Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2010, Article ID 658328, 14 pages
doi:10.1155/2010/658328

Research Article

A Joint Watermarking and ROI Coding Scheme for Annotating Traffic Surveillance Videos

Po-Chyi Su and Ching-Yu Wu

Department of Computer Science and Information Engineering, National Central University, Jhongli 32001, Taiwan

Correspondence should be addressed to Ching-Yu Wu, 965202105@cc.ncu.edu.tw

Received 21 March 2009; Revised 5 October 2009; Accepted 6 January 2010

Academic Editor: Alex Kot

We propose a new application of information hiding by employing the digital watermarking techniques to facilitate the data annotation in traffic surveillance videos. There are two parts in the proposed scheme. The first part is the object-based watermarking, in which the information of each vehicle collected by the intelligent transportation system will be conveyed/stored along with the visual data via information hiding. The scheme is integrated with H.264/AVC, which is assumed to be adopted by the surveillance system, to achieve an efficient implementation. The second part is a Region of Interest (ROI) rate control mechanism for encoding traffic surveillance videos, which helps to improve the overall performance. The quality of vehicles in the video will be better preserved and a good rate-distortion performance can be attained. Experimental results show that this potential scheme works well in traffic surveillance videos.

1. Introduction

The research of information hiding or digital watermarking in multimedia data has drawn tremendous attention these years [1]. Information hiding is the technique to embed an imperceptible signal into such host data as digital images and audio or video clips. The close integration of the host signal and the hidden information with unambiguous detection can benefit the applications of copyright protection, steganography, fingerprinting and authentication of digital data, and so forth. It should be noted that different applications will require varying functions of digital watermarking so that a practical design has to take a specific application into account and fine-tune the digital watermark to achieve the objectives in the target scenario.

In this research, we consider the application of managing the data related to traffic surveillance videos in intelligent transportation systems (ITSs). The development of ITS is in need and underway. Advanced ITSs usually employ multiple sensors to gather detailed information about traffic conditions for better traffic flow analysis, incident detection and tracking, and so forth. As there are more and more surveillance cameras deployed along the highways or local roads, the visual information provided by cameras plays an important role in ITS and should be effectively coupled with the information from other sensors to help ensure the safety of people or maintain the traffic order. Nevertheless, managing traffic surveillance videos may require considerable amounts of efforts. First of all, the data volume of a surveillance video is extremely large as the cameras function almost incessantly. In addition, developing an efficient method to find the correspondence between the visual information and the data gathered from different sensors about the same traffic scene may not be a trivial task. Furthermore, there may be many kinds of surveillance camera shots; so describing the scene effectively is not easy either. Therefore, many analyzing and indexing approaches have been proposed for querying traffic surveillance videos [2].

One major contribution of this work is to exploit the digital watermarking techniques for managing traffic surveillance videos in a rather convenient manner. To be more specific, the information related to vehicles, possibly provided by other sensors, will be embedded into the corresponding pixels in the traffic surveillance video. The main advantage is that the vehicle information will be

closely tied with the appearance of the car in the video. We cannot only eliminate the need of managing the extra meta data to describe the scene but also facilitate the information retrieval. Besides, if the information can be embedded effectively via digital watermarking techniques without severely increasing the video data size, the removal of meta data will be an even better motivation. Moreover, the camera- or video-related information can also be embedded into the video to further ensure its authenticity.

It should be noted that digital videos will always be compressed to facilitate data transmission and storage. As practical video codecs are usually lossy compression and have high complexity, the information hiding processes should be integrated with the coding procedures to achieve both the efficiency and reliable digital watermarking. The state-of-the-art video codec is H.264/AVC [3], which makes use of various coding tools to provide enhanced coding efficiency for a wide range of applications. We thus assume that the advanced ITS will adopt H.264/AVC to process the captured traffic scenes and our scheme will be designed under its framework. The proposed H.264/AVC watermarking scheme can be viewed as an object-based methodology since a video frame will be segmented into the background and the foreground, that is, vehicles, for subsequent information embedding and detection. Most of the existing researches on object-based watermarking are related to copyright protection in MPEG-4 videos [4, 5], which explicitly address the object coding. The existing works on digital watermarking in H.264/AVC focus on robust embedding/detection [6, 7], high bit-rate information hiding [8], and the efficiency issues [9]. In our opinion, the robustness of digital watermark in a specific coding standard for annotation purposes may not be required. Nevertheless, the payload should be high enough to carry the appropriate amount of information. The efficient execution is the other important issue since the coding process of H.264/AVC is computationally expensive; so the watermarking procedures should not cause further heavy burden. Moreover, the target bit-rate and video quality should be well preserved to meet the requirements of the applications. Therefore, we have to ensure that the watermark signal be imperceptible and the embedding/detection processes be reliable and efficient for data annotation.

The other contribution of this work is to propose a rate control mechanism tailored to traffic surveillance videos compressed with H.264/AVC. The issues of rate control are important in video compression [10] and such methodologies as operational R-D theory, model-based rate control, and the Rate-Distortion Optimization (RDO) are exploited to achieve good coding performances [11, 12]. In this research, we propose a new model-based rate control mechanism for encoding traffic surveillance videos. Since vehicles appearing in traffic scenes may contain significant information, we set the area covering vehicles as the Region of Interest (ROI) to better preserve its quality. For the ROI-based rate control, how to allocate bits in the Group of Pictures (GOPs), frames, ROI, and non-ROI in a frame may be a more complicated issue. Liu et al. [13, 14] used the Lagrange theory to compute the Quality Parameter (QP) of each Macroblock (MB) and control the complexity of encoding process for low-power mobile devices. Wu and Chen [15] utilized multiple encoders and the relationship between two independent encoders to predict the MB coding mode of the ROI in the video, which helps to maintain the quality of ROI. Li et al. [16] proposed a motion-based rate prediction model, which exploits the feature of Human Visual System (HVS), the prior knowledge of video content and RDO based on Lagrange Multiplier. Agrafiotis et al. [17] proposed a two-stage scheme. The first stage uses the coding result of the first two frames of the current GOP to determine the target buffer level for the remaining P frames in the current GOP. Then the second stage determines the amount of bits for the current P frame. Zheng et al. [18] proposed a so-called Adaptive Frequency Coefficient Suppression scheme, which can adaptively suppress the selective frequency coefficients of 4×4 subblocks in the non-ROI. The saved bits in non-ROI blocks are then reallocated to the ROI to improve its visual quality.

We aim at developing an efficient and accurate bit-rate determination mechanism for traffic surveillance videos. It is worth noting that, in addition to achieving a good rate-distortion performance, there are a couple of other reasons of incorporating the ROI-based rate control mechanism with the proposed digital watermarking scheme. First, the ROI coding can benefit the effective and reliable watermark detection, which will be explained later. Second, one may question that vehicles appearing in the surveillance videos are of great importance; so the changes of pixel values from the watermark embedding may not be appropriate. By using the ROI-based rate control, we can make the quality of the "watermarked vehicles" even better than that in the compressed video without using ROI coding so that the concern of quality degradation in vehicles can be eased.

The rest of the paper is organized as follows. The object-based watermarking is described in Section 2 and the ROI-based rate control mechanism is presented in Section 3. Experimental results are shown in Section 4 to demonstrate the feasibility of the proposed scheme. Concluding remarks will be given in Section 5.

2. The Proposed Digital Watermarking Scheme

The design of our watermarking scheme in H.264/AVC is described in this section. Like the previous video standards, H.264/AVC is based on the motion compensated, DCT-like transform coding methodologies. Each video frame is composed of macroblocks, which are blocks of 16×16 luma samples with the corresponding chroma samples. The macroblocks may be intra- or intercoded. In the intercoding process, the macroblocks are further divided into sub-macroblock partitions of several different sizes for effective motion estimation. In the intracoding process, the spatial prediction based on neighboring decoded pixels in the same slice will be applied. The residual data will be divided into 4×4 subblocks and processed by a spatial transform, which is an approximate DCT and can be implemented with integer operations and a few additions/shifts. The point-by-point multiplication in the transform step will be combined with

(a) (b) (c)

Figure 1: (a) A traffic scene, (b) the constructed background, and (c) the extracted vehicles.

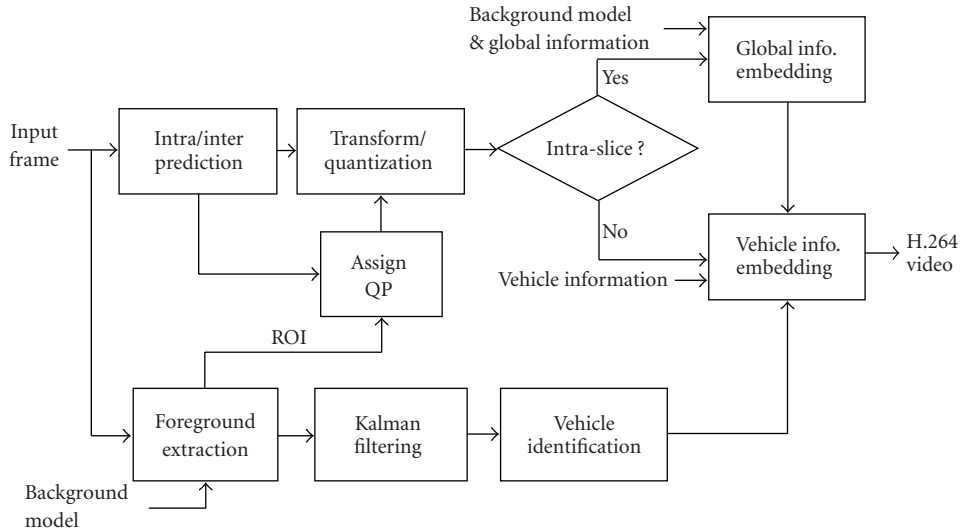


Figure 2: The diagram of information embedder.

the quantization step by simple shifting operations to speed up the execution. Next, we will detail the proposed scheme, which consists of three portions, that is, the analysis of the traffic scene, information embedding, and detection.

2.1. The Analysis of the Traffic Scene. For each incoming frame, we have to identify the vehicles and background for the subsequent processing. The procedures are mostly based on the system proposed by Yoneyama et al. [19]. Since the traffic surveillance cameras are fixed, the stationary background can be constructed by iteratively updating. That is, the background pixels are formed by

$$B_{x,y}^{i+1} = P_{x,y}^{i} \times \alpha M_{x,y}^{b} + B_{x,y}^{i} \times \left(1 - \alpha M_{x,y}^{b}\right), \quad (1)$$

where $P_{x,y}^{i}$ is the luminance pixel of incoming frame, $B_{x,y}^{i}$ and $B_{x,y}^{i+1}$ are the current and updated background pixel, respectively, and α is a small updating weighting factor. A binary mask $M_{x,y}^{b}$ is introduced in (1) to improve the quality of constructed background. To be more specific, $M_{x,y}^{b}$ can help to selectively turn on and off the updating procedure by checking whether the pixel at (x, y) is in the background ($M_{x,y}^{b} = 1$) or in the vehicle region ($M_{x,y}^{b} = 0$). A rough vehicle mask can thus be obtained by subtracting the background image from the captured video frame. Then the morphological operations including opening and closing are applied to remove the isolated noises and group the foreground pixels.

Next, the six-vertex model [19] is used to draw the contour of a vehicle approximately. The model is based on the perspective projection and the assumption that the shadows of vehicles only appear on one side of the cars. Since most vehicles in the scene are moving parallel to the lanes, we use the displacement vector of the vehicle between two frames as the slope of one slanted edge of the six-vertex model. The vertical and horizontal lines are superimposed to cover the rough vehicle mask generated from background subtraction. The other three lines can also be constructed as they are parallel with the three lines we just drew. After forming the six-vertex mask, we further remove the cast shadow according to the vehicle-shadow types defined in [19]. We basically decrease the area of the six-vertex mask by shortening the lengths of the selected two edges. An example

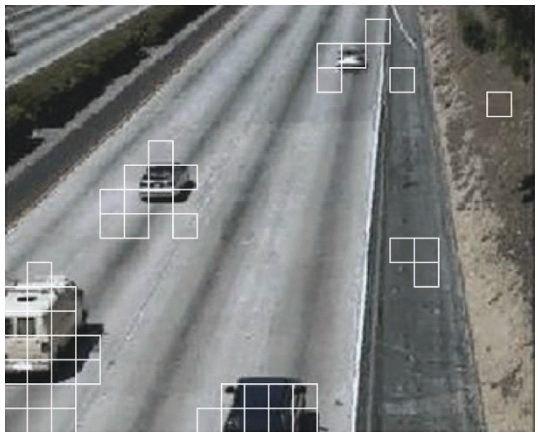


Figure 3: The intracoded macroblocks in a typical intercoded frame of a traffic surveillance video.

of a traffic scene with its constructed background and the extracted vehicle masks is shown in Figure 1.

It should be noted that we also collect other information related to the background, including the traffic lane information and the pixels covering the highway roads, which can be identified by training a few video frames [20]. The background image and the related information are called the background model, which will help us determine the correspondence between the car and its information.

2.2. Information Embedding. With the background model at hand, we can proceed to apply the information embedding. The block diagram of the watermark embedding is shown in Figure 2. The inputs to the information embedder are the captured video frames, the background model, and the information to be embedded. To be more specific, two types of information will be embedded, that is, the global and vehicle information. The global information may specify the data regarding to the camera and/or video, including the serial number of camera and/or video, the date/time of video recording, the sequence number of frames, and even the secure hash of video. By embedding the global information into the compressed video, the authenticity of the recorded video can be further ensured via the unambiguous information extraction. The vehicle information indicates the data of individual car collected by either the sensors or by the visual analysis of the recorded video.

After the vehicles in the input captured frames are extracted, we apply Kalman filtering to track the movement of a vehicle; so the appearance of a vehicle in frames can be identified. The next task is to link the information collected from the sensor to the corresponding vehicle in the video for effective information embedding. We assume that the sensors of ITS and the surveillance camera can obtain the information associated with a vehicle at the same time and that the information provided by the sensors will be available to the watermark embedder immediately. One solution is that each lane will be equipped with a separate sensor/detector and the information gathered in each lane will be matched with the vehicle mask determined in the same lane shown on the video frame. The watermark embedding can then be applied after both the vehicle mask and the associated information are obtained. It should be noted that we use the macroblocks covering the vehicle mask for the watermark embedding/detection to increase the stability of vehicle mask determination.

The vehicle information will be embedded into the quantized residual of 4×4 intracoded subblocks in H.264/AVC. The selection of intracoded subblocks is justified by Figure 3, in which the video is encoded at 350 Kbps and the intracoded macroblocks are highlighted. We can see that most of them cover the moving vehicles. In most of the traffic surveillance videos, the emergence of a car will always result in intracoded macroblocks. Besides, it is quite common that the size of vehicle will become larger or smaller (depending on the location of camera) in consecutive frames and this case violates the assumption of linear movements of a rigid body in motion estimation/compensation mechanism. Therefore, the intracoding is applied quite often in the duration of a vehicle's appearing in video frames.

Since the integrity of surveillance video is important, we take a rather conservative approach of watermark embedding. We will embed one bit information into a selected intracoded subblock by changing at most one quantization index. We may consider only m out of the 16 quantization indices in a 4×4 subblock for watermarking and exclude some low-frequency indices to further ensure a good visual quality. We calculate the sum of the m-selected quantization indices in a subblock numbered k, I^k_{sum}, and then compute $I^k_{\text{sum}}\%2$, where % is the modulo operation. Given that the bit to be embedded is b, one index in the subblock will be chosen to change the value by 1 if $I^k_{\text{sum}}\%2 \neq b$. The indices in a selected subblock will be kept the same when $I^k_{\text{sum}}\%2 = b$. A subblock will be skipped if the m-considered indices are all equal to 0. Besides, we also have to avoid generating a watermarked subblock with all the m-considered indices equal to 0 to maintain the synchronization between the watermark embedder and detector.

2.3. The Selection of Indices. When the data modification is necessary, we need to select a suitable quantization index. Since the 4×4 spatial transform adopted in H.264/AVC is closely related to DCT, we employ Watson's perceptual model [21] to guarantee the invisibility of the watermark. To be more specific, Watson's model helps in determining the maximum allowable change of coefficient value, that is, Just Noticeable Difference (JND). The model basically takes two masking effects into account, that is, the luminance masking and contrast masking. The luminance masking refers to the dependency of the visual threshold and the mean luminance of the local image region while the contrast masking indicates that the threshold for a visual pattern would be reduced in the presence of other patterns. For a subblock numbered k, the luminance-adjusted threshold is then formed by

$$a^k_{i,j} = t_{i,j}\left(\frac{c^k_{0,0}}{\bar{c}_{0,0}}\right)^{a_T}, \tag{2}$$

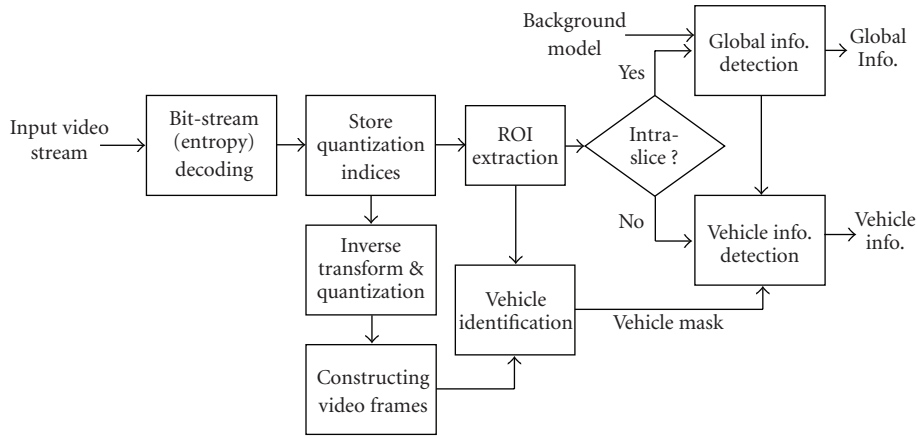


Figure 4: The diagram of information detector.

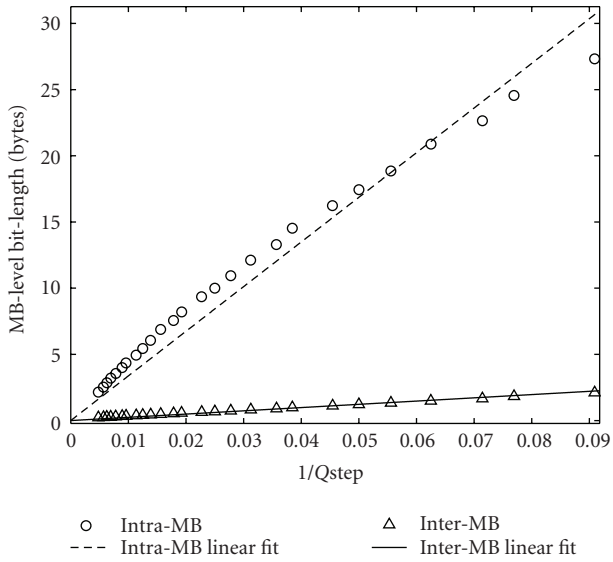


Figure 5: 1/Qstep versus the bit-stream length.

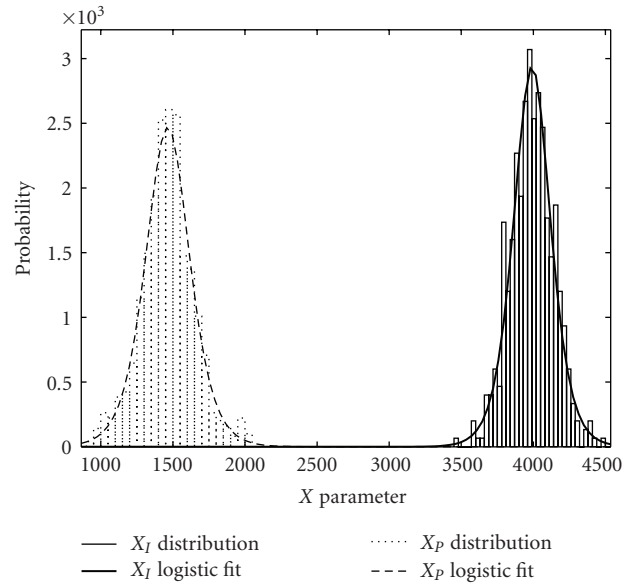


Figure 6: The probability distribution of $X_{I,\text{Frame}}$ and $X_{P,\text{Frame}}$.

where $t_{i,j}$ is a function of the global display and perceptual parameters such as the viewing distance, the display resolution, and the display luminance, a_T is a luminance-masking exponent with a typical value of 0.65, $\bar{c}_{0,0}$ is the average of DC coefficients for the image or a nominal value of 128× block size, corresponding to gray-level 8-bit images, and $c^k_{0,0}$ is the DC term of DCT for the subblock. In other words, the luminance masking, $a^k_{i,j}$, is determined by the DC term and the location (i, j) in a subblock. The luminance-adjusted threshold is then adjusted for the component contrast via

$$m^k_{i,j} = \max\left\{a^k_{i,j}, \left|c^k_{i,j}\right|^{s_{i,j}} \times a^{k(1-s_{i,j})}_{i,j}\right\}, \tag{3}$$

where $c^k_{i,j}$ is the DCT coefficient, $s_{i,j}$ is the exponent that typically has a value of 0.7, and $m^k_{i,j}$ is the resulting JND.

It should be noted that the exact value of a transform coefficient is required to determine $m^k_{i,j}$ in (3) but it is unavailable to the watermark embedder because of the intraprediction adopted in H.264/AVC. In other words, the additional DCT will be required to calculate $m^k_{i,j}$. Considering that the efficiency is important in the video watermarking and that the requirement of visual quality is higher in surveillance videos, we use a more conservative value, that is, the luminance masking $a^k_{i,j}$, as the JND, instead of $m^k_{i,j}$. As mentioned above, $a^k_{i,j}$ in Watson's model depend only on the DC value of transform block and some global settings. In the encoding process, we can calculate the average pixel value of a 4 × 4 subblock in the incoming frame to determine the DC value and derive the luminance masking afterwards.

In H.264/AVC [22], the quantization index, $I^k_{i,j}$, is calculated by

$$I^k_{i,j} = \left\{W^k_{i,j} \times MF^k_{i,j} + f \times 2^{q\text{bits}}\right\} \gg q\text{bits}, \tag{4}$$

(a) (b) (c) (d)

Figure 7: The views of 4 long videos.

where "≫" is the binary right-shift operation and $q$bits is equal to $15 + \lfloor \mathrm{QP}/6 \rfloor$. $W_{i,j}^k$ is the result of a linear transform with simple integer operations and $MF_{i,j}^k$ is the precalculated multiplication factor, which is equal to

$$\frac{PF_{i,j}}{\mathrm{Qstep}^{(\lfloor \mathrm{QP}/6 \rfloor)}} \times 2^{q\mathrm{bits}}, \tag{5}$$

where $PF_{i,j} > 0$ can be tabulated and $\mathrm{Qstep}^{(p)}$ is the quantization step size corresponding to a QP value p. It should be noted that $W_{i,j}^k \times PF_{i,j}$ is the exact value of the residual's transform coefficient and such division is to combine the scaling step of transform with the subsequent quantization. The parameter f can be determined by the encoder and is in the range of $[0, 1/2]$. Given that the quantization index of $W_{i,j}^k \times PF_{i,j}$ is $I_{i,j}^k$, then

$$\left(\left|I_{i,j}^k\right| - f\right)2^{q\mathrm{bits}} \le \left|W_{i,j}^k\right| PF_{i,j} < \left(\left|I_{i,j}^k\right| + 1 - f\right)2^{q\mathrm{bits}}. \tag{6}$$

Let $Dp_{i,j}^k = (|I_{i,j}^k| + 1 - f)2^{q\mathrm{bits}} - |W_{i,j}^k|PF_{i,j}$ and $Dn_{i,j}^k = |W_{i,j}^k|PF_{i,j} - (I_{i,j}^k - f)2^{q\mathrm{bits}}$. If we have to modify $I_{i,j}^k$ by 1, the watermarked index $\hat{I}_{i,j}^k$ will be formed by

$$\hat{I}_{i,j}^k = I_{i,j}^k + \mathrm{Sgn}\left\{W_{i,j}^k\right\}\mathrm{Sgn}\left\{Dn_{i,j}^k - Dp_{i,j}^k\right\}, \tag{7}$$

where $\mathrm{Sgn}\{x\} = 1$ if $x > 0$ and $\mathrm{Sgn}\{x\} = -1$ if $x \le 0$. The embedding process is as follows. For a subblock with nonzero I_{sum}^k, we trace the m selected indices with the backward zigzag scan until we meet a nonzero quantization index, $I_{m,n}^k$. Next we collect the remaining coefficients on the zigzag scan, including $I_{m,n}^k$ to form a set Ψ_k, and calculate the modification distance of each by $D_{i,j}^k = \min\{Dn_{i,j}^k, Dp_{i,j}^k\}$. We choose the position $\{x, y\}$ of the index for watermarking by

$$\{x, y\} = \arg\min_{I_{i,j}^k \in \Psi_k}\left\{a_{i,j}^k - D_{i,j}^k\right\} \tag{8}$$

and form the watermarked index, $\hat{I}_{x,y}^k$, according to (7). In addition, if $I_{m,n}^k$ is located on the diagonal, that is, the last four in the scan, then these four indices will all be considered. The other special case is when $I_{m,n}^k$ is the only nonzero index and its modification distance, $D_{m,n}^k$, is equal to $Dn_{m,n}^k$. We will force $D_{m,n}^k$ to be $Dp_{m,n}^k$ and then find the index to modify according to (8) to avoid generating a zero I_{sum}^k after watermarking.

The global information embedding is very similar to the design of vehicle information embedding with a few differences. First, as mentioned before, the global information is embedded into the background area of a frame. Second, as we use the H.264/AVC baseline profile, the video frames are classified into I- and P-frames. Since such global information

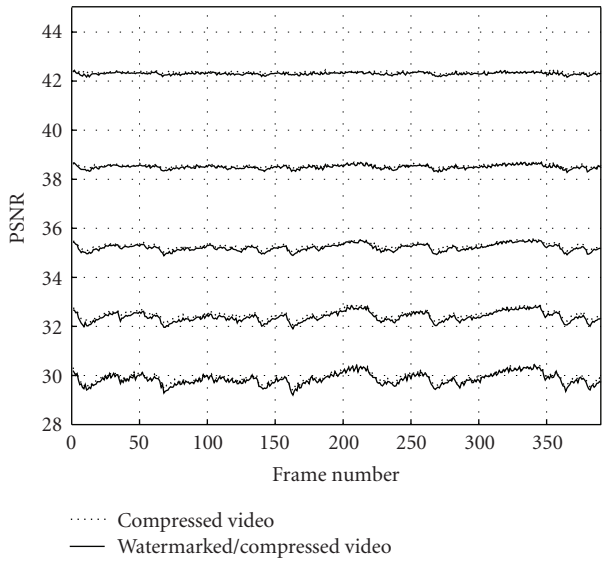


Figure 8: PSNR values of each frame in the Scene 0 video coded with fixed QP values.

as the camera/video serial number may be kept the same in consecutive frames, we choose to embed the global information only in I-frames, which appear periodically. Third, only the 4×4 intrapredicted subblocks, instead of the subblocks in the 16×16 intrapredicted macroblock, will be chosen for global information embedding to avoid generating visible artifacts in flat areas. In addition, although I-frames are expected to have more nonzero quantization indices, which are more suitable for digital watermarking, we have to avoid producing visible distortions from "over-watermarking." As the background area that we choose for global information embedding is usually kept steady in the video and occupies quite a large region, maintaining its quality is important. We sparsely select some subblocks with their locations known by both the embedder and detector and embed only one bit in each subblock to avoid successive modifications.

2.4. Information Detection. The flowchart of information detection is shown in Figure 4. After the entropy decoding, the quantization indices of a frame will be stored for the subsequent information extraction. In our design, Watson's model is calculated in the encoder but is not necessary in the decoder. Although it may seem a more elegant algorithm that both the encoder and detector calculate the JND to select or skip an index of subblock for watermarking, the possible difference between the JND's computed in both sides prevents us from designing this way. As mentioned before, the JND is determined by some global settings, the location of coefficient, and QP and DC values. The former three factors are the same in the encoder and the detector but their DC values may be different. A small change of DC value may result in the case that the encoder embeds one bit in a subblock but the detector ignores it because of a possible higher JND value and the errors from dropping bits will be difficult to correct. In addition, unlike the encoder, which can extract the vehicle masks from the raw video, the detector has to use the reconstructed, lossy-compressed, and watermarked video to identify the vehicles in the video. However, the slight difference in the shapes of vehicles between the encoder and decoder will result in the synchronization problem. Our solution is to explicitly inform the location of watermarking by using the ROI coding. To be more specific, in order to better preserve the visual quality of vehicles, we will assign a smaller QP to the vehicle area and a larger QP to the background. The different QP values in a frame can thus help to locate the hidden information for the watermark detector. This is also the reason why we choose 16×16 macroblocks to describe the ROI, instead of 4×4 subblocks, so that the shape of ROI can be more stable after coding.

By explicitly signalling with different QP values, the detection process can be simplified and, most of the time, the information can be extracted without resorting to the original video frame. However, given that the occlusion of vehicles may happen and that the detector may always expand the compressed bit-stream into frames to link the hidden information to the video content for the users to view, the frames will still be expanded for offering possible assistance of identifying the vehicles as shown in Figure 4. As in the encoder, only the data of intracoded macroblocks will be used for information extraction. Besides, the background model is also constructed on the fly to determine the area for global information extraction. The detection of both the global and vehicle information can then be applied in a rather straightforward manner. The decoder simply calculates the sum of m-considered quantization indices, I^k_{sum}, of the selected subblocks. The subblocks with all the selected indices being zero will be skipped. An even value of I^k_{sum} generates a bit "0" while an odd value of I^k_{sum} generates a bit "1."

3. The ROI-Based Rate Control Mechanism

As mentioned before, the ROI coding helps to improve the quality of vehicles and achieve the reliable watermarking. There are basically four steps in the proposed scheme. First, a linear R-Q models derived by training a segment of the traffic surveillance video will be used to decide the target bit-stream length in each frame. Then we allocate bits to GOP's and frames. Next, the QP or quantization step size, Qstep, associated with the macroblocks in the background and vehicle regions will be set accordingly to match the target bit-rate. Finally, a quick updating approach is adopted to cope with different traffic conditions.

3.1. The Linear R-Q Model. We need a fast and accurate model to map the bit-rate and quantization for the rate control. We found that the relationship between the bit-stream length and Qstep can be approximately expressed as a linear function in traffic surveillance videos. Figure 5 shows the correspondences of $1/Q$step and the bit-stream lengths of macroblocks. Every point is the average of the data in

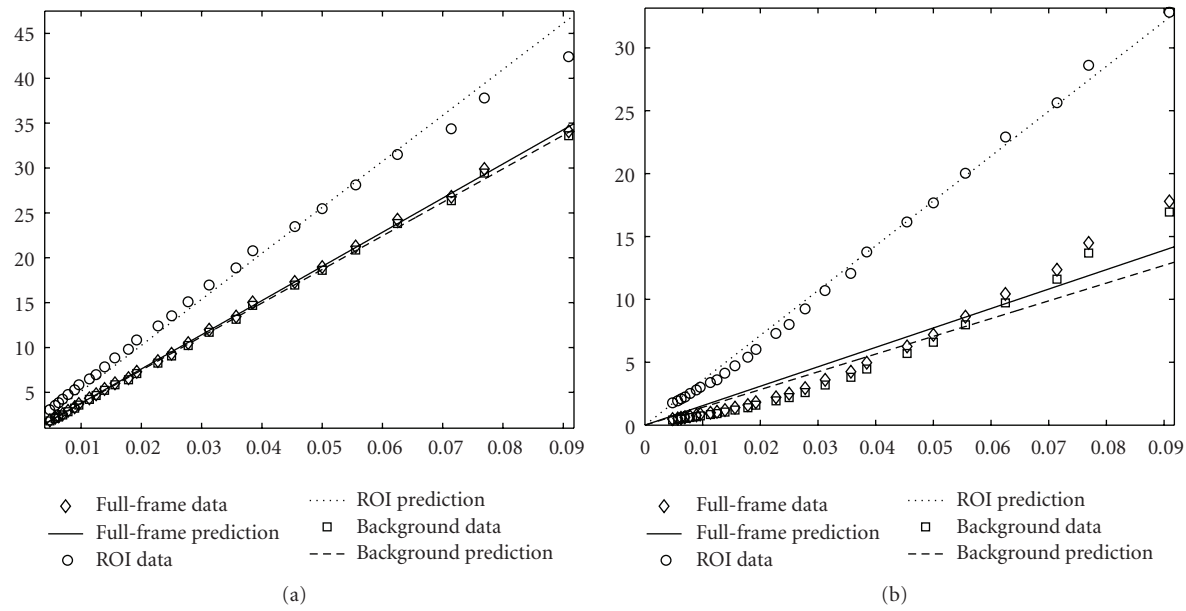


Figure 9: The linear R-Q models of (a) I frames and (b) P frames.

100 frames of a traffic surveillance video. Each frame in the test video is encoded by fixed Qstep values corresponding to the QP values ranging from 25 to 50. We separate the data of Intracoded MB's (marked by circles) and Intercoded MB's (marked by triangles) and then use the linear regression to fit these two groups of points. We can observe that the straight lines are reasonably close to those points from the experiment. In addition, the line can be made pass through the origin of the coordinates; so the constant in the linear function is not necessary. By changing the slope of the linear function, we can quickly adjust the R-Q model to make the scheme adaptive to the condition changes. We further extract twenty 100-frame video segments from five different video scenes and compress these video segments with varying QP values. By applying the linear regression, the average R-square values of Intra-MB's and Inter-MB's models are 0.93 and 0.97, respectively; so the use of linear model should be appropriate.

In our design, we calculate six linear models for different modes of prediction. The six models are classified into the frame-level models, that is, $M_{I,\text{Frame}}$ for I frames and $M_{P,\text{Frame}}$ for P frames, and region-level models, that is, $M_{I,\text{ROI}}$, $M_{I,\text{Bg}}$, $M_{P,\text{ROI}}$, $M_{P,\text{Bg}}$, which represent the models of ROI in the I-frame, the background in the I-frame, ROI in the P-frame, and the background in the P-frame, respectively. These linear R-Q models will help us predict the frame-level bit allocation and the region-level QP determination. The predicted bit-stream length R_{mode} will then be expressed as

$$R_{\text{mode}} = M_{\text{mode}}\left(\text{Qstep}_{\text{mode}}\right) = \frac{X_{\text{mode}}}{\text{Qstep}_{\text{mode}}}, \tag{9}$$

in which X_{mode} is the first-order coefficient, that is, the slope, of one of the six linear models. It should be noted that compressing the video with different QP values can generate more accurate data but is time-consuming. Here, we adopt a more practical approach by randomly assigning QP values in macroblocks in a training video segment and collecting their bit-stream lengths. By doing so, we can simply run the training process for a period of time to set up the model, instead of repeatedly compressing the same video segment with different QP values.

3.2. The Bit Allocation. With the R-Q model at hand, we can proceed to apply the bit allocation. We first determine the number of bits for a GOP. Given the target video bit-rate equal to V bits per second, the GOP size equal to S_{GOP} frames, and the frame rate equal to K frames per second, the target bit budget of the ith GOP, $R_{\text{GOP}^{(i)}}$, can be calculated by

$$R_{\text{GOP}^{(i)}} = \frac{V \times S_{\text{GOP}}}{K} + R_{\text{GOP}^{(i-1)}}, \tag{10}$$

where $R_{\text{GOP}^{(i-1)}}$ represents the remaining bits after processing the $(i-1)$th GOP and $R_{\text{GOP}^0} = 0$. If the coding process uses fewer bits than expected in the $(i-1)$th GOP, $R_{\text{GOP}^{(i-1)}}$ will be larger than 0; so we can consume more bits in the ith GOP. Else, fewer bits will be allowed in ith GOP. $R_{\text{GOP}^{(i)}}$ will then be reduced after we process each frame. Next, our scheme will set the target bit-stream length for the I-frame in the GOP. It should be noted that the number of bits assigned to the I frame in a GOP is very important. If the I-frame occupies too many bits, the quality of the following P-frames may be poor. However, if the I-frame uses too few bits, the quality of the following P frames will also be affected due to the inter-prediction process in video coding. Besides, the visible quality fluctuation may appear within a GOP. In our scheme, we require that the QP of the I frame, QP_I, should be smaller than the average QP values in P frames by 1. By our linear

Table 1: The Performance of information hiding.

	Unwatermarked video		*Watermarked video*			
Video Name (QP)	PSNR (dB)	Ave. Frame Length (bytes)	PSNR (dB)	Ave. Frame Length (bytes) {Increase}	Ave. Global Info. per Frame (bits)	Ave. Vehicle Info. per Car (bits)
Scene0 (20)	42.37	21967	42.31	22067 {0.46%}	199	3586
Scene1 (20)	41.86	26194	41.81	26293 {0.38%}	150	3497
Scene2 (20)	40.27	26314	40.24	26424 {0.42%}	131	3338
Scene3 (20)	42.54	20522	42.46	20652 {0.63%}	91	7246
Scene4 (20)	40.41	26150	40.36	26330 {0.69%}	181	5190
Scene0 (25)	38.56	13507	38.50	13605 {0.73%}	189	2482
Scene1 (25)	38.23	18139	38.16	18255 {0.64%}	145	2749
Scene2 (25)	37.46	17744	37.40	17866 {0.69%}	125	2559
Scene3 (25)	39.23	13830	39.13	13954 {0.89%}	72	5605
Scene4 (25)	37.43	17506	37.34	17687 {1.03%}	158	3805
Scene0 (30)	35.30	7697	35.21	7786 {1.16%}	142	1642
Scene1 (30)	34.67	11603	34.59	11713 {0.94%}	119	1819
Scene2 (30)	34.40	11207	34.31	11316 {0.98%}	107	1676
Scene3 (30)	35.89	8936	35.77	9040 {1.16%}	48	3717
Scene4 (30)	34.38	10991	34.27	11131 {1.27%}	118	2237
Scene0 (35)	32.50	44683	32.39	45360 {1.52%}	95	963
Scene1 (35)	31.38	73206	31.29	73958 {1.03%}	82	947
Scene2 (35)	31.43	69915	31.33	70712 {1.14%}	77	988
Scene3 (35)	32.66	58150	32.53	58832 {1.17%}	32	2290
Scene4 (35)	31.44	68283	31.33	69199 {1.34%}	84	1085
Scene0 (40)	29.93	2665	29.84	2697 {1.20%}	38	385
Scene1 (40)	28.24	4599	28.17	4637 {0.84%}	47	392
Scene2 (40)	28.44	4294	28.37	4339 {1.05%}	35	530
Scene3 (40)	29.43	3750	29.31	3797 {1.26%}	21	1265
Scene4 (40)	28.55	4261	28.46	4315 {1.26%}	50	503

R-Q model, for a given QP_I, the resulting bit-stream length of this GOP, $R_{\mathrm{GOP}}^{(\mathrm{QP}_I)}$, can be calculated by

$$R_{\mathrm{GOP}}^{(\mathrm{QP}_I)} = \left\{ \frac{X_{I,\mathrm{Frame}}}{Q\mathrm{step}^{(\mathrm{QP}_I)}} + (S_{\mathrm{GOP}} - 1) \times \frac{X_{P,\mathrm{Frame}}}{Q\mathrm{step}^{(\mathrm{QP}_I+1)}} \right\} \times N_{\mathrm{MB}}, \tag{11}$$

in which $Q\mathrm{step}^{(\mathrm{QP}_I)}$ and $Q\mathrm{step}^{(\mathrm{QP}_I+1)}$ are the Qstep corresponding to I and P frames, respectively. $X_{I,\mathrm{Frame}}$ and $X_{P,\mathrm{Frame}}$ of our linear models can be obtained from $M_{I,\mathrm{Frame}}$ and $M_{P,\mathrm{Frame}}$ as described in Section 3.1. Since the relationship between the bit-stream length and $1/Q$step is determined based on the data in MBs, N_{MB} representing the number of macroblocks in a frame has to be included in (11). The target QP value of the I frame, QP_I^t, in ith GOP will be set as

$$\mathrm{QP}_I^t = \underset{0 \le \mathrm{QP}_I \le 51}{\arg\min} \left| R_{\mathrm{GOP}^{(i)}} - R_{\mathrm{GOP}}^{(\mathrm{QP}_I)} \right|. \tag{12}$$

The target length of the I-frame, R_I^t, can then be derived by

$$R_I^t = \left\{ \frac{X_{I,\mathrm{Frame}}}{Q\mathrm{step}^{(\mathrm{QP}_I^t)}} \right\} \times N_{\mathrm{MB}}. \tag{13}$$

The bit budget in the ith GOP, $R_{\mathrm{GOP}^{(i)}}$, will be reduced by the actual bit consumption of processing each frame; so the target bit-stream length of P frame, R_P^t, will be dynamically determined by

$$R_P^t = \frac{R_{\mathrm{GOP}^{(i)}}}{F_p}, \tag{14}$$

where F_p is the number of remaining P frames in the current GOP. However, we found that, in order to stabilize the visual quality, the number of allocated bits cannot vary significantly during the encoding process of a GOP. For example, it may happen that the P frames at the end of GOP may be assigned with too few bits since many vehicles may appear just before this P frame and the number of the remaining bit budget is too small. We set a reference bit-stream length of P frames R_P^r by

$$R_P^r = \frac{R_{\mathrm{GOP}^{(i)}} - R_I^t}{S_{\mathrm{GOP}} - 1}. \tag{15}$$

We limit R_P^t in the range of $[R_P^r \times (1-\theta), R_P^r \times (1+\theta)]$, where the ratio θ is equal to 0.2 in the frames with vehicles and equal to 0.1 in the frames without vehicles. By this design, the scheme can assign more bits when the vehicles appear abruptly and prevent large quality fluctuations in the frames without vehicles.

Table 2: The performance of information hiding with the IPP rate control.

	150 Kbps			350 Kbps		
Video	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)
Scene 0	27.09	223	150.52	30.74	497	350.62
Scene 1	27.58	130	150.80	31.86	335	350.89
Scene 2	27.42	158	150.46	32.32	399	350.49
Scene 3	28.33	311	150.60	33.43	789	350.31
Scene 4	28.14	100	150.57	32.31	287	350.59
	550 Kbps			750 Kbps		
Video	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)
Scene 0	32.77	677	550.47	34.30	797	750.34
Scene 1	34.29	478	551.04	36.12	580	751.42
Scene 2	34.88	567	550.67	36.79	699	750.57
Scene 3	36.30	1110	550.56	38.46	1351	750.37
Scene 4	34.61	427	550.52	36.41	545	750.39

*3.3. The QP/Q*step *Determination.* After obtaining the frame-level bit-stream length prediction, R_T^t, where T is the frame type (I or P), we can proceed to determine the QP or Qstep values of the ROI and background. We enforce that $\mathrm{QP}_{T,\mathrm{Bg}}$ should be higher than $\mathrm{QP}_{T,\mathrm{ROI}}$ by a difference, $\mathrm{QP}d_T$, which should not be larger than $\mathrm{QP}d_T^{\max}$, so that the quality of the ROI and background can be maintained in a reasonable range. Then, we will find the best match of the bit-rates assigned to the ROI and background with the target frame bit-stream length. To be more specific, by testing with different $\mathrm{QP}_{T,\mathrm{ROI}}$ and $\mathrm{QP}d_T$ values, we can determine the QP of ROI, *that is*, $\mathrm{QP}_{T,\mathrm{ROI}}^c$, and the QP difference, *that is*, $\mathrm{QP}d_T^c$, of the current frame by

$$\begin{aligned}\left\{\mathrm{QP}_{T,\mathrm{ROI}}^c, \mathrm{QP}d_T^c\right\} &= \underset{\substack{\mathrm{QP}_{T,\mathrm{ROI}}^{\min}<\mathrm{QP}_{T,\mathrm{ROI}}<\mathrm{QP}_{T,\mathrm{ROI}}^{\max}\\ 0<\mathrm{QP}d_T<\mathrm{QP}d_T^{\max}}}{\arg\min} \Big\{\Big| R_T - \gamma\times f_{T,\mathrm{ROI}}\left(\mathrm{QP}_{T,\mathrm{ROI}}\right) - (1-\gamma) \\ &\qquad \times f_{T,\mathrm{ROI}}\left(\mathrm{QP}_{T,\mathrm{ROI}} + \mathrm{QP}d_T\right)\Big|\Big\},\end{aligned} \tag{16}$$

where γ is the ratio of the ROI area to the full frame. $\mathrm{QP}d_I^{\max}$ and $\mathrm{QP}d_P^{\max}$ are set as 3 and 4, respectively. It should be noted that we set a search range $[\mathrm{QP}_{T,\mathrm{ROI}}^{\min}, \mathrm{QP}_{T,\mathrm{ROI}}^{\max}]$ for choosing an appropriate QP value for ROI. In the case that the ROI exists in both the current and previous frames, we will set $\mathrm{QP}_{T,\mathrm{ROI}}^{\min}$ and $\mathrm{QP}_{T,\mathrm{ROI}}^{\max}$ as $\mathrm{QP}_{T,\mathrm{ROI}}^p - 2$ and $\mathrm{QP}_{T,\mathrm{ROI}}^p + 2$, respectively, where $\mathrm{QP}_{T,\mathrm{ROI}}^p$ is the QP of the ROI in the previous frame, to avoid changing the quality of the ROI too much. In other cases, $\mathrm{QP}_{T,\mathrm{ROI}}^{\min}$ and $\mathrm{QP}_{T,\mathrm{ROI}}^{\max}$ are set as 0 and $51 - \mathrm{QP}d_T^{\max}$ to find the best bit-stream length match.

3.4. The Adaptive Model Updating. We may have to adjust the R-Q model when the scene condition changes, such as the varying light or the effects from weather. A sliding window with 100 frames will be used to collect a set of data, including the actual number of bits and the corresponding Qstep values. As our model is a linear function, a linear regression will be applied to the data of the sliding window to obtain the new parameter X' by

$$X'_{\mathrm{mode}} = \frac{\sum\left(R^a_{\mathrm{mode}}/\mathrm{Qstep}_{\mathrm{mode}}\right)}{\sum\left(1/\mathrm{Qstep}_{\mathrm{mode}}\right)^2}, \tag{17}$$

where R^a_{mode} is the number of bits used in the ROI/background MB of I or P frame. Given that the previous parameter is X^{old}, the updated parameter, X^{new}, will be set as

$$X^{\mathrm{new}} = \beta \times X' + (1-\beta)\times X^{\mathrm{old}}, \tag{18}$$

where β is empirically set as 0.01, the reciprocal of the window size.

To improve the accuracy of the prediction, some outliers have to be removed during the model updating. Figure 6 shows the distributions of $X_{I,\mathrm{Frame}}$ and $X_{P,\mathrm{Frame}}$ by collecting the data from more than 5000 frames of a long video. We found that they match well with the logistic distribution:

$$g(x) = \frac{e^{-(x-\mu)/s}}{s\left[1+e^{-(x-\mu)/s}\right]^2}, \tag{19}$$

where μ is the mean of collected data, and $s = \sqrt{3}\times\sigma/\pi$ with σ being equal to the standard deviation. The match between X_{mode} and the logistic distribution indicates that our model works well and the outliers do not appear frequently. Our scheme collects X_{mode} within $[\mu-\sigma, \mu+\sigma]$ for updating and around 28% data will be viewed as the outliers. This method cannot only adjust the model dynamically but maintain the accuracy of the model.

4. Experimental Results

In our experiments, we use a 500-frame CIF video, which is shown in Figure 1 and labeled as Scene 0, to be the main test video for demonstrating the performances in frames. Four other 10-minute long videos will also be tested to illustrate the feasibility of the proposed scheme. The scenes

Table 3: The performance of the proposed rate control scheme.

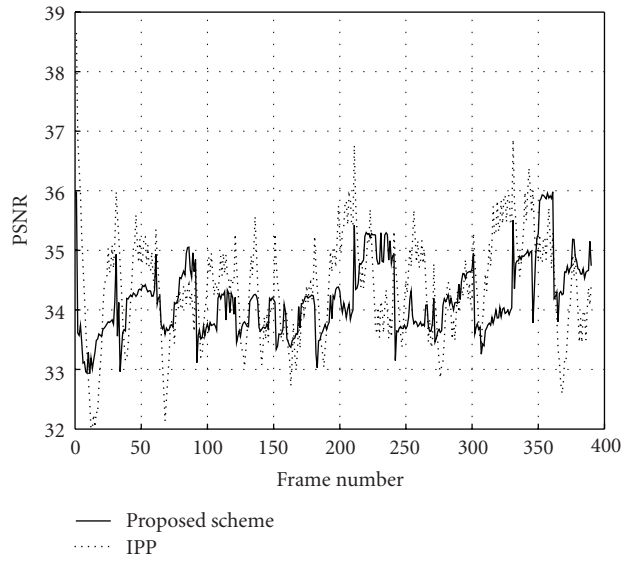
The Proposed Rate Control Scheme					
Video	Avg. PSNR (dB)	PSNR var.	Avg. ROI PSNR (dB)	ROI PSNR var.	Bit rate (Kbps)
Scene 0	34.17	0.3744	33.59	1.0278	350.31 (+0.088%)
Scene 1	33.83	0.6267	34.25	1.0796	349.83 (−0.049%)
Scene 2	34.82	0.4374	34.82	0.9415	350.29 (+0.083%)
Scene 3	36.34	0.3559	35.53	0.9324	350.33 (+0.093%)
Scene 4	34.11	0.4013	34.08	0.9267	350.32 (+0.091%)
The IPP Rate Control Scheme					
Video	Avg. PSNR (dB)	PSNR var.	Avg. ROI PSNR (dB)	ROI PSNR var.	Bit rate (Kbps)
Scene 0	34.29	0.8007	30.97	1.4775	350.63 (+0.179%)
Scene 1	33.96	1.4786	31.81	2.1151	350.49 (+0.140%)
Scene 2	35.12	0.7581	32.56	2.1910	350.60 (+0.170%)
Scene 3	36.90	1.3165	33.64	1.4582	350.62 (+0.186%)
Scene 4	34.55	0.8161	32.29	1.5951	350.57 (+0.162%)
The X264 Rate Control Scheme					
Video	Avg. PSNR (dB)	PSNR var.	Avg. ROI PSNR (dB)	ROI PSNR var.	Bit rate (Kbps)
Scene 0	34.51	0.5650	30.72	1.1926	353.92 (+1.120%)
Scene 1	34.42	0.4963	32.18	1.4177	359.06 (+2.588%)
Scene 2	35.81	0.3665	32.70	1.4449	363.40 (+3.829%)
Scene 3	37.32	0.5424	33.76	1.3004	359.87 (+2.819%)
Scene 4	35.00	0.4485	32.60	1.0186	358.29 (+2.370%)

Table 4: The performance of information hiding with the proposed rate control.

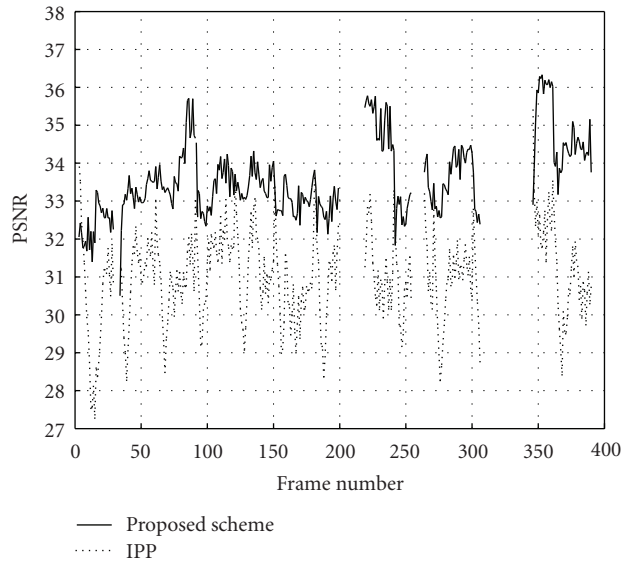
	150 Kbps			350 Kbps		
Video	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)
Scene 0	29.24	391	149.96	33.58	750	350.28
Scene 1	29.50	201	150.25	34.11	467	350.07
Scene 2	29.30	243	150.07	34.63	552	350.11
Scene 3	29.77	405	150.14	35.32	994	350.15
Scene 4	29.54	138	150.16	33.97	380	350.25
	550 Kbps			750 Kbps		
Video	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)	ROI PSNR (dB)	Payload (bits)	Bit Rate (Kbps)
Scene 0	35.91	936	550.28	37.67	1084	750.04
Scene 1	36.91	607	550.32	38.93	730	750.18
Scene 2	37.51	744	550.20	39.61	875	750.33
Scene 3	38.46	1347	549.92	40.76	1563	750.27
Scene 4	36.68	545	550.35	38.60	668	750.24

of these four long videos, labeled as Scene 1, 2, 3, and 4, are displayed in Figure 7. We adopt the H.264/AVC codec of Intel Integrated Performance Primitives (IPPs) to process the videos since the real-time processing is required. In the current implementation, the encoder can process more than 15 frames per second on an Intel P4 machine at 2.2 GHz and with 2 G RAM. It should be noted that a large portion of the complexity still resides in the execution of the ordinary H.264/AVC compression.

To begin with, we show the performance of information hiding. In a 4×4 subblock, the highest $m = 10$ quantization indices in a zigzag scan will be considered for watermark embedding/detection. To verify that the embedded signal in the proposed scheme will not affect the normal usage of the video, we first compress all the video frames with the intracoding so that each frame will be embedded with the global information and the vehicle information, if necessary. Figure 8 shows the PSNR values when the video of Scene 0 are compressed with QP = 20, 25, 30, 35, and 40. The dotted lines are the PSNR values between the original videos and the compressed videos while the solid lines are those between the original videos and the watermarked/compressed videos. We can see that the PSNR curves in each case are very close; so the visual quality degradation is very small at different bit-rates. We further demonstrate the average PSNR, the data volume of the global/vehicle information, and the bit-stream length in the four long videos, along with Scene 0, in Table 1, in which all the frames are intra-encoded. Again,



(a)



(b)

Figure 10: The comparison of PSNR values of the original compressed video at 350 Kbps by IPP and the watermarked video in (a) the full-frame and (b) the ROI.

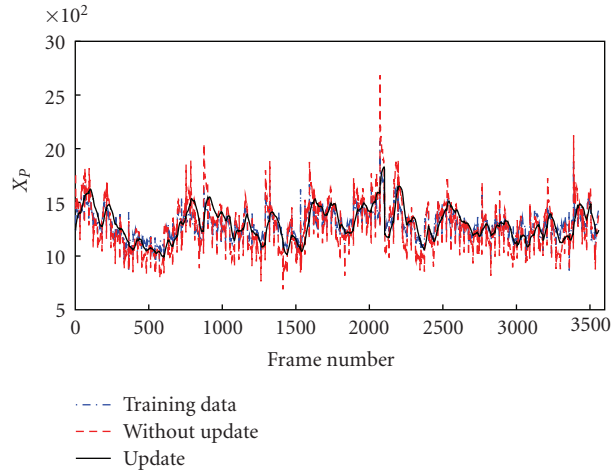


Figure 11: The variations of $X_{P,\text{Frame}}$.

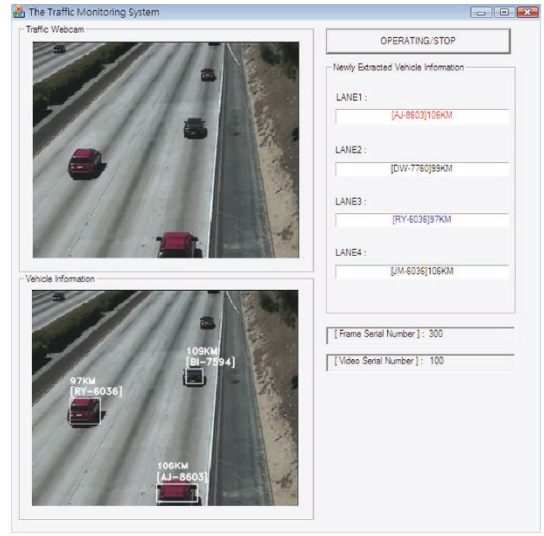


Figure 12: The interface of the watermark detector.

we can see that the PSNR decreases are usually less than 0.1 dB and the bit-stream size enlargements are less than 1.5% in these five videos. Table 1 also shows that the payload of global information will depend on the target bit-rates. We embed at most one bit in each macroblock so that the frame quality degradation is limited. Except Scene 3, more than 100 bits in average can be embedded into each I frame as the global information when the QP value is set below 30. It should be noted that the QP values above 35 have severely blurred the video. Scene 3 has the lowest payload because its background is quite smooth without much texture. If more global information is required, we may use more I frames to embed a piece of global information, especially in lower bit-rates. In our opinions, when I frames are used quite often and periodically, this strategy should be acceptable.

In addition, Table 1 lists the average vehicle information per car as a reference showing how its payload will be affected by using different QP values. Since it is impractical to compress the video with the intracoding only, we enable the rate control mechanism of IPP and assign different target bit-rates to see the performance of vehicle information embedding more clearly. The results are shown in Table 2. The GOP size is set as 30; so one I frame is followed by 29 P frames. Four bit-rates, that is, 150, 350, 550, and 750 Kbps are tested and we can see that each vehicle can be embedded with hundreds of bits; so a large amount of vehicle-related information can be embedded.

Next, we would like to check the performance of the proposed rate-distortion scheme. Figures 9(a) and 9(b) show the linear model for I frames and P frames of Scene 0, respectively. There are three lines in each case, which are the

models for the full-frame, ROI, and background. The near straight line in Figure 9(a) shows that the linear regression is quite accurate in I frames. Although the prediction in P frames in Figure 9(b) has larger deviations, we think that the linear model is good enough and the efficiency can be achieved.

Figure 10(a) demonstrates the PSNR curves of our scheme, shown as the solid line, and the PSNR curves of IPP, shown as the dashed line, of the frames in Scene 0. Although the full-frame PSNR values are usually lower than those in IPP, our scheme can effectively maintain the quality in frames so that the unnecessary quality fluctuation is minimized. In our opinion, the stable visual quality should be a requirement for traffic surveillance videos. Figure 10(b) shows the PSNR curves of the ROI only and we can see that the PSNR values in the proposed scheme are significantly higher than those in IPP since lower QP values are assigned to the ROI. According to our experiments on the five test videos compressed with varying bit-rates, the QP values assigned to the ROI in our scheme are lower than those in IPP by around 3 in average and the largest difference is 8. The lower QP values and the resultant higher PSNR of the watermarked ROI indicate that the visual information in ROI is better preserved and this should alleviate the concern that the information embedding process may affect the significant parts of the video. More detailed results in the five test videos are given in Table 3. We compare our scheme with IPP and the other popular real-time H.264 codec, X264. Given the target bit-rate equal to 350 Kbps, our scheme can perform a more accurate bit assignment. Besides, as mentioned before, the variations of the full-frame PSNR and the ROI PSNR are smaller in our scheme.

Table 4 shows the combined results of information hiding and rate control. By comparing Tables 2 and 4, we can see that the quality of ROI is significantly improved. The PSNR values in Table 4 are higher than those in Tables 2 by around 2 dB in each case. Although the global information embedding is affected by our strategy of ROI coding, its payload is still sufficient in our scenario. Besides, the information hiding process will not affect the performance of the proposed rate control scheme, which can be reflected from the accurate resulting bit-rate in each test as shown in Table 4.

The effectiveness of the proposed adaptive updating can be validated by Figure 11, which demonstrates the variation of $X_{P,\text{frame}}$ in 3600 frames. The solid line is the result of adaptive updating while the dashed line shows the result of the scheme that does not remove the outliers. We employ a 100-frame window and use the data in the window to determine the parameter of the next frame. The breaking line shows the parameters determined by training the video frames with different QP values. To be more specific, we collect average bit-stream lengths of macroblocks from different QPs of the target frame and then apply linear regression to obtain $X_{P,\text{Frame}}$ as the ground truth. We can see that the curve with the proposed updating approach will match the training data better. The parameters of the scheme without outlier removal will fluctuate a lot due to the abrupt content change in different frames and the accuracy of bit-stream length estimation will thus be affected.

The interface of watermark detection is shown in Figure 12. We use the speed (7 bits) and license plate number (27 bits) of a car as the vehicle information. The speed of car will be embedded first so that this value can be shown at the beginning of information detection in a vehicle. Two videos are displayed on the left side, that is, the watermarked video itself and the one superimposed with the bounding boxes of vehicles and the vehicle information including the simulated speed and the licence plate number, for better illustration. The right side shows the newly extracted vehicle information in each lane. In our opinion, to describe the scenes in traffic videos may require a lot of efforts, which may lead to a large metadata volume. By using digital watermarking, the correspondence between the vehicle and information is easy to be identified. Besides, as the bit-rate and distortion are not affected, the need of extra metadata is eliminated. The global information including the video serial number (16 bits) and a frame serial number (16 bits), which is an incremental value along with the information embedding, is used to ensure the correct order of video segments for authentication purposes.

5. Conclusion

We proposed to make use of digital watermarking techniques to facilitate annotating traffic surveillance videos. An H.264/AVC-based information hiding scheme is developed and the related issues are considered to achieve a reliable transmission of vehicle- and camera/video-related information. The ROI-based rate control mechanism is proposed to improve the visual quality of vehicles and achieve a good rate-distortion performance. The two schemes are combined to achieve the effective traffic data annotation in videos. Experimental results demonstrate the feasibility of the scheme. We believe that the proposed scheme can also be extended to other scenarios, such as the indoor/outdoor surveillance.

Acknowledgment

This research was supported in part by the National Science Council in Taiwan, under Grant NSC97-2752-E-008-001-PAE.

References

[1] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking: Principles and Practice*, Morgan Kaufmann, San Fransisco, Calif, USA, 2001.

[2] S.-C. Chen, M.-L. Shyu, S. Peeta, and C. Zhang, "Learning-based spatio-temporal vehicle tracking and indexing for transportation multimedia database systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp. 154–167, 2003.

[3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE*

Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560–576, 2003.

[4] M. Barni, F. Bartolini, and N. Checcacci, "Watermarking of MPEG-4 video objects," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 23–31, 2005.

[5] P. Bas and B. Macq, "A new video-object watermarking scheme robust to object manipulation," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 2, pp. 526–529, October 2001.

[6] J. Zhang, A. T. S. Ho, G. Qiu, and P. Marziliano, "Robust video watermarking of H.264/AVC," *IEEE Transactions on Circuits and Systems II*, vol. 54, no. 2, pp. 205–209, 2007.

[7] G.-Z. Wu, Y.-J. Wang, and W.-H. Hsu, "Robust watermark embedding/detection algorithm for H.264 video," *Journal of Electronic Imaging*, vol. 14, no. 1, pp. 1–9, 2005.

[8] M. Yang and N. Bourbakis, "A high bitrate information hiding algorithm for digital video content under H.264/AVC compression," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 935–938, August 2005.

[9] M. Noorkami and R. M. Mersereau, "Compressed-domain video watermarking for H.264," in *Proceedings of the International Conference on Image Processing (ICIP '05)*, vol. 2, pp. 890–893, September 2005.

[10] Z. Chen and K. N. Ngan, "Recent advances in rate control for video coding," *Signal Processing: Image Communication*, vol. 22, no. 1, pp. 19–38, 2007.

[11] "Joint Video Team of ISO/IEC MPEG and ITU-T VCEG document, JVT-G012," March 2003.

[12] "Joint Video Team of ISO/IEC MPEG and ITU-T VCEG document, JVT-H017," March 2003.

[13] Y. Liu, Z. G. Li, and Y. C. Soh, "Region-of-interest based resource allocation for conversational video communication of H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 134–139, 2008.

[14] Y. Liu, Z. G. Li, and Y. C. Soh, "A novel rate control scheme for low delay video communication of H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 1, pp. 68–78, 2007.

[15] P.-H. Wu and H. H. Chen, "Frame-layer constant-quality rate control of regions of interest for multiple encoders with single video source," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 7, pp. 857–866, 2007.

[16] H. Li, Z. Wang, H. Cui, and K. Tang, "An improved ROI-based rate control algorithm for H.264/AVC," in *Proceedings of the IEEE International Conference on Signal Processing (ICSP '06)*, vol. 2, pp. 16–20, 2006.

[17] D. Agrafiotis, D. R. Bull, N. Canagarajah, and N. Kamnoonwatana, "Multiple priority region of interest coding with H.264," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 53–56, October 2006.

[18] Y. Zheng, X. Tian, and Y. Chen, "Adaptive frequency coefficient suppression for roi-based H.264/AVC video coding," in *Proceedings of IEEE International Conference on Networking, Sensing and Control (ICNSC '08)*, pp. 714–718, April 2008.

[19] A. Yoneyama, C.-H. Yeh, and C.-C. J. Kuo, "Robust vehicle and traffic information extraction for highway surveillance," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 14, pp. 2305–2321, 2005.

[20] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor, "Detection and classification of highway lanes using vehicle motion trajectories," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 2, pp. 188–200, 2006.

[21] A. B. Watson Jr., "DCT quantization matrices visually optimized for individual images," in *Human Vision, Visual Processing, and Digital Display IV*, Proceedings of SPIE, pp. 202–216, 1993.

[22] H. S. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 598–603, 2003.