SEGMENTATION AND CLASSIFICATION OF MOVING OBJECTS FOR INTELLIGENT VIDEO SURVEILLANCE

B.SHRAVAN KUMAR,
Research Scholar, JNTUH,India,

Dr.V.USHA SHREE
Professor ECE Dept.

Abstract: Moving object segmentation and classification fromcompressed video plays an important role for intelligent videosurveillance. Compared with H.264/AVC, HEVC introduces ahost of new coding features which can be further exploited formoving object segmentation and classification. In this paper, wepresent a real-time approach to segment and classify movingobject using unique features directly extracted from the HEVCcompressed domain for video surveillance. In the proposedmethod, firstly, motion vector interpolation for intra-codedprediction unit and MV outlier removal are employed forpreprocessing. Secondly, blocks with non-zero motion vectors areclustered into the connected foreground regions using the fourconnectivitycomponent labeling algorithm. Thirdly, object regiontracking based on temporal consistency is applied to the connectedforeground regions to remove the noise regions. The boundary ofmoving object region is further refined by the coding unit size andprediction unit size. Finally, a person-vehicle classification modelusing bag of spatial-temporal HEVC syntax words is trained toclassify the moving objects, either persons or vehicles. Theexperimental results demonstrate that the proposed methodprovides the solid performance and can classify moving personsand vehicles accurately.

Keywords: Compression Domain, Object Segmentation,Object Classification, HEVC, Video Surveillance.

I. INTRODUCTION

Moving object segmentation and classification fromvideo data is one of the most important tasks forintelligent video surveillance. Most computer vision methodsfor moving object detection and classification assume that theoriginal video frames are available and extract descriptions or features from pixel domain [1-3]. Note that most video contentare received or stored in compressed formats encoded withinternational video coding standards, such as MPEG-2[4],H.264/AVC [5] and HEVC [6]. To obtain the original videoframe, we have to perform video decoding. In video analysis atlarge scales, such as content analysis and search for a largesurveillance network, the complexity of video decodingbecomes a major bottleneck of the real-time system. To addressthis issue, compression-domain approaches have been exploredfor video content analysis which extracts features directly fromthe bit stream syntax, such as motion vectors and block codingmodes [7]. The major advantage of compression-domainapproaches is their low computational complexity since thefull-scale decoding and reconstruction of pixels are avoided.Therefore, compressed domain methods are desired for realtimevideo analysis applications. In this paper, we focus onmoving object detection and classification from HEVCcompressed surveillance videos. Specifically, by extractingfeatures from HEVC compressed surveillance video bitstream,the moving objects are located and classified, such as personsor vehicles.

A. Related Work

Recently, a number of moving object segmentation andclassification algorithms using motion vector (MV) informationin H.264/AVC compression domain have been reported. R. V.Babuet al. [8] introduce a method to accumulate MVinformation over time for moving object segmentation.Temporally accumulated MVs are further interpolated spatiallyto obtain a dense field, and the expectation maximizationprocedure is then applied on the dense motion field for finalsegmentation. S. D. Bruyneet al. [9] develop a method toanalyze the reliability of MVs in H.264/AVC domain. Thisreliability information along with the MV magnitude is used tosegment foreground objects from the background. In [10] and[11], MVs are classified into multiple types, such asbackground, edge, foreground, and noise. Then, the MVs andtheir associated class information are used to segment eachblocks. In [12] and [13], global motion is first removed fromthe motion vector field, and moving object segmentationprocess is performed on the compensated motion vector filed.Y. Chen et al. [14] develop a method to extract moving objectregions from compressed domain by using global motionestimation and Markov random field classification.It should be noted that MVs extracted from the compressedbitstream are determined in terms of rate-distortion and theymay not represent the true object motion. Therefore, it isdifficult to find real moving objects solely based on the MVfields.

To address this issue, other information, such as DCTcoefficients and macro-block partitions, are used to detect andtrack moving objects. C. Poppeet al. [15] propose to use thenumber of bits consumed by each 4×4 block to detect themoving objects from H.264/AVC videos. F. Porikliet al. [16]present a segmentation method that takes advantage of the interframemotion and intra-frame DCT coefficients embedded inMPEG videos. M. Laumeret al. [17] propose to use (sub-)block types to refine the level of motion vector for each block.P. Dong et al. [18] develop a moving object segmentation andtracking method by adaptively using the information frommotion vectors, DCT coefficients and prediction modes. H.Sabirinet al. [19] propose a spatiotemporal graph-basedmethod for detecting and tracking moving objects by treatingthe encoded blocks with non-zero motion vectors and/or nonzeroresidues in H.264/AVC bit streams as potential parts ofmoving objects. S. H. Khatoonabadiet al. [20] present a methodto track moving objects in H.264/AVC compressed videosusing a spatiotemporal Markov random field (ST-MRF) model,which naturally integrates spatial and temporal aspects of theobject motion using motion vectors and block

coding modes.As the newest international standard for video coding, HEVCprovides equivalent subjective quality with about 50% bitratereduction compared to H.264/AVC high profile [21]. HEVChas adopted a host of new encoding features and tools, such ascoding unit and prediction unit, which can be exploited formoving object segmentation and classification in compresseddomain. However, little work has been done on moving objectanalysis directly from HEVC compressed videos.

H. Li et al.[22] present a rapid abnormal event detection method for videosurveillance by using the syntax features extracted from HEVCcompressed domain. Both the syntax information extractedfrom the HEVC compressed domain and the color informationin pixel domain are used to segment the foreground andbackground region [23, 31, 32]. D. Park et al. [24] propose toextend the ST-MRF model from H.264/AVC compresseddomain to HEVC compressed domain for moving objecttracking. M. Moriyama et al. [30] propose to conduct thesegmentation procedure after the temporal sub-sampling of thevideo sequence. However, the unique features, such as codingunit and prediction unit of HEVC, are not explored for movingobject segmentation and classification.In this paper, we develop a framework for moving objectsegmentation and classification by using the motion vectors andassociated modes directly extracted from HEVC compressedvideo. Specifically, we focus on surveillance videos whosecameras are stationary. Compared to existing methods in theliterature, our work has the following unique aspects andinnovations: (1) the unique features in the HEVC compresseddomain, such as coding unit and prediction unit, are employedto refine the moving object boundary; (2) the bag of wordsrepresentation in the HEVC compressed domain is applied toclassify the moving persons and vehicles.

B. Overview of the Proposed Method

The overall framework of our system is illustrated in Fig.1. It consists of two stages: moving object segmentation andperson-vehicle classification.For moving object segmentation, firstly, MV interpolationfor intra-coded prediction unit (PU) and MV outlier removal areemployed for preprocessing. Then, blocks with non-zeromotion vectors are clustered into the connected foregroundregions by using the four-connectivity component labelingalgorithm [25]. Finally, object region tracking with temporal consistency is applied to the connected foreground regions toremove the noise regions. The boundary of moving object region is further refined by using the coding unit (CU) and PUsizes of the blocks.

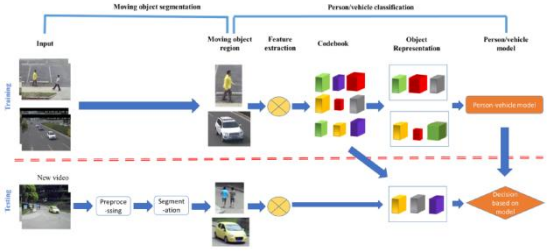


Fig. 1 The proposed framework for HEVC compressed domain moving object segmentation and classification.

For person-vehicle classification, it involves a training phaseto learn the person-vehicle model using "bag of spatiotemporalHEVC syntax words" and a testing phase to apply the learnedmodel to test videos. For the testing phase, we first extract thespatial and temporal information of each 4×4 block to obtainthe feature descriptor. Then, the descriptors of all blocks areclustered into a number of codeword. The foreground object isrepresented by a histogram of the codeword. Finally, for thesegmented moving object, we apply the learned person-vehiclemodel to determine which category to assign.The rest of paper is organized as follows. Section II presentsthe preprocessing steps. Section III presents the HEVCcompressed domain moving object segmentation, tracking andrefinement. The compressed domain object classification usingbag of HEVC syntax words is described in Section IV. SectionV presents the experimental results. Section VI concludes the paper.

II. PREPROCESSING

In HEVC compressed video, one MV is associated with aninter-coded prediction unit (PU). The motion vectors are scaledappropriately to make them independent of the frame type. Thisis accomplished by dividing the MVs according to thedifference between the corresponding frame number and thereference frame number (in display order). For example, oneMV has values (4,4) for reference frame -1 while another MVin a nearby block has values (8,8) for reference frame -2, thesetwo MV values will be corrected to both be (4,4) after thescaling process. For the PU with two motion vectors, the motionvector with larger length will be selected as the representativemotion vector of the PU. In the preprocessing process, the MVinterpolation for intra-coded blocks and MV outlier removal areemployed before the moving object segmentation andclassification.

A. Motion Vector Interpolation for Intra-coded PUs

In order to segment the foreground and background region, itis useful to assign a MV to an intra-coded PU. We propose toselect the representative MV from the MVs of its neighboringPUs as its MV. To be specific, the MVs of first-orderneighboring PUs (top-left, top, top-right, left, right, bottom-left,bottom, bottom-right) are employed. Fig. 2 shows an exampleof an intra-coded PU together with its first-order neighboringPUs. In Fig. 2, MVs of all neighboring PUs are collected andstored in MVList. Since one of the neighboring PUs is intracoded,MVListcontains seven vectors, which is shown asfollows:. (1)After MVListis constructed, the next step is to assign arepresentative MV for the intra-coded PU from MVList. IntracodedPUs usually occur when there is a large motion in thescene. Therefore, we propose to choose the maximum MV fromMVListas the MV of the current intra-coded PU. Specifically,when all the first-order neighboring PUs are encoded with intramode, in order to obtain the non-zero MV within the nearby

$$MVList = (MV_1, MV_2, MV_3, MV_4, MV_5, MV_6, MV_7) \quad (1)$$

region, we extend the range of neighborhood to 16x16 (blocks),which is set empirically as being optimal.

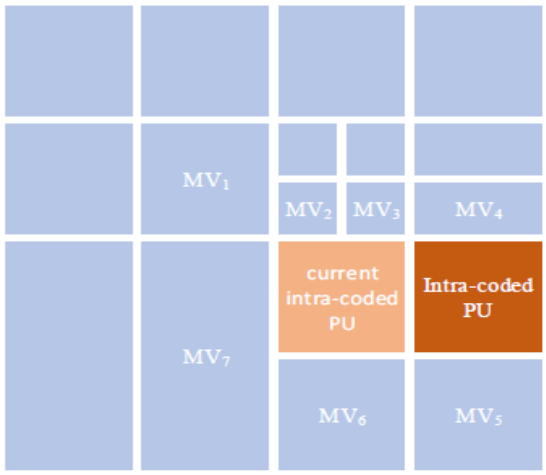


Fig. 2.MV assignment for an intra-coded PU. One of the first-order neighboringPUs is also intra-coded, and the remaining variably sized neighboring PUs haveMVs.

B. MV Outlier Removal

Because the MVs from the compressed bitstream aredetermined in terms of rate-distortion, they may not representthe true object motion. In other words, MV fields may be noisy.In this section, we propose to reduce the motion noise byreferring to motion continuity over time and motion coherencewithin the spatial neighborhood. Three steps are included in theMV outlier removal, which are MV filtering, MV refining, andisolated and small MV removal.

MV Filtering: The original MVs are filtered along the temporal direction.To be specific, the original MVs at the co-located position inthe m previous frames and m following frames are employed tofilter the original MVs at current frame t. Since the CU and PUsizes at the same position may be different among differentframes, MV filtering is operated at 4x4 block level, which is theminimum size of PU. Let $MV_t^x(k,l)$ and $MV_t^y(k,l)$ representthe original MVs along the horizontal direction and verticaldirection for a 4x4 block at position (k,l) at frame t. Then, thefiltered MVs: $MV_t^x(k,l)'$ and $MV_t^y(k,l)'$ can be estimated by

$$MV_t^x(k,l)' = floor\left(\frac{\sum_{i=t-m}^{t+m} MV_i^x(k,l)}{2m+1}\right) \quad (2)$$

$$MV_t^y(k,l)' = floor\left(\frac{\sum_{i=t-m}^{t+m} MV_i^y(k,l)}{2m+1}\right) \quad (3)$$

wherefloor(x) represents the operation to get the largest integerless than or equal to x. In the experiment, m is empirically setto 4.

MV Refining: Since the moving objects in the previous frames andfollowing frames have a displacement relative to the movingobject in current frame, the filtering process using co-locatedblocks in the neighboring frames may cause a few non-zero MVnoise adjacent to the moving object boundary in current frame.Although the original MVs may be noisy, but if the original MVof current block and most of its neighboring blocks are bothzero, current block has a high probability to belong tobackground. Figs. 3 shows an example of the original MVs andtheir associated filtered MVs on frame #53 of the Hall Monitorsequence. In Figs.3 (a) and (b), the blocks with non-zero MVsand intra mode are marked with green border and red borderrespectively. As it is shown, the blocks marked with orangecircles belongs to background. The original MVs of theseblocks and most of their neighboring blocks are zero MV. Buttheir associated filtered MVs become non-zero MV.

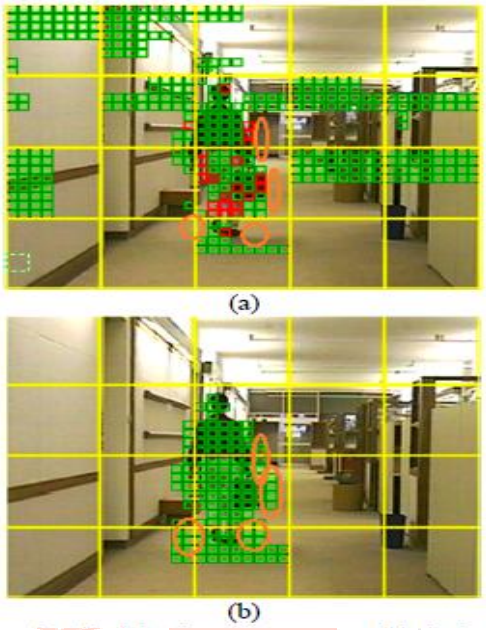


Fig.3. (a) an example of original motion vectors; (b) their associated filteredmotion vectors.

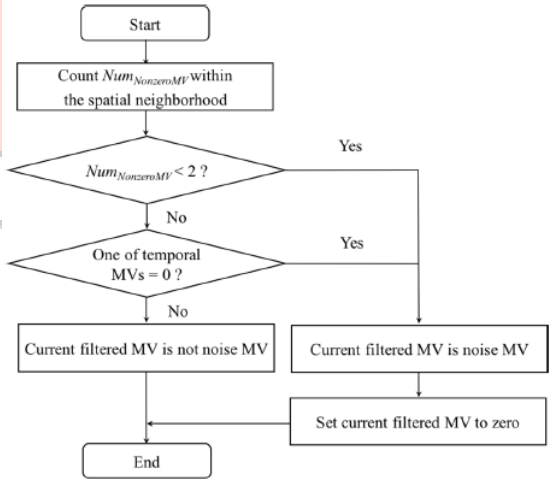


Fig. 4.MV refining for the filtered non-zero MVs.

Therefore, we can use the original MVs to reduce these nonzeroMV noise for the filtered MVs. According to the spatialcompactness and temporal continuity of the MVs, the originalMVs of spatial neighboring PUs and temporal co-located Pusare both involved to make a joint judgment. When the spatialneighboring PU of current PU is out of the frame, the originalMV of that PU will be set to zero. The flowchart of the MVrefining for filtered non-zero MVs is illustrated in Fig. 4. InFig.4, NumNonzeroMVdenotes the number of non-zero MVs withinthe spatial neighborhood, where top PU, bottom PU, left PU andright PU are considered. First, if NumNonzeroMVis less than 2, weassume the condition of spatial compactness is not

satisfied.Second, we check whether the MVs in the co-located temporalPUs from the previous frame and following frame are non-zeroor not. If one of the MVs is zero, we assume the condition oftemporal continuity is not satisfied. If the condition of thespatial compactness or the condition of temporal continuity isnot satisfied, the no-zero filtered MV will be marked as noisemotion and set to zero.

Isolated and Small MV Removal: For a foreground moving object, it usually has a connectednon-zero MV region and a relatively larger filtered MV, so thePUs with isolated non-zero MVs or smaller MVs have a highprobability to be the background PUs. Therefore, we proposeto label the PUs with isolated non-zero MVs or small MVs asthe background PUs. To be specific, we define one MV as anisolated MV when all the MVs of its spatial neighborhood arezero MV. In addition, we define one MV as a small MV whenthe MVs of current PU and more than half of its spatialneighboring PU are less than or equal to 1. If one PU isidentified as the PU with isolated or small MV, its associatedMV will be modified to zero.

III. MOVING OBJECT SEGMENTATION IN HEVC COMPRESSEDDOMAIN

After the preprocessing of the MVs, as described in SectionII, blocks with non-zero MVs are marked as foreground blocks.These foreground blocks are clustered to the connectedforeground regions using the four-connectivity componentlabeling algorithm [25]. For each foreground region, firstly, weexamine its temporal consistency by using object regiontracking. Secondly, we refine the boundary of moving objectregion by using CU and PU sizes of the blocks.

A. Object Region Tracking

In order to examine the temporal consistency of foregroundregions, these foreground regions are temporally tracked byusing the MVs extracted from HEVC compressed domain. Forthe ith foreground region O_t^i at frame t, if we find that itscorresponding object region continuously describes the sameobject in backward direction (from t to t - 4) and in forwarddirection (from t to t + 4), then we assume that currentforeground region satisfies the condition of temporalconsistency. Otherwise, this foregroud region will be labeled asthe noise region and removed from frame t.The flowchart of the object region tracking in backwarddirection is illustrated in Fig. 5, which is composed of thefollowing five main steps.

Step 1: variable bTemporalis used to indicate whether the corresponding foreground region of current foreground regioncontinuously describes the same object in backward directionor not.

Step 2: each 4x4 block $B_t(k,l)$ in the foreground regionis projected to the previous frame t - 1 by using its MV (MV^x, MV^y). Its projected position (k',l') in frame t - 1 is

$$(k',l') = (k,l) + (MV^x, MV^y) \quad \textbf{(4)}$$

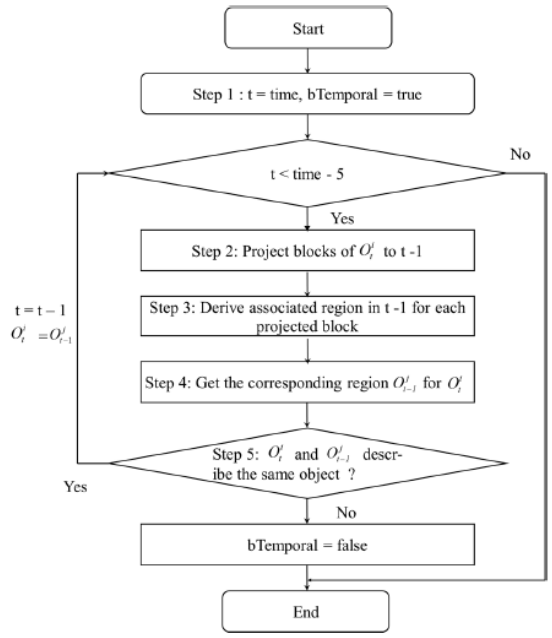


Fig. 5 The flowchart of object region tracking in backward direction.

Step 3: for each projected block $B_{t-1}(k',l')$ at time t - 1, if $B_{t-1}(k',l')$ belongs to background region, the region for $B_{t-1}(k',l')$will be set to ϕ.

Step 4: the number of projected blocks in each foregroundregion O_{t-1}^j at frame t – 1 is counted. Then, the foregroundregion O_{t-1}^j with the most projected blocks will be identified asthe corresponding foreground region for O_t^i at frame t.Step 5: the intersection of O_t^i and its correspondingforeground region O_{t-1}^j is computed and compared with anempirical threshold. In (5), size(object) denotes the number offoreground blocks in region Object and is empirically set to50%. If (5) is not satisfied, bTemporalis set to be false and thisprocess is finished. Otherwise, we assume and itscorresponding foreground region is assumed to describethe same object region, and then return to step 1 and repeat thewhole process.

$$\frac{size(O_t^i \cap O_{t-1}^j)}{size(O_t^i)} > \partial \quad \textbf{(5)}$$

B. Object Boundary Refinement

Figs. 6(a) and (b) show an example of block partitions fortwo surveillance video frames with moving persons andvehicles. Here, the largest square blocks with yellow border,smaller square blocks with green border and rectangular blockswith pink border represent coding tree unit (CTU), CU and PUrespectively. It is observed that blocks within the movingperson and vehicle region are encoded with smaller CU and PUsizes when compared to CU and PU sizes of the blocks withinthe background region. On the contrary, when the minimumsize of CU and PU in one block row (column) is larger than theaverage size of CU and PU of the moving object region, thisblock row (column) have a high probability to belong to thebackground region. So we propose to refine the object boundaryusing the CU

and PU sizes.In order to describe the object boundary refinement moreclearly, we use CU depth level in this section. The relationshipbetween CU size and depth level is illustrated in Fig.7. Thedepth range of CU is [0, 3]. The size of CTU in Fig. 7 is 64x64and the root of CTU corresponds to depth level 0. The depthlevel for CU size of 32x32, 16x16 and 8x8 are 1, 2 and 3respectively. One CU can be further split into one, two or fourPUs according to the PU splitting type. When the CU is onlysplit into one PU, the depth level for PU is 0; otherwise, thedepth level of PU is 1. Note that all 4x4 blocks inside a CUshare the same CU depth and PU depth. The depth of a 4x4block is defined as the sum of CU and PU depth.

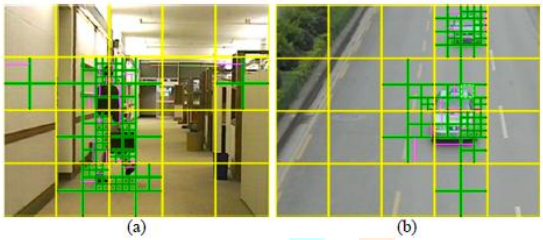


Fig. 6. Block partitions of the moving person and vehicles.

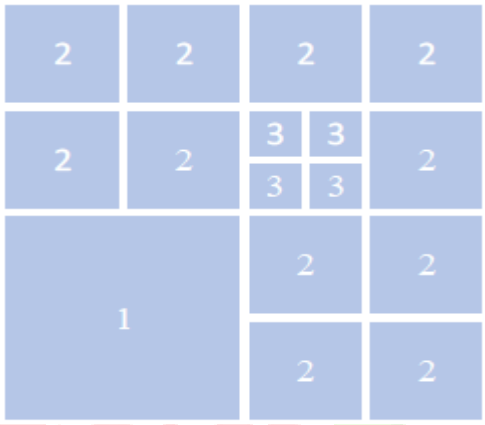


Fig. 7.The relationship between the depth level and CU size.

Our object boundary refinement is operated on the 4x4 blocklevel, and the main steps of object boundary refinement isdescribed as follows.

Step 1: the average depth of the foreground region iscalculated. To be specific, let H0 and H1 denote the leftboundary and right boundary of the foreground region, V0 andV1 denote the top and bottom boundary of the foregroundregion, CUdepth(k, l) denotes the CU depth of the block atposition (k, l), PUdepth(k, l) denotes the PU depth of the blockat position (k, l), depth(k, l) denotes the depth of the block atposition (k, l) and can be computed by (6), foregBlockNumdenotes the number of blocks within the foreground region, andfloor(x) represents the operation to get the smallest integerlarger than or equal to x. Thus, the average depth of theforeground region is computed by (7).

$$depth(k,l) = CUdepth(k,l) + PUdepth(k,l) \quad \textbf{(6)}$$

$$AvgDepth = floor\Big(\sum_{\substack{V0\le k\le V1 \\ H0\le l\le H1}} \frac{depth(k,l)}{foregBlockNum} + 0.5\Big) - 1 \quad \textbf{(7)}$$

Step 2: the maximum depth for each block row within themoving object region is calculated and compared withAvgDepth. Formally, let MaxDepth(k) denote the maximum
depth in block row k, and it is computed by

$$MaxDepth(k) = \max_{H0\le l\le H1} (depth(k,l)) \quad \textbf{(8)}$$

If (9) is satisfied, this block row will be marked as the candidateto be modified to background blocks.

$$MaxDepth(k) \le AvgDepth \quad \textbf{(9)}$$

Step 3: for each candidate block row, we need to furthercheck whether they are real background blocks or not. Formally,let IndexMin(k) denote the index of the block with the minimumdepth in row k, it is computed by

$$IndexMin(k) = \arg\min_{l} (depth(k,l)) \quad \textbf{(10)}$$

The block with IndexMin(k) is referred as the index block inrow k. The CU which contains the index block is referred as theindex CU. Then we get the top CU and the bottom CU of theindex CU. After that, the maximum block row number in thetop CU and the minimum block row number in the bottom CUare denoted as MaxRowTopand MinRowBottomrespectively.If MaxDepth(MaxNumTop) or MaxDepth(MinRowBottom) issmaller than AvgDepth, the candidate row k has a highprobability to belong to the background blocks. To be specific,if (11) or (12) is satisfied, we assume candidate row k belongsto background and need to be modified to background blocks.

$$MaxDepth(MaxRowTop) \le AvgDepth \quad \textbf{(11)}$$

$$MaxDepth(MinRowBottom) \le AvgDepth \quad \textbf{(12)}$$

Figs. 8 shows an example of the object boundary refinementon frame #27 of the Hall Monitor sequence. The AvgDepthofthe moving object region in Figs. 8(a) is 2. As illustrated in Figs.8(b), the rows belonged to region 1 and region 2 satisfy (9), sothe rows within region 1 and region 2 are marked as thecandidates. In Figs. 8, the rows within region 1 neither satisfy(11) nor (12), so the rows within region 1 are foreground blocks.The rows within region 2 satisfy (12), so they are realbackground blocks and need to be removed from the foregroundregions. After the boundary refinement, the moving objectregion are illustrated in Figs. 8(c).After checking each block row within the moving object, weflip the moving object region by 90 degree and use the sameway to check each block column.

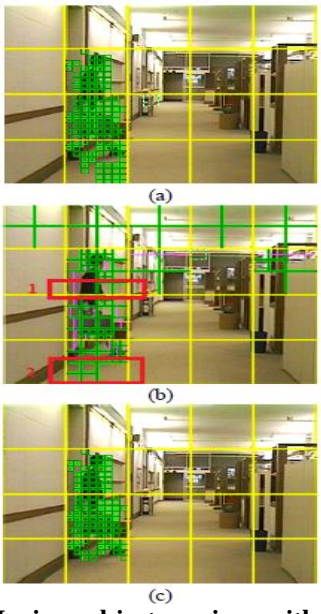


Fig. 8 (a) Moving object region without boundary refinement; (b) CU and PUsizes of the whole frame; (c) Moving object region after boundary refinement.

IV. MOVING OBJECT CLASSIFICATION IN HEVC COMPRESSEDDOMAIN

For object classification in surveillance videos, we aim toclassify the segmented moving objects into persons andvehicles using "bag of HEVC syntax words" in HEVCcompressed domain. The "bag of words" representation hasbeen successfully used for object classification in the pixeldomain [26-27]. R. V. Badu et al. [28] propose to use "bag ofwords" representation in H.264/AVC compressed domain toclassify the video content. The major contribution of this workis to establish a bag of words model in the HEVC domain formoving object classification.This proposed object classification has the following major

Steps:

- describing each coding block within the movingobject region using HEVC syntax features;
- constructing acodebook using a clustering method;
- representing eachmoving object using a normalized histogram of codeword fromthe codebook; and
- training a binary classifier to classify themoving objects into persons and vehicles. The main challengeis to select effective features in the HEVC compressed domain.

In this work, we have identified three types of features, thelength of motion vectors, prediction modes, and motion vectordifference (MVD), as effective features for our object classification. The length of the motion vector relates to the velocity of the object, which is a simple yet important featurefor discriminating persons and vehicles. This is becausevehicles usually move faster than persons. It is observed thatpersons often undergo non-rigid deformations, it is harder tofind a good match for each PU within the region of movingpersons. As a result, more blocks within the region of personsare coded with intra modes when compared to the blocks withinthe region of moving vehicles. Therefore, the prediction modecan be utilized as an effective feature. For example, as shownin Figs. 9 (a) and (b), blocks with red borders are encoded withintra mode, blocks with green borders are encoded with intermode and non-zero MVs, and other blocks are encoded with
inter mode and zero MVs. We can see that more blocks arecoded with intra modes within the moving person. Formally,
the prediction mode of current block at frame t is denoted as CurrModet.Because the MVs within vehicles are moreconsistent than those within persons, the MVD betweenneighboring blocks within the region of vehicles are usuallysmaller then that within the region of persons. Since we focuson the motion variation within the moving objects, the MVD ofcurrent block and its neighboring blocks is calculated onlywhen they both have non-zero MVs. Specifically, MVD ofcurrent block is computed by

$$CurrMVD_t^i = \begin{cases} 0, & if\ NeighMV_t^i \text{ or } CurrMV=0 \\ abs(NeighMV_t^i - CurrMV_t), & else \end{cases} \quad \textbf{(13)}$$

$$MaxCurrMVD_t = max(CurrMVD_t^i) \quad \textbf{(14)}$$

where $NeighMV_t^i$ denotes the MV of the ith neighboring blockat frame t, CurrMVtdenotes the MV of the current block atframe t, $CurrMVD_t^i$ denotes the MVD between current blockand the ith neighboring block, and MaxCurrMVDtdenotes themaximum MVD between current block and its neighboringblocks at frame t. In addition, the maximum MVD of the collocated block at frame t-1 and t+1 are also used as the featuresof current block, which are denoted as MaxCurrMVDt-1 andMaxCurrMVDt+1 respectively.

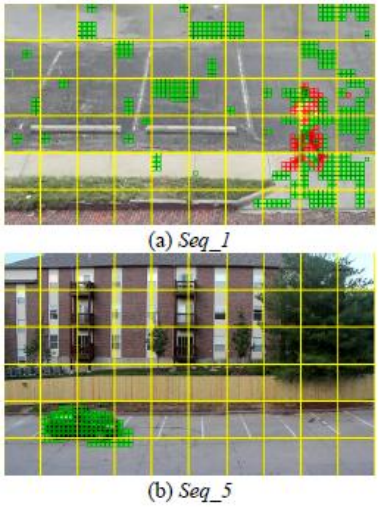


Fig. 9. The distribution of intra mode within the moving objects (blocks withred borders are encoded with intra mode, blocks with green borders are encodedwith inter mode and non-zero MVs, and other blocks are encoded with intermode and zero MVs).

The features are summarized in Table I. The total feature sizeis 5. Once the features of all blocks in the training dataset havebeen extracted, these feature vectors are clustered into Mclusters by using k-means clustering method. The center of eachcluster becomes a codeword. In total, the codebook will have Mcodewords. For example, in our experiments, we set M = 25.Each moving object, either a person or a vehicle, will contain alarge number of blocks. We compute the L2-norm distancebetween the feature vector of each block Bnand each codewordCm . We find the

codeword which has the minimum distance toBnand cast Bnto the bin of this codeword. In this way, wegenerate a codeword histogram for all blocks in the object.After normalized by its size, the histogram is used as the featureto describe the moving object. With this feature descriptionscheme and the training data, we train a binary linear SVMclassifier for person-vehicle classification. Our objectclassification algorithm is summarized in Algorithm 1.

TABLE I: Each Component Of The Feature Vector For Object Classification

Feature name	Definition	Feature Size
Motion vector	$CurrMV_t$	1
Prediction mode	$CurrMode_t$	1
MVD	$MaxCurrMVD_t$	1
	$MaxCurrMVD_{t-1}$	1
	$MaxCurrMVD_{t-1}$	1

Algorithm 1: Moving Object Classification

Algorithm 1: Moving Object Classification

Input: Bounding box of the moving object, pre-trained codebook and classification model

Output: Classification result of the moving object, either person or vehicle

Begin

1. For each 4x4 image patch within the bounding box, form its feature vector according to Table I, and find the nearest codeword in the pre-trained codebook.
2. Generate a codeword histogram for all image patches within the bounding box.
3. Normalize the histogram by its size.
4. Use the pre-trained classification model to classify the moving object into person and vehicle.

End

V. EXPERIMENTAL RESULTS

In order to train the person-vehicle model for moving objectclassification, 4 training sequences are used, which areillustrated in Figs. 10. To evaluate the performance of ourproposed moving object segmentation and classificationscheme in HEVC compressed domain, we have collected 2sequences from CDNet2012 dataset (Highway and Pedestrians),1 sequence from H.264/AVC standard sequence (Hall Monitor)and 6 sequences from our dataset.

4 of the test sequences havemore than one objects in one frame, which are Highway, Seq_3,Seq_4, and Seq_6. In addition, there are persons and vehiclespresent in one frame in Seq_6. Example frames of the testvideos are shown in Figs. 11 and Figs. 12. The resolutions andnumber of frames for the training and test videos are illustratedin Table II and Table III. Both the training and testing videosare encoded using the HEVC HM v10.0 encoder, at variousbitrates, with the GOP structure IBBBB, i.e., the first frame iscoded as intra (I), and subsequent frames are coded asgeneralized B frames. HEVC syntax features, such as motionvectors, prediction modes, CU sizes, and PU types, areextracted from HEVC compressed bitstream.

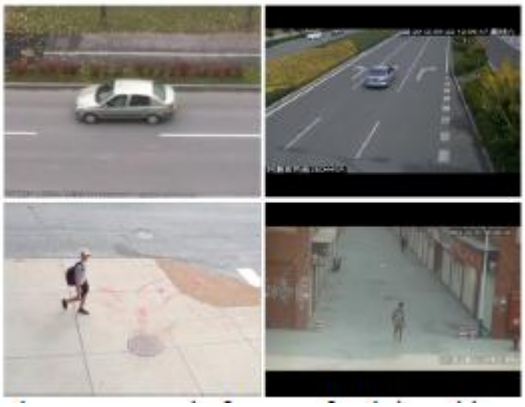
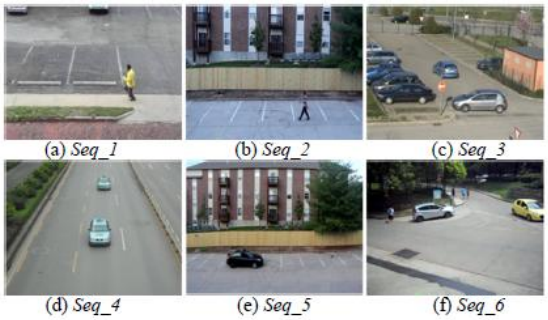


Fig. 10. Example frames of training videos.

(a) *Hall Monitor* (b) *Highway* (c) *Pedestrians*

Fig.11. Example frames of test videos from public dataset.



(a) *Seq_1* (b) *Seq_2* (c) *Seq_3*

(d) *Seq_4* (e) *Seq_5* (f) *Seq_6*

Fig. 12. Example frames of test videos from our dataset.

TABLE II: Resolutions And Number Of Frames For Each Training Sequence

Sequence	Resolution	Number of frames
Training_Seq_1	352x288	100
Training_Seq_2	640x480	200
Training_Seq_3	640x480	100
Training_Seq_4	640x480	100

TABLE III: Resolutions And Number Of Frames For Each Test Sequence

Sequence	Resolution	Number of frames
Hall Monitor	352x288	60
Highway	320x240	1100
Pedestrians	360x240	700
Seq_1	640x480	100
Seq_2	1920x1080	200
Seq_3	320x256	100
Seq_4	640x480	100
Seq_5	1920x1080	150
Seq_6	640x480	200

Fig. 13 shows several examples of the moving objectsegmentation results using our proposed method. As it is shown,the moving person or vehicle can be well detected andsegmented when they are not close to each other whereas themoving person and vehicle will be segmented as one wholeobject when they are close to each other.

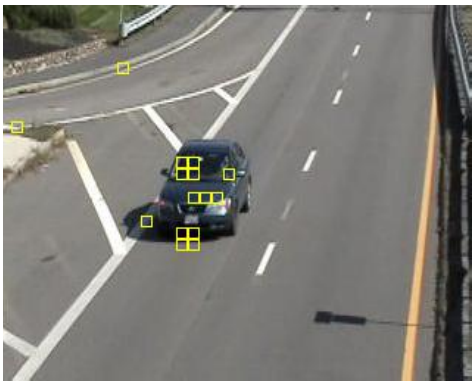


Fig. 13.aPreprocessing.

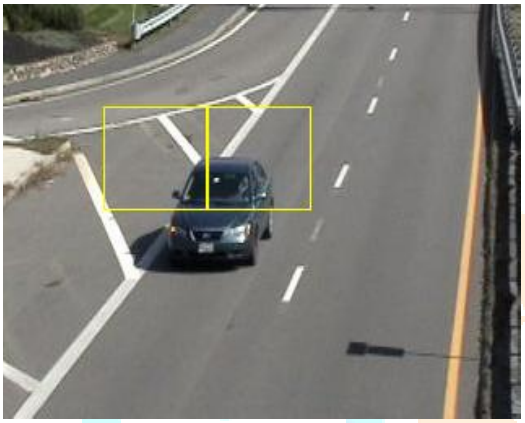


Fig. 13.bObject region tracking

Fig. 13.cObject Boundary Refinement

Fig. 13.dObject Segmentation

Fig. 13.eOutput

Moreover, the segmentation accuracy is measured bycomparing the segmented foreground and background blockswith the ground truth labels for each frame of the test sequences.Specifically, the proposed moving object segmentationalgorithm is evaluated in terms of precision, recall and Fmeasure,which are defined in (15) ~ (17). The notations TP,FP and FN are the total number of true positives, false positives,and false negatives respectively. Precision is defined as thenumber of TP divided by the total number of labeled 4x4 blocks.Recall is defined as the number of TP divided by the totalnumber of ground truth labels. F-measure is the harmonic meanof precision and recall.

$$Precision = \frac{TP}{TP + FP} \quad (15)$$

$$Recall = \frac{TP}{TP + FN} \quad (16)$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (17)$$

Table IV ~ Table VI summarize the performance of proposedsegmentation algorithm in terms of Precision, Recall and Fmeasurefor various values of QP between 22 and 32. It can beseen that the segmentation performance remains consistent fordifferent QPs. In addition, for different sequences, thesegmentation precision varies a lot. This is due to the differentdegree of motion vector noise produced from the encodingprocess for different sequences. As shown in Figs. 9, the motionvector noise of Seq_1 sequence is much larger than that ofSeq_5 sequence. As a result, the segmentation performance ofSeq_1 is relatively lower than Seq_5.Table VII compares the proposed method with the methodsof [20], [14], and [10] in terms of precision, recall, and Fmeasureon Hall Monitor sequence. Note that the methods of[20], [14] and [10] are performed on H.264/AVC bitstreamwhereas our proposed method is performed on HEVC bitstream.This is because little research has been done on HEVCbitstream. When compared to these methods, our method canachieve similar segmentation performance as [20], which hasthe best performance among these three methods.

TABLE IV: Precision (In Percentage) For Various Qp Values

QP	Hall Monitor	Highway	Pedestrians	Seq_1	Seq_2	Seq_3	Seq_4	Seq_5	Seq_6	Avg.
22	57.4	79.0	64.1	57.7	70.1	72.6	54.5	87.0	68.5	67.9
27	61.8	75.6	60.6	70.6	66.3	68.2	55.5	84.1	69.3	68.0
30	65.2	72.3	58.4	72.9	66.1	66.1	55.3	80.3	67.7	67.1
32	67.1	70.0	56.1	70.1	64.3	63.4	56.5	77.0	67.2	65.7

TABLE V: Recall (In Percentage) For Various Qp Values

QP	Hall Monitor	Highway	Pedestrians	Seq_1	Seq_2	Seq_3	Seq_4	Seq_5	Seq_6	Avg.
22	92.2	90.2	80.0	94.1	92.0	97.0	98.4	93.0	93.5	92.3
27	88.8	89.8	83.2	90.8	92.1	96.1	96.6	93.3	92.7	91.5
30	86.9	89.8	81.4	88.8	90.2	95.5	94.6	94.1	89.6	90.1
32	85.4	90.8	79.9	88.3	89.1	92.9	93.4	93.5	90.1	89.3

TABLE VI: F-Measure (In Percentage) For Various Qp Values

QP	Hall Monitor	Highway	Pedestrians	Seq_1	Seq_2	Seq_3	Seq_4	Seq_5	Seq_6	Avg.
22	70.8	84.2	71.2	71.5	79.6	83.0	70.1	89.9	79.1	77.7
27	72.9	82.1	70.1	79.4	77.1	79.8	70.5	88.5	79.3	77.7
30	74.5	80.1	68.0	80.1	76.3	78.1	69.8	86.7	77.1	76.7
32	75.2	79.1	65.9	78.2	74.7	75.4	70.4	84.5	77.0	75.6

At the meantime, the running speed of our segmentationmethod is much faster. For the test video with resolution352x288, the processing speed of the method [20] is about 10frames per second (fps), whereas the processing speed of ourproposed method is over 400 fps. Note that our system isimplemented by C language whereas the system of [20] is
implemented by matlab. The processing speed of our proposedmethod for each sequence is illustrated in Table VIII. Theseresults were obtained on a 2.30 GHz Intel Core i3 CPU with 8GB RAM.

TABLE VII: Comparison Of Several Methods In Terms Of Precision, Recall And F-Measure For Hall Monitor Sequence

Method	Precision	Recall	F-measure
Proposed	63.7	87.9	73.8
[20]	72.8	82.4	78.1
[14]	27.9	91.9	37.3
[10]	15.6	90.1	22.9

TABLE VIII: Running Speed Of The Proposed Moving Object Segmentation

Sequence	Resolution	Running Speed (fps)
Hall Monitor	352x288	437
Highway	320x240	645
Pedestrians	360x240	562
Seq_1	640x480	263
Seq_2	1920x1080	74
Seq_3	320x256	549
Seq_4	640x480	270
Seq_5	1920x1080	83
Seq_6	640x480	258

Table IX shows the comparison between the proposedsegmentation method (preprocessing + moving object trackingand objection boundary refinement) and the proposed methodonly with moving object tracking and object boundaryrefinement (without preprocessing) for all test sequences interms of F-measure. As it is shown, the preprocessing processimproves the F-measure accuracy about 14% for different QPs.In addition, Table X shows the comparison between theproposed segmentation method and the proposed segmentationonly with preprocessing (without moving object tracking andobject boundary refinement) for all test sequences in terms ofF-measure. The moving object tracking and object boundaryrefinement improves the F-measure accuracy about 5% fordifferent QPs.

TABLE IX: Comparison Of The Proposed Method And The Proposed Method Without Preprocessing In Terms Of F-Measure

QP	Proposed segmentation	Proposed method without preprocessing process
22	77.7	61.2
27	77.7	63.1
30	76.7	63.8
32	75.6	64.6

TABLE X: Comparison Of The Proposed Segmentation And The ProposedSegmentation With Only Preprocessing In Terms Of F-Measure

QP	Proposed segmentation	Proposed segmentation with only preprocessing
22	77.7	73.5
27	77.7	72.6
30	76.7	71.9
32	75.6	70.5

TABLE XI: Accuracy Of The Proposed Classification Algorithm

QP	Hall Monitor	Highway	Pedestrians	Seq_1	Seq_2	Seq_3	Seq_4	Seq_5	Seq_6	Avg.
22	94	71	88	99	93	85	84	89	89	88
27	96	76	94	99	94	91	94	95	86	92
30	94	80	83	96	85	94	94	87	84	89
32	91	81	87	94	86	95	93	95	88	90

Table XI shows the performance of person-vehicleclassification results on different test videos and quantizationsettings. For performance evaluations, we manually label eachmoving object, either a person or a vehicle, as the ground truth.The moving object region containing just one kind of object isused for classification whereas the moving object regioncontaining both the person and vehicle is not used forclassification. In addition, the detected regions, which belongto background but falsely detected as moving object regions,are not used for classification. In total, there are 1187 personsand 1415 vehicles. We use the accuracy as the performancemetric, which is defined as

$$Accuracy = \frac{TP}{TP + FP} \quad \textbf{(18)}$$

Here, TP and FP are true positive and false positive rates,respectively. We can see that the overall classification accuracyof the proposed method is about 90% for more than 2500moving objects. Furthermore, our system achieves consistentperformance across different QPs.Since the input of our classification algorithm is the movingobject region extracted from the video frame by oursegmentation algorithm, the classification accuracy of ourclassification algorithm depends on the segmentation accuracyof our segmentation algorithm. In Seq_3 and Seq_4, the objectsare not close to each other, our classification algorithm canclassify them with about 90% accuracy. In Highway sequence,the vehicles are near to each other, our segmentation algorithmfail to segment them correctly, which is illustrated in Figs. 13(c),this increases the difficulty for our classification algorithm.Even though, our algorithm can also classify them with about80% accuracy.The average

running speed of the proposed segmentation andclassification scheme for each sequence is illustrated in TableXII. Please note that the decoding time of MV and theassociated mode information is not included. As it is shown, ourproposed scheme can achieve more than 200fps for 640x480sequence.

Table XIII compares our proposed method against pixeldomainSVM classifier for person-vehicle classification usingthe same detected bounding boxes. In the pixel-domain SVMobject classification, histograms of oriented gradients (HOG)[29] feature is used to describe each 4×4 patch within themoving object region. Then the moving object region isrepresented by "Bag of Words". We can see that our proposedmethod achieves comparable performance with the pixeldomain SVM classifier. When compared to pixel domain SVMclassification, our method directly uses the information fromthe bitstream and do not need the full decoding of the bitstream,which is a computation-intensive task.To verify the performance improvement of our proposedfeatures in "bag of temporal-spatial words", we refer the systemonly including the motion vector of current block in "bag oftemporal-spatial words" representation as the baseline systemsince the motion vector is the straight-forward butdiscriminative feature to distinguish person and vehicles. TableXIII also compares our proposed system with the baselinesystem. It is revealed that the baseline system fails todistinguish the persons and vehicles when persons walk fast orvehicles move slowly, such as the fast-moving person in Seq_1and Seq_2, and the slow-moving vehicles in Seq_5. This isbecause that most vehicles moves faster than persons in thetraining dataset and the model trained from the dataset willmisclassify the fast moving persons as vehicles. Whencompared to baseline system, our proposed system candistinguish these persons and vehicles robustly and efficiently.

TABLE XII: Processing Speed Of Proposed Segmentation And Classification Method

Sequence	Resolution	Running Speed (fps)
Hall Monitor	352x288	389
Highway	320x240	533
Pedestrians	360x240	457
Seq_1	640x480	241
Seq_2	1920x1080	62
Seq_3	320x256	435
Seq_4	640x480	233
Seq_5	1920x1080	71
Seq_6	640x480	233

TABLE XIII: Comparison Of Our Proposed Method With Pixel Domain SVM Classifier And The Baseline System For QP = 27

Sequence	Proposed	Pixel domain SVM classifier	Baseline system
Hall Monitor	96	97	91
Highway	76	93	61
Pedestrians	94	96	88
Seq_1	99	96	15
Seq_2	94	99	17
Seq_3	91	95	94
Seq_4	94	95	76
Seq_5	95	98	37
Seq_6	86	95	56
Avg.	**92**	**96**	**60**

VI. CONCLUSION

In this paper, we have presented a novel approach to segmentand classify the moving objects from HEVC compressedsurveillance video. Only the motion vectors and the associatedcoding modes from the compressed stream are used in theproposed method. In the proposed method, firstly, MVinterpolation for intra-coded PU and MV outlier removal areemployed for preprocessing. Secondly, blocks with non-zeromotion vectors are clustered into connected foreground regionsby the four-connectivity component labeling algorithm. Thirdly,object region tracking based on temporal consistency is appliedto the connected foreground regions to remove the noise regions.The boundary of moving object region is further refined by thecoding unit size and prediction unit size. Finally, a personvehicleclassification model using bags of spatial-temporalHEVC syntax words is trained to classify the moving objects,either persons or vehicles. The proposed method has a fairlylow processing time, yet still provides high accuracy.

VII. REFERENCES

[1] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchicalgraph-based video segmentation," in Proc. IEEE Conf. Comput. Vis. AndPattern Recognit., pp. 2141–2148, Jun. 2010.

[2] S. Chien, W. Chan, Y. Tseng, and H. Chen, "Video object segmentationand tracking framework with improved threshold decision and diffusiondistance," IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 6, pp.921–634, Jun. 2013.

[3] H. Sakaino, "Video-based tracking, learning, and recognition method formultiple moving objects", IEEE Trans. Circuits Syst. Video Technol., vol.14, no. 5, pp. 1661–1674, Oct. 2013。

[4] "Generic Coding of Moving Pictures and Associated Audio Information- Part 2: Video," ITU-T and ISO/IEC JTC 1, ITU-T RecommendationH.262 and ISO/IEC 13 818-2 (MPEG-2), 1994.

[5] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview ofthe H.264/AVC video coding standard," IEEE Trans. Circuits Syst. VideoTechnol., vol. 13, no. 7, pp. 560–576, Jul. 2003.

[6] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of theHigh Efficiency Video Coding (HEVC) standard", IEEE Trans. CircuitsSyst. Video Technol., vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[7] R. V. Babu, M. Tom, and P. Wadekar, "A survey on compressed domainvideo analysis techniques," Multimedia Tools and Applications, vol. 75,no. 2, pp. 1043–1078, Jan. 2013.

[8] R. V. Babu, K. R. Ramakrishnan, H. S. Srinivasan, "Video objectsegmentation: a compression domain approach," IEEE Trans. CircuitsSyst. Video Technol., vol. 14, no. 4, pp. 462–474, Apr. 2004.

[9] S. D. Bruyne, C. Poppe, S. Verstockt, P. Lambert, and R. V. D. Walle,"Estimating motion reliability to improve moving object detection in theH.264/AVC domain," In Proc. IEEE Int. Conf. Multimedia Expo., pp.330–333, Jun. 2009.

[10] W. Zeng, J. Du, W. Gao, and Q. Huang, "Robust moving objectsegmentation on H.264/AVC compressed video using the block-basedMRF model," Real-Time Imaging, vol. 11, no. 4, pp. 290–299, Aug. 2005.

[11] W. Lin, M. Sun, H. Li, Z. Chen, W. Li, and B. Zhou, "Macroblockclassification method for video applications involving motions," IEEETrans. Broadcasting, vol. 58, no. 1, pp. 34–46, Mar. 2012.
[12] Z. Liu, Y. Lu, and Z. Zhang, "Real-time spatiotemporal segmentation ofvideo objects in the H.264 compression domain," J. Visual Commun.Image Represent., vol. 18, no. 3, pp. 275–290, Jun. 2005.
[13] Y. Chen and I. V. Bajic, "A joint approach to global motion estimationand motion segmentation from a coarsely sampled motion vector field,"IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 9, pp. 1316–1328,Sep. 2011.
[14] Y. Chen, I. V. Bajic, and P. Saeedi, "Moving region segmentation fromcompressed video using global motion estimation and Markov randomfields," IEEE Trans. Multimedia, vol. 13, no. 3, pp. 421–431, Jun. 2011.
[15] C. Poppe, S. D. Bruyne, T. Paridaens, P. Lambert, and R. V. D. Walle,"Moving object detection in the H.264/AVC compression domain forvideo surveillance applications," J. Visual Commun. Image Represent.,vol. 20, no. 6, pp. 428–437, Aug. 2009.
[16] F. Porikli, F. Bashir, and H. Sun, "Compression domain Video ObjectSegmentation," IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 1,pp. 2–14, Jan. 2010.
[17] M. Laumer, P. Amon, A. Hutter, and A. Kaup, "Compressed domainmoving object detection by spatio-temporal analysis of H.264/AVCsyntax elements," in Proc. IEEE Conf. Picture Coding Symposium (PCS),pp.282-286, May 2015.
[18] P. Dong, Y. Xia, L. Zhuo, and D. Feng, "Real-time moving objectsegmentation and tracking for H.264/AVC surveillance videos," in Proc.IEEE Int. Conf. Image Processing, Brussels, pp. 11–14, Sep. 2011.
[19] H. Sabirin and M. Kim, "Moving object detection and tracking using aspatio-temporal graph in H.264/AVC bitstreams for video surveillance,"IEEE Trans. Multimedia, vol. 14, no. 3, pp. 657–668, Jun. 2012.
[20] S. H. Khatoonabadi and I. V. Baji´c, "Video object tracking in thecompression domain using spatio-temporal Markov random fields," IEEETrans. Image Process., vol. 22, no. 1, pp. 300–313, Jan. 2013.
[21] J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand,"Comparison of the coding efficiency of video coding standards --including high efficiency video coding (HEVC)", IEEE Trans. CircuitsSyst. Video Technol., vol. 22, no. 12, pp. 1669–1684, Dec. 2012.
[22] H. Li, Y. Zhang, M. Yang, Y. Men, and H. Chao, "A rapid abnormal eventdetection method for surveillance video based on a novel feature incompressed domain of HEVC," in Proc. IEEE Int. Conf. Multimedia andExpo. (ICME), pp.1-6, Jul. 2014.
[23] B. Dey, and M. K. Kundu, "Efficient foreground extraction from HEVCcompressed video for application to real-time analysis of surveillance 'big'data," IEEE Trans. Image Process., vol. 24, no. 11, pp. 3574–3585, Nov.2015.
[24] D. Park, D. Lee, and S. Oh, "Object tracking in HEVC bitstreams,"Journal of Broadcast Engineering, vol. 20, no. 3, pp.449-463, May 2015.
[25] L. He, Y. Chao, and K. Suzuki, "A run-based two-scan labeling algorithm,"IEEE Trans. Image Process., vol. 17, no. 5, pp. 749–756, May 2008.
[26] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. "Visualcategorization with bags of keypoints," In Workshop on statisticallearning in computer vision, ECCV, pp. 1-22, May 2004.
[27] F. Wang, Z. Sun, Y. Jiang, and C. Ngo, "Video event detection usingmotion relativity and feature selection," IEEE Trans. Multimedia, vol. 16,no. 5, pp. 1303–1315, Aug. 2014.
[28] S. Biswas, and R. V. Babu. "H. 264 compressed video classification usinghistogram of oriented motion vectors (HOMV) ," in Proc. IEEE Conf.Acoustics, Speech and Signal Processing (ICASSP), pp. 2040-2044, May2013.
[29] N. Dalal, and B. Triggs, "Histograms of oriented gradients for humandetection," in Proc. IEEE Conf. Comput. Vis. and Pattern Recognit., pp.886-893, Jun. 2005.
[30] M. Moriyama, K. Minemura and K.S. Wong, "Moving object detection inHEVC video by frame sub-sampling," IEEE Int. Symposium on ISPACS,pp. 48-53, Nov. 2015.
[31] S. Giil, J. T. Meyer, T. Schierl, C. Hellge, and W. Samek, "Hybrid videoobject tracking in H.265/HEVC video streams," IEEE Int. conf on MMSP,2016.
[32] S. Pulare and S. S. Tale, "Implementation Implementation of Real TimeMultiple Object Detection and Classification of HEVC Videos," Int.Journal for IRST, vol. 2, no. 11, pp. 248-254, 2016