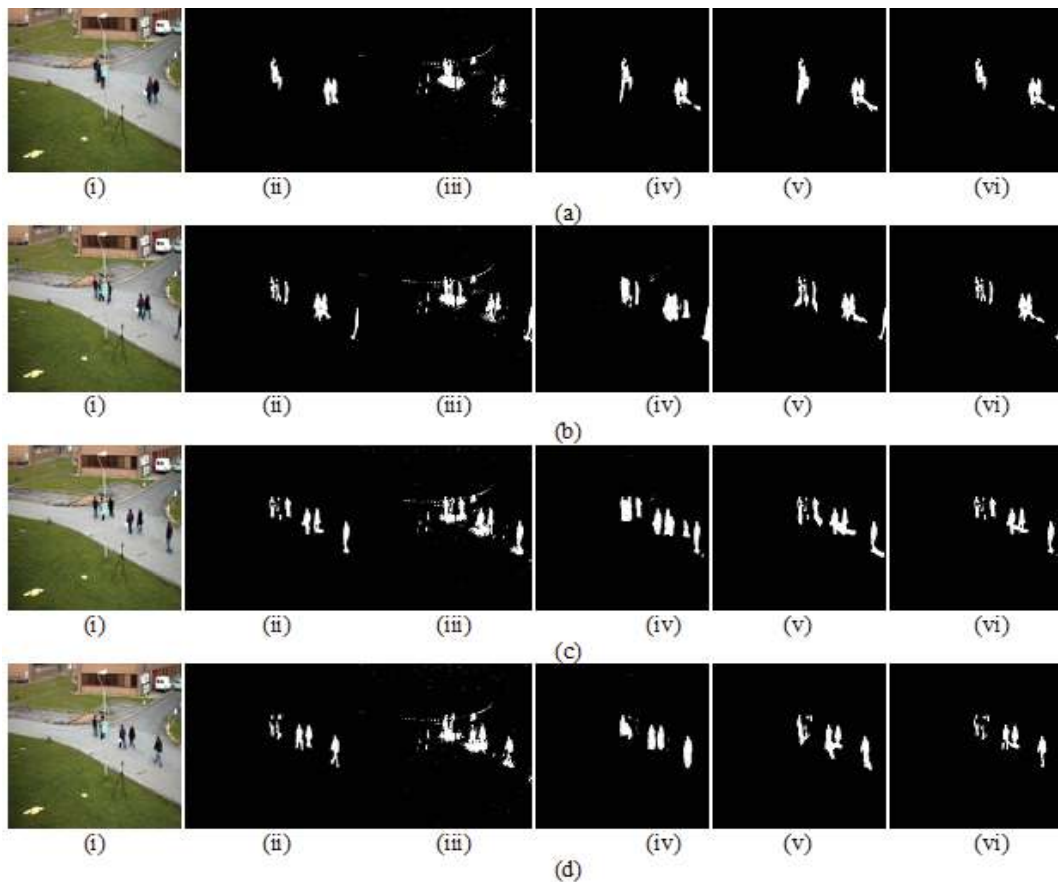


Figure 4.3: Segmentation results for People video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

(B) Experiment-2

Here in this experiment, we have taken Intelligent Room video sequence from the Visor dataset [130]. From Fig. 4.4, it is observed that better shape of moving object with least noise in segmented frame is obtained by using the proposed method as compared to segmented frame obtained by the other methods [33, 120, 121, 141]. From Fig. 4.4, it is also clear that method used in [33, 120, 121, 141] (see frame 125-200 (iv-vii)) results in comparable shape structure as compared to the proposed method but is poor in noise removal.

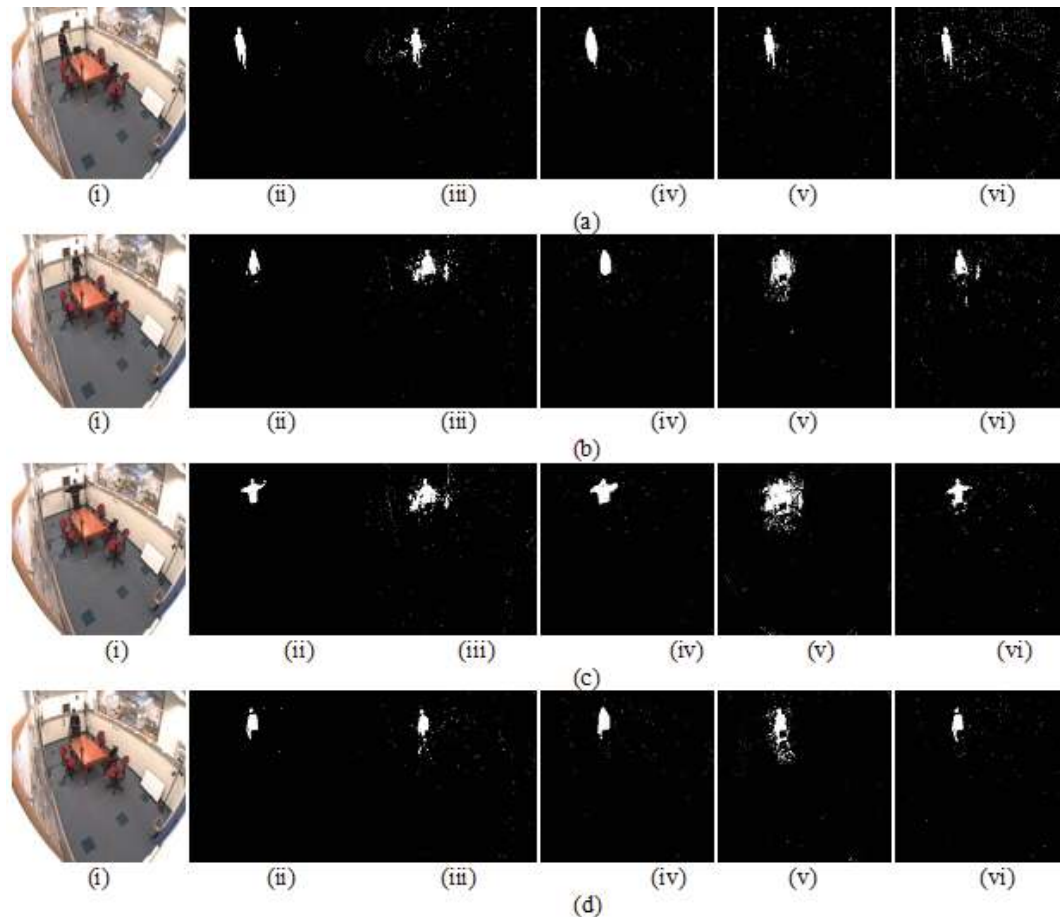


Figure 4.4: Segmentation results for Intelligent Room video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

(C) Experiment-3

Here in this experiment, we have taken One Step video sequence from the Caviar dataset [133]. In the Fig. 4.5, while comparing the segmentation results of the proposed method to other methods, one can easily observe the segmented frames obtained using proposed method are more accurate (see frame no. 125, 150, 175, and 200). The proposed technique handles the problem of ghosts and shadows unlike segmented frame obtained by the method used by [33, 120, 121, 141] (see frame 125-200 (iii-vi)).

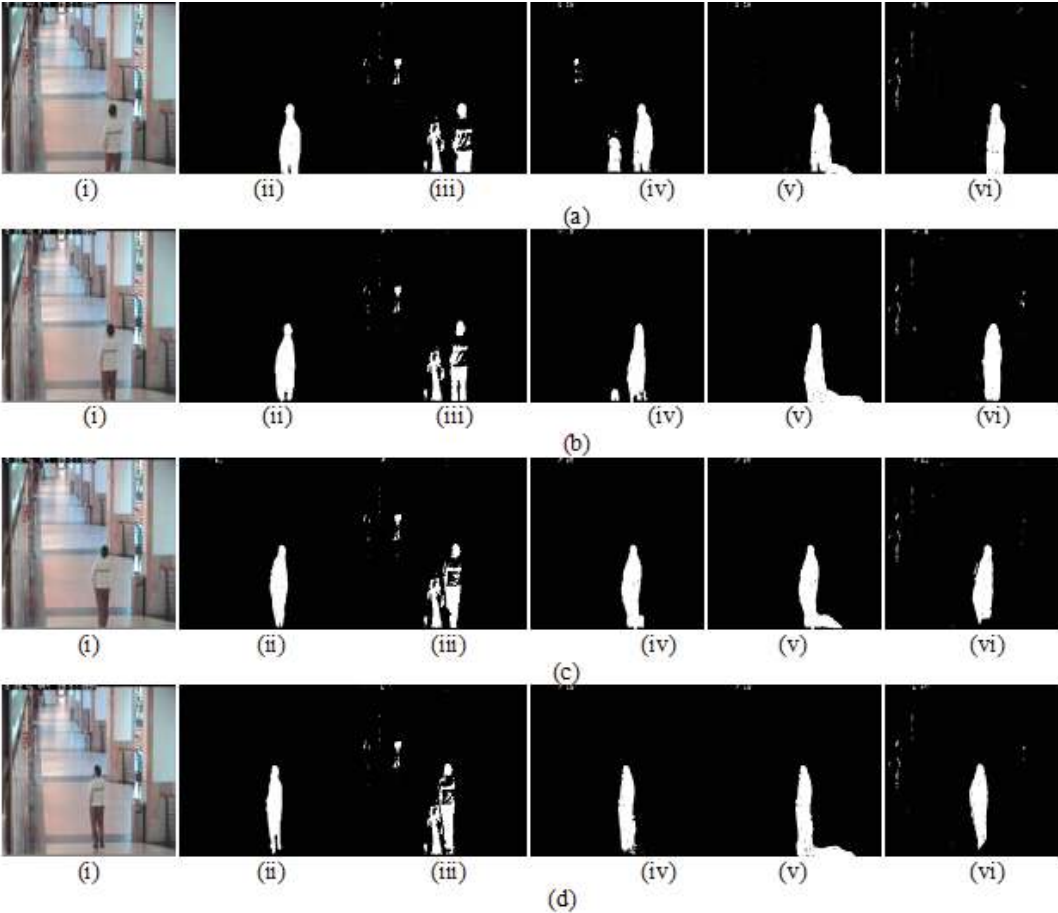


Figure 4.5: Segmentation results for One Step video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

(D) Experiment-4

Here in this experiment, we have taken Camera2_070605 video sequence from the visor dataset [131]. In the Fig. 4.6, it is clear that proposed method work properly in noisy environment and poor lighting condition. This video is performed at particular angle, it also seems that other methods [33, 120, 121, 141] have not properly segmented the object; some portion of objects distorts (see frame no. 125, 150, 175, 200 (iii-vi)) but the proposed method properly segment the whole object. From obtained result, it is also clear that most of the method suffer the problem of shadow region (see frame no. 125, 150, 175, 200 (iii, v, vi)) but proposed method suppress the shadow problem (see frame no. 125, 150, 175, 200 (ii)).

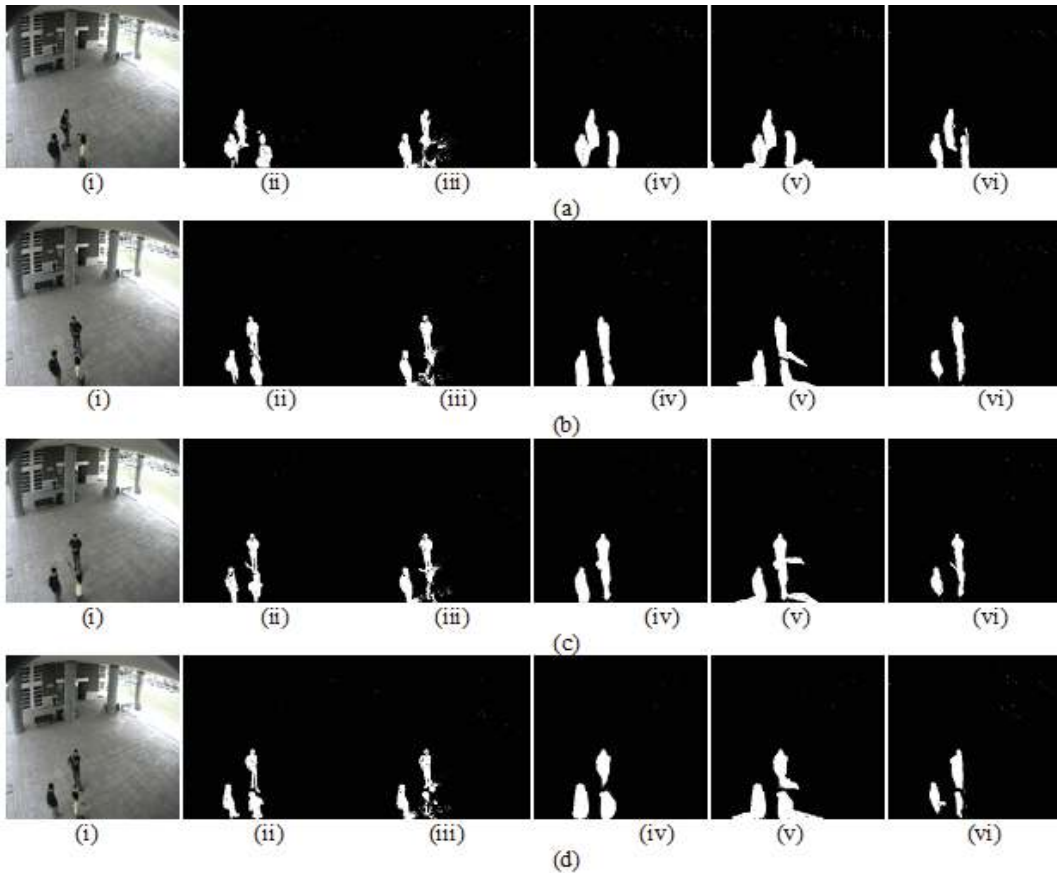


Figure 4.6: Segmentation results for Camera2_070605 video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

(E) Experiment-5

Here in this experiment, we have taken highwayI_raw video sequence from the visor dataset [132]. In these video sequence cars shadows are present (see frame 125-150). In the Fig. 4.7, it is clear that proposed method easily handle the problem of shadow (see frame 125-200 (i)) but other method [33, 120, 121, 141] suffer from the problem of shadows [see frame 125-200 (ii-viii)]. Due to fast moving speed of car the segmented result obtained by Khare *et al.* [121, 141] method is distorted (see frame 125-200 (vi)) but in this condition proposed method work properly.

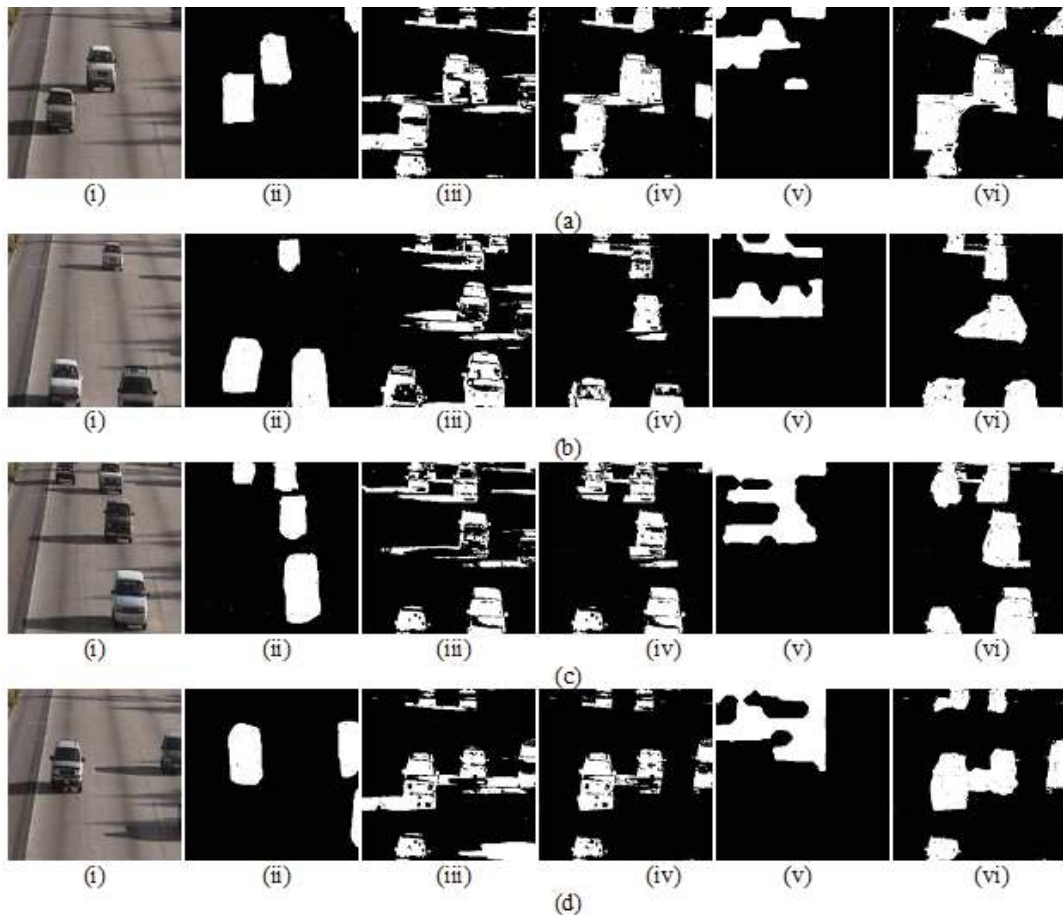


Figure 4.7: Segmentation results for highwayI_raw video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

(F) Experiment-6

Here in this experiment, we have taken 4917-5_70 video sequence from the CVCR dataset [134]. This is the most critical data set because the video was shot much more height and in highly crowded environment and also in this video full occlusions, shadow and noise are present. From the Fig. 4.8, it is clear that proposed method work properly in crowded environment (see frame 125-200 (ii)) but other methods [33, 120, 121, 141] suffer from the noise, shadow and ghost problem (see frame 125-200 (iii-viii))

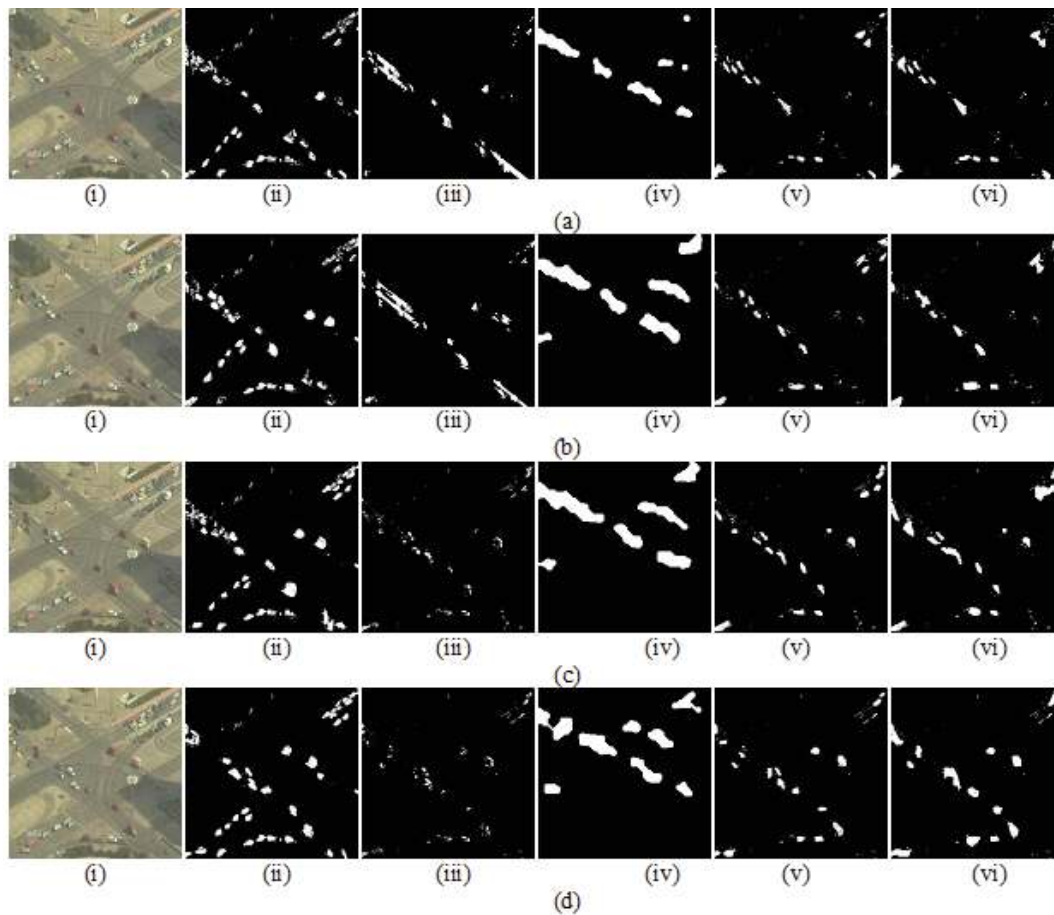


Figure 4.8: Segmentation results for 4917-5_70 video sequence corresponding to (a) Frame 125 (b) frame 150, (c) frame 175, (d) frame 200 (i) original frame, and the segmented frame obtained by various methods such as: (ii) the proposed method, (iii) Kim *et al.* [33], (iv) Hsia *et al.* [120] and (v) Khare *et al.* [121] (vi) Khare *et al.* [141].

Quantitative Analysis

It can be observed from the qualitative results presented in section 4.3.1.1 that none of the previously proposed segmentation algorithms give accurate segmentation result as compared to proposed approach. It is also very difficult to compare the segmentation results visually because human visual system can identify and understand scenes with different connected objects effortlessly. Therefore, quantitative performance metrics together with visual results are more appropriate. In this section of the chapter, the performances of the proposed method have been compared quantitatively with other state-of-the-art methods [33, 120, 121, 141]. For quantitative analysis, the various performance metrics considered include relative foreground area measure(RFAM) [84], relative position based measure (RPM) [84], normalized cross correlation (NCC) [85], peak signal-to-noise ratio (PSNR) [87] and misclassification penalty (MP) [84], normalized absolute error (NAE) [86], pixel classification based measure (PCM) [84], shadow detection rate (SDR) [88], shadow discrimination rate [88], execution time and memory consumption as discussed in chapter 2 (in section 2.5).

Performance Analysis

Tables 4.1-4.7 presents the values of RFAM, MP, RPM, NCC, NAE, PSNR, and PCM for the proposed method and other methods [33, 120, 121, 141] for six video sequences [129-134]. From Tables 4.1-4.7, one can conclude that the proposed method is associated to high value of RFAM, RPM, NCC, PSNR, PCM ; and low value of MP & NAE (see the result in table 4.1-4.7 in bold), in most of the frames in comparison to other method in consider [33, 120, 121, 141] for all dataset [129-134].

From Tables 4.8 and 4.9, it can be inferred that in cases of 4917-5_70 video sequence, HighwayI_raw video sequence, Camera2_070605 video sequence, One Step video sequence, and People video sequence the proposed method is found to be better than all

other methods [33, 120, 121, 141] in terms of shadow detection rate and shadow discrimination rate, and it can also be observed that the proposed method is found to be better than other methods [33, 120, 121, 141] in terms of shadow detection rate and shadow discrimination rate for six video sequence. Tables 4.8 and 4.9, we can say that the proposed method performs well for both indoor and outdoor types of video sequences [129-134].

In Table 4.10, average computation time (second/frame) and memory consumption for different methods for a video of frame size 480 x 272 with 100 frames [133] are shown. From the Table 4.10, it can be observed that the proposed method using complex wavelet transform is faster than [120, 121], takes approximately same time as comparison to the method proposed by Khare *et al.* [141] and more time as compared to the method proposed by Kim *et al.* [33]. Also from the Table 4.10, the proposed method consumes only 4.38 megabytes of RAM which is the least in comparison with the other methods discussed [33, 120, 121, 141]. Therefore, it can be concluded that the time required for the execution of the proposed method is average to other methods and consumes less amount of the system memory.

Therefore, after observing values of various quantitative measures, it can be concluded that the proposed method using Daubechies complex wavelet transform gives better results as compared to other methods.

Table 4.1: Values of Relative foreground area measure (RFAM)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.9456	0.4914	0.5870	0.7126	0.9459
150	0.9565	0.4665	0.6491	0.6005	0.9725
175	0.9643	0.5719	0.7457	0.6190	0.9949
200	0.9651	0.5084	0.6206	0.7508	0.9797
B-Camera2_070605 Video Sequence [131]					
125	0.9253	0.8631	0.9514	0.8757	0.9786
150	0.9435	0.9316	0.8087	0.8184	0.9469
175	0.9056	0.9365	0.7920	0.8500	0.9888
200	0.9645	0.8274	0.8949	0.8261	0.9995
C-One Step Video Sequence [133]					
125	0.9371	0.7535	0.8651	0.8293	0.9956
150	0.9567	0.7583	0.7607	0.8612	0.9814
175	0.9839	0.7866	0.8249	0.7893	0.9247
200	0.9752	0.7914	0.8199	0.8752	0.9685
D-Intelligent room Video Sequence [130]					
125	0.8540	0.5215	0.8961	0.6415	0.9517
150	0.9154	0.3084	0.7992	0.6320	0.8936
175	0.9048	0.4833	0.6747	0.5326	0.8956
200	0.8947	0.3492	0.7583	0.6739	0.9110
E-Car Video Sequence [132]					
125	0.8892	0.4708	0.7433	0.8302	0.9148
150	0.8358	0.4925	0.8287	0.8768	0.9307
175	0.8426	0.4674	0.7645	0.7423	0.8787
200	0.8672	0.4733	0.7213	0.7521	0.9027
F-Crowd Video Sequence [134]					
125	0.9239	0.6514	0.8396	0.6581	0.9186
150	0.9037	0.5445	0.8089	0.6064	0.9795
175	0.9152	0.5629	0.8762	0.6183	0.9483
200	0.9325	0.6103	0.8114	0.6835	0.9886

Table 4.2: Values of Misclassification Penalty (MP)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.0452	0.0069	0.0060	0.0011	1.5893e-004
150	0.0360	0.0040	0.0030	1.4639e-003	1.6412e-005
175	0.0394	0.0067	0.0020	8.4168e-003	4.0949e-005
200	0.0673	0.0189	0.0095	1.3720e-004	8.8613e-005
B-Camera2 070605 Video Sequence [131]					
125	0.0039	0.0035	2.0025e-004	4.3669e-004	1.0918e-004
150	0.0050	1.4146e-004	0.0032	0.0010	1.4768e-004
175	0.0086	3.6778e-004	0.0034	5.4249e-004	3.4264e-005
200	0.0054	0.0013	0.0066	2.2011e-004	3.0940e-005
C-One Step Video Sequence [133]					
125	0.0027	0.0502	0.0069	0.0275	1.3841e-004
150	0.03	0.0846	0.0034	0.0043	0
175	0.0071	0.0581	0.0094	0.0046	1.8262e-004
200	0.0400	0.0988	0.0078	0.0400	4.7875e-004
D-Intelligent room Video Sequence [130]					
125	0.0061	0.0064	0.0031	0.0035	0.0016
150	0.0028	0.0052	6.0313e-004	8.4038e-004	2.0028e-004
175	0.0068	0.0107	5.0712e-005	1.6023e-004	0
200	0.0021	0.0097	2.0408e-004	0.0024	1.9837e-004
E-Car Video Sequence [132]					
125	0.0094	0.0242	0.0975	0.0463	9.8308e-004
150	0.0086	0.0307	0.0136	0.0498	0.0025
175	0.0011	0.0217	0.0285	0.0444	6.3444e-004
200	0.0092	0.0894	0.0204	0.0424	0.0029
F-Crowd Video Sequence [134]					
125	0.0086	0.0021	0.0698	0.3256	0.0399
150	0.0595	0.0201	0.0970	0.3195	0.0083
175	0.0483	0.0085	0.0296	0.4456	0.0066
200	0.045	0.0120	0.0178	0.1864	0.0060

Table 4.3: Values of Relative position based measure (RPM)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.8903	0.9537	0.7840	0.9268	0.9561
150	0.8707	0.8440	0.8616	0.8602	0.9867
175	0.8746	0.8328	0.8053	0.8842	0.9824
200	0.8877	0.8775	0.7890	0.9789	0.9815
B-Camera2_070605 Video Sequence [131]					
125	0.7585	0.9525	0.9800	0.9125	0.9906
150	0.8475	0.9900	0.9490	0.9576	0.9816
175	0.7688	0.9775	0.9495	0.9107	0.9950
200	0.8817	0.9754	0.9359	0.9690	0.9951
C-One Step Video Sequence [133]					
125	0.9087	0.7854	0.9244	0.9002	0.9939
150	0.8992	0.7802	0.9411	0.9667	1
175	0.9767	0.8277	0.9269	0.9904	0.9919
200	0.9105	0.8004	0.9299	0.9705	0.9900
D-Intelligent room Video Sequence [130]					
125	0.9489	0.8608	0.9403	0.9515	0.9752
150	0.9817	0.8503	0.9459	0.9395	0.9737
175	0.9295	0.8864	0.9876	0.9735	1
200	0.9319	0.8521	0.9652	0.9245	0.9826
E-Car Video Sequence [132]					
125	0.8877	0.8773	0.8910	0.7437	0.9922
150	0.8817	0.8549	0.8804	0.6589	0.9960
175	0.9616	0.9521	0.9471	0.6896	0.9845
200	0.9686	0.8325	0.8670	0.7230	0.9544
F-Crowd Video Sequence [134]					
125	0.8707	0.9894	0.9149	0.7202	0.9401
150	0.8475	0.9659	0.9145	0.7447	0.9596
175	0.8968	0.9475	0.9451	0.7647	0.9534
200	0.9124	0.9484	0.9594	0.7474	0.9623

Table 4.4: Values of Normalized cross correlation (NCC)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.9206	0.7598	0.8739	0.7505	0.9804
150	0.9672	0.7371	0.7991	0.7785	0.9830
175	0.9803	0.7692	0.8980	0.7799	0.9551
200	0.9655	0.7890	0.7379	0.7121	0.9407
B-Camera2 070605 Video Sequence [131]					
125	0.9163	0.7980	0.7215	0.7799	0.9472
150	0.9069	0.7621	0.7411	0.7070	0.9242
175	0.9013	0.7034	0.7446	0.7030	0.9303
200	0.9224	0.7245	0.7053	0.7091	0.9174
C-One Step Video Sequence [133]					
125	0.9257	0.8162	0.8407	0.7755	0.9767
150	0.9273	0.8435	0.8928	0.7298	0.9723
175	0.9011	0.8762	0.8847	0.7594	1
200	0.9444	0.8029	0.8093	0.7726	0.9661
D-Intelligent room Video Sequence [130]					
125	0.8369	0.7477	0.7769	0.7154	0.8676
150	0.8178	0.7928	0.7672	0.7182	0.8621
175	0.8733	0.7033	0.7475	0.7529	0.8295
200	0.8455	0.7038	0.7368	0.7302	0.9176
E-Car Video Sequence [132]					
125	0.8206	0.7631	0.7543	0.6263	1
150	0.8672	0.7174	0.7374	0.6394	0.9998
175	0.8803	0.7590	0.7364	0.6830	0.9628
200	0.8655	0.7215	0.7583	0.6596	0.9544
F-Crowd Video Sequence [134]					
125	0.7314	0.6496	0.6128	0.5027	0.8760
150	0.7673	0.6029	0.6093	0.5217	0.8983
175	0.7412	0.6837	0.6944	0.5133	0.8982
200	0.7673	0.6865	0.6578	0.5126	0.9013

Table 4.5: Values of Peak Signal-to-Noise Ratio (PSNR)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	67.2287	62.2629	66.8536	69.6700	76.8092
150	59.6538	62.2629	66.0358	67.2586	77.7967
175	63.9703	62.2500	67.8870	67.0565	75.9444
200	67.5842	62.3044	64.4048	68.5922	72.4405
B-Camera2_070605 Video Sequence [131]					
125	67.1911	62.0224	64.5609	63.2390	73.8846
150	64.7191	66.7869	68.0527	67.8788	73.4895
175	63.7259	64.8574	68.3232	66.8596	71.8194
200	68.5521	64.4200	66.4702	65.9246	71.1882
C-One Step Video Sequence [133]					
125	65.1845	61.2879	68.6556	64.5716	75.9043
150	64.6888	61.0113	67.9695	68.3547	74.3613
175	63.3983	61.1819	66.5266	67.5512	73.8112
200	69.0326	61.7630	67.3078	68.4575	73.2359
D-Intelligent room Video Sequence [130]					
125	65.2835	65.1092	73.2923	72.7520	75.5231
150	64.4941	66.5547	75.3112	74.3599	77.7936
175	63.7211	67.9265	71.4456	71.7210	76.8141
200	70.5598	63.9785	73.3295	73.9095	80.5499
E-Car Video Sequence [132]					
125	66.5151	55.0504	54.8410	55.636	69.0886
150	64.947	54.9127	55.6995	54.1830	68.3215
175	63.9195	55.4390	56.2767	54.1909	65.7199
200	67.4324	54.5799	56.1901	54.5214	68.1365
F-Crowd Video Sequence [134]					
125	65.997	63.1445	62.6475	58.887	69.3576
150	64.9319	62.2401	62.344	57.1533	68.2658
175	63.4649	61.5444	61.4541	56.6778	68.5928
200	67.7056	61.7444	62.1407	57.0360	67.9130

Table 4.6: Values of Normalized absolute error (NAE)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.8734	0.6239	0.5522	0.4978	0.0962
150	0.7253	0.6559	0.5463	0.6141	0.0631
175	0.7138	0.6925	0.5441	0.6588	0.0851
200	0.7944	0.6732	0.4399	0.5109	0.2106
B-Camera2 070605 Video Sequence [131]					
125	0.7797	0.7751	0.3107	0.5836	0.0831
150	0.7697	0.7781	0.3572	0.3718	0.1022
175	0.7808	0.7510	0.3381	0.4736	0.1512
200	0.7818	0.7953	0.4960	0.5624	0.1674
C-One Step Video Sequence [133]					
125	0.5238	0.4856	0.2724	0.3975	0.0513
150	0.3963	0.6223	0.3268	0.2991	0.0750
175	0.4752	0.5092	0.4408	0.3482	0.0824
200	0.3655	0.5289	0.3997	0.3655	0.1021
D-Intelligent room Video Sequence [130]					
125	0.8011	0.3988	0.3645	0.3029	0.2181
150	0.8683	0.2158	0.3217	0.2914	0.1816
175	0.906	0.2197	0.5869	0.3750	0.1705
200	0.9129	0.3276	0.4444	0.2929	0.0843
E-Car Video Sequence [132]					
125	0.6567	0.3876	0.5055	0.6860	0.0942
150	0.6252	0.4452	0.4726	0.7462	0.0750
175	0.623	0.4875	0.3266	0.7827	0.1394
200	0.641	0.4619	0.6993	0.7953	0.1085
F-Crowd Video Sequence [134]					
125	0.3108	0.7742	0.8681	0.7635	0.1705
150	0.2088	0.8175	0.7981	0.7374	0.1827
175	0.2129	0.8542	0.8721	0.7194	0.1535
200	0.241	0.8321	0.7595	0.7604	0.1866

Table 4.7: Values of Pixel Classification Based Measure (PCM)

A-People Video Sequences [129]					
Frame No.	Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
125	0.9081	0.6731	0.8076	0.8454	0.8904
150	0.8852	0.6998	0.7610	0.7790	0.8491
175	0.8578	0.7186	0.7443	0.7598	0.8304
200	0.8595	0.6780	0.7640	0.8127	0.8304
B-Camera2_070605 Video Sequence [131]					
125	0.8148	0.8377	0.8125	0.8076	0.7960
150	0.7879	0.8284	0.8419	0.8180	0.8464
175	0.8066	0.8294	0.8462	0.8230	0.8435
200	0.8093	0.8375	0.8177	0.7960	0.8371
C-One Step Video Sequence [133]					
125	0.9578	0.6709	0.5762	0.6810	0.7711
150	0.9483	0.6803	0.5646	0.7645	0.7852
175	0.9262	0.6797	0.6817	0.7462	0.7814
200	0.9449	0.6583	0.5317	0.9449	0.7923
D-Intelligent room Video Sequence [130]					
125	0.9569	0.7716	0.9015	0.7810	0.9460
150	0.967	0.7271	0.9155	0.7645	0.9631
175	0.9561	0.8343	0.9041	0.7462	0.9573
200	0.9468	0.7441	0.9315	0.7583	0.9582
E-Car Video Sequence [132]					
125	0.7848	0.7707	0.7245	0.7490	0.8375
150	0.8218	0.7316	0.7859	0.7911	0.8822
175	0.8368	0.7371	0.7354	0.7363	0.8922
200	0.8251	0.7333	0.7747	0.7494	0.8644
F-Crowd Video Sequence [134]					
125	0.8595	0.8451	0.7755	0.9106	0.9608
150	0.8093	0.8392	0.7718	0.8490	0.9247
175	0.9468	0.8332	0.7546	0.8322	0.9139
200	0.8251	0.8372	0.7490	0.8470	0.9199

Table 4.8: Shadow detection rate $\%$ for representative video sequences

A-People Video Sequences [129]				
Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
84.71	83.25	82.67	85.46	95.35
B-Camera2 070605 Video Sequence [131]				
84.65	78.83	86.65	86.52	93.14
C-One Step Video Sequence [133]				
85.91	80.18	88.62	86.87	98.27
D-Intelligent room Video Sequence [130]				
85.68	80.82	86.59	87.73	96.48
E-Car Video Sequence [132]				
86.47	81.39	88.54	87.72	97.43
F-Crowd Video Sequence [134]				
72.12	70.56	74.83	74.57	91.27

Table 4.9: Shadow discrimination rate ζ for representative video sequences

A-People Video Sequences [129]				
Proposed by Khare <i>et al.</i> [141]	Proposed by Kim <i>et al.</i> [33]	Proposed by Hsia <i>et al.</i> [120]	Proposed by Khare <i>et al.</i> [121]	Proposed Method
86.34	78.81	86.56	87.16	97.15
B-Camera2 070605 Video Sequence [131]				
84.29	76.97	85.73	85.69	96.37
C-One Step Video Sequence [133]				
83.93	77.28	87.24	86.21	97.82
D-Intelligent room Video Sequence [130]				
82.67	79.59	86.55	84.17	96.63
E-Car Video Sequence [132]				
85.48	77.28	87.47	87.27	98.37
F-Crowd Video Sequence [134]				
74.12	68.86	76.32	75.29	92.54

Table 4.10: Computational Time and Consumption Memory for One step video sequence [133]

S.no.	Methods	Computational Time (in second/frame)	Memory Consumption (MB)
1	Proposed by Khare <i>et al.</i> [121]	1.682	7.65
2	Method Proposed by Kim <i>et al.</i> [33]	0.722	22.92
3	Method Proposed by Hsia <i>et al.</i> [120]	1.912	8.64
4	Method Proposed by Khare <i>et al.</i> [141]	1.753	7.08
5	The Proposed Method	1.486	4.38

4.3.2. Dynamic Background Modelling and Shadow Suppression in Case of Dynamic Water Background

In this section, a fast and robust moving object segmentation method on a water surface for maritime surveillance using dynamic background modeling and shadow suppression (DBMSS) is proposed in the complex wavelet domain which is capable of addressing most of the real time practical problem. These problems include the waves on the water surface, boat wakes, and weather issues (such as bright sun, fog, heavy rain) which contribute to generate a highly dynamic background, gradual and sudden illumination changes, noise, camera jitter, shadows, and sun reflections that can provoke false detections. In the proposed approach (DBMSS), the above mentioned issues are addressed using dynamic background modeling and shadow detection step in the complex wavelet domain. After applying complex wavelet transform on a video frame, it is decomposed into four parts. The first part is known as approximation coefficients matrix LL (Low-Low) and other three parts are known as details coefficients matrices LH (Low-High), HL (High-Low), and HH (High-High) i.e. horizontal, vertical, and diagonal, respectively.

Seven major steps are applied on the given video frames which include: wavelet decomposition of frame using complex wavelet transform; use of change detection on detail coefficients (LH, HL, HH); use of dynamic background modeling on approximate co-efficient (LL sub-band); use of soft thresholding for noise removal; cast shadow suppression; inverse wavelet transformation for reconstruction; and finally using closing morphology operator. For dynamic background modeling, we have used frame difference, background registration, background difference, and background difference mask in the complex wavelet domain. For shadow detection and suppression problem in water surface, we exploit the high frequency sub-band in the complex wavelet domain. All these steps are iteratively applied until the result does not surpass the set

threshold value for object segmentation. Applying the change detection in Daubechies complex wavelet domain has following advantages (a) it is shift invariant and have a better directional selectivity as compared to real valued wavelet transforms [4]; (b) it has perfect reconstruction property [4]; (c) it provides true phase information [4]; contrary to other complex wavelet transforms [4]; (d) Daubechies complex wavelet transform has no redundancy [4]. The working of the proposed (DBMSS) framework is illustrated in Figs. 4.9 & 4.10:

Step 1: Wavelet Decomposition of frames

In the proposed approach, a 2-D Daubechies complex wavelet transform is applied on the current frame and the previous frame to get wavelet coefficients in four sub-bands: LL, LH, HL and HH. The generating Daubechies complex wavelet transform is described as follows:

The basic equation of multiresolution theory is the scaling equation [4]

$$\phi(u) = 2 \sum_i a_i \phi(2u - i) \quad (4.21)$$

where a_i 's are coefficients, and $\phi(u)$ is the scaling function. The a_i 's can be real as well as complex valued. Daubechies's wavelet bases $\{\psi_{j,k}(t)\}$ in one-dimension is defined using the above mentioned scaling function $\phi(u)$ and multi-resolution analysis of $L_2(\mathfrak{R})$ [4]. The generating wavelet $\psi(t)$ is defined as:

$$\psi(t) = 2 \sum_n (-1)^n \overline{a_{1-n}} \phi(2t - n) \quad (4.22)$$

where $\psi(t)$ share same compact support $[-L, L+1]$.

Any function $f(t)$ can be decomposed into complex scaling function and mother wavelet as:

$$f(t) = \sum_k C_k^{j_0} \phi_{j_0,k}(t) + \sum_{j=j_0}^{j_{\max}-1} \sum_k d_k^j \psi_{j,k}(t) \quad (4.23)$$

where, J_o is a given at low resolution level, $\{C_k^o\}$ is called approximation coefficient and $\{d_k^j\}$ is known as detail coefficient.

Step 2: Application of change detection method on wavelet coefficients

In step 2, a change detection based method is applied on detail wavelet coefficients i.e. on sub bands: LH, HL, and HH. Let $Wf_{n,d}^f(i,j)(d=\{LH,HL,HH\})$ and $Wf_{n-1,d}^f(i,j)(d=\{LH,HL,HH\})$ be the wavelet coefficients at location (i, j) of the current frame and previous frame. Instead of assigning a fixed *a priori* threshold $V_{th,d}$ to each frame difference, this method uses the fast Euler number computation technique [126] to automatically determine $V_{th,d}$ from the video frame. The stable Euler number technique is one of the most effective algorithms for determining thresholds for change differences. However, its high computational complexity has always precluded its employment in real-time applications. A fast Euler number computation method was proposed in [126] to overcome the high computational complexity of the stable Euler number method.

The fast Euler numbers algorithm calculates the Euler number for every possible threshold with a single raster of the frame difference image using following equation:

$$E(i) = \frac{1}{4}[(q_1(i) - q_3(i) - 2q_d(i))] \quad (4.24)$$

where q_1 , q_3 , and q_d is the quads (quad is a 2*2 masks of bit cells) contained in the given image.

The output of the algorithm is an array of Euler numbers: one of each threshold value.

The Zero crossings find out the optimal threshold. Detailed algorithms for the fast Euler number computation method can be found in [126].

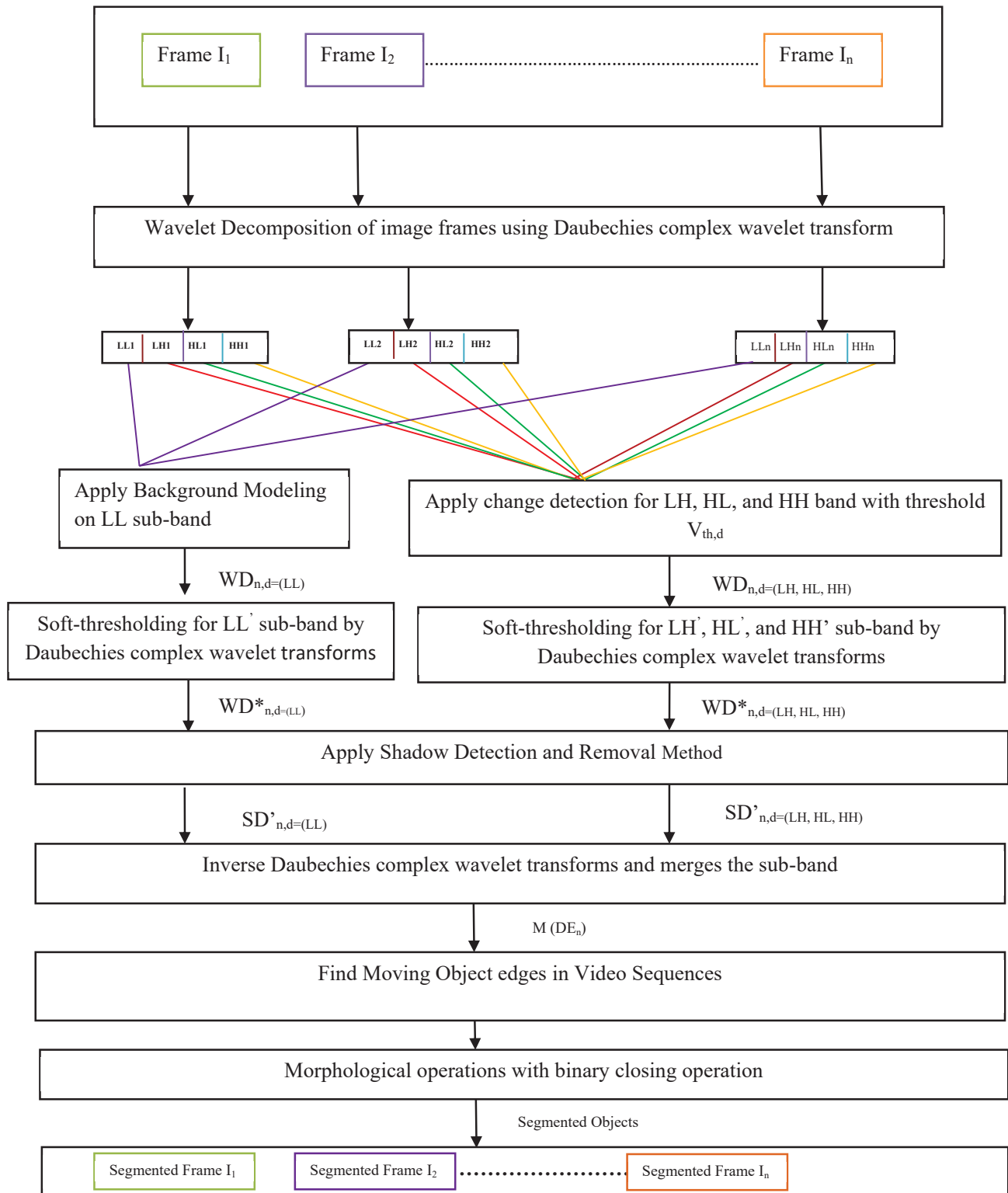


Figure 4.9: Block Diagram of the DBMSS

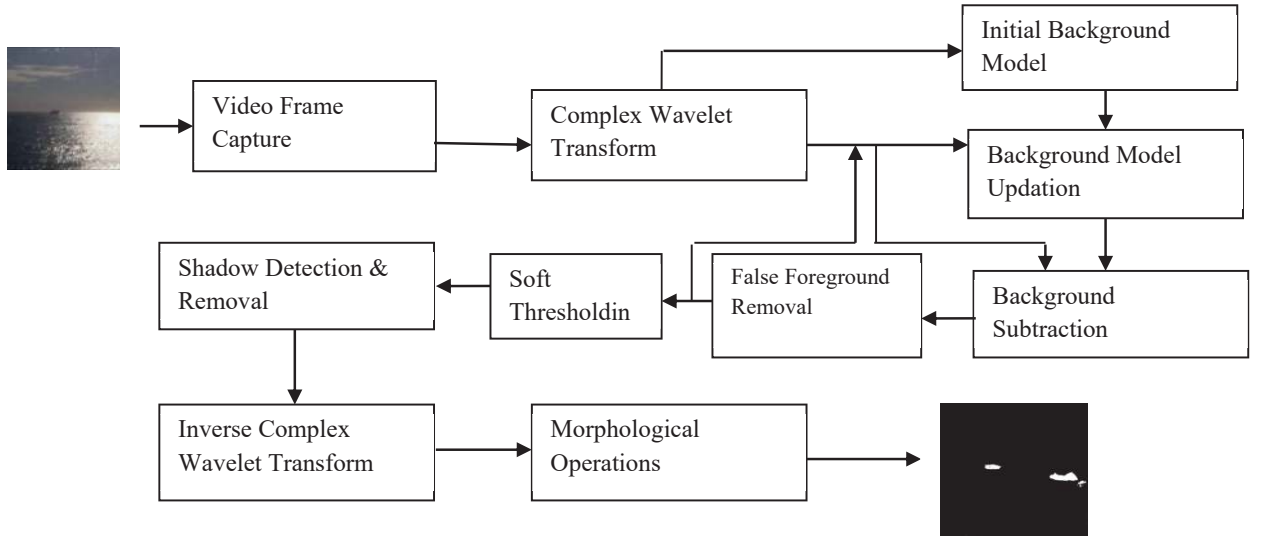


Figure 4.10: Sub-block Diagram of the DBMSS

The wavelet domain frame differences $WD_{n,d}(i, j)$ for respective sub-bands are computed as:

for every pixel location (i, j) in the co-ordinate of frame

$$WD_{n,d}(i, j) = \begin{cases} 1 & \text{if } |Wf_{n,d}(i, j) - Wf_{n-1,d}(i, j)| > V_{th,d} \\ 0 & \text{otherwise} \end{cases} \quad (4.25)$$

Step 3: Proposed background modeling using LL sub-band

Here in step 3, we have applied background modeling on water background surface. For dynamic background modeling, we have used frame difference, background registration, background difference, and background difference mask in the complex wavelet domain using LL sub-band. The background modeling step is divided into five major phases. The first phase calculates the frame difference mask $WD_{n,LL}(i, j)$ of the LL image which is obtained by thresholding the difference between coefficients in two LL sub-bands as follows:

$$WD_{n,LL}(i, j) = \begin{cases} 1 & \text{if } |Wf_{n,LL}(i, j) - Wf_{n-1,LL}(i, j)| \geq V_{th,WD} \\ 0, & \text{if } |Wf_{n,LL}(i, j) - Wf_{n-1,LL}(i, j)| < V_{th,WD} \end{cases} \quad (4.26)$$

$V_{th,WD}$ is a threshold determined automatically from the video frame by the fast Euler number computation method as explained in [126].

The second phase of dynamic background modeling maintains an up-to-date background buffer as well as background registration mask indicating whether the water background information of a pixel is available or not. According to the frame difference mask of the past several frames, water background pixels that are not moving for a long time are considered as reliable background and registered in the background buffer. The background registration process uses the following two equations:

$$S_n(i, j) = \begin{cases} S_n(i, j) + 1 & \text{if } WD_{n,LL}(i, j) = 0 \\ 0 & \text{if } WD_{n,LL}(i, j) = 1 \end{cases} \quad (4.27)$$

$$\mu_n(i, j) = \begin{cases} Wf_{n,LL}(i, j) & \text{if } S_n(i, j) \geq N_f \\ Undefined & \text{if } S_n(i, j) < N_f \end{cases} \quad (4.28)$$

where $S_n(i, j)$ is a stationary index and $\mu_n(i, j)$ is the background buffer value of a pixel with position (i, j) in the n^{th} frame. The initial values of $S_n(i, j)$ and $\mu_n(i, j)$ are set to 0 and $Wf_{n,LL}(i, j)$, respectively. If a pixel is masked as stationary for N_f successive frames (i.e., if the accumulated value in registration stationary index exceeds N_f), then that pixel is classified as part of the background region. Here experimentally set the value of N_f is 30. According to our experiments, N_f may be set at a larger value for fast moving object.

In the third phase of background modeling, a registered background buffer pixel is updated using the following equation.

$$\text{if} \quad |Wf_{n,LL}(i, j) - \mu_n(i, j)| < 2\sigma_n(i, j) \quad (4.29)$$

$$\text{then} \quad \begin{cases} \mu_n(i, j) = \chi\mu_{n-1}(i, j) + (1 - \chi)Wf_{n,LL}(i, j) \\ \sigma_n^2(i, j) = \chi\sigma_{n-1}^2(i, j) + (1 - \chi)(Wf_{n,LL}(i, j) - \mu_n(i, j))^2 \end{cases} \quad (4.30)$$

where $\sigma_n(i, j)$ is the standard deviation of a pixel with position (i, j) in the n^{th} frame and χ is the predefined constant and we considered four different maritime sequences, and recorded 10 different ocean pixel observations over 800 frames for each of the sequences. This resulted in 50 samples of size 800 each. The test statistic was calculated for each of the samples and the value of χ is set to 0.7

In the fourth phase of background modeling, we find the background difference mask with the help of background difference, which distinguishes moving objects from the background, and its operation are shown as follows:

$$BD_{n,LL}(i, j) = |Wf_{n,LL}(i, j) - \mu_n(i, j)| \quad (4.31)$$

$$BDM_{n,LL}(i, j) = \begin{cases} 1, & BD_{n,LL}(i, j) \geq V_{th,WD} \\ 0, & BD_{n,LL}(i, j) < V_{th,WD} \end{cases} \quad (4.32)$$

where $BD_{n,LL}(i, j)$ is the background difference and $BDM_{n,LL}(i, j)$ is the background difference mask of a pixel with position (i, j) in the n^{th} frame. The threshold value $V_{th,WD}$ is also automatically determined by the fast Euler number computation method [126].

In the fifth phase of background modeling, a background model is constructed on water surface using the frame difference, background registration, background difference, and background difference mask.

Step 4: Noise removal using wavelet based soft thresholding

After applying change detection based method and background modeling, the obtained result may have noise. This step deals with the noise reduction from the data obtained in steps 2 and 3. In presence of noise, the equation is expressed as:

$$WD_{n,d=(LL,LH,HL,HH)}(i, j) = WD_{n,d=(LL,LH,HL,HH)}^*(i, j) + \eta \quad (4.33)$$

where $WD_{n,d=(LL,LH,HL,HH)}^*(i, j)$ is frame difference without noise, $WD_{n,d=(LL,LH,HL,HH)}(i, j)$ is the original frame difference with noise, and η is the additive noise. T is applied on wavelet

coefficients for noise reduction. The value of soft thresholding parameter T for de-noising is computed as [127]

$$T = \frac{1}{2^{j-1}} \left(\frac{\psi}{\xi} \right) \omega \quad (4.34)$$

where j is wavelet decomposition level and ψ , ξ and ω are standard deviation, absolute mean and absolute median of wavelet coefficients of a sub-band.

Step 5: Shadow detection and removal

Daubechies complex wavelet transform provides significant amount of structural information along the edges and are useful for the shadow detection and suppression tasks. We have observed the fact that, compared to the moving object having large discontinuities at edge locations, the shadow regions are relatively smooth and have less variation in the chromacity. Therefore, shadow region contains a less high frequency information. This suggests that analysis of high frequency coefficients may prove useful in detecting shadows. Therefore, in shadow detection and removal step high frequency wavelet coefficients are analyzed in the complex wavelet domain to detect and suppress the cast shadow. The basic property of complex wavelet transform used here is that wavelet transform provides the local regularity of functions and singularities of an image $f(x, y)$ represent its edges. So edges can be detected effectively by the local maxima of the wavelet transform modulus [145]. The edge map of any sub-band can be computed by using thresholding. For example the edge map of LH sub-band can be computed as follows:

$$SD_{\eta^{ll}}^l(\alpha, \beta) = \begin{cases} 1 & \text{if } WD_{\eta^{ll}}^l F(\alpha, \beta) \geq Th(\alpha, \beta) \\ 0 & \text{Otherwise} \end{cases} \quad (4.35)$$

where $Th(\alpha, \beta)$ is a threshold determined automatically from the video frame by the fast Euler number computation method as explained in [126].

In a similar way, we can compute the edge maps for HL and HH sub-band.

Shadow regions are relatively smooth; therefore the edges of the shadow are normally suppressed in high frequency coefficients and are evident only in one of the directional coefficients [145]. So, in the next step horizontal and vertical projection vectors in high frequency sub bands (directional coefficients) are analyzed to extract the actual object region measurements.

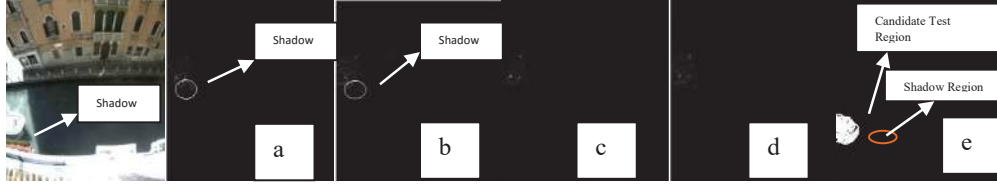


Figure 4.11: Wavelet decomposition of image from Venice-3 video sequence [146] (a) Approximation coefficients (b) Horizontal coefficients (c) Vertical coefficients (d) Diagonal coefficients (e) Candidate test region and shadow.

The shadow detection and removal method consists of the following steps:

(A) Compute the Edge Test Regions (ETR) by applying mask operation in directional coefficients using ‘Candidate Test Region’ using Eq. 4.35

$$SD_{ER}^H(\alpha, \beta) = WD_{\eta^H}^L(\alpha, \beta) * F_{obj+sha}(\alpha, \beta) \Big|_{mask} \quad (4.36)$$

$$SD_{ER}^V(\alpha, \beta) = WD_{\eta^V}^L(\alpha, \beta) * F_{obj+sha}(\alpha, \beta) \Big|_{mask} \quad (4.37)$$

$$SD_{ER}^D(\alpha, \beta) = WD_{\eta^D}^L(\alpha, \beta) * F_{obj+sha}(\alpha, \beta) \Big|_{mask} \quad (4.38)$$

where $WD_{\eta^H}^L(\alpha, \beta)$, $WD_{\eta^V}^L(\alpha, \beta)$, and $WD_{\eta^D}^L(\alpha, \beta)$ represent the high frequency coefficients in horizontal, vertical and diagonal direction respectively at L^{th} level. The superscript H, V and D represent the Horizontal, Vertical and Diagonal directional sub-band respectively.

(B) Compute the vertical and horizontal projection vectors from each direction edge region.

$$\mathbf{v}^H = \sum_i^{m^H} \{SD_{ER}^H(\alpha_i, \beta_1), \dots, SD_{ER}^H(\alpha_i, \beta_{n^H})\} \quad (4.39)$$

$$h^H = \sum_i^{nH} \{SD_{ER}^H(\alpha_1, \beta_j), \dots, SD_{ER}^H(\alpha_{mH}, \beta_j)\} \quad (4.40)$$

$$v^V = \sum_i^{mV} \{SD_{ER}^H(\alpha_i, \beta_1), \dots, SD_{ER}^H(\alpha_i, \beta_{nV})\} \quad (4.41)$$

$$h^v = \sum_i^{nV} \{SD_{ER}^V(\alpha_1, \beta_j), \dots, SD_{ER}^V(\alpha_{mV}, \beta_j)\} \quad (4.42)$$

$$v^D = \sum_i^{mD} \{SD_{ER}^D(\alpha_i, \beta_1), \dots, SD_{ER}^D(\alpha_i, \beta_{nD})\} \quad (4.43)$$

$$h^D = \sum_i^{nD} \{SD_{ER}^D(\alpha_1, \beta_j), \dots, SD_{ER}^D(\alpha_{mD}, \beta_j)\} \quad (4.44)$$

where mH, nH, mV, nV, mD and nD are the number of pixels along the horizontal, vertical and diagonal directions.

(C) Now using the following equations we can find the best fit measurements of object region (without shadow).

$$W_{obj}^x = \min \{\max(h^H), \max(h^V), \max(h^D)\} \quad (4.45)$$

$$W_{obj}^y = \min \{\max(v^H), \max(v^V), \max(v^D)\} \quad (4.46)$$

The proposed shadow suppression technique is only applied to foreground pixels rather than the entire image, thus saving significant processing time.

Step 6: Application of inverse Daubechies complex wavelet transform

After shadow detection and removal in wavelet domain from step 5, inverse wavelet transform is applied to get moving object edges in spatial domain i.e. E_n .

Step 7: Application of closing morphological operation to sub-band

As a result of step 6, the obtained segmented object may include a number of disconnected edges due to non-ideal segmentation of moving object edges. Extractions of object using these disconnected edges may lead to inaccurate object segmentation. Therefore, some morphological operation is needed for post-processing of object edge map to generate connected edges. Here, a binary closing morphological operation is used [128]

which gives $M(E_n)$ i.e. the set of connected edges. In this step, the final segmented output is obtained.

Algorithm of the proposed (DBMSS) method is given below:

Algorithm:

Input: Sequence of image frames; Output: Segmented Sequence of image frames

1. Load the Sequence of image frames.
2. Apply Wavelet Decomposition of image frames using Daubechies complex wavelet transform and find out LL, LH, HL and HH sub-band.
3. Apply change detection on detail wavelet coefficients i.e. on sub-bands: LH, HL, and HH with threshold $V_{th,d}$ (Eq. (4.25)).
4. Apply Background Modeling using LL sub-band to update the background pixels (Eq. (4.26-4.32)).
5. Apply Soft-thresholding on LL, LH, HL, and HH sub-band to remove the noise (Eq. (4.33)).
6. Apply Shadow Detection and Removal using high frequency sub-band (Eq. (4.35-4.46)).
7. Apply inverse Daubechies complex wavelet transform to get moving object edges in spatial domain.
8. Apply closing morphological operation to generate connected edges.
9. The segmented image sequences are obtained.

4.3.2.1. Experimental Results and Analysis

The proposed method (DBMSS) for dynamic background modelling and shadow suppression has been applied on a number of video clips dataset [146] discussed below.

Material and Methods

For experimentation purpose of Maritime Detection and Tracking, four datasets namely Reflections-1_2video sequence [146], Venice-3 video sequence [146], Venice-7 video sequence [146] and Ir-2 video sequence were taken [146]. The brief descriptions of each datasets are given as follows:

Reflections-1_2video sequence [146]

Reflections-1_2video sequence contains 961 frames of frame size 704 x 576. This video sequence is captured in the presence of sun reflection water and suffering from the presence of noise.

Venice-3 video sequence [146]

Venice-3 video sequence dataset contains 8476 frames of frame size 480 x 320. In this video sequence boat shadows are present and it is also suffering from the problem of noise.

Venice-7 video sequence [146]

Venice-7 video sequence dataset contains 6754 frames of frame size 480 x 320. In this video sequence water wakes and multiple direction shadows are present and it's also suffering from the problem of noise.

Ir-2 video sequence [146]

Ir-2 video sequence contains 1053 frames size 704 x 576. This video is captured in night mode environment using night vision camera with camera jittering conditions and it's also suffering from the problem of noise.

A summary of descriptions of above datasets are also given in Table 4.11.

Qualitative Analysis

This section presents the qualitative analysis of the proposed method with other methods.

For qualitative analysis of the video object by various methods, four video sequences as

given in Table 4.11 and as discussed in sub-section 4.3.2.1 were taken. For experimentation purposes, whole video sequences for each of the four test cases were examined and the visual results for two frames viz. 125 and 150 are provided.

Experiment-1

For this experiment, the Reflections-1_2 video sequence [146] from the MAR data sets have been taken, which are actually captured in the presence of some reflection and suffering from the presence of noise. From Fig.4.12, one can conclude that the segmentation result obtained by the proposed method has better segments in sun reflection water. From the obtained results in Fig. 4.12, it can be observed that the other methods [121, 136] suffers from the problems of object distortion, false acceptance of sun reflection, shadow and noise problem but the proposed method (DBMSS) is able to suppress these problems (see frame 125 – 150 (ii)). From Fig. 4.12, one can conclude that Hsia *et al.* [120] method depends on the fast motion of objects, so it is not segmenting the frame properly (see frame 125 – 150 (v)). The proposed method (DBMSS) handles all these critical condition and segments the result properly.

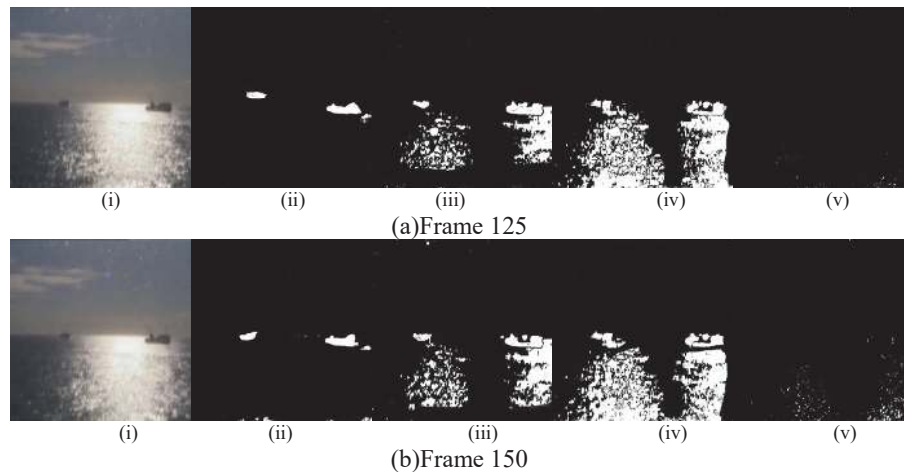






Figure 4.12: Segmentation results for high-view video sequence corresponding to (a) Frame 125, (b) frame 150 (i) original frame, and the segmented frame obtained by four methods such as: (ii) DBMSS, (iii) Bloisi *et al.* [136], (iv) Khare *et al.* [121], and (v) Hsia *et al.* [120].

Table 4.11: Description of Datasets Used for Experimentation

Video Sequence	Information with respect to Object				Information with respect to Scene			Information with respect to Shadow	
	Object Type	Object Size	Scene Type	Details of frames and Objects	Noise Level	Complexity	Shadow Size	Shadow Strength	Shadow Direction
 <p>Reflections-1_2video sequence [146]</p>	Boat/Ship	Small	Outdoor	No. of frames=961 Size of frames=704 x 576 No. of objects in a frame=2	High	Presence of sun reflection	Medium	Weak	Multiple
 <p>Venice-3 video sequence [146]</p>	Boat	Large	Outdoor	No. of frames=8476 Size of frames=480 x 320 No. of objects in a frame=1	Low	Cast shadow	Large	Strong	Multiple
 <p>Venice-7 video sequence [146]</p>	Boat	Large	Outdoor	No. of frames=6754 Size of frames=480 x 320 No. of objects in a frame=5	Low	Wakes	Large	Weak	Multiple
 <p>Ir-2 video sequence [146]</p>	Ship	Large	Outdoor	No. of frames=1053 Size of frames=704 x 576 No. of objects in a frame=1	Low	Captured by night vision camera in night (i.e. mire noisy)	Medium	Weak	Multiple

Experiment-2

Here, we have taken venice-3 video sequence [146] from the MAR data sets. In these video sequence boat shadows are present (see frame 125-150 (i)) and it's also suffering from the problem of noise. From Fig. 4.13, one can conclude that the segmentation result obtained by the proposed method (DBMSS) has better segments in shadow conditions. From the obtained results, it can be observed that the other methods [120, 121, 136] suffer from the problems of cast shadow, object distortion and noise (see frame 125 – 150 (iii-v)) but the proposed method (DBMSS) is able to suppress these problem

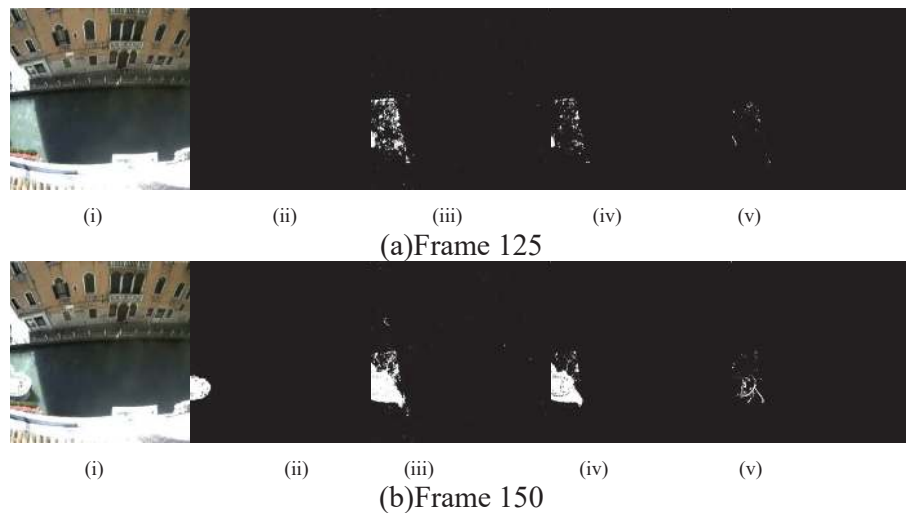


Figure 4.13: Segmentation results for venice-3 video sequence corresponding to (a) Frame 125, (b) frame 150 (i) original frame, and the segmented frame obtained by four methods such as: (ii) DBMSS, (iii) Bloisi *et al.*[136], (iv) Khare *et al.*[121], and (v) Hsia *et al.*[120].

Experiment-3

Now, the venice-7 video sequence [146] has been taken from the MAR data sets. In these video sequence water wakes and multiple direction shadows are present (see frame 125-150 (i)) and it is also suffering from the problem of noise. From the Fig. 4.14, it is clear that proposed method (DBMSS) works properly in water wakes and multiple direction shadows environment but other methods [120, 121, 136] suffer from the noise, shadow, ghost and dynamic background problems (see frame 125-150 (iii-v)).

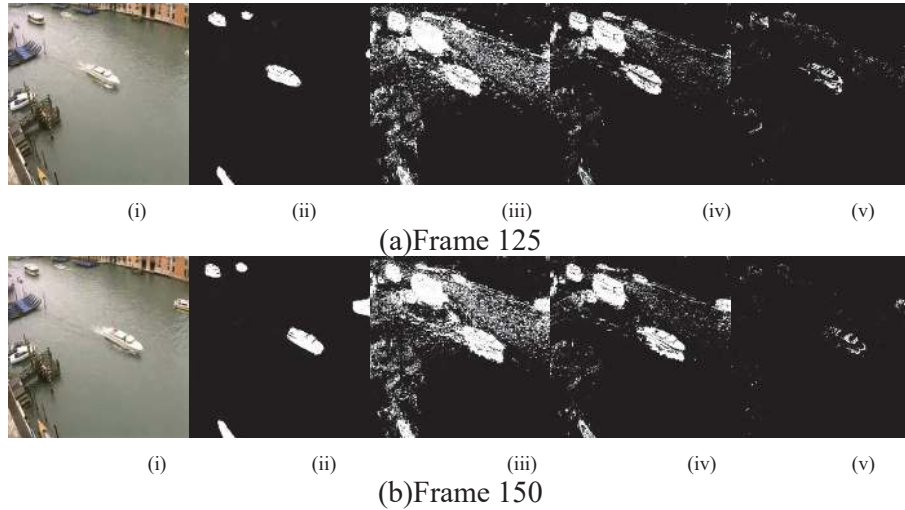


Figure 4.14: Segmentation results for venice-7 video sequence corresponding to (a) Frame 125, (b) frame 150 (i) original frame, and the segmented frame obtained by four methods such as: (ii) DBMSS, (iii) Bloisi *et al.* [136], (iv) Khare *et al.* [121], and (v) Hsia *et al.* [120].

Experiment-4

Finally, the Ir-2 video sequence [146] has been taken from the MAR data sets and used for this experiment. This video is captured in night mode environment using night vision camera with camera jittering conditions and it is also suffering from the problem of noise. From the obtained results in Fig. 4.15, it can be observed that the other methods [121, 136] suffers from the problems of object distortion, shadow, and noise problem but the proposed method (DBMSS) is able to suppress these problems (see frame 125 – 150 (ii)). From Fig. 4.15, one can conclude that Hsia *et al.* [120] method depends on the fast motion of object, so it is not segmenting the frame properly (see frame 125 – 150 (v)). The proposed method (DBMSS) handles all these critical conditions and segments the result properly (see frame 125 – 150 (ii)). From Fig. 4.15, it is deduced that the segmentation result obtained by the proposed method (DBMSS) has better segments in night mode condition.

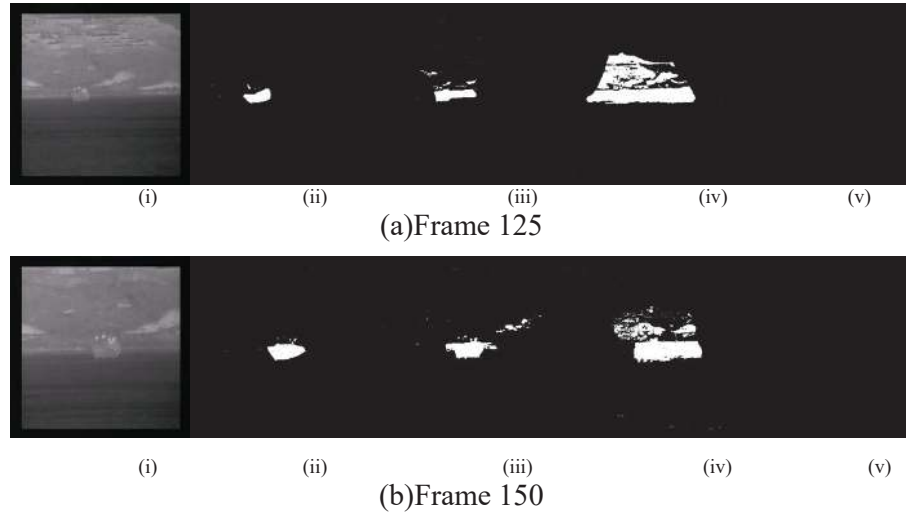


Figure 4.15: Segmentation results for Ir-2 video sequence corresponding to (a) Frame 125, (b) frame 150 (i) original frame, and the segmented frame obtained by four methods such as: (ii) DBMSS, (iii) Bloisi *et al.*[136], (iv) Khare *et al.*[121], and (v) Hsia *et al.*[120].

Quantitative Analysis

In this section of the chapter, the performances of the proposed method (DBMSS) have been compared quantitatively with other state-of-the-art methods [120, 121, 136] in terms of various performance measures: Relative foreground area measure (RFAM) [84], and Misclassification penalty (MP) [84], Relative position measure (RPM) [84], Normalized Cross Correlation (NCC) [85], Peak signal-to-noise ratio (PSNR) [87], Pixel classification based measure (PCM) [84], Normalized absolute error (NAE) [86], shadow detection rate (SDR) [88], Shadow discrimination rate [88], Precision (PR) [89], Recall (RE) [89], F-Measure [89], Execution time and Memory consumption.

For experimentation purposes, four video sequences as given in Table 4.11 and as discussed in sub-section 4.2.4 were taken in to consideration. For large video sequences, the video frames were tested at the interval of 100 frames and for small video sequences, the video frames were tested at the interval of 15 frames. The first video sequence *Reflections-1_2video sequence* [146] consists of 961 frames and total of 63 frames were tested for at the interval of 15 frames from frame no. 1 to 961 frames. The second video

sequence *Venice-3 video sequence* [146] consists of 8476 frames and total of 84 frames were tested for at the interval of 100 frames from frame no. 1 to 8476 frames. The third video sequence *Venice-7 video sequence* [146] consists of 6754 frames and total of 67 frames were tested for at the interval of 100 frames from frame no. 1 to 6754 frames. The fourth video sequence *Ir-2 video sequence* [146] consists of 1053 frames and total of 70 frames were tested for at the interval of 15 frames from frame no. 1 to 1053 frames.

All experimentations were performed on a machine with Intel 2.53GHz core i3 processor with 4 GB RAM using Matlab 13b software. Tables 4.12 present the values of RFAM, MP, RPM, NCC, PSNR, PCM, NAE, shadow detection rate, Shadow discrimination rate, PR, RE, F-Measure, Execution time and Memory consumption for the proposed method and other methods [120, 121, 136] for four video sequences [146].

Performance Analysis

Table 4.12 shows the average value of RFAM, RPM, NCC, PSNR, PCM, MP, NAE, shadow detection rate, shadow discrimination rate, precision, recall and F-measure for the whole video sequence for each test cases at for each frame taken at the interval of 15, 100, 100, 15 respectively. From Tables 4.12, one can conclude that the proposed method (DBMSS) is associated with a high average value of RFAM, RPM, NCC, PSNR, PCM, Precision, Recall, F-Measure ; and low value of MP and NAE (see the result in table 4.12 in bold) in comparison to other methods [120, 121, 136] for all datasets [146].

From Table 4.12, it can be inferred that in cases of Reflections-1_2 video sequence [146], Venice-3 video sequence [146], Venice-7 video sequence [146], and Ir-2 video sequence [146] the proposed method (DBMSS) is found to be better than all other methods [120, 121, 136] in terms of shadow detection rate and shadow discrimination rate.

In Table 4.13, average computation time (second/frame) and memory consumption for different methods for Venice-7 video sequence [146] of frame size 480 x 320 with 6754 frames are shown. From the Table 4.13, it can be observed that the proposed method (DBMSS) using complex wavelet transform is faster than the method reported in [120, 136]. The proposed method (DBMSS) takes approximately same time as the Khare *et al.* [121] but the performance of the proposed method in terms of other parameters is better in comparison to other methods.

Also from the Table 4.13, it is observed that the proposed method (DBMSS) consumes only 6.18 megabytes of RAM, which is the least in comparison with the memory requirements of other methods [120, 121, 136].

Therefore, after observing the values of twelve quantitative measures, it can be concluded that the proposed method using complex wavelet transform give better results as compared to other methods and also takes less processing time and memory in comparison to others.

4.4. Conclusions

Two new methods for dynamic background modeling and shadow suppression in Daubechies complex wavelet transform have been presented. The first method handles the small movements of non-static objects such as tree branches and bushes blowing in the wind, waving trees, shadow regions that are projected by foreground objects and are detected as moving objects. In the first proposed method, we have improved the Gaussian mixture model and used mode value to find the variance of K-Gaussian for dynamic background modeling. For shadow detection and removal we have used saturation component from HSV model and Grey level model and ratio of standard deviation and mean in complex wavelet domain. The second method deals with highly dynamic background such as moving object in water surface, boat wakes, and weather issues (such

as bright sun, fog, heavy rain), moving object in rain fall, and maritime object detection in night. In the second proposed method, we have used frame difference, background registration, background difference, and background difference mask for dynamic background modeling. For shadow detection and removal, we exploit the high frequency sub-band in the complex wavelet domain. Finally, it was concluded that the two proposed methods in Daubechies complex wavelet transform performs better visually as well as quantitatively, in comparison to other state-of-the-art methods and has been tested for different types of videos.

Table 4.12: Average Performance of different methods for all video frames of each dataset

Datasets	Methods	Average Performance for Over All Video Frames Of Each Dataset											
		RFA M	MP	RP M	NC C	PSN R	PC M	NAE	Shadow Discrimi nation Rate ζ	Shado w Detecti on Rate x	Precis ion	Recall	F- measu re
Reflectio ns-1_2 video sequence [146](961 frames)	Bloisi <i>et al.</i> [136]	0.7350 85	0.0055 22	0.70 0788	0.63 547	68.47 045	0.82 4735	0.088 643	79.22795	76.3215 9	0.8002 83	0.7429 24	0.7701 32
	Hsia <i>et al.</i> [120]	0.1581 6	0.1741 61	0.19 7895	0.17 2981	37.29 577	0.45 3747	0.941 489	30.79981	30.6682	0.5162 7	0.4145 9	0.4582 7
	Khare <i>et al.</i> [121]	0.6426 94	0.0734 21	0.58 6334	0.59 558	64.35 813	0.72 7871	0.257 711	74.68252	71.7389 8	0.7199 46	0.6591 77	0.6877 69
	DBMSS	0.8912 42	0.0027 21	0.89 2746	0.90 8262	75.28 204	0.91 6858	0.040 55	91.46176	93.2610 1	0.9273 18	0.9423 16	0.9345 96
Venice-3 video sequence [146](847 6 frames)	Bloisi <i>et al.</i> [136]	0.7434 91	0.0137 42	0.69 7718	0.76 1546	68.19 047	0.80 7248	0.088 792	78.8588	75.1456	0.7968 02	0.7416 05	0.7679 48
	Hsia <i>et al.</i> [120]	0.3630 67	0.8697 56	0.19 5995	0.44 6572	37.21 795	0.45 9936	0.693 226	30.22631	29.9818	0.5216 7	0.4055 1	0.4545 1
	Khare <i>et al.</i> [121]	0.7170 77	0.0175 82	0.59 7819	0.64 5534	64.13 214	0.71 3382	0.252 687	74.74371	71.6646 9	0.7210 24	0.6601 01	0.6888 93
	DBMSS	0.9021 02	0.0035 33	0.88 9889	0.90 2847	75.34 242	0.91 0781	0.039 875	90.48646	93.0366 7	0.9217 32	0.9420 2	0.9315 96
Venice-7 video sequence	Bloisi <i>et al.</i> [136]	0.7321 55	0.0111 33	0.71 1606	0.68 833	68.49 647	0.81 2584	0.088 548	78.67723	75.7052 9	0.8004 99	0.7460 35	0.7719 03
	Hsia <i>et al.</i> [120]	0.2678 67	0.1685 18	0.19 6681	0.34 2247	37.43 86	0.46 5583	0.699 671	30.31893	30.0337	0.5246 6	0.4259 3	0.4687 4
	Khare <i>et al.</i> [121]	0.7278 95	0.0211 51	0.58 5823	0.45 52	64.01 594	0.71 6882	0.269 222	74.31774	71.6634 8	0.7168 21	0.6525 21	0.6827 92

[146](675 4 frames)	DBMSS	0.9054 82	0.0068 11	0.89 0771	0.90 47	75.77 964	0.91 5672	0.039 835	90.70287	93.0105 3	0.9278 77	0.9404 65	0.9339 05
Ir-2 video sequence [146](105 3 frames)	Bloisi <i>et al.</i> [136]	0.7367 68	0.0150 63	0.69 9871	0.54 5711	47.74 326	0.81 8705	0.088 919	78.45618	75.7826 4	0.8023 99	0.7440 46	0.7717 19
	Hsia <i>et al.</i> [120]	0.1835 03	0.2402 42	0.19 829	0.27 3953	25.93 171	0.35 0929	0.694 22	29.28651	29.7168	0.5114 3	0.4113 6	0.4541 6
	Khare <i>et al.</i> [121]	0.6775 64	0.0406 41	0.59 1698	0.46 7109	42.74 249	0.71 9074	0.261 881	74.20675	71.8106 2	0.7150 3	0.6605 58	0.6863 27
	DBMSS	0.9196 87	0.0086 38	0.90 1842	0.73 8976	71.97 8	0.91 1328	0.039 458	90.93223	93.5795 2	0.9225 9	0.9401 75	0.9311 11

Table 4.13: Computational Time and Consumption Memory for Venice-7 Video Sequence [146]

Methods	Computational Time (in second/frame)	Memory Consumption (MB)
Bloisi <i>et al.</i> [136]	3.945	22.75
Hsia <i>et al.</i> [120]	1.867	11.75
Khare <i>et al.</i> [121]	1.427	8.36
DBMSS	1.312	6.18

