Object-Based Approach for Adaptive Source Coding of Surveillance Video

Tung-Ming Pan [1,2], **Kuo-Chin Fan** [1] and **Yuan-Kai Wang** [3,*]

[1] Department of Computer Science & Information Engineering, National Central University, Chung-Li 320, Taiwan; macgyver@mail.fju.edu.tw (T.-M.P.); kcfan@csie.ncu.edu.tw (K.-C.F.)

[2] Holistic Education Center, Fu Jen Catholic University, New Taipei 242, Taiwan

[3] Department of Electrical Engineering, Fu Jen Catholic University, New Taipei 242, Taiwan

[*] Correspondence: ykwang@fju.edu.tw; Tel.: +886-2-2905-2101

Abstract: Intelligent analysis of surveillance videos over networks requires high recognition accuracy by analyzing good-quality videos that however introduce significant bandwidth requirement. Degraded video quality because of high object dynamics under wireless video transmission induces more critical issues to the success of smart video surveillance. In this paper, an object-based source coding method is proposed to preserve constant quality of video streaming over wireless networks. The inverse relationship between video quality and object dynamics (i.e., decreasing video quality due to the occurrence of large and fast-moving objects) is characterized statistically as a linear model. A regression algorithm that uses robust M-estimator statistics is proposed to construct the linear model with respect to different bitrates. The linear model is applied to predict the bitrate increment required to enhance video quality. A simulated wireless environment is set up to verify the proposed method under different wireless situations. Experiments with real surveillance videos of a variety of object dynamics are conducted to evaluate the performance of the method. Experimental results demonstrate significant improvement of streaming videos relative to both visual and quantitative aspects.

Keywords: moving object detection; adaptive source coding; video quality; regression algorithm; linear model

1. Introduction

Network-based video surveillance has become a dominant architecture from the second generation of video surveillance. Intelligent video surveillance as the mainstream of the third generation needs to analyze the network-transmitted videos and detect and recognize objects and events in the videos [1]. The intelligence of the system relies on the high accuracy of detection and recognition, which demands high video quality. Streaming high-quality videos requires significant bandwidth and is not achievable in many surveillance situations, especially for the wireless environment of mobile video surveillance [2]. Wireless networks have an inherent radio signal attenuation problem, which makes guaranteeing appropriate video quality difficult. A source coding method that cannot only compress surveillance videos but also adapt to network situations is crucial.

Source coding employs a compression technique to reduce video size, such as frame-type selection [3], macroblock (MB) partition size [4], and quantization parameters (QP) [5,6]. Optimized selection of I-, P-, and B-frame types through motion estimation can greatly reduce video size. An MB is linear block transform for video compression and is a base unit for motion prediction. In many codecs, MB data are transformed and quantized prior to coding and rescaling. QPs regulate how much spatial information is saved.

However, increased picture complexity and object dynamics can induce quality degradation caused by the intrinsic property of video coding. Video streaming with MPEG-based video coding

formats has a base coding structure that includes block-based motion compensation. However, this requires more bits to encode motion information. Under a predefined bitrate constraint, video quality declines because some coding parameters, such as QPs, must be adjusted adaptively to satisfy the bitrate constraint. Although the source coding approach has been well studied and applied successfully to improve video streaming efficiency, high-level video characteristics, such as picture complexity and object dynamics, should be incorporated into the source coding scheme to control the bitrate adaptively.

Moreover, video quality can degrade significantly when objects move rapidly. In compressed video, rapid movement of large objects induces multiple changes in pixel values between successive frames, which reduces video quality significantly. In addition, the radio signal attenuation associated with wireless networks introduces a higher packet loss rate. Packet loss may occur in the critical building blocks of the coding structure, such as I-slices and I-frames. Packet loss errors can destroy intra MBs and propagate to subsequent video frames. The combination of large object motion and errors inherent in wireless networks reduces video quality dramatically.

Figure 1 illustrates such deterioration with respect to object dynamics under wireless and wired network conditions. Two surveillance video frames containing different sizes of objects moving at similar speed are shown in Figure 1a. Here, the network environment is simulated using network simulator version 2 (NS-2) [7]. The peak signal-to-noise ratio (PSNR) shown in in Figure 1a is reduced when object motion occurs. For both moving object events, the reduction is more obvious under wireless conditions. In addition, the larger object (i.e., the car) produces sharper deterioration. The negative effect of object movement relative to speed is shown in Figure 1b. In this video, the same object moves at different speeds, that is, walking and running. A serious decline in PSNR is evident when the object moves at higher speed, particularly under the wireless condition.
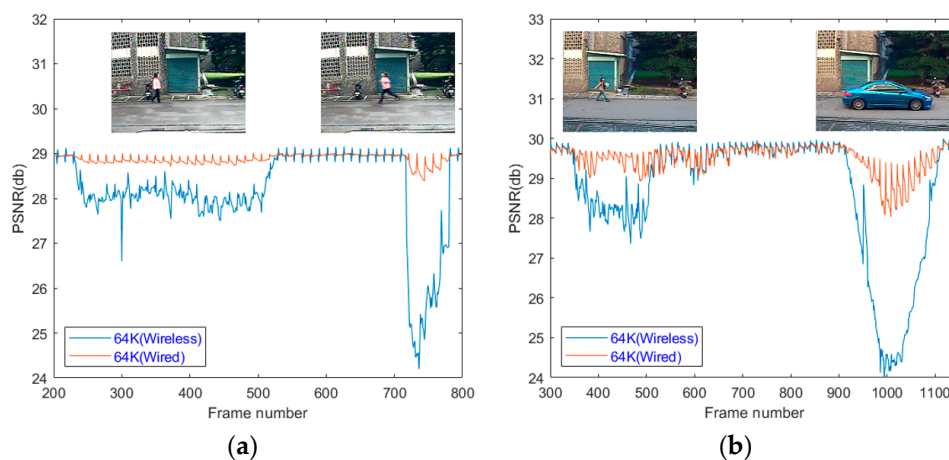

Figure 1. Degraded video quality with respect to object and network conditions: (**a**) moving objects with different speeds and (**b**) moving objects with different sizes.

Providing reliable video quality over networks can be achieved by bitrate control approach that can be classified to be constant bitrate (CBR) and variable bitrate (VBR) controls. Some adaptive methods employ low-level metrics to measure frame complexity and adaptively control bitrates. Many such methods use the mean absolute difference (MAD) of predictive residuals to measure texture coding complexity [8,9]. Low-level metrics predict frame bitrate allocation recursively by calculating QPs and rate-distortion optimization (RDO). However, such low-level metrics are not robust and are very susceptible to interference due to noisy motion, such as scene changes. High-level metrics use object dynamics (content-driven) to acquire more frame information to calculate frame complexity. To reduce the extra computational complexity of High Efficiency Video Coding (HEVC) intra encoding, a previous study [10] proposed a content-driven adaptive scheme that depends on frame texture and

combines smaller prediction units into larger units to reduce time complexity. That method can decrease RDO encoding complexity; however, it can only be applied to a single type of partitioning structure.

Traditional bitrate control methods are designed with the only goal to fit the network throughput constraint by CBR or VBR, but not with the goal to give constant and reliable video quality for intelligent applications [11,12]. Considering the video quality degradation with respect to the dynamics of picture complexity, moving objects and wireless networks, constant quality control becomes a more challenging goal.

In this paper, a constant quality control method is proposed for surveillance video streaming over wireless networks. We propose an object-based source coding method that adapts the bitrate according to object dynamics relative to size and speed. The relationship between video quality degradation and object dynamics is first analyzed and modeled by a linear system. A set of linear models that correspond to different bitrates is then developed to predict quality reduction relative to bitrate increments. When a moving object is detected, this model predicts the encoding bitrate increment to enhance video quality. A robust estimator is applied to estimate the parameters of the linear regression because of the outliers in the statistical modeling.

The remainder of this paper is organized as follows. Section 2 reviews related source coding schemes. The adapted object-based coding method is presented in Section 3. Experimental results are provided in Section 4, and conclusions and suggestions for future work are presented in Section 5.

2. Related Works

Here, we review source coding techniques that control bitrates for video quality, including the rate-distortion (R-D) model, QP adjustment, region of interest (ROI) and frame layer control methods.

Bitrate control methods can be divided into constant bitrate (CBR) and variable bitrate (VBR) approaches. Generally, live streaming broadcasting over the Internet adapts CBR. The sender chooses a constant bitrate to encode and transmit video data, and the receiver does its best to receive video data. However, this approach cannot guarantee video quality because the actual bandwidth or the network transmission quality are other factors. Hence, using CBR to achieve constant quality video transmission is difficult. Fortunately, VBR can be used to adjust bitrate dynamically based on the demanding conditions. Several rate control strategies have been proposed in VBR to provide constant video quality, such as QP parameters adjustment, RD function pre-design, 2-pass optimization design, structural similarity (SSIM)-base analysis and others. Most algorithms use information about group of pictures (GOP), frame layer and macro-block (MB) layer.

Adjusting bitrate on the fly based on the actual demanding conditions can help VBR achieve the result of a constant quality video. Han et al. [13] adopted the VBR approach to adjust image quality. These methods are the hope of achieving consistent visual quality or constant PSNR. In [14], a VBR incremental rate control algorithm to reduce the computational complexity of H264/AVC was proposed. It combined picture complexity estimation and an exponential rate-complexity-quantization model together in the design of a H.264/AVC coding algorithm. This paper also proposes a buffer control method to prevent buffer overflow and underflow by adjusting the quantization parameter. Wang et al. [15] proposed SSIM-motivated perceptual two-pass VBR rate control algorithm for HEVC. They used video quality assessment (VQA) to optimize perceptual video coding protocols.

Many rate control schemes that use source coding technology to adjust the bitrate have been proposed for video transmission. Thus, source coding is considered an efficient approach to improve the quality of streaming videos. The R-D model, QP step size determination, and MB size prediction are source coding techniques to enhance the flexibility of rate control and guarantee video quality. In contrast, ROI methods are high-level techniques that employ object dynamics to guarantee quality.

A testbed and computed motion activity to achieve a real-time variable frame rate for live video has been proposed [16], and frame layer control, which is considered a low complexity method, can achieve an approximately real-time effect. Motion detection is done, but high-level information, such as moving object, is not. Some methods use frame selection to reduce the overall amount of

image transmission, which enhances meaningful information and improves image transmission quality. Another study [11] developed a cooperative framework that involves semi-dynamic environment processing and simple event surveillance. That method used semantic filtering to perform frame selection to adjust frame transmission for back-end monitoring and querying, which was able to achieve better image quality monitoring results over limited bandwidth. Although that method can reduce the number of transmitted images, objects and scenes are restricted. When moving objects appear, it is impossible to determine whether image quality has been enhanced.

Many source coding schemes for encoders are based on the R-D model, and bitrate and particularly quantization are important factors in this model. Although the quadratic R-D model is more accurate than the linear R-D model, the quadratic R-D model has higher computational complexity. A previous study [17] modified the relationship between QPs and the rate quantization step size from non-linear to linear mode, and also proposed a complexity-adjustable two-pass rate control scheme based on statistical and theoretical analyses of the quantization scheme. A recent study [18] proposed a perceptual distortion-based RDO video coding scheme for HEVC, in which a new SSIM Lagrange multiplier λ was computed for RDO to decide the optimal coding unit size.

At the MB layer, to fit given target bits accurately, the QPs must be adjusted to fit video transmission. A previous study [19] analyzed the relationships among the QPs, MAD, and the coded bits to propose a weighted-window model to reduce computational complexity at the MB layer, which is critical to construct an accurate rate-quantization (R-Q) model that can achieve high bitrates. Another previous study [20] exploited both spatial and temporal correlations among neighboring MBs and proposed a context-adaptive model parameter prediction scheme. This scheme improves estimation accuracy of the MAD of texture with R-Q model-based MB layer rate control for real-time low bitrate applications. However, in high-motion videos, the temporal correlations among the MBs between two contexts cannot provide sufficient information to predict QPs for this R-Q model. It is important to remember that RDO is critical in video compression. A previous study [21] found that R and the Lagrange multiplier λ can provide more robust correspondence than the R-Q model and proposed a λ-domain model based on an R-λ model that did not require complex iterative computation. To reduce the high computational complexity of the rate control algorithm in HEVC. Atta, R. [22] proposed a single-pass joint temporal-quality rate control algorithm. In this algorithm, the predefined target bitrate at each quality layer used in the existing rate control algorithm was not used. Instead, the overall target bitrate adaptively distributed between the quality layers having the same and different temporal resolutions was used. A set of empirical values was first derived to estimate the initial values of the R-D model parameters for the joint temporal and quality layers. Then, a prediction mechanism to update these model parameters during the encoding process was then presented to further improve rate control performance.

ROI methods can enhance image quality in target parts of a video while the remaining parts are transmitted at low quality. A previous study [23] used dynamic background modeling to divide MBs into foreground and background regions and proposed strategies to increase the transcoding speed of surveillance video. Another study [24] used a superpixel-based MB selection method to obtain accurate shape information when detecting motion objects using a low bit ROI coding system. This method has been used to monitor road traffic [25]. In addition, an R-λ model has been proposed [26] to present an ROI scheme based on both frame and coding tree unit levels (where QPs are computed independently for different regions) at HEVC standards.

Adaptively adjusting the R-Q model to obtain a group of pictures, frame bit allocation, and QP values can enhance ROI video quality. Another proposed method [27] added coefficient ω to each frame type and calculated QPs to control the extent to which the ROI was protected, where ROI protection increases as ω increases. In addition, a method that uses a ROI with rate control technology to balance video quality and data size has been proposed [28]. Bitstream length and quantization step size can be expressed approximately as a linear function to predict frame-level bit allocation and

ROI QP determination. The adaptive updating model uses a linear regression method to update the number of bits of each target frame and a corresponding quantization step size to enhance ROI quality.

An adaptive method based on an optimized effort strategy that can achieve constant surveillance video transmission quality in consideration of video content should be investigated. Precise and rapid dynamic adjustment of the image quality of moving objects in videos transferred over a wireless network is the key concept of this study. Based on our literature survey, frame rate control source coding parameters, MB size prediction, ROI, and QP determination can be used to adjust the bitrate and enhance video quality. However, a high-level metric using an ROI to adjust source coding parameters has not been well studied to date.

Frame layer rate control is a high-level adaptive quality control method. It uses frame selection approach to reduce the overall amount of video transmission, which enhances meaningful information to improve video quality. Lam et al. [11] created a cooperative framework that uses semantic filtering to select frames, which can achieve better image quality monitoring results over limited bandwidth. Fiandrotti et al. [12] proposed a content-adaptive traffic prioritization strategy for H.264/SVC communications over IEEE 802.11e wireless networks. This strategy estimated the perceptual impact of data losses in the different types of enhancement layers for a large set of videos first and proposed a content-adaptive traffic prioritization strategy based on the identification of the most important parts of the enhancement layers of the video sequence by means of a low complexity macroblock analysis process. If a layer, temporal or spatial, has motion detected, it should have high priority to preserve than others. The above two studies pointed out that by scalable coding and by dropping different types of scalability layers or frames based on the content characteristics could achieve post-encoding to adapt content dynamics.

High-level quality control methods adopt content dynamics and picture complexity to help the coding of bitrate, which has less computational demanding than VBR scheme. Dynamics of moving objects is a kind of high-level metrics that can greatly improve the high-level quality control with low-level metrics, including the motion detection used in [12].

3. Proposed Adaptive Coding Method Using Object Dynamics

The proposed adaptive bitrate control method employs a statistical model that describes the linear relationship between PSNR reduction and object dynamics. In this method, the bitrate is controlled adaptively when a new object appears in the video and is increased to sustain PSNR quality based on the prediction of the statistical model. This section explains the proposed model estimation algorithm and the prediction method employed to adaptively increase the bitrate.

3.1. Modeling the Statistical Relationship

Assume p random variables, X_i, $1 \leq i \leq p$, corresponding to the characteristics of object dynamics. Video quality is set as response variable Y being a function of X_i as follows:

$$Y = f(X_1, \ldots, X_p) + e, \tag{1}$$

where e is a residual term representing modeling error and the random effect of the system. The response surface $E(Y|x_1, \ldots, x_p) = f(X_1, \ldots, X_p)$, where $(x_1, \ldots, x_p) \in \{(X_1, \ldots, X_p)\}$, can be explained by parameters $\beta_0, \beta_1, \ldots, \beta_p$. If regress equation is β (i.e., the regression coefficient) system of linear equations, we can describe a multi-variable linear regression model as follows:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + e, \tag{2}$$

where $\beta_i = \partial E(Y|x_1, \ldots, x_p)/\partial x_i$, $i = 1, 2, \ldots, k$.

In this linear regression model, Y is the PSNR that numerically represents video quality and X_i denotes an object's size and speed, that is, $p = 2$.

Typically, the least squares method is used to obtain the optimal solution of a linear regression model. Here, consider the residual value between observed values y_i and the estimated value $\hat{y}_i$ expressed as $e_i = y_i - \hat{y}_i$. The sum of square errors (SSE) can be written as follows:

$$\sum_{i=1}^{n}(e_i)^2 = \sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{1,i} - \beta_2 x_{2,i})^2 \tag{3}$$

The least squares method is used to solve $\beta_0, \beta_1, \beta_2$ to obtain the minimum SSE value. Note that this method is a simple, efficient, and common estimator.

However, least squares estimates can perform poorly when the error distribution is not normal. Please note that outliers are sample values; however, they may be correct but should always be checked for transcription errors. In addition, they can seriously interfere with standard statistical methods. For a more precise estimation, we require a robust regression method to enhance our linear regression that is less sensitive to outliers.

The most common and robust regression method is M-estimation [29,30]. M-estimation uses an objective function and a weight function to enhance regression. In addition, M-estimation uses an M-estimator to minimize the objective function. For example, the least squares method's objective function is e_i^2 and weight function is 1. We use Tukey's bisquare (biweight) [31] algorithm to adjust our estimation. The bisquare defines new objective and weights function, which are expressed as follows:

$$O(e) = \begin{cases} \frac{k^2}{6}\left\{1 - \left[1 - \left(\frac{e}{k}\right)^2\right]^3\right\} & for\ |e| \le k \\ \frac{k^2}{6} & for\ |e| > k \end{cases} , \tag{4}$$

$$W(e) = \begin{cases} \left[1 - \left(\frac{e}{k}\right)^2\right]^2 & for\ |e| \le k \\ 0 & for\ |e| > k \end{cases} , \tag{5}$$

where e is the previously mentioned residual and k is a positive constant. We use multiple coefficients of determination typically denote R^2 to evaluate the goodness of fit of our linear regression. Here, we define the sum of squares total as $SST = \sum_{i=0}^{n}(y_i - \overline{y})^2$, where $\overline{y}$ is the sample mean of y_i, and the sum of squares due to regression is expressed as $SSR = \sum_{i=0}^{n}(\hat{y}_i - \overline{y})^2$. The total difference of variable y_i can be expressed as $y_i - \overline{y} = \hat{y}_i - \overline{y} + y_i - \hat{y}_i$, i.e., SST = SSR + SSE. This is expressed as follows:

$$\sum_{i=0}^{n}(y_i - \overline{y})^2 = \sum_{i=0}^{n}(\hat{y}_i - \overline{y})^2 + \sum_{i=0}^{n}(y_i - \hat{y}_i)^2 \tag{6}$$

Please note that R-square is caused by the *SSR* in the *SST* proportion:

$$R^2 = \frac{SSR}{SST} = \frac{\sum_{i=0}^{n}(\hat{y}_i - \overline{y})^2}{\sum_{i=0}^{n}(y_i - \overline{y})^2} = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=0}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=0}^{n}(y_i - \overline{y})^2} \tag{7}$$

Here, an adjusted R-square measure method is used to solve the R-square method for a small sample or independent variable increase, which can cause regression degrees of freedom decrease will get higher R-square value. This can be expressed generally as follows:

$$R^2_{adj} = 1 - \left(1 - R^2\right)\frac{n-1}{n-p-1} = 1 - \frac{SSE(n-1)}{SST(n-p-1)} \tag{8}$$

Here, variable n is the observed value number (i.e., the regression equation input data points) and p is the number of independent variables. Please note that the adjusted R-square value may be less than zero, and values that are closer to 1 indicate a better fit.

Figure 2a shows the moving object's area, speed, and PSNR results obtained at a 64 Kbps bitrate after robust linear regression. Here, the SSE is 0.7183, and the adjusted R-square is 0.6202. Under different coding bitrates, we use the same reference moving object in 64 Kbps video to build robust linear regression equations for each bitrate using the above mentioned method.
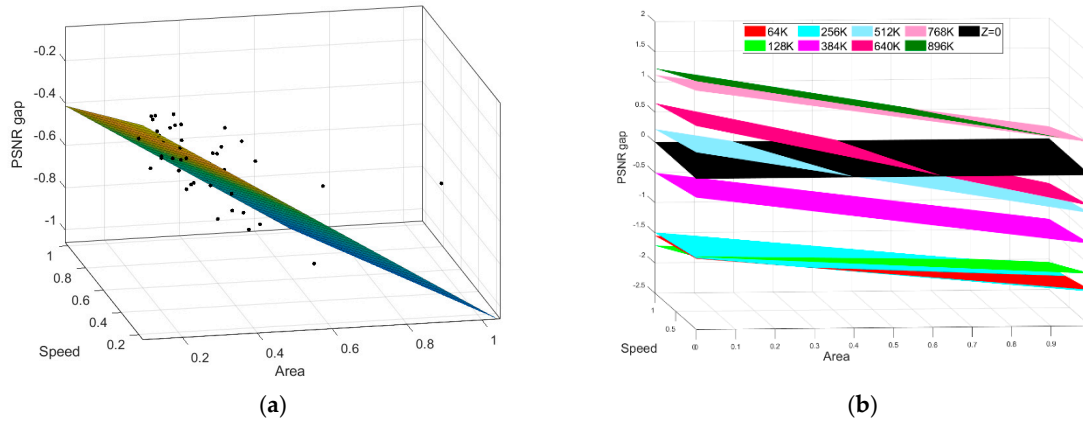

(**a**)   (**b**)

Figure 2. The moving object's area, speed, and PSNR results. (**a**) Plane generated from 43 moving objects using robust linear regression at 64 Kbps coding bitrate and (**b**) Planes created from each individual coding bitrate and the $z = 0$ plane (black) show the same PNSR as the 64 Kbps video transferred by the wired network. In order not to make the graph look too cluttered, the layer interval is set to 128 Kbps.

Using different bitrates, we compared the PSNRs to that of the 64 Kbps video transferred by a wired network. By observing those equations synchronously, we created the results shown in Figure 2b. Here, we can see that the planes created by those equations are nearly parallel. Figure 2b also shows that the area and the speed of the moving object are critical points that affect the PSNR in a linear relation. The $z = 0$ plane (black) shows the same PNSR as the 64 Kbps video transferred by the wired network.

3.2. Adaptive Bitrate by Prediction

The adaptive bitrate function is given as follows:

$$B^t = B^{t-1} + \varphi(A_i, S_i), \tag{9}$$

where the area and speed of object i are denoted A_i and S_i, respectively. $\varphi(A_i, S_i)$ is an adaptive parameter function that is described in the following. B^t represents the current coding bitrate at time t. The bitrate of B^t is determined by the previous bitrate B^{t-1} with the $\varphi(A_i, S_i)$ calculation.

When the moving object appears, its area and moving speed will degrade the video quality. To maintain the same quality as the previous time point, the coding bitrate must be increased; however, the bitrate cannot be greater than the maximum available bandwidth. In contrast, when the moving object slows down, stops, or disappears, a higher coding bitrate is not required to maintain video quality; thus, the bitrate should be reduced to the default value.

The proposed coding control method increases the coding bitrate to the necessary level rather than simply increasing it to the maximum. Please note that wired network transmission and the default coding bitrate are considered the standard. When a moving object appears in the video in the wireless network environment, we attempt to maintain quality that is equal to the established standard. Thus, we can achieve constant quality in every situation. However, tuning the adaptive parameter function $\varphi(A_i, S_i)$ remains a very important issue.

To maintain constant video quality when a moving object presents, we input the speed and area values into the model to determine which plane is closest to the $z = 0$ plane. By doing so, we are able

to find the bitrate of the closest plane that can be used to maintain the constant video quality. Thus, we define $\varphi(A_i, S_i)$ as follows:

$$\varphi(A_i, S_i) = \hat{B}_t - B^{t-1}\mathbf{a} = 1 \tag{10}$$

Here, $\hat{B}_t$ is the predictive bitrate of object *i* defined as follows:

$$\hat{B}_t = argmin(dP_1(A_i, S_i), \ldots\ldots, dP_n(A_i, S_i)), \tag{11}$$

where $dP_j(A_i, S_i)$ is the distance of object *i* from $z = 0$ to the *j*th bitrate regression plane.

Figure 3 shows that, when moving objects appear, sufficient bandwidth for moving objects can be obtained using Equations (9)–(11) could let 64 K wireless has almost the same image quality as 64 K wired. The proposed method determines the required bitrate according to the moving object's area and speed for saving network bandwidth. In Figure 4, the blue diamond line shows the network bandwidth used by the proposed method, and the pink star line shows the efficient packet delivery methods used. Generally, the other methods will adjust to the maximum bandwidth to improve image quality when the moving object appears.
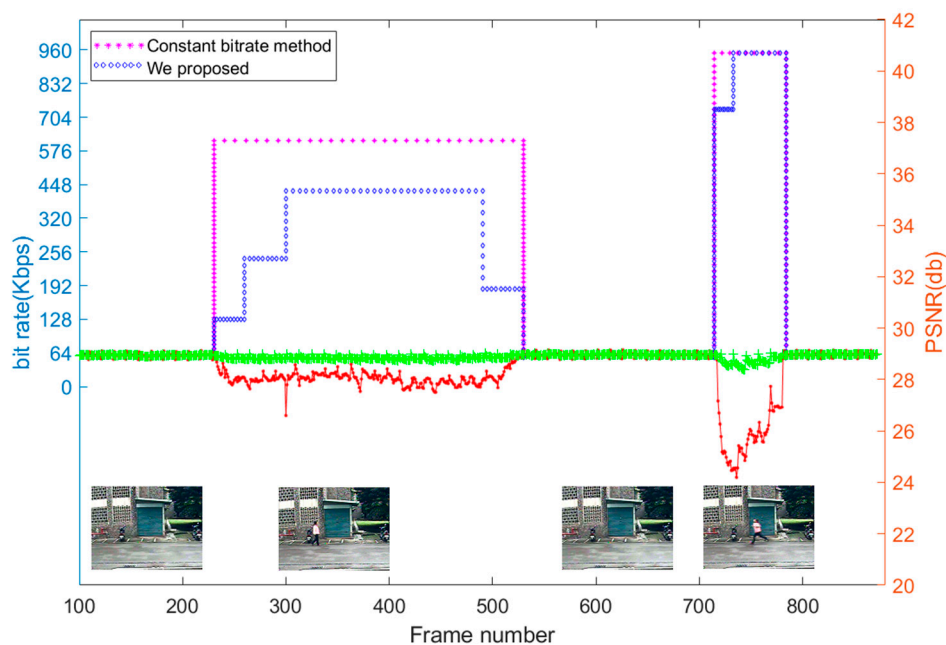

Figure 3. Using the adaptive algorithm, the blue diamond line shows that the bitrate changes when a moving object appears.
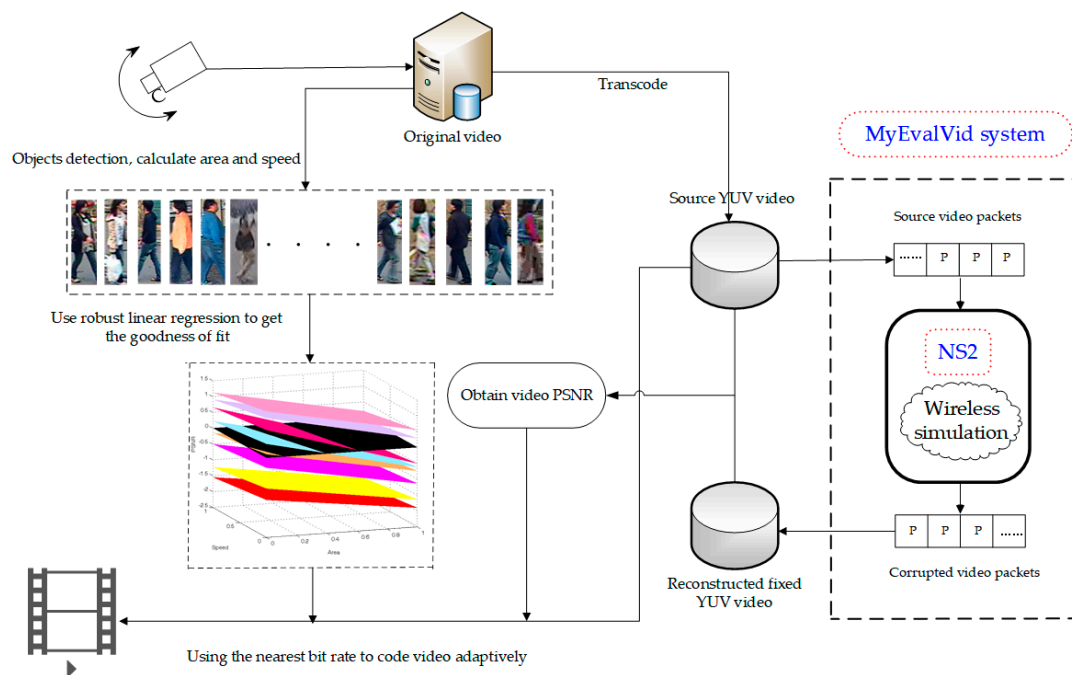
Figure 4. The experimental framework of our system.

4. Experimental Results

We set up an outdoor surveillance camera to record all training and experimental videos by ourselves. These videos' resolution are 640×480 and 30 frames per second. Their codecs are set to MPEG-4, the group of picture (GOP) size is 9 and quantization scale is 31. There are two B frames between the I and P frame or the P and P frame. We randomly recorded 24 videos and most of the people in the videos are our lab's members and students. We use an object tracking algorithm [32] to automatically mark moving objects.

In addition, we used the IEEE 802.11 wireless environment. The NS-2 simulator was used in our simulation experiments. We referenced Enhancement of EvalVid (MyEvalVid) [33] to simulate video transmission in a wireless network environment. MyEvalVid is a set of tools consisting of EvalVid [34] and NS-2. EvalVid is a multimedia quality assessment tool. It provides an architecture that allows verifying the impact of the proposed network-related issues associated with the quality of multimedia streaming via physical or analog networks. Since the analog network model provided by EvalVid is too simple, it was necessary for MyEvalVid to add three agent programs, including MyTrafficTrace, MyUDP, MyUDPSink, to provide a more comprehensive multimedia quality assessment in conjunction with running NS-2. In the NS-2 simulation environment, the maximum transfer unit is set to 1 Mbps. For the packet lost and jitter conditions on the wireless network, the random uniform error model is used and the error rate is set to 0.01. The transmission of the network was done through multicasting. Figure 4 shows the experimental framework.

To establish the linear regression model, we randomly recorded 24 videos (Figure 5a shows some of the training videos) and captured 43 video scenarios with a human as the moving object as training samples (Figure 5b shows some of the moving objects). The initial coding bitrate was 64 Kbps, which increased by 64 Kbps per unit until the coding bitrate reached 960 Kbps. Through that process, we could obtain video in which each frame had a different PSNR.

(**a**)

(**b**)

Figure 5. Training videos scenes. (**a**) Training videos and (**b**) moving objects in training videos.

Many studies have been conducted on video quality assessment [35,36]. The PSNR is most commonly used as a measure of the quality of the reconstruction of lossy compression codecs [37]; thus, we used it as an evaluation standard in our experiments. It is most easily defined via the mean squared error, which is defined as follows:

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{i=m-1} \sum_{j=0}^{j=n-1} [I(i,j) - P(i,j)]^2 \tag{12}$$

Which for two $m \cdot n$ images I (original image) and P (approximate image).
The *PSNR* is defined as follows:

$$PSNR = 10 \cdot log_{10}\left(\frac{(MP)^2}{MSE}\right), \tag{13}$$

where *MP* is the maximum possible pixel value of the image.

We recorded 17 videos (Figure 6a shows some of the experimental video scenes) and captured 57 scenarios in which the moving object has different area and speed as experimental samples (Figure 6b shows some of the moving objects). We calculated the area and speed of each moving object, using the framed and tracked software tool.
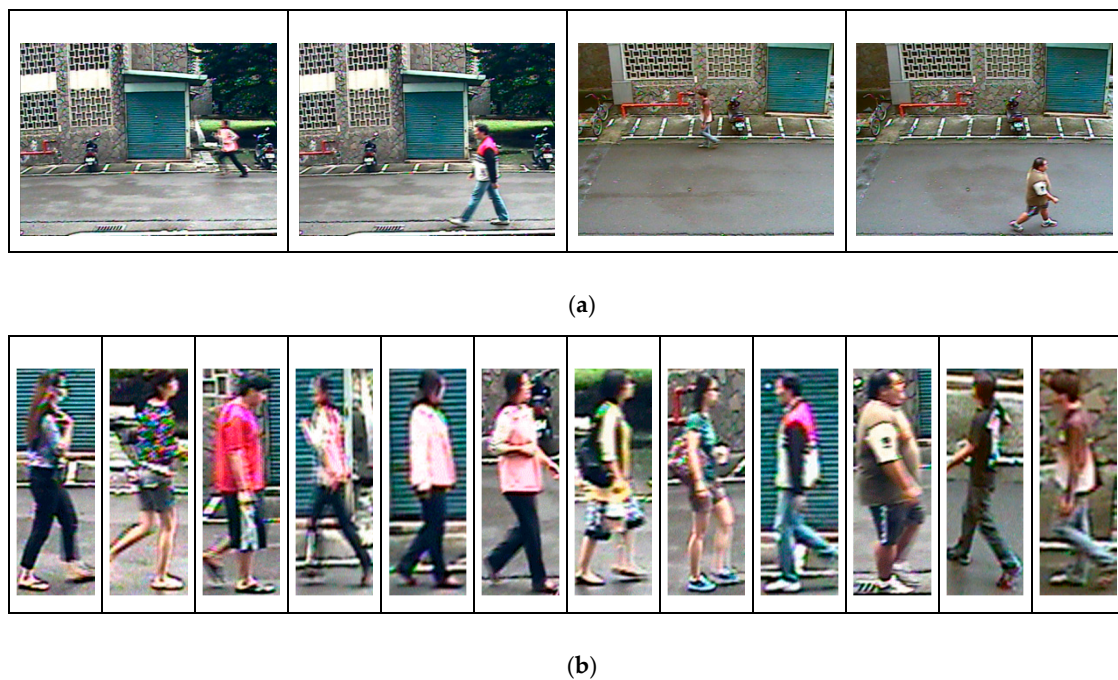
(**a**)

(**b**)

Figure 6. The experimental video scenes. (**a**) Experimental videos and (**b**) moving objects in experimental videos.

The speed and area values were input to the linear regression model, and we generated the average PSNR value for each moving object at different bitrate coding values. By comparing the obtained values to the standard values, we can obtain an estimated coding bitrate for each moving object with the most similar PSNR. Then, we compared the estimated value to the real coding bitrate with the closest PSNR in the wireless transfer environment. Figure 7 shows an error distribution chart.

There are 57 scenarios in our experiment. We used MPEG-4 as a codec that uses a standard baseline profile [38]. We set every 64 Kbps as a coding unit. The 57 scenarios were compared using the average PSNR of the 64 K wired network. The error distribution chart Figure 7 shows the difference between the coding bitrate estimated by the linear regression model and the original coding bitrate for each scenario. The reduced value is positive, which means that our model predicts coding bitrate better than the original method in terms of obtaining the same video quality. In addition, the linear regression method was used. The results show that 28 scenarios running at lower coding bitrate can achieve the required image quality; 11 scenarios running at the same bitrate have no change in image quality, and 18 scenarios need to run on a larger coding bitrate to complete the task. The results of our experiment show that for the 18 scenarios, we need to increase the bitrate on average by 96 Kbps when using our method, whereas for the 28 scenarios we can reduce the bitrate by 194.29 Kbps on average when using our linear regression model.

The moving object in scenario 9 has a small area and slow speed, and its estimated coding bitrate is 512 Kbps, the same as the exact value. In Figure 8, the x-axis is the frame number at which the moving object appears, and the y-axis shows the PSNR value of each frame. The red line (+) is the PSNR value for coding at 64 Kbps without a network transfer. The green line (*) is the PSNR value for coding at 64 Kbps transferred through the network environment. As can be seen, when the moving object appears, the PSNR values decrease significantly. The blue line (−) represents the estimated PSNR value for a 512 Kbps coding bitrate transferred by the network environment. As shown, the blue line is closer to the red line than the green line.
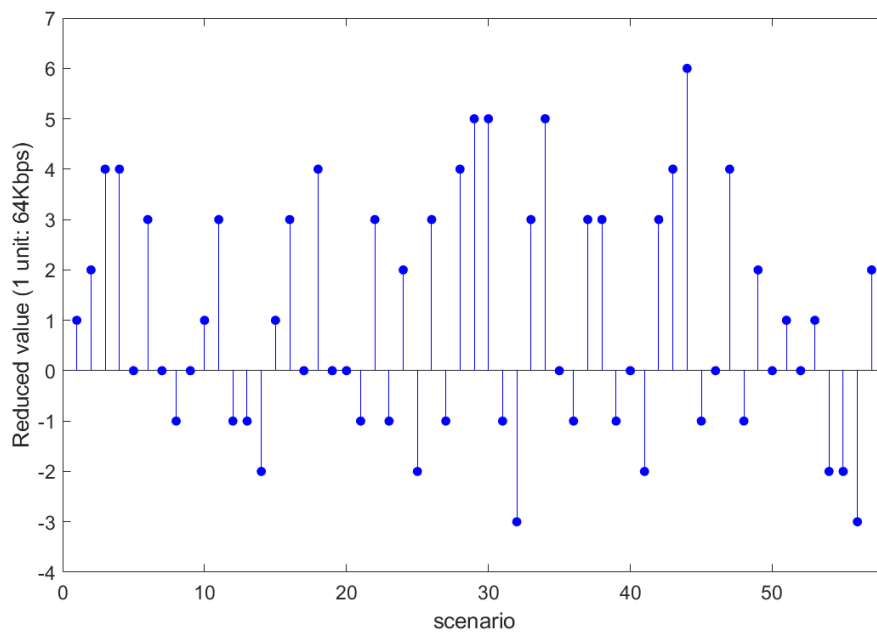
Figure 7. The error distribution chart shows the difference between the coding bitrate estimated by the linear regression model and the original coding bitrate for each scenario. The positive reduced value means that the bitrates predicted by our model is better than those predicted by the original method. Each unit in the reduced value is 64 Kbps. Taking scenario 3 as an example, our method saves 256 Kbps.

We captured frames 119, 154, and 236 of the moving object from the video and show them in Figure 8a. In Figure 8a, the bottom row shows the distortion result of those frames at a coding bitrate of 64 Kbps under network transfer, and the upper row shows the results of the same frames at an estimated coding bitrate (i.e., 512 Kbps) and network transfer. The video quality obviously improved by increasing the coding bitrate.

Figure 8b shows the results of scenario 15. Here, the area of the moving object is similar to that of scenario 9 but the object has slower speed. Please note that the estimated and real values show no differences.
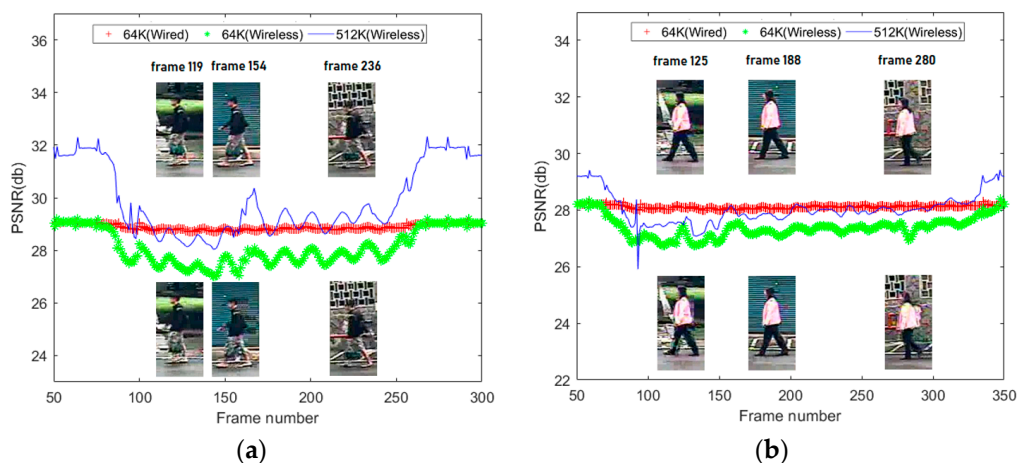

(a)

(b)

Figure 8. (**a**) The red line (+) represents the PSNR value for coding at 64 Kbps without network transfer. The green line (*) represents the PSNR value for coding at 64 Kbps transferred through the network environment. The blue line (−) represents the estimated PSNR value for a 512 Kbps coding bitrate transferred by the network environment. As shown in the figure, the blue line is closer than the green line to the red line. It is clear that frames 119, 154, and 236 show a better image quality. (**b**) Another set of the moving objects. Similarly, the result shows that the moving objects at the frame 125, 188, and 280 positions have better image quality except that their moving speed is slower.

Figure 9a shows the experiment results of scenario 38. Here, there is a 192 Kbps difference (i.e., three units). The moving object has a large area and higher speed. When the moving object appears in the video, the PSNR value is reduced significantly. When we input this scenario into our model, we obtain an estimated coding bitrate of 768 Kbps (blue line). Please note that there is a significant gap between the blue and red lines, where the red line is the experimental goal. In theory, it should fall in 960 Kbps, even if we set the limit to 1 Mbps. We plot the PSNR value of the 960 Kbps coding rate under the network transfer condition with the magenta line (-o-), which shows that most of its pattern overlaps the 768 Kbps coding rate result. Here, only two sectors had obvious differences. The proposed model uses the average PSNR value for comparison. The average difference between the 768 Kbps and 960 Kbps results obtained by our model is 0.11, and the line of the 768 Kbps result is closer to the red line (+). This example shows that the proposed model may result in significant differences for some scenarios; however, the average value it can obtain shows very little difference compared to the real average PSNR.
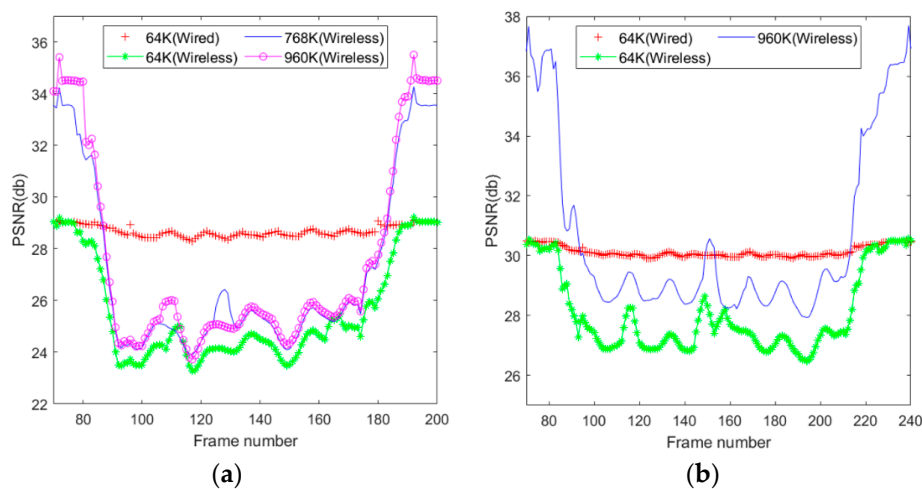

Figure 9. (**a**) In scenario 38, the moving object occupies a large area and moving at higher speed. When a moving object appears in the video, the PSNR value is reduced significantly. When we input this scenario into our model, we obtain an estimated coding bitrate of 768 Kbps. See the blue (−) line. Please note that there is a significant gap between the blue (−) and red lines (+). The red line is our experimental goal. In theory, it should fall in 960 Kbps, even if we set the limit to 1 Mbps. The magenta (-o-) represents a coding rate of 960 Kbps and shows that most of its pattern overlaps the 768Kbps coding rate result. The proposed model uses the average PSNR value for comparison. (**b**) The area occupied by the moving object is a little bigger than that used in (a) but the object is moving at slower speed. The gap shown between the blue (−) and red (+) line is much smaller than (a).

Figure 9b shows the results for scenario 56. Here, the moving object has a large area but a very slow speed. Compared to a moving object with same area but higher speed, this scenario demonstrates better video reconstruction results. We estimated that the coding bitrate in this experiment uses the largest coding unit (i.e., 960 Kbps).

Figure 10 shows the frames in which the moving object appears. The first row is the frame number, and the second row shows the original images (64 Kbps without network transfer). The third row shows the reconstruction images after network transfer with coding at 960 Kbps by the proposed model. The fourth row shows the reconstruction images after network transfer with coding at 64 Kbps. Here, we found that the images reconstructed using a coding value obtained by the proposed model have a very similar quality compared to the original images.
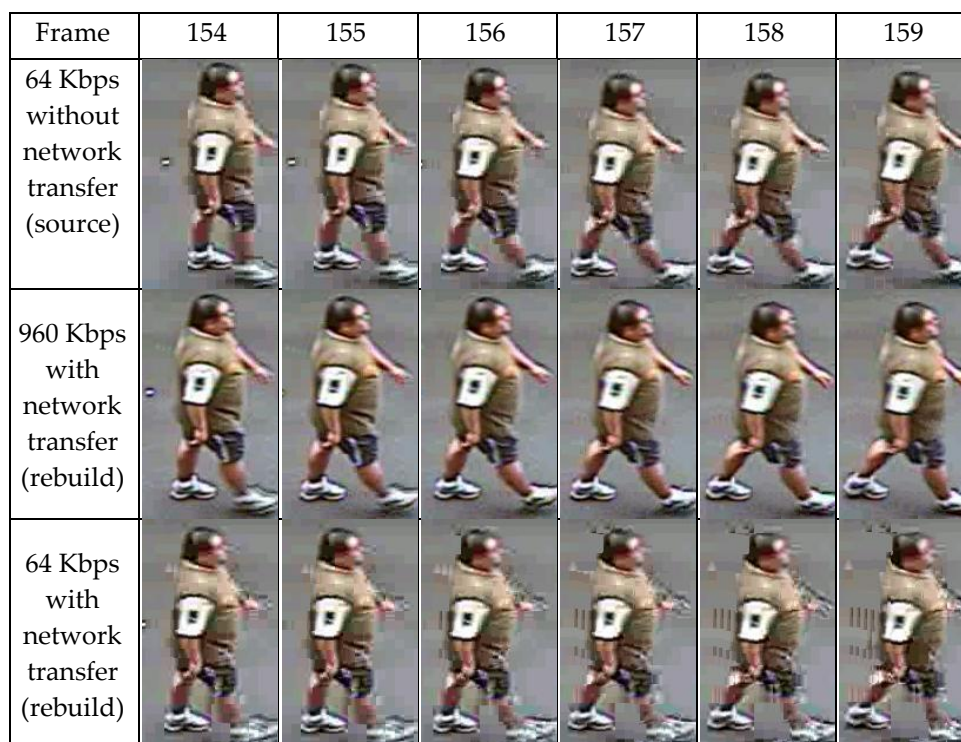
Frame	154	155	156	157	158	159
64 Kbps without network transfer (source)						
960 Kbps with network transfer (rebuild)						
64 Kbps with network transfer (rebuild)						

Figure 10. Successive frames in scenario 56 (frames 154–159).

5. Conclusions

In this paper, we proposed a new adaptive source coding method to enhance the quality of surveillance video transmission under limited bandwidth conditions. The advantage of the proposed method is the reduction of both network bandwidth usage and decoder complexity, and the range of coding bitrate can be adaptively determined as required. We develop a linear model to represent the relationship between video quality and moving object's dynamics. The relationship was learned from a regression algorithm. This learned model could be applied to enhance video quality, by adaptively adjusting coding bitrate according to object dynamics. Our experimental results show that more than 68% of the scenarios can save coding bitrate and can be reduced by 194.29 Kbps on average.

The proposed method is a general framework that can be extended to any type of network environment, with new learning data provided for the specific network environment. However, more object dynamics, such as background changes, texture complexity, and the color of moving objects, can be incorporated into the proposed linear model and could enhance the accuracy of the bitrate prediction. Nonlinear modeling and more advanced regression methods are also promising for future work.

Author Contributions: Conceptualization, T.-M.P. and Y.-K.W.; methodology, T.-M.P. and Y.-K.W.; software, T.-M.P.; validation, T.-M.P. and Y.-K.W.; formal analysis, T.-M.P. and Y.-K.W.; writing—original draft preparation, T.-M.P.; writing—review and editing, T.-M.P., K.-C.F. and Y.-K.W.; supervision, K.-C.F. and Y.-K.W.

References

1. Wang, Y.K.; Chen, H.Y. Intelligent Mobile Video Surveillance System with Multilevel Distillation. *J. Electron. Sci. Technol.* **2017**, *15*, 133–140.
2. Fan, C.T.; Wang, Y.K.; Huang, C.R. Heterogeneous information fusion and visualization for a large-scale intelligent video surveillance system. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *47*, 593–604. [CrossRef]

3. Pan, Z.; Jin, P.; Lei, J.; Zhang, Y.; Sun, X.; Kwong, S. Fast reference frame selection based on content similarity for low complexity HEVC encoder. *J. Vis. Commun. Image Represent.* **2016**, *40*, 516–524. [CrossRef]

4. Dey, B.; Kundu, M.K. Enhanced Macroblock Features for Dynamic Background Modeling in H.264/AVC Video Encoded at Low Bitrate. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *28*, 616–625. [CrossRef]

5. Lee, C.; Jung, Y.; Lee, S.; Oh, Y.; Kim, J. Real-Time Frame-Layer H.264 Rate Control for Scene-Transition Video at Low Bit Rate. *IEEE Trans. Consum. Electron.* **2007**, *53*, 1084–1092. [CrossRef]

6. Chen, X.; Lu, F. A reformative frame layer rate control algorithm for H.264. *IEEE Trans. Consum. Electron.* **2010**, *56*, 2806–2811. [CrossRef]

7. Network Simulator-2. Available online: http://www.isi.edu/nsnam/ns/ (accessed on 11 January 2019).

8. Chen, J.Y.; Chiu, C.W.; Li, G.L.; Chen, M.J. Burst-aware dynamic rate control for H.264/AVC video streaming. *IEEE Trans. Broadcast.* **2011**, *57*, 89–93. [CrossRef]

9. Choi, H.; Yoo, J.; Nam, J.; Sim, D.; Bajic, I.V. Pixel-wise unified rate-quantization model for multi-level rate control. *IEEE J. Sel. Top. Signal Process.* **2013**, *7*, 1112–1123. [CrossRef]

10. Khan, M.U.K.; Shafique, M.; Henkel, J. An adaptive complexity reduction scheme with fast prediction unit decision for HEVC intra encoding. In Proceedings of the 2013 IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 1578–1582.

11. Lam, K.Y.; Chiu, C.K. The design of a wireless real-time visual surveillance system. *Multimedia Tools Appl.* **2007**, *33*, 175–199. [CrossRef]

12. Fiandrotti, A.; Gallucci, D.; Masala, E.; De Martin, J.C. Content-adaptive traffic prioritization of spatio-temporal scalable video for robust communications over QoS-provisioned 802.11e networks. *Signal Process Image Commun.* **2010**, *25*, 438–449. [CrossRef]

13. Han, B.; Zhou, B. VBR rate control for perceptually consistent video quality. *IEEE Trans Consum. Electron.* **2008**, *54*, 1912–1919. [CrossRef]

14. Sun, Y.; Zhou, Y.; Feng, Z.; He, Z.; Sun, S. Incremental rate control for H.264/AVC video compression. *IET Image Process.* **2009**, *3*, 286–298. [CrossRef]

15. Wang, S.; Rehman, A.; Zeng, K.; Wang, J.; Wang, Z. SSIM-motivated two-pass VBR coding for HEVC. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *27*, 2189–2203. [CrossRef]

16. Bajic, I.V.; Ma, X. A testbed and methodology for comparing live video frame rate control methods. *IEEE Signal Process. Lett.* **2011**, *18*, 31–34. [CrossRef]

17. Ma, S.; Gao, W.; Lu, Y. Rate-distortion analysis for H.264/AVC video coding and its application to rate control. *IEEE Trans. Circuits Syst. Video Technol.* **2005**, *15*, 1533–1544. [CrossRef]

18. Lee, B.; Choi, J.Y. A rate perceptual-distortion optimized video coding HEVC. *IEICE Trans. Inf. Syst.* **2018**, *101*, 3158–3169. [CrossRef]

19. Zhong, H.; Shen, S.; Fan, Y.; Zeng, X. A Low Complexity Macroblock Layer Rate Control Scheme Base on Weighted-Window for H.264 Encoder. In Proceedings of the International Conference on Multimedia Modeling, Klagenfurt, Austria, 4–6 January 2012; pp. 563–573.

20. Dong, J.; Ling, N. A Context-Adaptive Prediction Scheme for Parameter Estimation in H.264/AVC Macroblock Layer Rate Control. *IEEE Trans. Circuits Syst. Video Technol.* **2009**, *19*, 1108–1117. [CrossRef]

21. Li, B.; Li, H.; Li, L.; Zhang, J. λ Domain Rate Control Algorithm for High Efficiency Video Coding. *IEEE Trans. Image Process.* **2014**, *23*, 3841–3854. [CrossRef]

22. Atta, R.; Ghanbari, M. Low-Complexity Joint Temporal-Quality Scalability Rate Control for H.264/SVC. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *28*, 2331–2344. [CrossRef]

23. Geng, M.; Zhang, X.; Tian, Y.; Liang, L.; Huang, T. A fast and performance-maintained transcoding method based on background modeling for surveillance video. In Proceedings of the IEEE International Conference on Multimedia and Expo, Melbourne, Australia, 9–13 July 2012; pp. 61–66.

24. Meuel, H.; Reso, M.; Jachalsky, J.; Ostermann, J. Superpixel-based segmentation of moving objects for low bitrate ROI coding systems. In Proceedings of the 2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance, Krakow, Poland, 27–30 August 2013; pp. 395–400.

25. Kim, N.V.; Chervonenkis, M.A. Situation control of unmanned aerial vehicles for road traffic monitoring. *Mod. Appl. Sci.* **2015**, *9*, 1–13. [CrossRef]

26. Meddeb, M.; Cagnazzo, M.; Pesquet-Popescu, B. Region-of-interest-based rate control scheme for high-efficiency video coding. *APSIPA Trans. Signal Inf. Process.* **2014**, *3*, e16. [CrossRef]

27. Chen, X.; Wu, Z.; Zhang, X.; Xiang, Y.; Xie, S. One Novel Rate Control Scheme for Region of Interest Coding. In Proceedings of the International Conference on Intelligent Computing Methodologies, Lanzhou, China, 2–5 August 2016; pp. 139–148.

28. Wu, C.Y.; Su, P.C. A Region of Interest Rate-Control Scheme for Encoding Traffic Surveillance Videos. In Proceedings of the 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kyoto, Japan, 12–14 September 2009; pp. 194–197.

29. Muthukrishnan, R.; Radha, M. M-Estimators in Regression Models. *J. Math. Res.* **2010**, *2*, 23–27.

30. Huber, P.J. Robust Estimation of Location Parameter. *Ann. Math. Stat.* **1964**, *35*, 73–101. [CrossRef]

31. Tukey, J.W. *Exploratory Data Analysis*; Addison-Wesley Publishers: Boston, MA, USA, 1977.

32. Godbehere, A.B.; Matsukawa, A.; Goldberg, K. Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation. In Proceedings of the 2012 American Control Conference (ACC), Montreal, QC, Canada, 27–29 June 2012; pp. 4305–4312.

33. Ke, C.H.; Shieh, C.K.; Hwang, W.S.; Ziviani, A. An Evaluation Framework for More Realistic Simulations of MPEG Video Transmission. *J. Inf. Sci. Eng.* **2008**, *24*, 425–440.

34. Klaue, J.; Rathke, B.; Wolisz, A. EvalVid—A Framework for Video Transmission and Quality Evaluation. In Proceedings of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Urbana, IL, USA, 2–5 September 2003; pp. 255–272.

35. Kahaki, S.M.M.; Nordin, M.J.; Ashtari, A.H.; Zahra, S.J. Invariant feature matching for image registration application based on new dissimilarity of spatial features. *PLoS ONE* **2016**, *11*, e0149710.

36. Kahaki, S.M.; Arshad, H.; Nordin, M.J.; Ismail, W. Geometric feature descriptor and dissimilarity-based registration of remotely sensed imagery. *PLoS ONE* **2018**, *13*, e0200676. [CrossRef]

37. Bondzulic, B.P.; Pavlovic, B.Z.; Petrovic, V.S.; Andric, M.S. Performance of peak signal-to-noise ratio quality assessment in video streaming with packet losses. *Electron. Lett.* **2016**, *52*, 454–456. [CrossRef]

38. Kwon, S.K.; Tamhankar, A.; Rao, K.R. Overview of H.264/MPEG-4 part 10. *J. Vis. Commun. Image Represent.* **2006**, *17*, 186–216. [CrossRef]