International Journal of Natural and Engineering Sciences 2 (3): 69-74, 2008
ISSN: 1307-1149 www.nobel.gen.tr

Unsupervised Moving Object Edge Segmentation Using Dual-Tree Complex Wavelet Transform

Turgay CELİK[1] Zeki YETGİN[2*]

1 Temasek Laboratories, Nanyang Technological University, Singapore

2 Dokuz Eylul University, Computer Engineering Department, Izmir, Turkey

[*]Corresponding Author
E-mail: zeki@cs.deu.edu.tr

Received: May 08, 2008
Accepted: July 15, 2008

Abstract

Unsupervised moving object edge segmentation in dual-tree complex wavelet transform domain is presented. The interframe change detection method with automatic thresholds in six complex wavelet subbands is used with edge map detected by Canny edge detector in the corresponding frame in order to detect moving edges. We propose automatic threshold selection algorithm which are used for thresholding magnitude of subbands of dual-tree complex wavelet transform. The motivation behind using dual-tree complex wavelet transform rather than discrete wavelet transform is the fact that the directionality information of dual-tree complex wavelet transform is better than the discrete wavelet transform and it is nearly shift-invariant. The performance of the proposed algorithm is demonstrated both numerically and visually.

Key words: Moving object edge segmentation, change detection, dual-tree complex wavelet transform, discrete wavelet transform.

INTRODUCTION

The MPEG-4 Video supports the representation of video objects (VOs) of natural or synthetic origin, coded as separate entities in the bit-stream, which the user can access and manipulate such as cut, paste, replace and rotate [1]. In the MPEG-4 context, a VO can still be the traditional case of a sequence of rectangular frames formed by pixels, but a VO can also correspond to a sequence of arbitrarily shaped sets of pixels possibly with a semantic meaning, given that this higher level information is somehow made available (e.g. by providing shape or transparency information).

The way VOs are identified is not within the scope of the MPEG 4 standard - it is considered as a pre-processing step. MPEG-4 provides the means to represent any composition of objects, whatever the methods used to achieve the composition information. The arbitrarily shaped VOs can be obtained by a variety of means such as: automatic or assisted segmentation of natural data, chroma key techniques, or synthetic computer generated data. Currently the MPEG-4 Video bit stream syntax consists of a hierarchy of classes: Video Session (VS), Video Object (VO), Video Object Layer (VOL) and Video Object Plane (VOP). The VS class is the highest entity in the class hierarchy and may contain one ore more VOs (VO1, VO2, ..., VON), while each VO can consist of one or more layers (VOL1, VOL2,…, VOLN) which can be used to enhance the temporal or spatial resolution of a VO. Thus the VOL class is used to support temporal and spatial scalabilities. An instance of a VOL at a given time instant is called a Video Object Plane (VOP). However, it is very difficult to extract VOPs [2]; a pre-processing becomes an important issue. Since VOs usually have different motion features from the background and from other VOs, most existing automatic VO extraction schemes use motion information in video sequences as an important cue to produce semantic objects, which is simply extraction of moving objects.

Using interframe difference in change detection is a popular method to extract moving objects [2], [3], [4]. The idea behind this is to use the image difference between consecutive frames and identify significant differences with post-processing. Huang *et al*. used the interframe difference in wavelet domain to extract moving objects edges [5]. They have employed the three consecutive frame differences in discrete wavelet transform (DWT) domain for frames at times (n-1), (n) and (n+1) and edge map of frame at time (n) to extract moving edges in frame (n). DWT has shift variance and poor directionality [6]. Recently, Kingsbury introduced dual-tree complex wavelet transform (DT-CWT) [6]. DT-CWT poses properties of shift invariance and better directional information with respect to DWT. The DT-CWT is a form of DWT that generates complex valued coefficients, where it is implemented with dual-trees of filters that independently generates real and imaginary parts of complex coefficients. At each scale, the DT-CWT produces six directional subbands which are oriented at ±15°, ±45° and ±75° where DWT produces three directional subbands oriented at 0, 45 and 90 degrees. We modify the structure proposed by Huang *et al*. method which used DWT to detect moving object edges. We employ characteristics of DT-CWT in moving object edge detection in video sequences. The proposed algorithm is modified version of the algorithm proposed by Huang *et al*. which uses DWT. The change detection in DT-CWT domain is used to find significant changes between consecutive frames in video sequences. Change detection is performed in a pre-defined scale of DT-CWT, and Canny edge detector is applied on the difference coefficients for each directional subband to detect moving object edges which are significant differences in difference image. Detected moving edges in each subband are merged together to detect final moving edges at specified scale.

Detected moving edges are transformed into image domain by using nearest neighboring interpolation technique. Thick edges caused from interpolation technique are removed using edge detection result on the current frame found by applying Canny edge detector. We propose an automatic threshold selection criterion used for subband change detection part, which makes our algorithm free of thresholds. Further more; we compare our algorithm and Huang *et al*. method in the manner of correct edge detection and false alarm in well-known video sequences.

This paper is organized as follows; in the following section we give moving regions detection algorithm in DT-CWT domain, this section is followed by the performance comparison with Huang *et al*. method and we finish our paper with conclusion.

MATERIALS AND METHODS

Change Detection Between Two Consecutive Frames in DT-CWT Domain

The proposed method uses three frame differences in DT-CWT domain to extract moving object edges, namely frames (n-1), (n) and (n+1). The moving object edges between frames (n-1) and (n) is combined with the moving object edges between frames (n) and (n+1). The object's edges at frame (n) are detected using the result of this combination and the edge detector applied in frame (n). The proposed algorithm's flow chart is shown in Fig. 1.

The change detection method is employed to extract significant pixel differences in DT-CWT domain between two consecutive frames namely (n) and (n-1) in corresponding subbands which can be formulated as follows;

$$D_{n,n-1,d,s} = \begin{cases} \left|\grave{u}_{n,s,d} - \grave{u}_{n-1,s,d}\right|, \text{f} \ \left|\grave{u}_{n,s,d} - \grave{u}_{n-1,s,d}\right| \geq Th_{n,n-1,d,s} \\ 0, \text{otherwise} \end{cases} \quad (1)$$

where $CD_{n,n\text{-}1,d,s}$ is change detection in DT-CWT domain between frames (n) and (n-1) at scale s and directional subband d. $\grave{u}_{n,s,d}$ is magnitude of complex coefficients in DT-CWT domain for frame n at scale s and directional subband d, similarly $\grave{u}_{n-1,s,d}$ is magnitude complex coefficients in DT-CWT domain for frame (n-1) at scale s and directional subband d. $Th_{n,n\text{-}1,d,s}$ is a threshold used for each subbands, which will be automatically determined by analyzing distribution of coefficients in $CD_{n,n\text{-}1,d,s}$. Note that the scale s can be any value meanwhile d Є {± 15, ± 45, ± 75}.

The idea behind equation (1) is to suppress small differences and leave the high differences between consecutive frame differences in DT-CWT domain coefficients. Using $CD_{n,n\text{-}1,d,s}$, we can employ Canny edge detector to find significant edges in $CD_{n,n\text{-}1,d,s}$ as follows;

$$CDM_{n,n-1,d,s} = \Phi\left(D_{n,n-1,d,s}\right) \quad (2)$$

where $CDM_{n,n\text{-}1,d,s}$ is binary map edge map with edges represented with 1 and non-edges are 0, and $\Phi(\,)$ is a function used to detect edges, which is Canny edge detector in our case. $CDM_{n,n-1,d,s}$ represents moving object edges between consecutive frames detected in scale s in directional subband d. Note that in order to detect all edges, Canny edge detector with a threshold 0 is used.

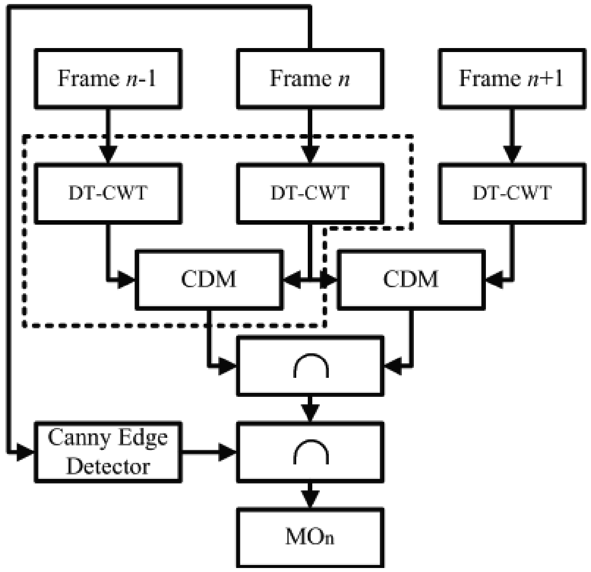


Figure. 1: Proposed moving object edge segmentation algorithm.

Moving Object Edge Detection in DT-CWT Domain

After finding edges in subbands, the overall moving object edge map, $CDMO_{n,n-1s}$, between frames (n) and (n-1) is found by combining moving edge maps in all subbands which is formulated as follows;

$$CDMO_{n,n-1s} = \bigcup_{all\ d} CDM_{n,n-1,d,s} \quad (3)$$

After, the DT-CWT of frames (n-1), (n), (n+1) are calculated and change detection maps between frames (n-1) and (n) ($CDMO_{n,n-1s}$) and frames (n+1) and (n) ($CDMO_{n,n+1s}$) are found using the equation (3), the next stage is the combination of change detection maps to find final moving object boundaries at scale s. From the perspective of moving object representation concept, $CDMO_{n,n-1s}$ represents moving object edges in frame (n) with respect to frame (n-1) and $CDMO_{n,n+1s}$ represents moving object edges in frame (n) with respect to frame (n+1).

In order to find $CDMO_{n,n-1s}$ or $CDMO_{n,n+1s}$ (the process enclosed in dashed square in Fig. 1), the flowchart in Fig. 2 is used where only generation of $CDMO_{n,n-1s}$ is demonstrated and application for $CDMO_{n,n+1s}$ is straight forward. In order to find moving object edges in frame (n) using only $CDMO_{n,n-1s}$ and $CDMO_{n,n+1s}$ at scale s, binary AND operation is used, i.e.;

$$TMO_{n,s} = CDMO_{n,n-1,s} \cap CDMO_{n,n+1,s} \quad (4)$$

where $TMO_{n,s}$ represents the moving object edges found using intersection of $CDMO_{n,n-1s}$ and $CDMO_{n,n+1s}$ for frame (n) in scale s.

It is clear that, edges in the current frame (n) found using Canny edge detector should exist in the final map $TMO_{n,s}$ with sampled version. Using this idea, false edge detections in $TMO_{n,s}$ *is* suppressed by combing Canny edge detector in frame (n), i.e.;

$$MO_n = \Phi(Frame\ n) \cap \grave{e}\,(TMO_{n,s}) \qquad (5)$$

where MO_n represents moving edges in frame (n), è () is a function to upscale input data to the original image size. The meaning of equation (5) is that, the moving object edges detected by using $TMO_{n,s}$ may have false alarms, on the other hand the edges detected by $TMO_{n,s}$ should also be edges in current frame (n) detected by Canny edge detector. Using equation (5), one may expect a reduction in false alarm rate. Note that, since moving object's edges are detected using DT-CWT, detected edge map $TMO_{n,s}$ is downscaled version of original image size. For instance, if we take 2 level DT-CWT, then the size of $TMO_{n,s}$ is quarter of original image size. In order to detect moving object edges in original image spatial grid, upscaling operation is applied where in upscaling operation nearest neighboring type interpolation is used. This operation also produces false alarms because each edge pixel in lower resolution is transformed to finer resolution by copying its values to 4 pixels in finer resolution. This effect produces thick edges in finest resolution which is combated by using equation (5).

The change detection used in equation (1) directly depends on the value of $Th_{n,n-1,d,s}$. The selection of this value is important for overall performance of the proposed algorithm. Huang *et al*. used fixed value of thresholds for overall video analysis which is optimized with respect to input video. Here, we propose automatic threshold selection criterion for $Th_{n,n-1,d,s}$. For each subband $CD_{n,n-1,d,s}$ at scale s, $Th_{n,n-1,d,s}$ is found as follows;

$$Th_{n,n-1,d,s} = 2.5\ m_i \qquad (6)$$

where m_i is center of highest frequency bin of the histogram of $CD_{n,n-1,d,s}$. While creating histogram for $CD_{n,n-1,d,s}$ the number of bins, B, is found as follows;

$$B = \left\lfloor \frac{\max\left(CD_{n,n-1,d,s}\right) - \min\left(CD_{n,n-1,d,s}\right)}{3.49\ \acute{o} P^{-1/3}} \right\rfloor - 1 \qquad (7)$$

where ó is standard deviation of $CD_{n,n-1,d,s}$, P is total number of pixels in $CD_{n,n-1,d,s}$, and $\lfloor\ \rfloor$ rounds to nearest integer. The calculation of $Th_{n,n-1,d,s}$ is straight forward for $CD_{n,n-1,d,s}$. In Fig. 3 we give visual demonstration of proposed algorithm using three consecutive frames for well-known video test sequence Trevor.

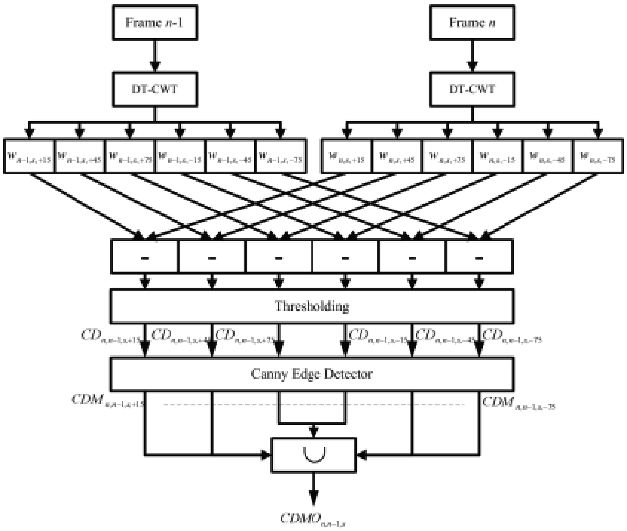


Figure. 2: Flow chart to generate $CDMO_{n,n-1s}$.

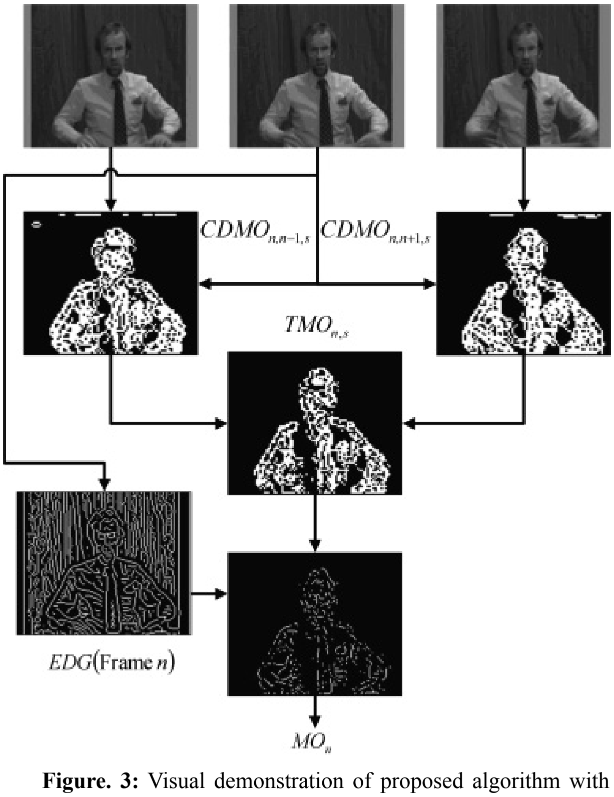


Figure. 3: Visual demonstration of proposed algorithm with frames from Trevor sequence.

RESULTS

In this section we give performance analysis of our algorithm in well known video sequences, Trevor and Claire video sequences. In order to evaluate our algorithm's and Huang *et al.*'s algorithm performance, it is needed to select appropriate implementation for DT-CWT and DWT. For DWT Haar filters are used to detect edges. Note that the filters are intentionally selected as short as possible in order to make the algorithm sensitive for small changes in edge detection process. On the other hand, length-10 filters given in [7] is used for DT-CWT. For both techniques, the algorithms are performed for two level transform, i.e. s = 2 and their results are upscaled to original image's size by using nearest-neighboring interpolation.

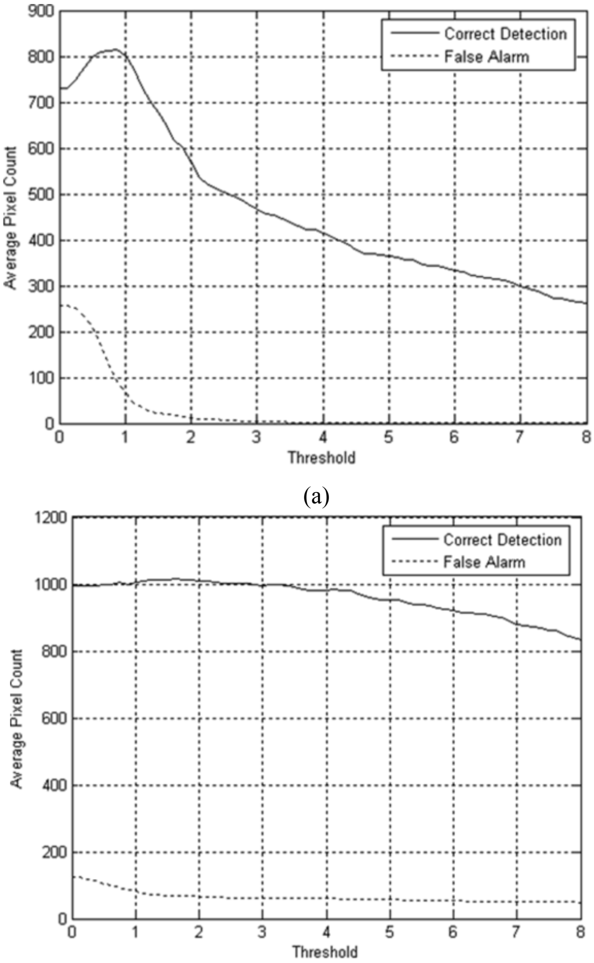
Performance Criterion

Performance is evaluated in the manner of correct moving edge detection rate and false alarm rate. The false alarm rate and correct detection rate is evaluated using the following criterion; each video sequence is manually segmented into moving objects and non-moving objects parts and the detected edges in non-moving parts are counted as false alarm, meanwhile detected edges in the moving parts are counted as correct moving edges.

Finding Optimum Threshold Value for DWT-based Moving Object Edge Segmentation Method

In order to compare our method with Huang *et al.*'s method a threshold is needed for their method, note that our proposed method does not need such a threshold selection scheme. For this purpose we do the following; for a range of thresholds, we counted correct detected edges and false detected edges for each sequence. In counting process, we set a threshold value then for a range of video sequences we find averages of correct and false detected edges. For Claire video sequence, frames from 2 to 51 are used for this process, average of 50 frames is evaluated and for Trevor sequence, frames from 61 to 110 are used. Results of this process are summarized in Fig. 4. Using Fig. 4 (a) and (b), threshold values are selected for Huang *et al.*'s method. Thresholds are settled to 2.0 where for Claire sequence average of correct detection rate is around 756 pixels and average of false detected edges is around 36 pixels, and for Trevor sequence average of correct detection rate is around 1007 pixels and average of false detected edges is around 66 pixels.

In Fig. 5 detected moving object edges for Claire sequence is shown for frames 92 to 94; the first column shows the original video sequence, the second column shows moving edges detected by our algorithm where automatic thresholding is applied and the third column shows the moving edges detected by Huang *et al.*'s algorithm. It is clear that moving object edge detection capacity of our algorithm is much better than the Huang *et al.*'s algorithm. This is apparent by visual observation of number of detected moving object edge pixels. In order to show numerical performance results Fig. 6 is presented. In Fig. 6 correct detection and false alarms for proposed algorithm and Huang *et al.*'s algorithm is shown for the Claire sequence for frames from 100 to 200. Fig. 6 depicts that, our algorithm outperforms Huang *et al.*'s algorithm both in high detection rate and low false alarm rate. Proposed algorithm performs the average correct detection rate of 1104 pixels with 28 false alarm pixels; on the other hand the correct detection rate for Huang *et al.*'s algorithm is around 645 pixels with 37 false alarm pixels.



(b)

Figure. 4: Average correct detection and false detection counts for Huang *et al.*'s algorithm ; (a) Claire video sequence, and (b) Trevor video sequence.

Quantitative and Visual Comparisons

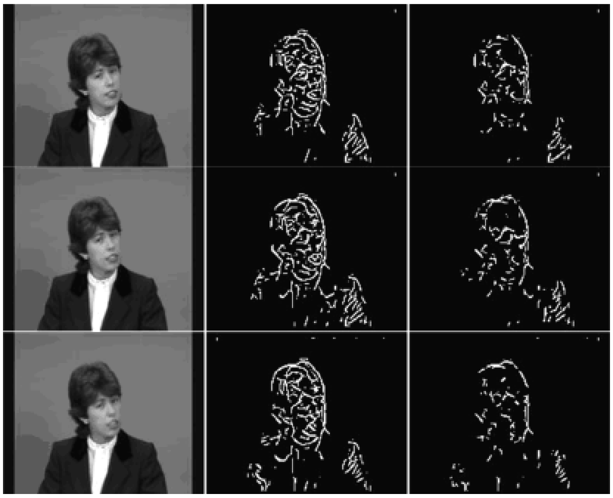


Figure. 5: Moving object edges of Claire video sequence for frames 92 to 94; first column is the original frame, second column is moving edges found using proposed algorithm, third column is the moving edges using Huang *et al.*'s algorithm.

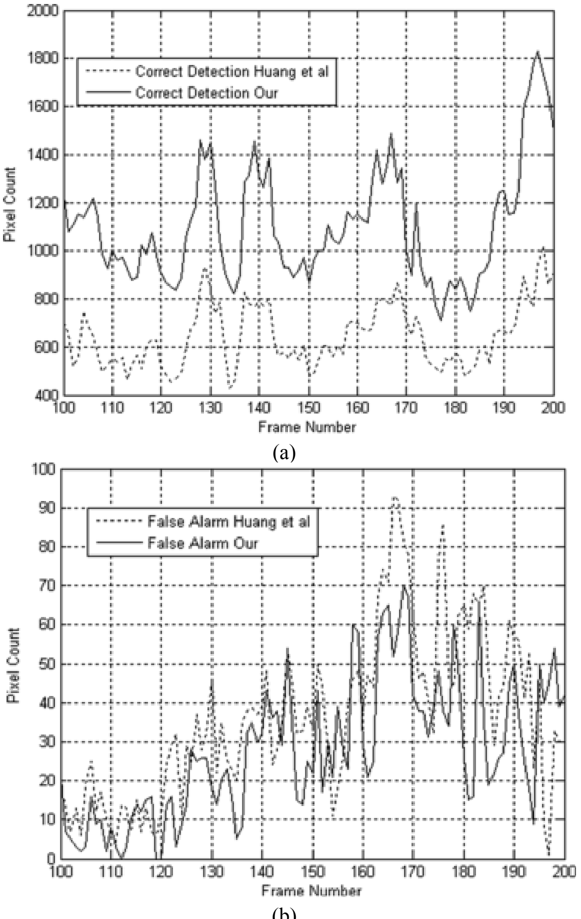


Figure. 6: Correct detection and false alarm comparisons for our algorithm and Huang *et al.*'s algorithm in Claire video sequence; (a) Correct detection comparison graph, and (b) False alarm comparison graph

Fig. 5 reflects the directional moving edge detection strength of proposed algorithm. Moving edges around face area are much more detailed than the Huang *et al.*'s algorithm. Huang *et al.*'s algorithm misses some edges around facial features, e.g. mouth, eyes, eyebrows and nostrils, and face boundary. Proposed algorithm performs very well in detecting moving edges around both facial features and face boundary. This is mainly caused from the fact that DT-CWT has better directional response than DWT. Proposed method detects weak directional moving edges around the face area and left-shoulder of Claire. Claire sequence has very small motion in it; head motion, shoulder motions, and both of head and should motions. Fig. 6 (a) shows that both methods have similar detection characteristics but proposed method detects more correct edges with respect to Huang *et al.*'s method with fewer false alarms. Peaks in Fig. 6 (a) come from the motion of Claire. In some frames, she both moves her head and some parts of her shoulders and arms at the same time where both algorithms performs high detection rates with respect to other frames (peaks in graphs). However the proposed algorithm detects more details with respect to Huang *et al.*'s algorithm.

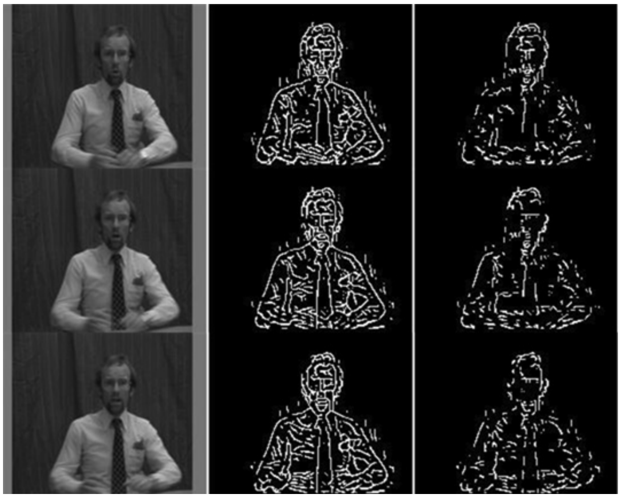
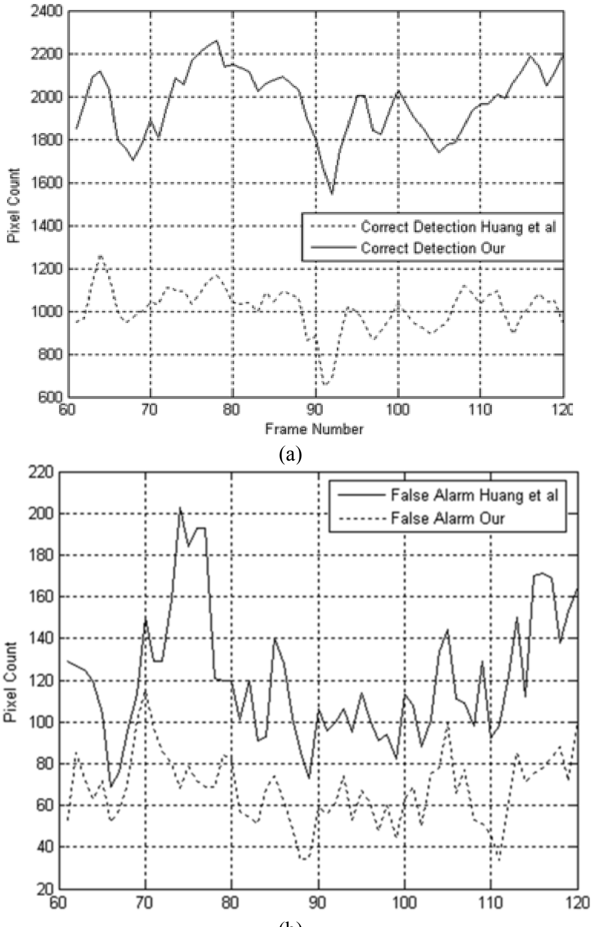


Figure. 7: Moving edges of Trevor video sequence for frames 85 to 87; the first column is the original frame, the second column is moving object edges found using our algorithm, and the third column is the moving edges using Huang *et al.*'s method.

Similar to Fig. 5, detected moving object edges for Trevor sequence is shown in Fig. 7 for frames 85 to 87; the first column shows the original video sequence, the second column shows moving edges detected by our algorithm where automatic thresholding is applied and the third column shows the moving edges detected by Huang *et al.*'s algorithm. It is clear that moving object edge detection capacity of our algorithm is much better than the Huang *et al.*'s algorithm which was the case for the Claire video sequence too. This is apparent by visual observation of number of detected moving object edge pixels. In order to show numerical performance results Fig. 8 is presented. In Fig. 8 correct detection and false alarms for proposed algorithm and Huang *et al.*'s algorithm is shown for the Trevor sequence for frames from 61 to 120. Proposed algorithm performs the average correct detection rate of 1970 pixels with 68 false alarm pixels; on the other hand the correct detection rate for Huang *et al.*'s algorithm is around 1009 pixels with 121 false alarm pixels. The correct detection rate for the proposed algorithm is almost twice the correct detection rate of Huang *et al.*'s algorithm, meanwhile false detection rate is almost twice better than the false detection rate of Huang *et al.*'s algorithm.

Fig. 6 reflects the directional moving edge detection strength of proposed algorithm. Moving edges around foreground object are much more detailed than the Huang *et al.*'s algorithm. Huang *et al.*'s algorithm misses some edges around foreground objects, e.g. head boundary, right and left arms and shoulders boundary. Proposed algorithm performs very well in detecting moving edges of foreground object boundaries. Similar to Claire sequence, this is mainly caused from the fact that DT-CWT has better directional response than DWT. Proposed method detects even weak directional moving edges around foreground object. Trevor sequence has almost equal motion in its frames; head motion, shoulder motions, and both of head and should motions are almost available at each frame of sequence. This fact is reflected in Fig. 8 (a) and (b). The number of detected moving object edges is almost the same for each frame with some fluctuations in it. It is also clear from Fig. 8 that proposed algorithm outperforms Huang *et al.*'s algorithm both in high detection rate and low false alarm rate.



(a)

(b)

Figure. 8: Correct detection and false alarm comparisons for our algorithm and Huang *et al*.'s algorithm in Trevor video sequence; (a) Correct detection comparison graph, and (b) False alarm comparison graph.

The shape characteristics of Fig. 8, 9 (a) and (b) reflect that both algorithms detect some major features which are very strong difference coefficients in wavelet domain. The improvement in high detection rate of proposed algorithm comes from the fact that DT-CWT more sensitive to directional information than DWT do.

DISCUSSION

In this paper, we proposed unsupervised moving object edge detection in DT-CWT domain. The proposed algorithm is modified version of the algorithm proposed by Huang *et al*. which uses DWT. The change detection in DT-CWT domain is used to find significant changes between consecutive frames in video sequences. Change detection is performed in a pre-defined scale of DT-CWT, and Canny edge detector is applied on the difference coefficients for each directional subband to detect moving object edges which are significant differences in difference image. Detected moving edges in each subband are merged together to detect final moving edges at specified scale. Detected moving edges are transformed into image domain by using nearest neighboring interpolation technique. Thick edges caused from interpolation technique are removed using edge detection result on the current frame found by applying Canny edge detector.

The proposed algorithm outperforms Huang *et al*.'s algorithm both in high detection rate and low false alarm rate. The performance improvement is mainly caused from shift-invariance and directionality properties of DT-CWT. We also propose an automatic threshold selection criterion which makes proposed algorithm free of threshold selection for subband change detection map extraction which is not the case for Huang *et al*. method.

Since DT-CWT is implemented in two parallel DWT with different low-pass and high-pass filters for lower and upper DWT branches, it can be applied for real-time applications. This makes our algorithm suitable for real-time application where moving object edges are required.

REFERENCES

[1] N. Brady, "MPEG-4 standardized methods for the compression of arbitrarily shaped video objects," *IEEE Trans. on Circ. and Sys. for Video Technology*, vol. 9, pp. 1170–1189, Dec. 1999.

[2] F. Long, H. F. D. P. and W. Siu, "Extracting semantic video objects,", *IEEE Digital Media*, pp. 48–55, 2001.

[3] C. Kim and J. Hwang, "Fast and automatic video object segmentation and tracking for content-based applications," *IEEE Trans. on Circ. and Sys. for Video Technology*, vol. 12, pp. 122–129, 2002.

[4] C. Kim and J. Hwang, "Object-based video abstraction for video surveillance systems," *IEEE Trans. on Circ. and Sys. For Video Technology*, vol. 12, pp. 1128–1138, 2002.

[5] J. Huang and W. Hsieh, "Double-change-detection method for wavelet-based moving-object segmentation," *IEE Electronic Letters*, vol. 40, 2004.

[6] N. Kingsbury, "Image processing with complex wavelets," *Philos. Trans. Roy. Soc. Lon.*, vol. 357, pp. 2543-2560, 1999.

[7] N. G. Kingsbury and J. F. Magarey, "Wavelet transforms in image processing," *Proceedings of First European Conference on Signal Analysis and Prediction*, pp. 23–34, 1997.