Improving Key frame Encoding and Rate Distortion Performance of BCAWZ for Surveillance video compression

[1]M.Sailaja, PG Student, Dept of ECE, VBIT, JNTUH, TELANGANA
[2]T. Sravanthi, Assistant Professor, Dept of ECE, VBIT, JNTUH, TELANGANA
[3]Brahma Reddy, Professor & HOD, Dept of ECE, VBIT, JNTUH, TELANGANA

Abstract—**Video surveillance has been widely used in recent years to enhance public safety and privacy protection. A video surveillance system needs efficient transmission and storage of video data. Video compression techniques can be used to achieve this. State-of-the-art video compression methods such as H.264/AVC often lead to high computational complexity at the encoder, which is generally implemented in a video camera in a surveillance system. This can significantly increase the cost of a surveillance system, especially when a mass deployment of end cameras is needed.**

In this paper, we discuss the specific considerations for surveillance video compression. We present a SVC system with low-complexity encoder based on Wyner–Ziv coding. In addition, we propose a backward-channel aware Wyner–Ziv (BCAWZ) video coding approach using both full search and cross search motion estimation to improve the coding efficiency while maintaining low complexity at the encoder. The experimental results show that for surveillance video contents, BCAWZ using cross search gives better results than full search and can achieve significantly higher coding efficiency than H.264/AVC INTRA coding as well as existing Wyner–Ziv video coding methods and is close to H.264/AVC INTER coding, while maintaining similar coding complexity with INTRA coding.

Index Terms—**Backward Channel Aware Wyner-Ziv video coding (BCAWZ) ,low complexity video compression, video surveillance, Wyner–Ziv video coding (WZVC), Surveillance video compression (SVC).**

I. INTRODUCTION

VIDEO SURVEILLANCE is an important tool to enhance public safety and privacy protection. It has long been deployed in places of homeland security interests such as airports, train and subway stations, city centers, and major sports events. Surveillance cameras in London provided key photos of the men who bombed the underground system in July 2005. The latest terrorist attack in London was foiled in 2007, partly thanks to the millions of surveillance cameras that London authorities have installed across the city. According to a recent Reuters report, France will triple the number of surveillance cameras by 2009 as part of the fight against terrorism and street crimes. Video surveillance is also used in commercial locations such as banks, automated teller machines (ATMs), supermarkets, and parking areas to prevent and track criminal activities. Consumer adoptions of video surveillance also have soared in recent years due to the increasing concern on privacy protection.

Fig. 1 shows a basic diagram of video surveillance systems [1], [2]. A video surveillance system consists of a client side and a server side. At the client side, the video is first captured by a surveillance camera. Such cameras can be either analog or digital. Digital capturing has become more and more popular, mainly because the captured video by digital surveillance cameras is easier to track and analyze with object detection and content analysis tools. The captured video is sent to the server for further processing. At the server side, video data is used for object detection, activity tracking, and event analysis. In a small video surveillance system, this can be done by sending the video to a closed-circuit television server room, where a security staff can monitor several input feeds. Such procedures are subject to human errors. In addition, this is very cost inefficient for a larger scale video surveillance system where many feeds are monitored simultaneously, and it is impractical for consumer applications as well. Video surveillance systems are also often used as an after-attack forensic tool to identify suspects. These applications require the storage of video data over a period of time for automatic analysis and future use. The storage of raw video data captured directly from the cameras can be very expensive. For example, a VGA (640×480 pixels with 4:2:0 format) camera at 24 frames/s consumes more than 80 Mbits in storage per second, which leads to nearly 1 tera bytes a day for one single camera input. This is cost prohibitive even with the steadily decreasing storage price. Video compression can be used to achieve efficient representation and transmission of surveillance video by reducing the size of the video with no or small quality loss. The availability of efficient and low-cost compression, transmission, and storage of surveillance video data will make it possible to deploy a larger scale of end surveillance cameras, which will in turn improve the accuracy and efficacy of surveillance systems.

In this paper we present a new method for surveillance video compression. We show that while it is straightforward to use existing coding standards, the different requirements on the design of surveillance video compression compared with other video compression applications call for a more careful treatment. The rest of this paper is organized as follows.

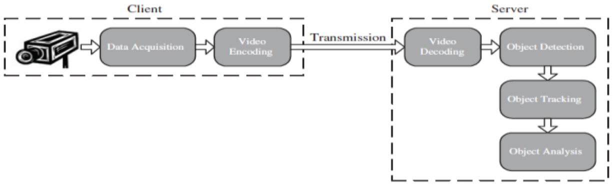


Fig.1.Architecture of surveillance video system

We discuss the design principles and approaches for surveillance video compression. In Section III we present a low-complexity encoding method for surveillance video compression based on Wyner–Ziv coding principle. In addition, we propose a backward-channel aware Wyner–Ziv (BCAWZ) video coding approach to improve the coding efficiency while maintaining low complexity at the encoder. Section IV addresses the backward-channel using cross search motion estimation algorithm. The experimental results are shown in Section V. Section VI concludes the paper.

II. SURVEILLANCE VIDEO COMPRESSION: PRINCIPLES AND APPROACHES

A. Video Compression Design Considerations for Surveillance Video

It is straightforward to use existing image and video coding standards for surveillance video compression. Video compression used in video surveillance system spans from Motion JPEG, MPEG-1, MPEG-2, MPEG-4, H.261, H.263, and currently state-of-the-art H.264/AVC. Motion JPEG encodes each frame independently, which removes only the spatial redundancy. Video compression standards further take advantage of the temporal correlation with motion estimation. The frames coded in this way are referred to as INTER frames, whereas in INTRA frames only spatial redundancies are exploited. While the high coding efficiency of INTER frame coding can significantly reduce the size of the captured video with no or small loss of the video quality, it is also desirable to insert an INTRA frame periodically in a group of pictures (GOP) to enable easy access to the contents, which is important for surveillance video search and analysis. INTRA frames can also provide protection against transmission errors such that the errors are not propagated to the following GOPs. The combination of INTER coding and INTRA coding in video compression standards satisfies the needs of high coding efficiency and easy content accessibility for surveillance video compression. In a typical H.264/AVC encoder, the input video frame is divided into slices and further divided into macro blocks, which is the basic coding unit. A macro block can be further divided into blocks of smaller sizes. A block is predicted from the spatial or temporal neighbouring blocks. For INTRA coding, the block is predicted from its spatial neighbouring blocks. For INTER coding, the block is predicted from previously reconstructed frames and motion vectors that denote the location of the reference block are obtained. The motion vector is searched at the encoder among all candidate motion vectors within a search range of $M \times N$. The prediction error between the current block and its reference block is transformed, quantized, and entropy-encoded and sent to the decoder, along with the motion vector differences, which are also entropy-encoded. At the decoder, the bit stream is entropy-decoded and the motion vectors are extracted to construct the reference block, which is then used to reconstruct the current block with the decoded prediction error.

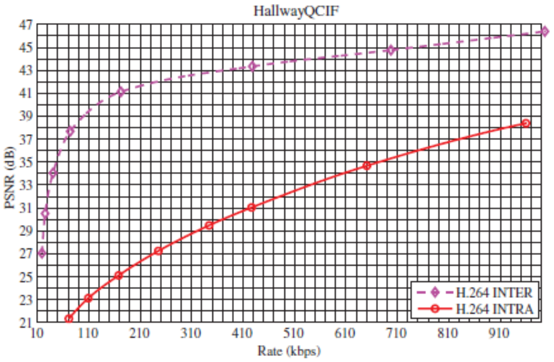


Fig2: Rate distortion performance comparison.

In Fig. 2 we show the rate distortion performance comparison for H.264 INTRA and INTER coding in *Vipmen* sequence, which is a sequence with content similar to indoor surveillance video. As shown in Fig. 2, INTER coding shows considerably a higher coding efficiency than INTRA coding. It is hence of interest to use

INTER coding more frequently to achieve higher reduction of video size, where as INTRA coding should only be used as an anchor picture periodically to stop error propagation and provide fast access to the contents. In a typical GOP for videos captured at 24 frames per second, generally only one or two frames are INTRA coded for every 24 frames, while the rest are INTER coded. The superior coding efficiency of INTER coding, however, comes at the cost of much higher computational complexity at the encoder. The coding complexity of INTER encoding is typically 5–10 times higher than INTRA encoding.

This is primarily due to the fact that motion estimation is done at the encoder where a number of candidate motion vectors are compared with rate-distortion-optimized motion search. A "heavy" encoder with a "light" decoder may not be a problem, and at times even desirable, for many consumer video applications such as video playback and video streaming. In these cases, which is part of the main target applications for existing video coding standards, the content is encoded once and decoded many more times by the end users. For example, a movie studio only needs to encode a movie once to put it on a DVD, which will then be decoded and played tens of thousands or millions of times at different decoders by the consumers. A simple decoder makes further sense in these cases because the encoder, which is used by content owners such as studios and cable carriers in their professional facilities, commands much higher computational capability compared to the decoder, which can be a consumer player on a PC, cell phone, DVD player, etc. Such mainstream application scenarios have deeply affected the design philosophy of video compression standards, where a greater emphasis is put on the low complexity of decoders.

The video surveillance systems, however, are very different from the above applications. In a video surveillance system, the encoder is implemented in a simple and low-cost video surveillance camera and a number of such surveillance cameras are installed in a surveillance system. On the other hand, as we discussed in Section I, the video is decoded and analyzed at the central server in the data processing centres, which have much more capable computational resources. This asymmetry in video surveillance systems compared to other consumer for surveillance video compression. A good surveillance video compression method needs to not only satisfy the high coding efficiency, easy content accessibility, and error resilience requirements discussed in Section I, but it also needs to achieve these goals with a low-complexity encoder such that the surveillance cameras can be massively deployed. In this paper, we aim to build a surveillance video compression system that meets these requirements and achieves coding efficiency comparable to INTER coding with encoding complexity close to INTRA coding.

B. Low-Complexity Video Encoding Approaches for Surveillance Video Compression

This goal is achievable theoretically as a result of two theorems discovered in 1970s, referred to as Slepian–Wolf Theorem and Wyner–Ziv Theorem. Consider two correlated information sources *X* and *Y* encoded by two separate encoders *A* and *B*, as shown in Fig. 3. When the switch is off, neither has the access to the other source. If joint decoding is allowed, the Slepian–Wolf Theorem shows that the admissible rate region for the system is

$$\begin{aligned} RX &\geq H(X|Y) \\ RY &\geq H(Y|X) \qquad (1) \\ RX + RY &\geq H(X, Y) \end{aligned}$$

Where *H(X|Y)* and *H(Y|X)* denote the conditional entropy and *H(X, Y)* denotes the joint entropy of *X* and *Y*. Fig. 4 shows the admissible rate region for the Slepian–Wolf theorem.

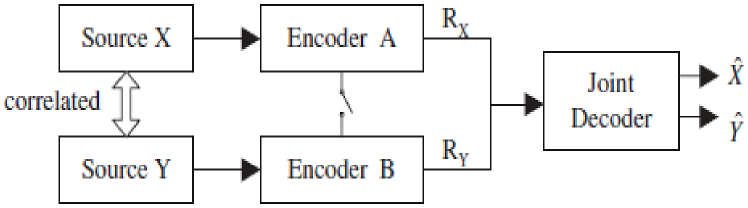


Fig.3. Source coding theorem

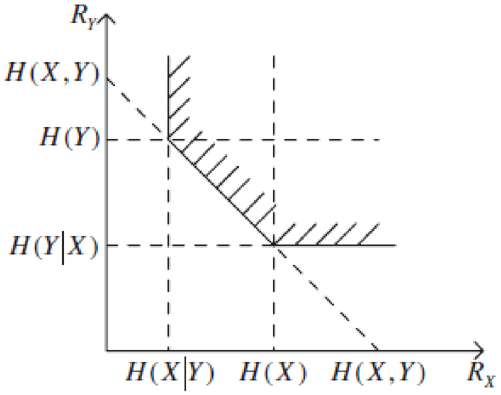


Fig.4.Admisible rate region for Slepian-Wolf theorem

This is the same as the case when the switch is on and both have access to the other source at the encoder. Slepian–Wolf theorem shows that, regardless of its access to side information *Y*, encoder *A* can encode *X* with arbitrarily high fidelity as long as the decoder *A* has access to *Y* and the rate of *X* is equal to or greater than conditional entropy *H(X|Y)* and the overall rate is equal to greater than *H(X, Y)*. This result is extended to lossy compression in [20]. Denote *R*(d)* as the rate distortion function of *X* when the side information *Y* is only available at the decoder, and

$RX|Y\ (d)$ as the rate distortion function when side information Y is available at both the encoder and decoder. In [20] it is shown that $R^*(d) \geq RX|Y\ (d)$, with equality achievable for certain cases, e.g., for a Gaussian source and a mean square error distortion metric. This result was recently further extended to more general distributions in by invoking the duality between source coding and channel coding. These results show that the side information at the encoder is not always necessary to achieve the rate distortion bound in lossy compression. This leads to a new coding scheme, referred to as Wyner–Ziv video coding (WZVC) where the temporal correlation among frames is exploited at the decoder instead of the encoder. Each frame is independently encoded at the encoder and the computationally intensive job of motion estimation is shifted to the decoder. Due to the "light" encoder characteristics, WZVC can be an ideal choice for surveillance video compression, where the decoding is done at the data processing centres with more powerful computational capabilities. Such coding scheme is particularly useful for Surveillance application with a strict constraint on the encoder Such as the battery-powered wireless end camera.

III. VIDEO SURVEILLANCE USING BACKWARD-CHANNEL AWARE WYNER–ZIV CODING

A. Video Compression Design Considerations for Surveillance Video

Since the theoretic results of Slepian–Wolf Theorem and Wyner–Ziv Theorem were revisited in the late 1990s, several video codecs have been developed based on Wyner–coding principles. In this section we build such a compression system for surveillance video based on our Wyner–Ziv video codec as shown in Fig. 5. The captured video sequence is divided into two groups which are encoded by two different methods. All frames are encoded independently. These frames are denoted as key frames and will serve to generate side information at the decoder for other frames. The remaining frames are encoded using channel coding methods and the parity or syndrome bits are sent to the decoder. These frames are referred to as Wyner–Ziv frames. Two channel coding methods are supported in the system: Turbo code [29] and low-density-parity-check (LDPC) code. The Wyner–Ziv frames can be encoded either in the pixel domain or the transform domain with the integer transform. Significant bit plane is first coded by the channel encoder, followed by the other bit planes in the order of descending significance. The entire bit plane in a frame is encoded with the channel coder. The output of the channel coder, which are parity bits in the case of Turbo encoder and syndrome bits in the case of LDPC encoder, is sent to the decoder.

At the decoder, the key frames are independently decoded by the H.264 INTRA decoder. For the Wyner–Ziv frames, its initial estimate is first derived from the previously reconstructed key frames, which serves as the side information.

There are multiple ways to obtain the initial estimate. Suppose the nth frame is the current frame and $(n-1)$th and $(n+1)$th frames are the previously decoded key frames. The simplest way to extract the initial estimate is to use the co-located pixel value in the previous reconstructed $(n-1)$th frame as the side information for the pixel in the current frame. Another method is to take the average of the co-located pixel values at the $(n-1)$th and $(n+1)$th reconstructed frames. In these two cases, the quality of the initial estimate is low and the temporal correlation of the motion field is not fully exploited. To obtain a higher quality initial estimate, motion estimation can be done at the decoder. Initial estimate can be obtained by extrapolating the previous reconstructed frame as shown in Fig. 6. For every block in current nth frame, we search for the motion vector $MVn-1$ of the co-located block in previous frame $(n-1)$. Assuming the motion field is continuous, we can use $MVn-1$ as the predictor for the motion vector of the current block in the nth frame. Applying the predicted motion vector MVn to the reconstructed frame $n-1$, we can find its reference block, which can be used as the initial estimate, as shown in Fig. 6.

We can also use motion-compensated interpolation to obtain the side information. As shown in Fig. 7, motion search is done between the $(n-1)$th frame $s^\wedge(n-1)$ and $(n+1)$th frame $s^\wedge(n+1)$. For each block in the current frame, as shown in Fig.7, the initial estimator first uses the co-located block in the next reconstructed frame $s^\wedge(n+1)$ as the source and the previous reconstructed frame $s^\wedge(n-1)$ as the reference to perform forward motion estimation.

We denote the obtained motion vector as **MVF**. We then use the co-located block in the previous frame as the source and the next reconstructed frame as the reference to perform backward motion estimation. Denote the obtained motion vector as **MVB**. The side estimator uses **MVF/2** from $s^\wedge\ (n-1)$ to find the corresponding reference block **PF1**, and **−MVF/2** from $s^\wedge(n+1)$ to find the corresponding reference block ***PF*2**. We also use **MVB /2** from $s(^\wedge n+1)$ to find the corresponding reference block ***PB*1**, and − **MVB /2** from $s^\wedge(n-1)$ to find the corresponding reference block ***PB*2**. The final reference is

$$P = (PF1 + PF2 + PB1 + PB2)/4 \tag{2}$$

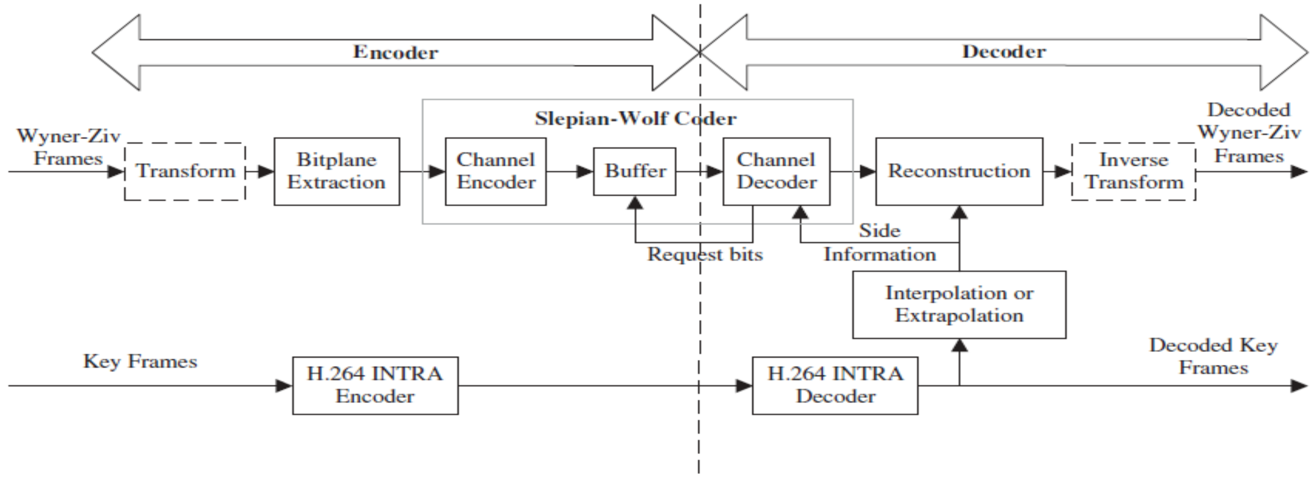


Fig.5. WZVC structure.

This average of the four references can be used as the initial estimate of the current block. Experimental results show that the initial estimate derived from motion compensated interpolation generally performs better than that from motion-compensated extrapolation, which is further confirmed in our theoretic analysis in.

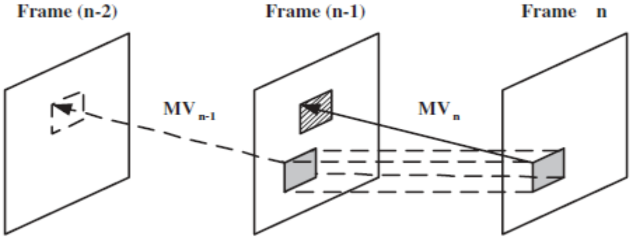


Fig.6. Derivation of side information by extrapolation.

After the initial estimate is obtained, if the system is encoded in transform domain, the initial estimate is also integer-transformed. The initial estimate is then represented by bit plane, as done in the encoder. The channel decoder uses the initial estimate and the incoming parity or syndrome bits at each bit plane received from the encoder to decode the frame. Similar to other WZVC schemes, here we assume that the decoder can communicate with the encoder to request more bits until the bit plane is correctly decoded.

We note that the Wyner–Ziv video codec formulates the video decoding problem as an error correction problem. This gives the surveillance video compression based on WZVC additional error resilience capability compared to the codec based on existing video coding standards. If the parity or syndrome bits are lost or corrupted during the transmission, it only adds to the error correction complexity due to the increasof errors in the decoding. This is different from the standard- based video codec when the bits are lost or corrupted, a block may not be correctly decoded, and the error can propagate to the following frames until an INTRA frame is met.

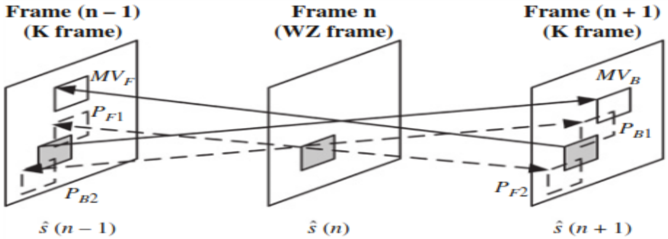


Fig.7. Derivation of side information by interpolation

B. Backward-Channel Aware Wyner–Ziv Video Coding for Video Surveillance

One problem for this WZVC with INTRA key frames, as discussed in Section III-A and also used in many other existing Wyner–Ziv video codecs, is that while its coding efficiency is higher than INTRA coding, it is still significantly lower compared to state-of-the-art INTER video coding, as we shall see in Section V. In such WZVC schemes, INTRA frames have to be used frequently to ensure that the initial estimate derived from the key frames is accurate enough for the decoding of Wyner–Ziv frames. Increasing the distance between two neighboring key frames results in a lower coding efficiency compared to closer key frames at the same bit rate. This is because the motion-compensated interpolation is less accurate when the distance between two key frames increases, and hence degrading the quality of the initial estimate for the Wyner–Ziv frames. It is for this reason that many current Wyner–Ziv video codec's encode every other frame as INTRA frames. However, the frequent use of INTRA frames reduces the coding efficiency, and hence the quality of INTRA frames at the same bit rate, which in turn results in lower quality side information for Wyner–Ziv frames. To address this problem, we extend the above coding method by encoding the key frames with the use of backward channel aware motion estimation (BCAME) .We shall refer to our WZVC method based on BCAME, as BCAWZ. The basic idea of BCAME is to perform motion estimation at the decoder and send the motion information back to the encoder through a backward-channel. With the received motion vectors from the decoder, we are able to improve the coding efficiency of the key frames and the side estimation quality without significantly increasing the encoder complexity. Since the motion vectors are needed at the server for decoding as well as object tracking and event analysis, it causes only minimal increase of decoding complexity. Furthermore, surveillance systems typically allow the transmission of meta -data to the surveillance cameras through feedback channel to control the camera behaviors. The motion information needed at the encoder can be transmitted as part of these meta-data. While this may increase the latency and cause problems for certain type of surveillance applications such as high-speed scanning and aero surveillance, such latency can be addressed in many other surveillance systems. In the following we describe the coding procedure of backward-channel aware WZVC, as shown in Fig. 8.

Assuming that in natural video sequences the motion field is continuous, it is possible to predict the motion of the current frame using the information of its adjacent frames. For a sequence we encode the first and the third frame as INTRA frames. All the other odd frames are encoded with BCAME, and we refer to these backward predictively coded frames as BP frames. All the even frames are encoded as a Wyner– Ziv frame. A BP frame is coded as follows. Assume that the two BP frames prior to the current BP frame, as shown in Fig. 9, have been decoded at the decoder. For each block in the current BP frame, we use its co-located block in one of the previous two BP frames as the source and the other BP frame as the reference. Block-based motion search is done at the decoder to estimate the motion vector.

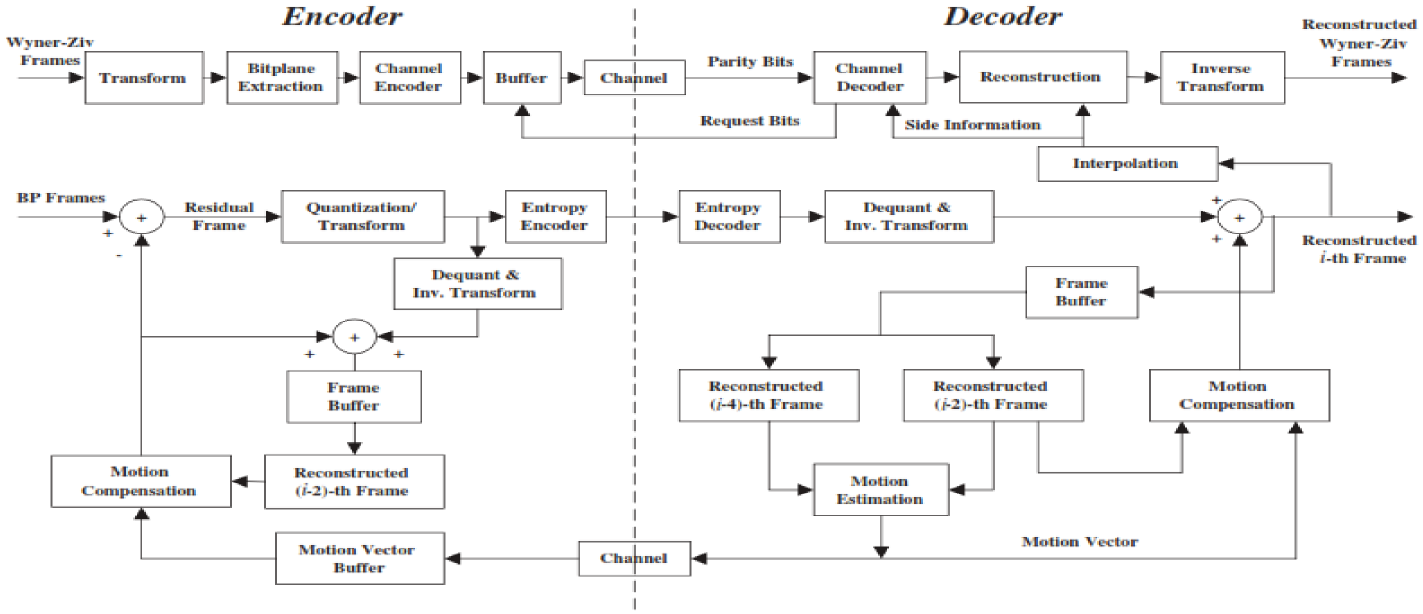
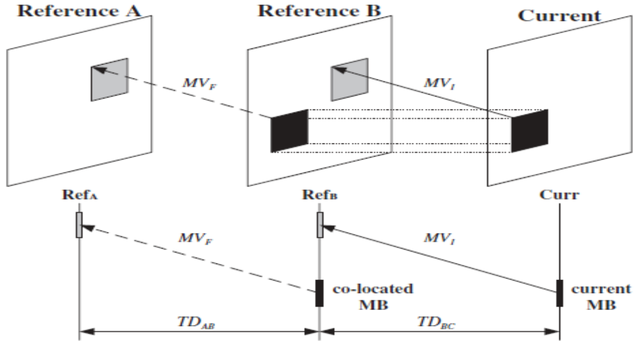


Fig.8 Backward Channel Aware WZVC

The motion vectors are sent back to a motion-vector buffer at the encoder through the backward-channel. This buffer is updated when the next frame's motion vectors are received. At the encoder, we use the received motion vectors with the previous reconstructed BP frames to generate the motion-compensated reference for the current BP frame. The residual between the current BP frame and its motion-compensated reference is then transformed and entropy-coded.

Depending on which of the previous decoded BP frames is used as the source and reference at the decoder, we can obtain two sets of motion vectors. The first motion vector is shown in Fig. 9

Frame A and B are the previous two reconstructed BP frames stored in the frame buffer at the decoder. The temporal distances between adjacent frames are denoted as *TDAB* and *TDBC*.



.

Fig.9 Mode I: Forward motion vector for BCAWZ

To find the motion vector in the current macro block, we use the motion information of the co-located macro block in the previous frame by assuming that a constant translational motion velocity remains across the frames. For each block in the current frame, we use the co-located block in B and search for its best match in A and obtain the forward motion vector *MVF*. Assuming linear motion field, the motion vector for the block in the current BP frame is then

MVI= **[***TDBC/ TDAB*] MVF

Since we encode the BP frame and Wyner–Ziv frame alternately, *TDAB* = *TDBC* = 2, *MVI* = *MVF.*

The second motion vector is obtained in a similar way as Mode I but we use the co-located frame in *A* as the source, as shown in Fig. 10. We search for the best matched block in B. This motion vector is referred to as the backward motion vector *MVB*. Again assuming linear motion, the motion vector for the current frame is

MVII=- [*TDAC/ TDAB*] MVB

Here *TDAB* = 2 and *TDAC* = 4, *MVII* = −2*MVB*.

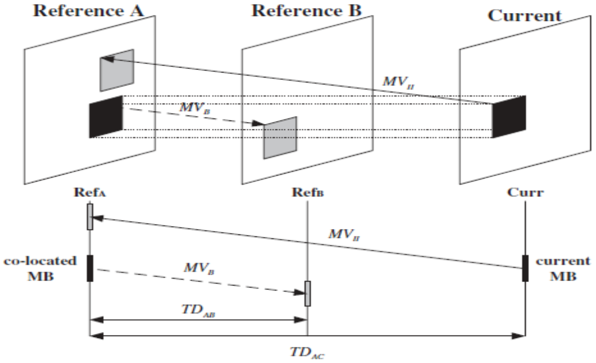


Fig.10.Mode II Backward motion vector for BCAWZ.

These two sets of motion vectors are sent back to the encoder, where the original current BP frame is available. The encoder can then do a mode selection to choose the best matched motion vector based on metrics such as mean squared error (MSE) or sum of absolute difference (SAD)

$$\text{Optimal Mode} = \arg \min_{k\in\{I,II\}} \frac{\sum_{(i,j)} Dx(i,j) - \tilde{x}^{(k)}(i,j)}{N \times N} \tag{3}$$

where *k* denotes the index of the modes, *x(i, j)* denotes the original pixel value in the position *(i, j)*, x^k(i,j)denotes the reference pixel value using mode *k*, *N* represents the size of the macro block, and the summation is over all the pixels of the current macro block. The mode decision result is sent to the decoder along with the transform coefficients of the residual frame using the selected motion vector. Compared to the WZVC method in Section III-A, BCAWZ provides an efficient approach to predictively encode the key frames without greatly increasing the encoder complexity. We note that since the motion vectors are also needed for object tracking and event analysis at the server, as well as for generating the interpolated initial estimate at the decoder for the Wyner–Ziv frame between *A* and *B*, the increase of the server's complexity is marginal.

IV. BACKWARD-CHANNEL AWARE WYNER–ZIV CODING USING DIAMOND AND CROSS SEARCH MOTION ESTIMATION

A. Cross Search Motion estimation

Cross search motion estimation algorithm achieves less number of computations required to processing motion estimation process, more visual quality and less time required to execute the program when compared to the full search motion estimation algorithm.

ARPS [4] algorithm makes use of the fact that the general motion in a frame is usually coherent, i.e. if the macro blocks around the current macro block moved in a particular direction then there is a high probability that the current macro block will also have a similar motion vector. This algorithm uses the motion vector of the macro block to its immediate left to predict its own motion vector. An example is shown in Fig. 11. The predicted motion vector points to (3, -2). In addition to checking the location pointed by the predicted motion vector, it also checks at a rood pattern distributed points, as shown in Fig 11, where they are at a step size of S = Max (|X|, |Y|). X and Y are the x-coordinate and y-coordinate of the predicted motion vector. This rood pattern search is always the first step. It directly puts the search in an area where there is a high probability of finding a good matching block. The point that has the least weight becomes the origin for subsequent search steps. From this step it moves to next least weight by searching points in '+' shape that means cross shape.

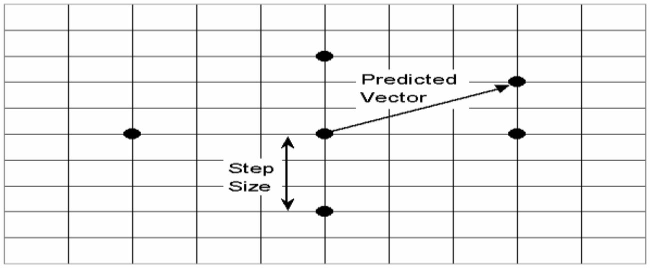


Fig.11 Cross search pattern

Cross search algorithm is fastest algorithm because it requires less number of computations and more image visual quality than compared to the full search algorithm. The objective quality is measured by the peak signal-to-noise ratio (PSNR), which is commonly used in the objective quality comparison. Since the SAD (sum of absolute difference) operation dominates the complexity of the ME algorithm, we can measure the algorithm complexity by the number of SAD operations, which is identical to the number of search points.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section we show the experimental results. We implemented our scheme based on the H.264/AVC reference with main profile. In this paper for WZVC with INTRA, INTER, or BP key frames, every odd frame is encoded as key frames. And every even frame is encoded as Wyner–Ziv frames, which is first encoded with the integer transform in H.264/AVC. Each transform coefficient is represented by bit planes. The bits at the same bit plane are encoded by the LDPC code with a code block length equal to the number of pixels in the frame. At the decoder, an initial estimator for each frame is constructed with motion compensated interpolation using a block size of 16×16. This initial estimate is also integer-transformed and represented by bit planes. Motion vectors derived at the decoder are predictively and entropy encoded with the methods used in H.264/AVC reference software. We evaluate the performance of BCAWZ compared with WZVC with INTRA key frames, WZVC with INTER key frames, as well as H.264/AVC INTER coding and INTRA coding. For WZVC with INTER key frames, all the key frames are encoded as P frame except the first key frame, which is INTRA coded. For H.264/AVC INTER coding, only the first frame is INTRA coded, and all the other frames are coded with IBPBP GOP structure.

Fig.12 Decoded frame 5 by BCAWZ using full search

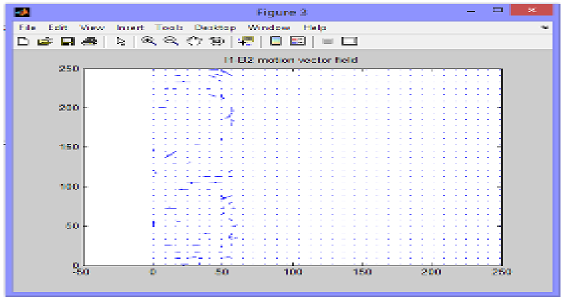


Fig.13 Motion vectors for frame 5 by full search

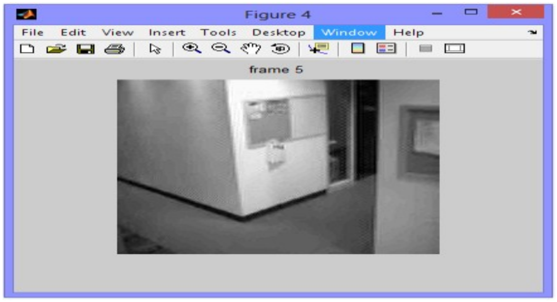


Fig.14 Decoded frame 5 by BCAWZ using cross search

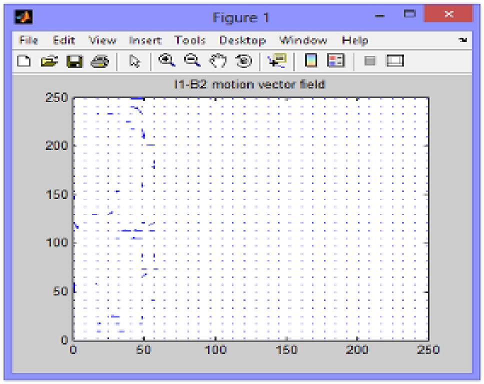


Fig.15. Motion Vectors for frame 5 by cross search

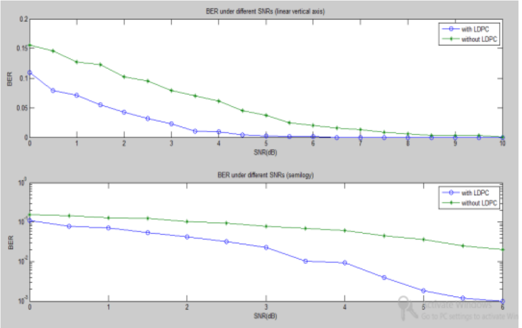


Fig.16.BER for Wyner-Ziv with LDPC without LDPC

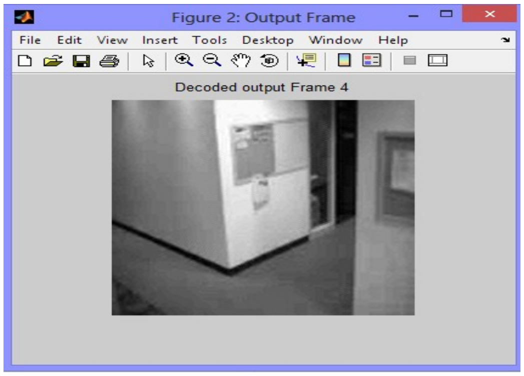


Fig.17.Decoded even frame by LDPC.

VI. CONCLUSIONS

In this paper we present a low-complexity video encoding frame work for surveillance video compression. The "light" encoder characteristics of WZVC make it suitable for surveillance video compression, but the coding efficiency of WZVC with INTRA coding is less when compared to WZVC with INTER.

To address the coding performance problem, we further propose a backward-channel aware Wyner–Ziv (BCAWZ) video coding approach to improve the coding efficiency while maintaining low complexity at the encoder. We propose to send the motion vectors derived at the decoder back to the encoder. Since the motion information is needed at the decoder and server for decoding and, object detection, and event analysis, this scheme only incurs marginal complexity increase for the decoder and encoder and a small usage of backward-channel, which is generally available in video surveillance system.

The experimental results show that for surveillance video contents, BCAWZ can achieve significantly higher coding efficiency than H.264/AVC INTRA coding and WZVC with INTRA key frames and is within as low as 1 dB compared to H.264/AVC INTER coding, while maintaining similar coding complexity with INTRA coding. Further we proposed BCAWZ with cross search motion estimation. With this cross search number of computations for motion vectors is reduced to 6.36 when compared to full search for which number of computations is 51076.The quality of frames by BCAWZ using cross search is increased to 81.4 and total program elapsed time is reduced to 43 seconds from 62 when compared to BCAWZ by full search.

BCAWZ using full search

1. Computations on processing on f5 Frame 51076
2. Computations on processing on f7 Frame 51076
3. PSNR5= 81.3607
4. Elapsed time is 62.047692 seconds.

BCAWZ using cross search

1. Computations on processing on f5 Frame 6.3633
2. Computations on processing on f7 Frame 6.2734
3. PSNR5=81.4
4. Elapsed time is 43.85 seconds.

REFERENCES

[1] M. Valera and S. Velastin, "Intelligent distributed surveillance systems: A review," in *IEEE Proc. Vis.Image, Signal Process.*, vol. 152, no. 2,pp. 192–204, Apr. 2005.

[2] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, A. Senior, C. Shu, and Y. Tian, "Smart video surveillance: Exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Process. Mag.*, vol. 22, no. 2, pp. 38–51, Mar. 2005.

[3] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 34, no. 3, pp. 334–352, Aug. 2004.

[4] Yao Nie, and Kai-Kuang Ma, "Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation", *IEEE Trans. Image Processing*, vol 11, no. 12, pp. 1442-1448, December 2002.

AUTHOR'S PROFILE

M.Sailaja pursuing his PG in VBIT Hyderabad with specialization of Communication Systems. His areas of interest are to do hardware engineer, also in the image processing and video surveillance

Ms.**T.SRAVANTHI** ME from Chaitanya Bharathi Insistute of Technology (Affiliated to Osmania University) & B.Tech from Bhoj Reddy Engineering College for Women (Affiliated to J NT U, Hyderabad).Her areas of interests are image registration, tracking, data fusion and video compression.