Motion Segmentation in Compressed Video Using Markov Random Field Classification

Yue-Meng Chen, Ivan V. Bajić, and Parveneh Saeedi
School of Engineering Science, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada

***Abstract*—In this paper, we propose an unsupervised segmentation algorithm for extracting moving objects/regions from compressed video using Markov Random Field (MRF) classification. First, motion vectors (MVs) are quantized into several representative classes, from which MRF priors are estimated. Then, a coarse segmentation map of the MV field is obtained using a maximum a posteriori estimate of the MRF label process. Finally, the boundaries of segmented moving regions are refined using color and edge information. The algorithm has been validated on a number of test sequences, and experimental results are provided to demonstrate its superiority over previously proposed methods.**

***Index Terms*—Motion segmentation, Markov Random Field.**

I. Introduction

Moving object segmentation is important in a variety of applications such as video surveillance, video database browsing, object-based video transcoding, etc. During the last two decades, a number of approaches have been proposed to tackle this problem. Especially interesting is the problem of moving object segmentation in compressed video, due to the abundance of compressed video content.

State-of-the-art object segmentation methods can be broadly grouped into pixel domain approaches (e.g., [1–3]) and compressed domain approaches (e.g., [4-11]). The former extract objects by exploiting visual features such as shape, color and texture. In this case, the compressed video has to be fully decoded prior to segmentation. The high computational load and over-segmentation of possible moving objects are two major drawbacks of these methods. On the other hand, compressed domain methods exploit compressed domain data, such as motion vectors (MVs) and DCT coefficients, to facilitate segmentation. Some methods [4–5] operate directly on sparse (block-based) MV field. These methods have low complexity, but often suffer from poor localization of object boundaries, and inconsistency in the number or segmented regions from frame to frame. Alternatively, one can create a dense (pixel-base) MV filed by interpolation, and then run segmentation on the dense field, at the cost of significantly higher complexity [6–8]. Combinations of compressed-domain and pixel-domain operations have also been proposed to balance complexity and accuracy [9–11]. These methods first create a coarse segmentation from the sparse MV field, and then refine it in the pixel domain. Although these methods generally have higher segmentation accuracy near object boundaries than purely compressed domain approaches, maintaining a consistent number of segmented regions across frames is still a challenge for them.

The segmentation method proposed in this paper is a combined compressed domain and pixel domain approach. A distinctive feature of our method is the use of MV quantization based on local motion similarity to find the most likely number of moving objects/regions, and use the statistics of the resulting clusters to initialize prior probabilities for subsequent Markov Random Field (MRF) classification. This way, the proposed method is able to overcome some of the difficulties faced by previous methods, such as over-segmentation [1-3], under-segmentation [4], and segmented region inconsistency [9-11]. Further, pixel domain boundary refinement yields more accurate region boundaries than can be achieved by purely compressed domain methods [4-5], while still having a much lower complexity than pixel domain methods [6-8].

The paper is organized as follows. The segmentation framework and its major components are elaborated in Section II, followed by the experimental results in Sections III. The conclusions are drawn in Section IV.

II. Markov Random Field Motion Segmentation

Block diagram of the proposed segmentation system is shown in Fig. 1. The system incorporates two major segmentation components: coarse segmentation from motion, which can be carried out in the compressed domain, and fine segmentation, carried out in the pixel domain. Coarse segmentation further consists of two units: MV quantization, which generates the preliminary segmentation map, and MRF MV classification.

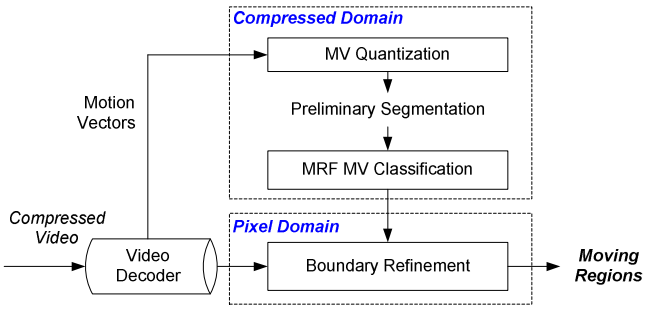

Figure 1: Overview of the MRF motion segmentation system

This work was supported in part by the NSERC/CCA New Media Initiative grant number STPGP 350740. E-mail: yuemengc@sfu.ca, ibajic@ensc.sfu.ca, psaeedi@sfu.ca.

A. Markov Random Field motion model

Our approach to coarse motion segmentation is based on a Markov Random Field (MRF) motion model [1] [3] [8]. In this model, motion vectors $\mathbf{MV} = (MV^X, MV^Y)$ within a given moving region ω follow a conditional distribution $P(\mathbf{MV} \mid \omega)$, while region labels ($\omega$s) follow a 2-D MRF distribution based on a given neighborhood system. The goal is to infer region labels (ωs) from the observed MV field.

To simplify calculations, we assume that within each region, MVs form an independent bivariate Gaussian process. Under this assumption, the likelihood function for the j-th block in the frame is

$$P(\mathbf{MV}_j \mid \omega_j) = \frac{1}{\sqrt{2\pi}\sigma^X_{\omega_j}\sigma^Y_{\omega_j}} \cdot \exp\left(-\frac{1}{2}\left(\frac{\left(MV_j^X - m^X_{\omega_j}\right)^2}{\left(\sigma^X_{\omega_j}\right)^2} + \frac{\left(MV_j^Y - m^Y_{\omega_j}\right)^2}{\left(\sigma^Y_{\omega_j}\right)^2}\right)\right), \quad (1)$$

where $m^X_{\omega_j}$ and $m^Y_{\omega_j}$ are the means of the horizontal and vertical MV component within the region labeled ω_j, while $\sigma^X_{\omega_j}$ and $\sigma^Y_{\omega_j}$ are the corresponding standard deviations. The dependence among the labels of neighboring blocks is modeled by a MRF which follows the Gibbs distribution:

$$P(\omega_j) = \frac{1}{Z}\prod_C \exp(-V(C)), \quad (2)$$

where Z is the normalizing constant ensuring that $\Sigma P(\omega_j) = 1$, C is a *clique* (a set of neighboring blocks) and $V(C)$ is the *clique potential*. We only consider 4-adjacency cliques. In other words, two blocks form a clique if one is immediately to the North, South, East, or West of the other, as shown in Fig. 2. If ω_1 and ω_2 are the region labels of the two blocks in the clique C, the potential of C is defined to be

$$V(C) = \begin{cases} -\beta, & \text{if } \omega_1 = \omega_2, \\ +\beta, & \text{otherwise,} \end{cases} \quad (3)$$

where $\beta > 0$ is a parameter controlling the homogeneity of the regions. Based on (2) and (3), nearest neighbors are more likely to have the same region label.

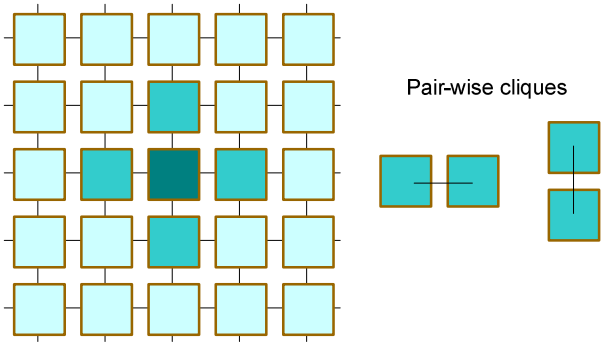

Figure 2: First-order neighborhood system and clique configuration.

B. MV quantization and MRF parameter estimation

The main difficulty in MRF segmentation is to determine the parameters that specify the MRF, particularly the number of motion segments and their statistics. Our approach is to first perform vector quantization of MVs in order to estimate these parameters. To achieve robust quantization, we suppress the influence of possibly inaccurate MVs by examining the smoothness of the MV field [12]. A MV that is very different from its neighbors, and therefore suspected to be inaccurate, will have less influence on the resulting quantization. A similar idea was studied in [2] in the context of color quantization. We first apply a 3×3 vector median filter to the MV field. Then, for each motion vector $\mathbf{MV}_j$, find the maximum Euclidean distance $D_{MAX,j}$ from its 8-adjacent neighbors, and assign it the weight $W_j = \exp(-D_{MAX,j})$. Using these weights, we run a generalized Lloyd algorithm for vector quantization:

1) Start with a single cluster (all MVs in the frame), compute its centroid $\mathbf{MV}_{cent}$ as
$$\mathbf{MV}_{cent} = \left(\sum_j W_j \mathbf{MV}_j\right) / \left(\sum_j W_j\right), \quad (4)$$
then split it into two clusters by deriving two new centroids as $\mathbf{MV}_{cent} \pm \mathbf{MV}_{cent}/2$.
2) Quantize all MVs in the frame into existing clusters using the nearest neighbor criterion. Then, for the i-th cluster C_i, update the centroid MV as:
$$\mathbf{MV}^{C_i}_{cent} = \left(\sum_{\mathbf{MV}_n \in C_i} W_n \mathbf{MV}_n\right) / \left(\sum_{\mathbf{MV}_n \in C_i} W_n\right). \quad (5)$$
3) Compute the weighted distortion of each cluster C_i:
$$WD^{C_i} = \sum_{\mathbf{MV}_n \in C_i} W_n \left\| \mathbf{MV}_n - \mathbf{MV}^{C_i}_{cent} \right\|. \quad (6)$$
Let C_k be the cluster with the maximum weighted distortion, and let $X_{\max}$, $X_{\min}$, $Y_{\max}$, and $Y_{\min}$ be, respectively, the maximum and minimum horizontal and vertical component among the centroids. Split cluster C_k into two clusters with centroids $\mathbf{MV}^{C_k}_{cent} \pm \mathbf{P}$, where
$$\mathbf{P} = \left(\frac{X_{\max} - X_{\min}}{2(N-1)}, \frac{Y_{\max} - Y_{\min}}{2(N-1)}\right), \quad (7)$$
and N is the total number of clusters prior to splitting.
4) Repeat steps 2) and 3) until the total weighted distortion (sum of all WD^{Ci}) becomes less than a given threshold (in our experiments, 5% of its initial value in step 1), or the smallest cluster size becomes less than another threshold (in our experiments, 5% of the total MV field size).

Upon completion, a preliminary segmentation map is obtained: MVs in cluster C_i obtain the region label ω_i, which enables us to compute $m^X_{\omega_i}$, $m^Y_{\omega_i}$, $\sigma^X_{\omega_i}$, $\sigma^Y_{\omega_i}$ and $P(\omega_i)$.

C. MRF motion segmentation

For block j, based on the Bayes' theorem, the posterior probability $P(\omega_j \mid \mathbf{MV}_j)$ is proportional to $P(\mathbf{MV}_j \mid \omega_j)P(\omega_j)$, so the Maximum A Posteriori (MAP) estimate of ω_j is given by:

$$\hat{\omega}_j = \arg\max_{\omega_j} P(\mathbf{MV}_j \mid \omega_j) P(\omega_j), \quad (8)$$

where $P(\mathbf{MV}_j \mid \omega_j)$ is computed as in (1), and $P(\omega_j)$ is computed as in (2)–(3). The MAP segmentation for the entire MV field corresponds to maximizing

$$\prod_j P(\mathbf{MV}_j \mid \omega_j) P(\omega_j), \tag{9}$$

and is obtained using the method of Iterated Conditional Modes (ICM) [13], by iteratively solving (8) for each block in the frame. We use the ICM implementation from [3] (modified for MV segmentation rather than pixel segmentation), with six iterations. The final step is to identify small regions whose size is less than 2% of the total MV field, and group each block in those regions to the neighboring large region with the closest centroid MV.

D. Boundary refinement

Segmentation map obtained in the previous section is block-based. Since real object/region boundaries rarely follow block boundaries, segmentation map needs to be refined. Fig. 3 illustrates the refinement steps on frame #22 from *Flower Garden*. The steps involve boundary block identification, edge detection, and interior region growing.

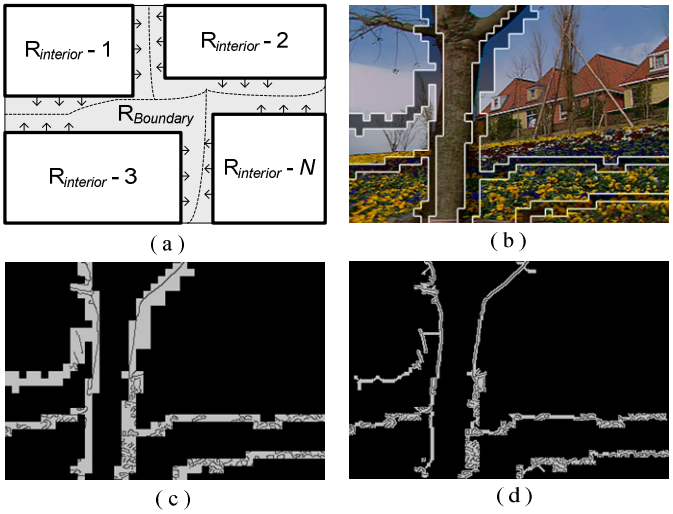

Figure 3: (a) Interior regions grow within boundary regions, (b) Coarse segmentation and identified boundary blocks, (c) Initial boundary region and edges within them, (d): Result of interior region growing.

Boundary blocks are identified based on the segmentation map from Section II-C and the Region Motion Deviation (RMD) map. The RMD value I_j^C of $\mathbf{MV}_j$ within region C is the normalized deviation of $\mathbf{MV}_j$ from the centroid MV of region C: $I_j^C = 255 \cdot \left(D_j^C / D_{\max}^C \right)$, where $D_j^C = \left\| \mathbf{MV}_{cent}^C - \mathbf{MV}_j \right\|$, and $D_{\max}^C = \max_j D_j^C$. A two-pass procedure is employed to classify a block as either a *boundary block* or *interior block*. In the first pass, we scan all the blocks in the raster scan order, and for each block we check its East (E), South (S), and South-East (SE) neighboring blocks, if available. If any of these blocks belong to a different region than the one the current block belongs to, we compare the RMD values of all four blocks (current, E, S, SE), and label the block with the highest RMD value as a boundary block. In the second pass, we seek to extend the boundary to be at least 2 blocks (16 pixels) wide, to improve the chance that the real region boundaries lie within boundary blocks. To do this, we check 4-adjacency neighbors of all boundary blocks found so far, and check if they have at least one horizontal (vertical) neighbor classified as a boundary block. If not, we label the horizontal (vertical) neighbor with the higher RMD value as a boundary block. At the end, all blocks not classified as boundary blocks are labeled interior blocks. An example is shown in Fig. 3(b), where boundary blocks are indicated in darker color.

Canny edge detector on the Y-component is used to identify edges within boundary blocks as shown in Fig. 3(c). Then, interior regions are grown towards each other via morphological erosion of the boundary blocks using a 3×3 structuring element. The structuring element is not allowed to cross an edge. Hence, this restricted erosion will move the interior region boundaries up to the nearest edge(s). In this process, some boundaries of neighboring interior regions may meet, in which case the pixel-wise boundary between these regions is identified. In other cases, boundaries do not meet due to a complicated edge pattern between them, so we further employ region growing based in color information as in [11] to finalize region boundaries.

III. Experimental Results

The proposed segmentation algorithm has been tested on a variety of standard sequences with different motion characteristics. Sequences are CIF (352×288) and SIF (352×240) resolution with a frame rate of 30 frames per second. In this work, we use the XviD MPEG-4 codec (http://www.xvid.org/) for compression, using the IPPP… GOP structure, at 512 kbps. We point out that the segmentation framework is generic and easily adapted to other video compression standards. The MVs extracted from the bitstream are normalized to form a uniformly sampled MV field, where each MV corresponds to an 8×8 block.

A. Estimation of the number of MRF classes

We first evaluate MV quantization as a way to determine the number of MRF classes, and to provide the initial segmentation map. Figs. 4(a) and (b) show how the weighted quantization distortion changes as a function of the number of classes on sample frames from *Flower Garden* (frame #2) and *Table Tennis* (frame #4). Fig. 4(a) indicates that three classes seem to be appropriate for the frame #2 of *Flower Garden*, while Fig. 4(b) indicates that two classes are appropriate for frame #4 of *Table Tennis*. The corresponding initial segmentation maps are shown in Figs. 4(c) and (d), respectively. These initial segmentation maps enable us to calculate the means and variances of horizontal and vertical MV components within each region.

B. MRF motion segmentation and boundary refinement

Next, we evaluate MRF segmentation, especially the number of ICM iterations and the role of parameter β in (3) which influences the spatial structure of the MRF. In the top left part of Fig. 5, we show the posteriori energy (the sum of potentials in (3) of all cliques in the field) vs. the number of iterations of ICM implementation from [3], when $\beta = 3.5$. The graph indicates that 4-6 iterations are sufficient, as suggested in [3]. Hence, we used 6 iterations in all our experiments.

The rest of Fig. 5 shows the segmentation of frame #2 of *Flower Garden* obtained by setting β to 0, 1.5, and 3.5,

respectively. When $\beta = 0$, no spatial constraints are imposed on the MRF, so the segmentation does not change from its initial layout obtained by MV quantization (Fig. 4(c)). As β increases, neighboring blocks are more likely to be in the same region, so region boundaries end up being more compact. Our experiments indicate that $\beta = 3.5$ provides a good balance between boundary compactness and segmentation accuracy, so we use this value in the remaining experiments.

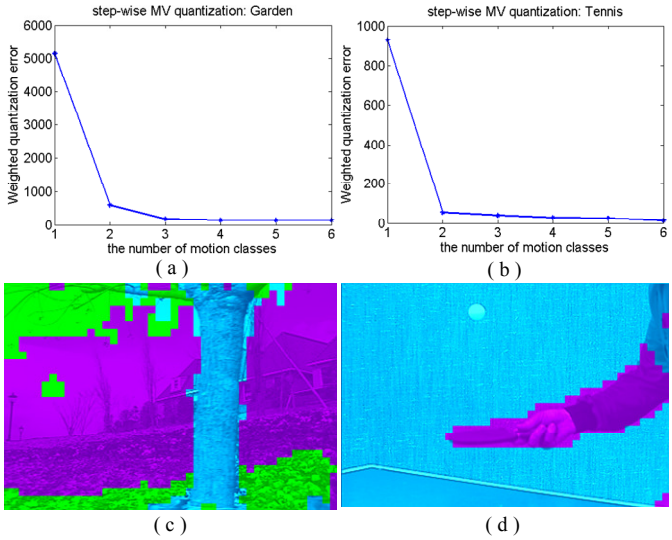

Figure 4: (a) and (b): weighted quantization error vs. the number of motion classes, (c) and (d): the corresponding segmentation map after MV quantization, where segments are distinguished by different colors.

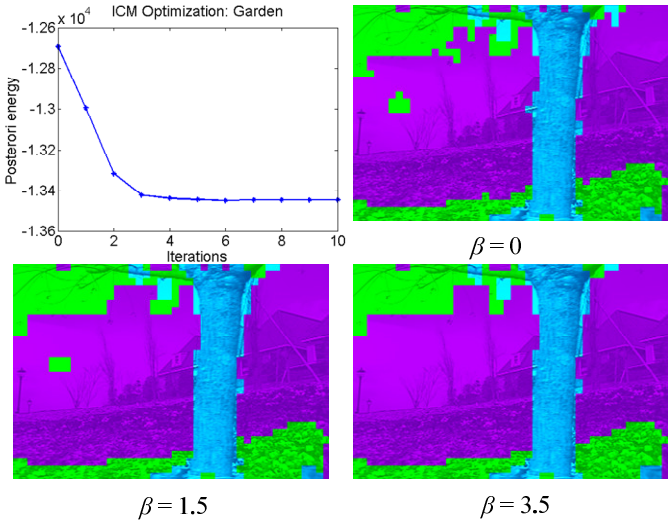

Figure 5: [Top Left]: Posteriori energy vs. iterations; Other figures: MRF segmentation with $\beta \in \{0, 1.5, 3.5\}$.

In Fig. 6(a), we illustrate the final MRF segmentation of frame #2 of *Flower Garden* (after merging blocks from small regions to neighboring regions), and in Fig. 6(b) we show the boundary refinement results. We also show the results from four other state-of-the-art segmentation algorithms: [2], [4], [7], and [11] for comparison. Fig. 6(c) shows the segmentation result using the algorithm from [2], which is image-based, and does not use motion information, and hence results in over-segmentation. This problem has been mitigated to some extent by our earlier work [11], shown in Fig. 6(d), which utilizes *k*-means clustering and motion consistency. However, the scene is still over-segmented. Fig. 6(e) shows the result of using the method from [4], which is based on MRF with two motion classes (background and foreground). The result is an under-segmented scene, with part of the background (garden) included in the same segment as the foreground (tree trunk). Finally, Fig. 6(f) shows the segmentation result from [7], which is a MV-based approach using the Expectation Maximization algorithm on a dense MV field. This method ends up with same number of motion classes as ours, but these motion classes (segments) are less compact that in our case, and some are not even spatially connected. A few other segmentation results of our method are shown in Fig. 7.

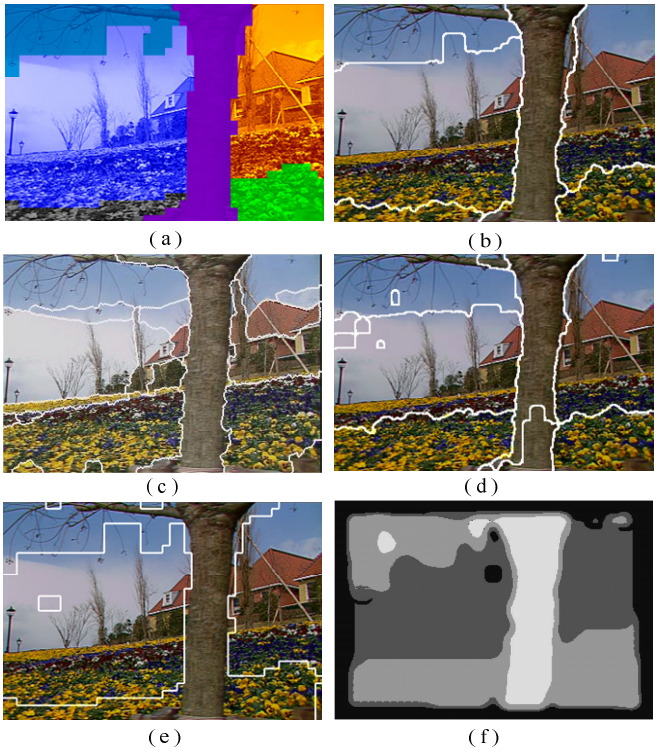

Figure 6: (a): Coarse MRF segmentation, (b):boundary refinement. (c, d, e, f): segmentation result from Ref. [2], [11], [4] and [7], respectively.

Figure 7: Final segmentation results, [Left to Right]: Sequence *Table Tennis* (frame # 5), *Coastguard* (frame # 40), and *Hall Monitor* (frame # 50).

C. *Quantitative evaluation*

In addition to the visual results above, we provide a quantitative evaluation of our method, using the manually segmented sequences *Flower Garden* and *Table Tennis* (available at: http://www.sfu.ca/~ibajic/datasets.html). We test how accurately the fastest moving objects (tree trunk in *Flower Garden*, player's hand and ball in *Table Tennis*) can be segmented. By counting the pixels correctly identified as moving region pixels (True Positives - TP), the pixels correctly identified as the background (True Negatives – TN), the pixels wrongly identified as moving region pixels (False Positives - FP), and the pixels wrongly identified as

background (False Negatives – FN), we can compute several quantities for measuring segmentation accuracy: Precision = TP / (TP + FP), Recall = TN / (TN + FN), and F-measure as the harmonic mean of Precision and Recall. In terms of these quantities, we compare our method and the method from [4], which is the latest work addressing MRF motion segmentation in block-based compressed video. In our implementation, 8×8 uniformly sampled MV field is used.

TABLE I
AVERAGE PRECISION, RECALL, AND F-MEASURE.

Sequence	*Flower Garden*		*Table Tennis*	
	Proposed	**Ref. [4]**	**Proposed**	**Ref. [4]**
Precision	0.86	0.41	0.91	0.79
Recall	0.94	0.98	0.67	0.69
F-measure	0.90	0.56	0.75	0.72

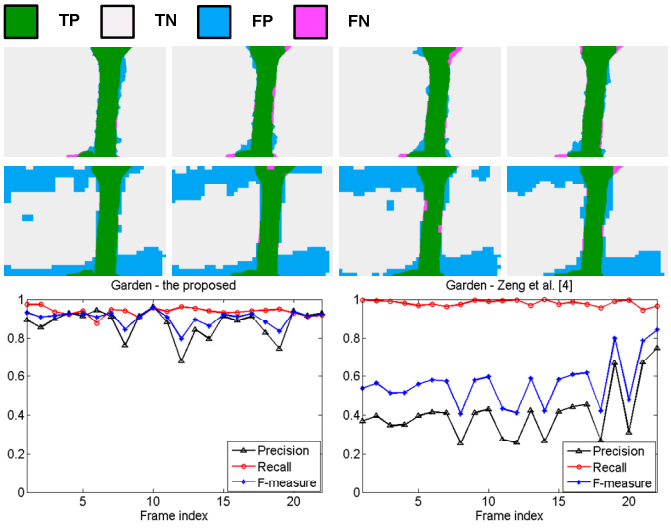

Figure 8: Quantitative evaluation – *Garden.* [Top]: the proposed method, frame #1, #3, #5, #7, from left to right, [Middle]: corresponding segmentation using method from [4], [Bottom]: the quantitative evaluation for the proposed method (left) and method from [4] (right).

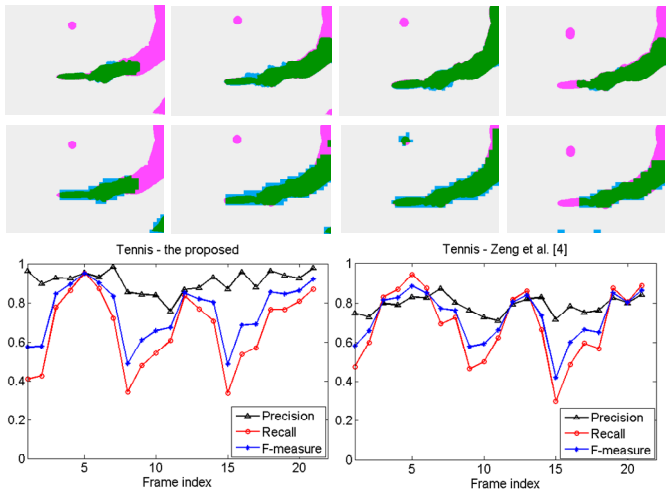

Figure 9: Quantitative evaluation – *Tennis.* [Top]: the proposed method, frame #1, #3, #5, #7, from left to right, [Middle]: corresponding segmentation using method from [4], [Bottom]: the quantitative evaluation for the proposed method (left) and method from [4] (right).

The top and middle rows in Figs. 8-9 show the segmented objects in frames #1, #3, #5, and #7, extracted by our method and the one from [4]. TP, TN, FP, and FN pixels are also shown. The last row in both figures shows the quantitative measures for the first 25 frames of *Flower Garden* and *Table Tennis*, while their averages are listed in Table I. For *Flower Garden*, the method from [4] has an average precision of 0.41 due to under-segmentation (background pixels included in the foreground, shown as blue pixels in Fig. 8), while our method maintains a much higher precision of 0.86. The performance of the two methods is more similar on *Table Tennis*, where the assumption made in [4] about two motion classes (foreground and background) is more appropriate. Nonetheless, our boundary refinement yields more accurate boundaries, which again leads to higher precision (0.91 vs. 0.79). Finally, note that our segmentation method has a reasonably low complexity. On a standard desktop PC with Intel Pentium CPU at 3.0 GHz, with 2 GB of RAM, on a CIF sequence, motion segmentation (in Matlab) takes about 80 ms per frame, while boundary refinement (in C/C++) takes about 20 ms.

IV. CONCLUSION

In this paper, we have presented an unsupervised moving region/object segmentation algorithm for compressed video, which includes MRF-based coarse segmentation, and boundary refinement using color and edge information. The proposed method achieves a good balance between accuracy and complexity, and compares favorably against other state-of-the-art segmentation methods.

REFERENCES

[1] Z. Kato, "Segmentation of color images via reversible jump MCMC sampling," *Image and Vision Computing*, vol. 26, issue 3, pp. 361-371, Mar. 2008.

[2] Y. Deng, and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 23, issue 8, pp. 800-810, Aug. 2001.

[3] Z. Kato, T. C. Pong, and J. C. M. Lee. "Color Image Segmentation and Parameter Estimation in a Markovian Framework," *Pattern Recognition Letters*, 22(3-4):309--321, March 2001.

[4] W. Zeng, J. Du, W. Gao, and Q. Huang, "Robust moving object segmentation on H.264/AVC compressed video using the block-based MRF model," *Real-Time Imaging*, vol. 11, pp. 290-299, Jun. 2005.

[5] M. Ritch and N. Canagarajah, "Motion-based video object tracking in the compressed domain," *Proc. IEEE ICIP'07,* vol. 6, pp. VI-301-VI-304, Oct. 2007.

[6] J. Wang and E. Adelson, "Representing Moving Images with Layers," *IEEE Trans. Image Processing*, vol. 3, pp. 625-638, Sept. 1994.

[7] R. V. Babu, K. R. Ramakrishnan, and S. H. Srinivasan, "Video object segmentation: a compressed domain approach," *IEEE Trans. Ciruits Syst. Video Technol*. vol. 14, no. 4 , pp. 462-474, 2004.

[8] N. Vasconcelos and A. Lippman, "Empirical Bayesian motion segmentation," *IEEE Trans. on Pattern Anal. Mach. Intell*., vol. 2, issue 2, pp. 217-221, Feb. 2001

[9] D. Zhong and S. F. Chang, "An integrated approach for content-based video object segmentation and retrieval," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 9, no. 8, pp. 1259-1268, Dec. 1999.

[10] X. Shi, Z. Zhang, L. Shen, "Multiresolution segmentation of video objects in the compression domain," *Optical Engineering.,* vol. 46, no. 9, 097401, Sep. 2007.

[11] Y.-M. Chen, I. V. Bajić, and P. Saeedi, "Coarse-to-fine moving region segmentation in compressed video," *Proc. IEEE WIAMIS'09*, pp. 45-48, London, UK, May 2009.

[12] A. Dante and M. Brookes, "Precise real-time outlier removal from motion vector fields for 3D reconstruction," *Proc. IEEE ICIP'03*, pp. 393-396, Sep. 2003.

[13] J. Besag, "On the statistic analysis of dirty pictures," *J. Roy. Statist. Soc. B*. vol. 48, no. 3, pp. 259-302, 1986.