Distributed Video Coding in Wireless Multimedia Sensor Network for Multimedia Broadcasting

Zhuo Xue[1], K.K. Loo[1], J. Cosmas[1], P.Y. Yip[2]
[1] School of Engineering and Design, Brunel University
[2] School of Computer Science, University of Hertfordshire
UNITED KINGDOM
{Zhuo.Xue, Jonathan.Loo, John Cosmas}@Brunel.ac.uk, p.y.a.yip@herts.ac.uk

Abstract: - Recently the development of Distributed Video Coding (DVC) has provided the promising theory support to realize the infrastructure of Wireless Multimedia Sensor Network (WMSN), which composed of autonomous hardware for capturing and transmission of quality audio-visual content. The implementation of DVC in WMSN can better solve the problem of energy constraint of the sensor nodes due to the benefit of lower computational encoder in DVC. In this paper, a practical DVC scheme, pixel-domain Wyner-Ziv(PDWZ) video coding, with slice structure and adaptive rate selection(ARS) is proposed to solve the certain problems when applying DVC into WMSN. Firstly, the proposed slice structure in PDWZ has extended the feasibility of PDWZ to work with any interleaver size used in Slepian-wolf turbo codec for heterogeneous applications. Meanwhile, based on the slice structure, an adaptive code rate selection has been proposed aiming at reduce the system delay occurred in feedback request. The simulation results clearly showed the enhancement in R-D performance and perceptual quality. It also can be observed that system delay caused by frequent feedback is greatly reduced, which gives a promising support for WMSN with low latency and facilitates the QoS management.

Key-Words: - Wireless Multimedia Sensor Network, Distributed Video Coding, Wyner-Ziv video coding, Laplacial Distribution, RCPT.

1. Introduction

The recent advances in micro-electro-mechanical systems (MEMS) technology, wireless communications, and digital electronics of sensors have promoted the development of wireless sensor networks (WSN) [1,2,3]. By using low-cost, low-power and multifunctional sensor nodes, the WSN has been deployed in many civil or military applications. Most current deployed WSNs measure scalar physical phenomena like temperature, humidity, pressure, or location of objects, which usually have low bandwidth requirement and usually delay tolerant. However, applications such as environmental monitoring, health-care monitoring, emergency response and security/surveillance, the multimedia information especially video stream is indispensable. Therefore, wireless "multimedia" sensor networks (WMSN) [4,5], which is capable of delivering multimedia information with high data rate have drawn much attention in research community in recent years.

A basic WMSN infrastructure can be created by using autonomous hardware composed of advanced wireless technology and inexpensive components such as a simple processing unit, CMOS cameras and microphones that enables capturing of multimedia data ubiquitously. Many factors that influence the design of WMSN are discussed in [6]. In WMSN, the multimedia sensor nodes usually have lower power/energy and limited processing capability which make it not feasible to use the conventional video encoders e.g. H.264 at the source sensor node due to their energy constraint. The conventional video standards are based on predictive coding that exploit the source statistics at encoder to realize high compression. However, massive computation due to motion estimation and other processes during encoding discourages the application of standard video compression system in WMSN.

Recently, it was shown that the traditional concept of complex encoder and simple decoder can be reversed within the framework of Distributed Video Coding (DVC) [6,7]. DVC is based on distributed source coding theory [7,8], which states that efficient compression can also be achieved by exploiting source statistics partially or wholly at the decoder only. As a result, the massive computation can be shifted from encoder to decoder. This new video coding paradigm encourages deployment of networks composed of video sensors for capturing and transmission.

A wireless video capturing facility using WMSN is proposed for used in multimedia broadcasting. Video information is gathered by a network of video

sensors and delivered to end-users via various transmission techniques e.g. digital TV broadcasting internet etc.

Fig.1 illustrates the concept of WMSN for multimedia broadcasting. In this scenario, for example, a football fan can watch the goal from different angle via his personal communication equipments like mobile phone, PDA etc. From the figure, a main camera with higher processing ability will capture major field of interesting area and send major on site video information directly back to the sink server on a reliable transmit channel. In addition, several wireless video sensor nodes are deployed to cover different viewing angle and areas. The video content gathered by sensor nodes is relayed to a wireless gateway, through which it is sent back to a sink server. The sink server can compose video information based on the correlated information received from main camera and sensor nodes. Finally, the video stream is broadcasted to heterogeneous end-users via standard digital TV broadcasting. On the other hand, many researches exploring the delivery of information to Ad hoc network, the design of protocol layer and the QoS management etc. have been carried in [10-20], which support for delivering high quality multimedia information to heterogeneous end-users. In [21], the author utilized the IEEE 802.22 to realize interconnection between wireless gateways and the sink server in energy constraint sensor network.

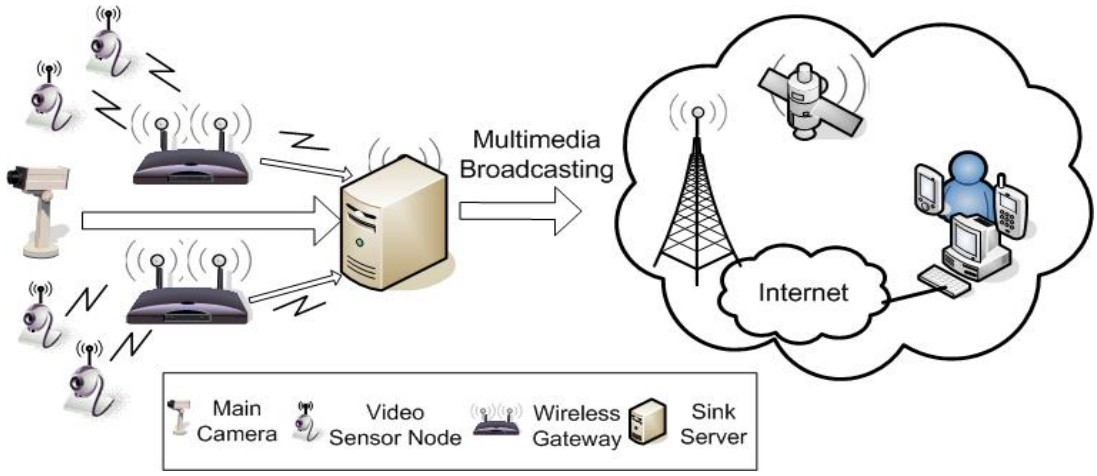


Fig.1. Wireless multimedia sensor network for multimedia broadcasting

In this paper, the focus is on the design of an efficient DVC architecture where an improved PDWZ video coding with slice structure and adaptive rate control is proposed for realizing the construction of WMSN. The rest of this paper is organized as follows. Section II discusses the motivation and background of this work. Section III, the proposed enhanced PDWZ system with slice structure is introduced, and the enhancement is verified by comparing to classic WZ coding. In Section IV, the ARS scheme is described in details and the performance of reducing the time delay and improving the QoS is tested by comparing to that of traditional ERS and QRS scheme. The final part concludes this paper.

2 Motivation and Background

In order to deliver a high quality video to end-users, the quality of video at sink server has to be guaranteed by using effective video coding scheme, efficient rate control and QoS management etc. As advised above, traditional predictive video coding is not applicable for WMSN, thus DVC is introduced here due to its competitive compression performance and lower energy consumption. A popular DVC scheme known as PDWZ [22] can be easily applied in the wireless video capturing network shown in Fig.1 where the main camera will be supplying the video frame as side information via a traditional intra-frame coding like JPEG, and the video sensor nodes will be supplying video frames as Wyner-Ziv (WZ) frame. Multiple video sensor nodes can be applied as an extension of WZ frame.

However, there are still several challenges need to be resolved before PDWZ can be applied in WMSN to deliver high quality video stream. Firstly, in the conventional PDWZ, the interleaver size used in Slepian-Wolf turbo code is ideally assumed to be large enough to encode a whole WZ frame. This is not feasible for practical use as significant computation latency will be added to the system end-to-end. In other words, the system will not be able to provide a timely WZ decoded output due to

the enormous Slepian-Wolf turbo coding and decoding delay. Secondly, in most WZ video coding either in pixel domain or transform domain [22,23,24], the bitrate of WZ frame is determined by statistic dependency between the side information and the original WZ frame. Since the encoder has no knowledge of side information during encoding, it has no idea of how many parity bits would be enough for decoding. Most previous literatures like [22,23,24] solve this rate control problem by using the process called "decode and request" in conjunction with rate compatible punctured turbo codec (RCPT) [25]. After encoding, all generated parity bits are saved into buffer and all systematic bits are discarded. The encoder will first send a few parity bits, which is produced by puncturing according to the highest code rate in RCPT. The decoder performs decoding by using side information and received parity bits. If decoding fails, a request for more parity bits will be sent back to encoder via feedback channel. However, the request is not capable of telling the encoder even approximately how many parity bits really should be sent.

To overcome this problem, there are three candidate methods:

Method 1: In order to guarantee the least parity bits to be sent and obtain the maximum compression performance, the encoder will lower the code rate by a very small step and increase the number of parity bits for transmission. This process will continually repeat until a successful decoding is obtained or all parity bits are sent. The smaller is the step size of increase in parity bits, namely, the smaller step size is in decrease in code rate(from 8/9 to 8/10), the better compression performance can be guaranteed. We refer this kind of 'decode and request' process to 'exhaustive rate selection' (ERS) through the whole paper. Although ERS scheme achieves better compression performance, on the other hand, it would produce significant system latency due to the frequent feedback request for seeking the least parity bits to be sent exhaustively and increase huge decoding complexity. Each request would cause a whole new round of retransmission and decoding process therefore huge decoding complexity is caused. This will exceed the delay tolerance in WMSN, and has significant negative influence on QoS.

Method 2: As suggested in [10], the code rate can be dropped by power of 2. For instance, if the puncturing period is set to l, the code rate will be $R=l/(l+2^n)\{ n=0, 1, 2..l/2\}$. We refer this kind of scheme as quick rate selection (QRS). However, the step of dropping the code rate is too rapid(like from 8/9 to 8/12) therefore much redundancy parity bits will possibly be sent and results in the waste of bandwidth.

Method 3: Using ARS solution, which is main idea proposed in this paper. The decoder will evaluate the quality of side information, which subsequently is feed back to encoder via feedback channel. The encoder then will have an initial estimation of the code rate required and send the corresponding parity bits.

In this paper, two major improvements are proposed to overcome the system latency and rate control problems, which eventually improve the delivery of video to the sink server and provide better QoS management in the terms of output video quality when applying PDWZ in WMSN. Firstly, a novel slice structure is introduced to adapt to an appropriate interleaver size of Slepian-wolf turbo code in which system latency can be greatly reduced. Furthermore, the slice structure has the benefit in providing an optimal rate control solution for PDWZ. The performance of this improvement is evaluated by comparing to that of classic WZ architectures. In addition, driven by the motivation of minimizing the delay caused by the feedback channel, an adaptive rate selection (ARS) is suggested. The proposed ARS scheme still works under the model of 'decode and request', but the request would bring back the evaluation of side information quality. As is known in WZ video coding, the quality (similarity between the side information and the original WZ frame) of side information determines the amount of parity bits to be sent. The encoder can choose the appropriate code rate based on the side information quality and send corresponding number of parity bits. The whole decoding can be expected to be finished in very few rounds of feedback loops. Comparing to ERS, the system delay can be greatly decreased without losing too much compression performance.

3. Enhanced PDWZ architecture

3.1 PDWZ video coding with slice structure

Fig.2 shows the overall architecture of the proposed PDWZ video coding system with slice structure which mainly inherited from the PDWZ proposed in [22]. Note that in the application of WMSN, WZ frame is from one of video sensor nodes while side information is from main camera. To simplify, in this paper, the proposed PDWZ with on slice structure is based on our previous work used in [25]. As in Fig.2 shows, WZ frame sequence is composed by each even frame X_{2i} (i=1,2,3…n) of video sequence, and each two adjacent odd frames X_{2i+1} and X_{2i-1} are used

as key frames; the WZ frame is intra-frame coded and inter-frame decoded at decoder with the help of side information. Side information Y_{2i} is obtained by performing bidirectional motion compensated interpolation [23] between two key frames, which are assumed to be transmitted and reconstructed perfectly at decoder. In practice, this information is obtained from main camera which used traditional intra-frame video coding technology.

A slice structure for WZ frame here is used to adapt the limited interleaver size. We use the interleaver of 3GPP turbo codec with the limited size between 40~5114 bits as a special example to illustrate the principle, by which the application to other different interleavers can be derived.

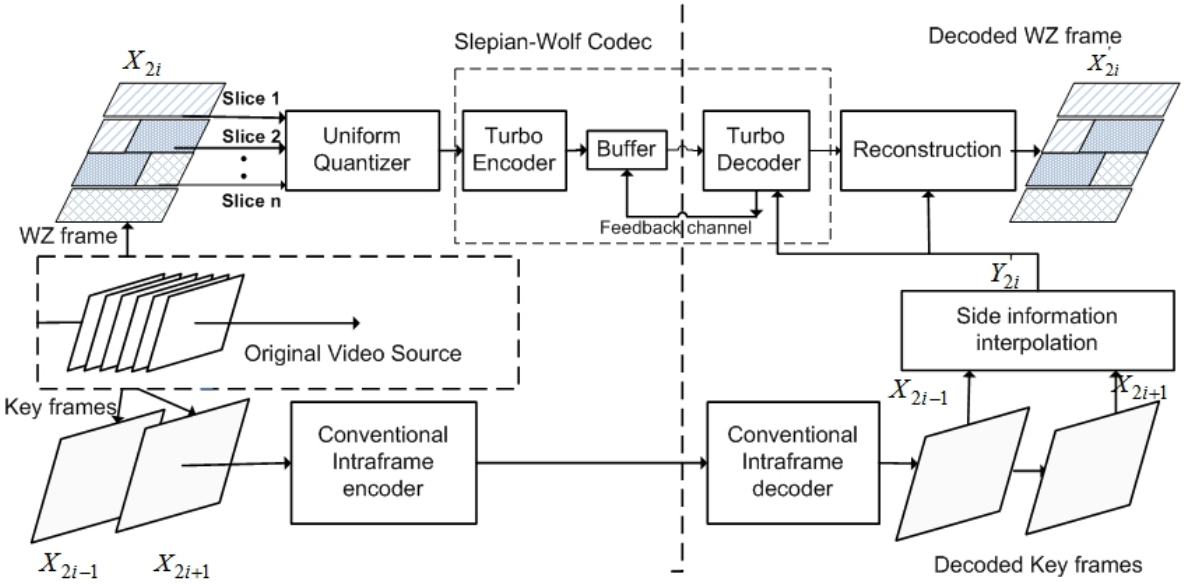


Fig.2 Proposed PDWZ with slice structure

The current WZ frame is divided into slices, in which the slice length, more precisely the number of macroblock (MB) in a slice, is dynamically decided by the quantization levels and MB size. Each slice should be no more than 5114 bits after the quantization. Given the quantization level is 2^M and MB size is b^2, then the maximum length l of each slice can be obtained by:

$$l = \left\lfloor \frac{5114}{b^2 * M} \right\rfloor \quad (1)$$

where l is the rounded integer value. Therefore, a WZ frame with r rows and c columns can be divided into n slices with the length of l, and a remaining 1 slice with the length of l1 where $0<l_1<l$. The relationship between n and l_1 is expressed as:

$$(n, l_1) = Quo\_rem(\frac{r * c}{b^2 * l}) \quad (2)$$

The Quo_rem function returns quotient of the division to n and the remainder to l_1. For an example, with 8-level quantization and 16x16 MB, a QCIF frame will be divided into 16 slices with length of 6 MBs and 1 slice with length of 3 MBs. Therefore, the total number of slices to be coded and transmitted is n+1(or n where l_1 =0). Note that the following case (l_1=1, b^2=4x4 and 2^M =2) is invalid since the target minimum interleaver size is 40 bits. Each pixel in a slice is quantized by using a uniform scalar quantizer with 2^M ={2, 4, 8, 16} levels. The quantized symbols are fed into Slepian-Wolf codec. Here, the 3GPP turbo codec with polynomial generator of (13,11) and constraint length of 4 is implemented as the Slepian-Wolf codec with the rate compatible puncturing method [26] to achieve flexible code rate and adapt the statistical change of side information, Y_{2i}. The turbo coded parity bits are saved into buffer for transmission and all the systematic bits are discarded. At the receiver, Laplacial distribution is used to model statistic dependency between original WZ frame and side information Y_{2i}. The WZ frame is decoded and reconstructed slice-by-slice by the turbo decoder with the help of Y_{2i} and calculated Laplacial parameter. Encoder initially sends a small fraction of parity bits to commence the decoding. A slice is considered successfully decoded when the resulting bit-error-rate (BER) is lower than a predefined 10^{-3} and the next slice is sent for decoding subsequently. If decoding fails, a request for more parity bits will be sent back to encoder via feedback channel. This process is repeated until the BER of the current slice decoding is lower than 10^{-3}. Reconstruction of a complete WZ frame is performed after all successive slices received.

3.2 Performance of proposed PDWZ

Firstly, the performance of the proposed PDWZ is evaluated in comparison with two other conventional PDWZ from [22] and [23]. H.264+ inter-frame coding with IBIB structure is simulated as a reference. The "Foreman" QCIF video sequence is used in the simulation. WZ frame rate is set to 15 fps. Each pixel is quantized with levels of 2^M={2,4,8,16} to give four rate-distortion points. The side information is interpolated by bidirectional motion compensated interpolation. The block size is set to 16, full search motion estimation (ME) window size is ±7 pixels and the bidirectional ME window size is ±4 pixels.

Fig.3 and Fig. 4 show that the proposed PDWZ has significantly enhanced the system R-D performance comparing with the conventional PDWZ systems in [22] and [23]. It is observed that the PSNR gain of proposed PDWZ can achieved up to 1.6 dB in comparison with [22] and 0.6 dB with [23] for "Foreman" sequence.

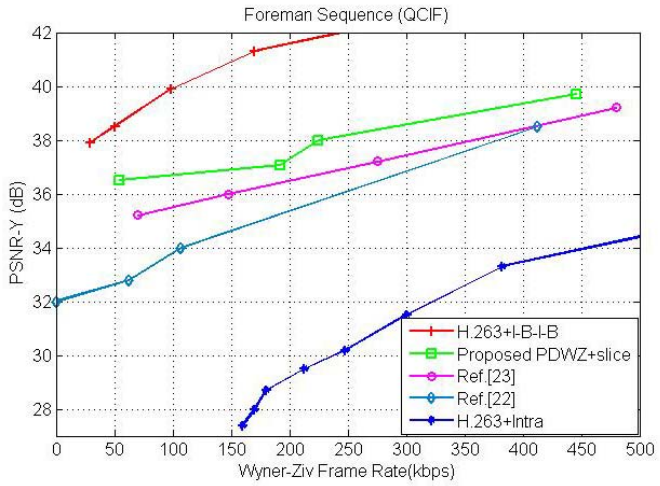


Fig. 3 RD performance of Foreman sequence

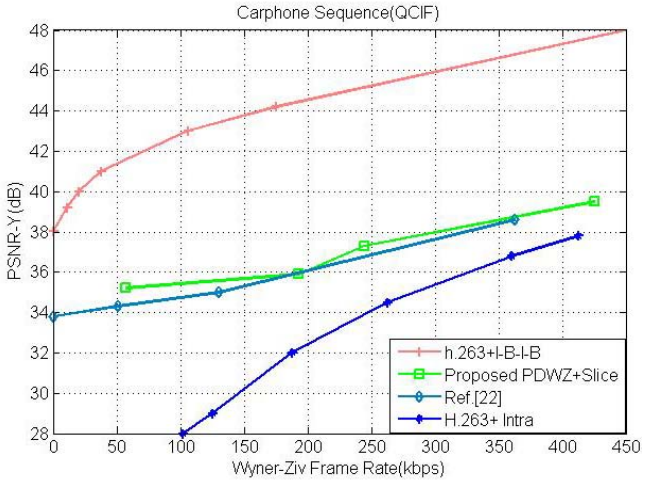


Fig. 4 RD performance of Carphone sequence

The system improvement achieved is largely attributed to the proposed slice structure. Unlike the proposed PDWZ, the conventional PDWZ will always transmits the parity bits of whole WZ frame repetitively during the "request and decode" process, which also the cause of increased data rate. However, most parts of the current WZ frame could have been successfully decoded, thus majority of the parity bits sent are unnecessary and waste of the bandwidth. In proposed PDWZ, the encoding of WZ frame is carried out in slice structure and only the parity bits of the current slice is transmitted during the "request and decode" process for decoding of the current WZ frame slice-by-slice. In this way, the required data rate is reduced and the rate control is optimized due to the tighter control of the parity bits transmission. In other words, the proposed PDWZ does not sent unnecessary parity bits to ensure the decoding of WZ frame as compared to the conventional PDWZ. It can be observed from Fig.3 that much bitrate of the proposed architecture is saved to achieve same level of PSNR when comparing to the PDWZ [22,23]. The bitrate could be saved up to 150kbps comparing to [23] and 200 kbps comparing to [22] for "Foreman" sequence.

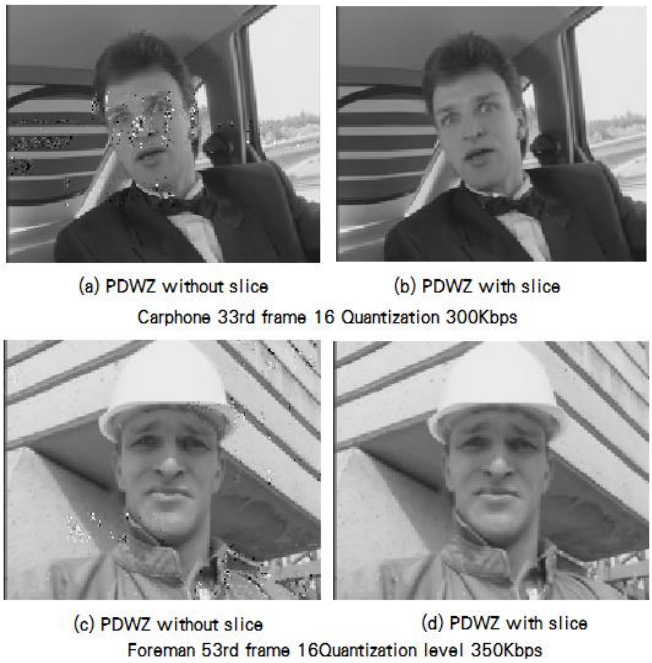


Fig.5 Perceptual quality comparison

Fig.5 shows the perceptual quality comparison for Foreman and Carphone sequence. Given the same bitrate of WZ frame, the amount of parity bits sent is fixed. It is apparent in the picture that the proposed PDWZ efficiently used received parity bits and correct most errors in frame due to the slice structure. While for conventional PDWZ, many received parity

bits are wasted while many errors can still be spotted and need further amount of parity bits to correct.

4. Adaptive Rate Selection Scheme

4.1 General description of ARS

The ARS starts to work when a slice is received and decoded unsuccessfully. Unlike the previous PDWZ in [22], the side information quality will be evaluated and sent back to encoder via feedback channel to aid the encoder to make judgment on how many parity bits should be sent. In this paper, the ARS scheme is performed by so called ARS table at encoder, which is built by experimental results and stores the mapping relationship between side information quality and proper code rate choices that can successfully decode the slice. Once the slice is received and considered fail in decoding, the side information quality will be evaluated and sent back to encoder via feedback channel. The encoder will look up the table according to evaluation and find the appropriate code rate to send correct number of parity bits to decoder. If decoding still fails, the encoder will choose next candidate code rate in the table and this process repeated until the whole decoding is finished. Comparing to the conventional WZ coding, the decoding process can be finished in very few rounds and much system delay are expected to be eliminated.

4.2 Side information evaluation

Before the side information evaluation criteria is introduced, we first illustrate the process of calculating Laplacian distribution parameter which used in decoding process. In previous PDWZ architecture, a virtual statistics dependency channel between the side information denoted by Y_{2i} and the original WZ frame denoted by X_{2i} is assumed. Particularly, Laplacian distribution is adopted to describe the statistics correlation between Y_{2i} and X_{2i}. The residual r(r=|X_{2i}-Y_{2i}|) between Y_{2i} and X_{2i} can be considered as Laplacian variable, which can be described as

$$f(X_{2i}-Y_{2i})=\frac{a}{2}e^{-a|X_{2i}-Y_{2i}|} \quad (1)$$

By using previously received key frame, the Laplacian parameter a is estimated by follows:

$$a^2=\frac{2}{\delta^2} \quad (2)$$

$$\delta^2=E[(r-E(r))^2]=E(r^2)-E(r)^2=E(r^2)-\mu^2 \quad (3)$$

where μ denotes the mean value (expected value) of the residual r and δ^2 denotes the variance of the residual r. However, since X_{2i} is not available at decoder during decoding, r usually is obtained by using side information and the one of adjacent key frames, r(r=|X_{2i-1}-Y_{2i}|) It is proved in [7] that using received key frame to estimate a won't degrade the system performance. With estimated a and side information, all possible input symbols of X_{2i} will be tested by (1), after which the symbol with highest possibility will be chosen as discarded systematic bits and fed into turbo decoder for decoding with received parity bits. In the proposed ARS scheme, the variance of each slice in the side information is evaluated as the criteria to measure the slice quality.

$$\delta_s^2=E[(r_s-E(r))^2]=E(r_s^2)-E(r)^2=E(r_s^2)-\mu^2 \quad (4)$$

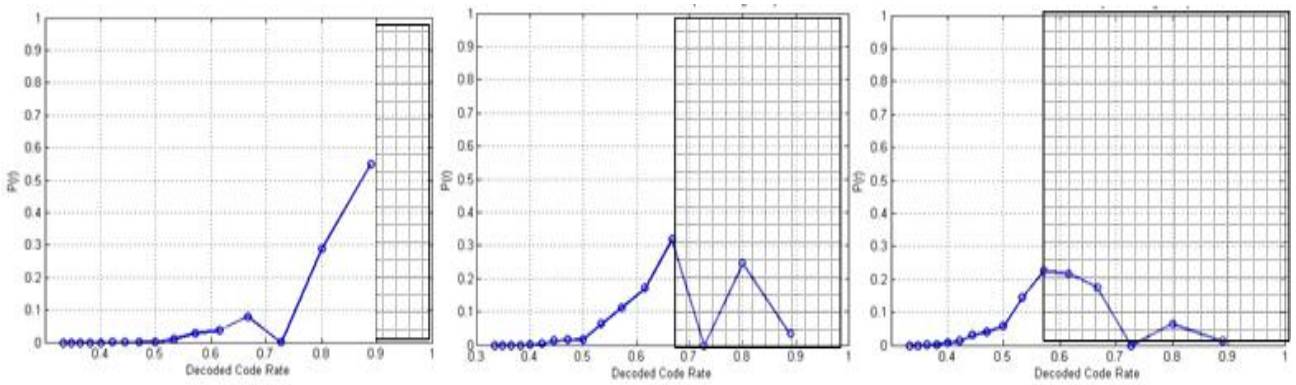
where r_s (r_s=|$S_{2i-1,k}$-$S_{2i,k}$| denotes the residual between the k-th slice of the key frame X_{2i-1} denoted by $S_{2i-1,k}$ and the k-th slice in the side information Y_{2i} denoted by $S_{2i,k}$. μ and r keep same definition as above. The variance δ_s^2 is the measure of statistical disperse degree of r_s around μ. Larger δ_s^2 denotes there are some residual values in r_s that are far from μ, which also implies the larger difference between some pixels in $S_{2i-1,k}$ and $S_{2i,k}$ In the case of $\delta_s^2<\delta^2$, the residual r_s more likely locate close around μ thus the input symbol of X_{2i} can be predicted correctly with higher probability Vice versa, when $\delta_s^2>\delta^2$,it is quite easily make wrong prediction and need to request more parity bits to finish decoding. It is observed that larger is the δ_s^2/δ^2, more parity bits are needed. The evaluation process at this stage can be regarded as a kind of estimation of virtual channel condition essentially.

4.3 Process to build ARS table

Based on above analysis, we will build the ARS table by experimental results with the purpose of deriving the relation between the range of δ_s^2/δ^2 and the number of parity bits (namely the code rate) which can successfully decode the slice. Once the relationship is found, the encoder can look up the ARS scheme table to find the correct code rate to send the parity bits according to δ_s^2/δ^2 from feedback. The target relationship can be found by running intensive simulation. 4 video sequences "Foreman", "Carphone", "Coastguard" and "Slient" with different natures were simulated. For each video sequence, 140 WZ frames are encoded and transmitted. Each WZ frame is quantized by 16 quantization levels thus each frame can be divided into 25 slices. There are total 14,000 sample slices used to provide enough data for statistics. The

puncturing period of RCPT is set to 8 to provide candidate code rate range of {8/9,8/10,8/11…8/24}. The range of δ_s^2/δ^2 and final successful decoding code rate of each slice is monitored during simulation. Fig.3 (a,b,c) shows the probability distribution *P(r)* of each code rate that has sucessuflly decode the slice within certain range of δ_s^2/δ^2 based on simulation results, the point {0.889, 0.8, 0.727..0.333} located at *x* axis is corresponding with the code rate of {8/9,8/10,8/11,….8/24}.

It can be seen from Fig.3(a), when $\delta_s^2 < \delta^2$, most slices can be decoded by the code rate of 8/9, with the increase in the value of δ_s^2, as shown in (b) and (c), the final decoding code rate decreased correspondingly, in other words, more parity bits are sent. Based on Fig.3, the ARS scheme is built by the following rules and shown in Table 1. Firstly, the highest code rate 8/9 should be added into the list since encoder has no knowledge of quality of side information at first transmission. The encoder will assume that the side information quality is good enough to perform successful decoding. Then, for each slice under certain range of δ_s^2/δ^2, the second code rate candidate will be the code rate with highest probability in the figure. The next candidate rate is a lower code rate but with higher probability to decode successfully. Note the higher code rate but with lower probability (shadow area in Fig.6.) and the code rate with probability of zero will not be eligible as candidate. Finally, the lowest code rate 8/24(1/3) will be added in case of the complex situation that all previous code rates are unable to decode successfully. Table I also includes the code rate list of ERS and QRS for comparison. It should be noticed that this scheme is built for the case that puncturing period is set to 8. Scheme for the other cases can be derived by the same way.



a) $\delta_s^2/\delta^2 < 1$, 9614 sample slices **b)** $1 \le \delta_s^2/\delta^2 \le 2$ 2597 sample slices **c)** $2 < \delta_s^2/\delta^2$ 1789 sample slices

Fig.6 Probability distribution of final decoding code rate with different range of δ_s^2/δ^2 obtained using Foreman, Coastguard, Carphone and Slient Sequences

The ARS scheme works as follows: the encoder first send a few punctured parity bits according to the code rate of 8/9 for the current slice, if decoding fails, the decoder will evaluate the range of δ_s^2/δ^2 for the current slice and feed it back to the encoder. Based on the range of δ_s^2/δ^2, the encoder will look up the Table I and choose the next code rate from the list and send the corresponding amount of parity bits. The process repeats until the current slice is successfully decoded.

4.4 Experimental Results

In order to test the performance of ARS scheme and make fair comparison with ERS, 4 other video sequences with those 4 video sequences used to build up ARS table are together used to verify the scheme. The simulation parameters are same as abvoe section , the WZ frame rate is 15fps, quantization level only choose 16 for simulation. More importantly, the number of times that the request happens for each slice to decode successfully is recorded during simulation. Table II shows the final result of ARS scheme in comparison with ERS scheme.

It is evident from Table 2 and Fig.7 that the R-D performance of ARS is very close to that of ERS. The average PSNR of ARS is even higher than ERS slightly, only the corresponding bitrate of WZ frame is slightly higher than that of ERS scheme. This is due to the ARS scheme tries to send the most probably appropriate amount of parity bits but can not guarantee the least parity bits at same time. The advantage of ARS scheme is apparent and significant when we compare the average request times per slice.

Much system delay has been eliminated due to the efficient and accurate rate selection. For Salesman sequence, the proposed scheme reduced system delay up to 94.2%, and has greatly improved the system processing speed and efficiency. To further explore the performance of ARS in WMSN, we assume that the system can only tolerant the maximum system delay up to 3 times of feedback. Under this circumstance, each scheme will only have first 4 of code rate as choices during decoding process.

Table 1: ARS table

Scheme		Code rate Selection list
ERS		8/9, 8/10, 8/11, 8/12, 8/13, 8/14, 8/15, 8/16, 8/17, 8/18, 8/19, 8/20, 8/21, 8/22, 8/23, 8/24
QRS		8/9, 8/10, 8/12, 8/16, 8/24
ARS	$\delta_s^2 / \delta^2 < 1$	8/9, 8/10, 8/12, 8/13, 8/14, 8/15, 8/24
	$1 \le \delta_s^2 / \delta^2 \le 2$	8/9,8/12, 8/13, 8/14, 8/15, 8/16, 8/17, 8/18, 8/24
	$2 < \delta_s^2 / \delta^2$	8/9,8/14,8/15,8/16,8/17, 8/18, 8/19, 8/20,8/24

Table 2: R-D performance comparison for ARS and ERS scheme for 16 Quantiztion levels

Video	Average Code Rate		Average number of request happened per slice		Average PSNR(dB)		Delay reduction ratio
	ERS	ARS	N_{ERS}	N_{ARS}	ERS	ARS	$\frac{N_{ERS} - N_{ARS}}{N_{ERS}}$
Foreman	0.769	0.7509	1.6618	0.6768	39.7346	39.7346	59.2%
Carphone	0.73848	0.7228	2.2461	0.979	39.52	39.1884	56.4%
Coastguard	0.7531	0.7415	1.9281	0.7907	39.1385	38.8908	58.9%
Slient	0.8286	0.8181	0.8941	0.2295	43.125	43.125	74.3%
Claire	0.874	0.827	0.172	0.061	49.320	49.321	64.5%
Mother	0.860	0.819	0.388	0.045	45.967	45.970	88.4%
News	0.841	0.828	0.692	0.106	44.426	44.426	84.7%
Salesman	0.868	0.831	0.279	0.0162	45.950	45.953	94.2%

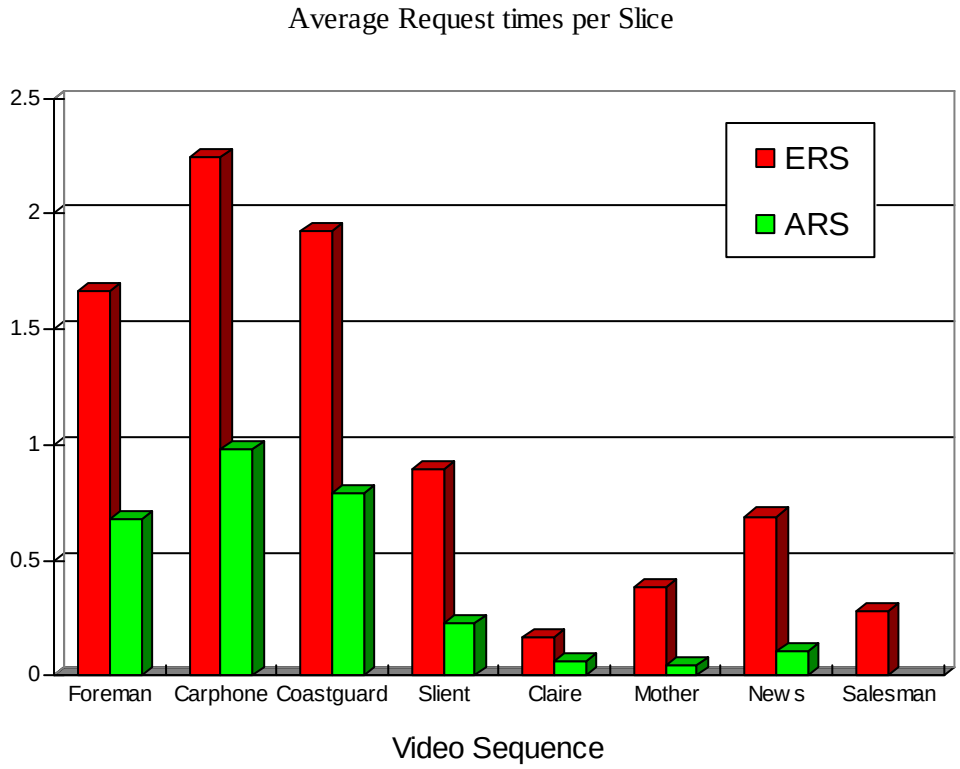


Fig.7 Average Request times per slice for different video sequences with ERS and ARS schemes

(a) ERS (b) ARS (c) QRS

Fig 8. Perceptual quality comparison News sequence 123-th frame, 16 Quantization level

Fig.8 shows the perceptual quality comparison for news sequence by three rate selection schemes. It can be noticed that due to the limitation in delay tolerance, the ERS is too slow to send enough and appropriate number of parity bits therefore many slices were not decoded successfully and gave the worst perceptual quality. Conversely, the QRS dropped the code rate rapidly. No matter the side information quality is good or not, after 3 rounds of feedback loop, there have already been half of all parity bits are sent for decoding. Too much redundant parity bits are sent, which result in the much higher bitrate than other two schemes. Even though the perceptual quality obtained is very high, the bandwidth is not efficiently utilized. Comparing to ERS and QRS, the proposed ARS scheme find the correct code rate quickly based on the evaluation of side information, the whole decoding is done quickly within 3 rounds of feedback.Under the condition of low latency given, not only is a satisfying perceptual quality delivered, the bitrate is also saved and the bandwidth is also efficiently utilized.

5. Conclusion

This paper has presented an improved PDWZ video coding architecture in view of constructing WMSN for multimedia broadcasting. The slice structure and adaptive rate selection scheme have been proposed

for PDWZ. The proposed slice structure help optimizing the rate control operation of PDWZ which enhanced the R-D performance and improved the receiving perceptual quality. With the slice structure, PDWZ is adaptable to any interleaver size used in Slepian-Wolf turbo codec, which has greatly reduced the coding and decoding latency. Based on the innovative quality evaluation of side information, the proposed adaptive rate selection solution adaptively estimates the virtual channel condition and has significantly reduced the delay caused by frequent feedback during decoding. The proposed ARS scheme achieved exceptional perceptual quality with lower latency and bitrate as compared to traditional ERS and QRS schemes.

References:

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam , E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no 4,pp 393-422, March 2002.

[2] A. Sluzek, P. Annamalai, M.S.A Islam, "wireless sensor network for visual detection and classification of intrusions", *WSEAS Transactions on Circuits and Systems*, 4 (12), pp. 1855-1860, 2005.

[3] M. Rizzi, M ,Maurantonio, B. Castagnolo, "A wireless sensor network for security systems adopting the bluetooth technology," *WSEAS Transactions on Circuits and Systems*, 5 (5), pp. 652-657,2006.

[4] I. F. Akyildiz, T. Melodia, K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Computer Networks*, vol. 51, no 4, pp. 921-960, March 2007.

[5] G. Raja, M.J. Mirza, "A new scheme of suppressing blocking artifacts in H.264/AVC deblocking filter for low bit rate video coding," *WSEAS Transactions on Circuits and Systems*, 6 (1), pp. 182-186, 2007.

[6] R. Puri, A. Majumdar, P. Ishwar, K. Ramchandran, "Distributed video coding in wireless sensor networks," *IEEE Signal Proc. Magazine*, vol. 23, pp. 94-106, 2006.

[7]B. Girod, A. M. Aaron, S. Rane, D. Rebollo-Monedero, "Distributed video coding," *IEEE Proc.*, vol. 93, pp. 71-83, 2005.

[8] J. D. Slepian, J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, pp. 471–480, 1973.

[9] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inf. Theory*, vol. IT-20, no 1, pp. 2–10, 1976.

[10]M. Kornfeld, G. May, "DVB-H and IP datacast - Broadcast to handheld devices", *IEEE Trans. Broadcasting*, vol. 53, no1, pp. 161-170, 2007.

[11G. Faria, J.A. Henriksson, E. Stare, P. Talmola, "DVB-H: Digital Broadcast Services to Handheld devices," *IEEE Proc.*, vol. 94, no 1, pp. 194-209, 2006.

[12]M.D. Colagrosso, "A classification approach to broadcasting in a mobile ad hoc network," *IEEE Conf. Communications*, vol. 2, pp. 1112-1117, 2005.

[13]M. Sheng, J. Li, Y. Shi, "Relative degree adaptive flooding broadcast algorithm for Ad hoc networks," *IEEE Trans. Broadcasting*, vol. 51, no 2, pp. 216-222, 2005.

[14]G. M. Muntean, N. Cranley, "Resource efficient quality-oriented wireless broadcasting of adaptive multimedia content," *IEEE Trans. Broadcasting*, vol. 53, no 1, pp. 362-368, 2007.

[15]G. Berger, R.Goedeken, J. Richardson, "Motivation and implementation of a software H.264 real-time CIF encoder for mobile TV broadcast applications" *IEEE Trans. Broadcasting*, vol. 53, no 2, pp. 584-587, 2007.

[16]S. R. Gulliver, G. Ghinea, "The perceptual and attentive impact of delay and jitter in multimedia delivery," *IEEE Trans. Broadcasting*, vol. 53, no 2, pp. 449-458, 2007.

[17]M. Rezaei, M. M. Hannuksela, M. Gabbouj, "Tune-in time reduction in video streaming over DVB-H," *IEEE Trans. Broadcasting*, vol. 53, no 1, pp. 320-328, 2007.

[18]F. Hartung, U. Horn, J. Huschke, M. Kampmann, T. Lohmar, M. Lundevall, "Delivery of broadcast services in 3G networks," *IEEE Trans. Broadcasting*, vol. 53, no 1, pp. 188-198, 2007.

[19]I.V. Bajić, "Efficient cross-layer error control for wireless video multicast," *IEEE Trans. Broadcasting*, vol. 53, no 1, pp. 276-285, 2007.

[20]L. Huang, K. A. Chew, S. Thilakawardana, Y. Liu, K. Moessner, R. Tafazolli, "Efficient group-based multimedia-on-demand service delivery in wireless networks ," *IEEE Trans .Broadcasting*, vol. 52, no 4, pp. 492-504, 2006.

[21]A. Durresi, V. Paruchuri, "Broadcast protocol for energy-constrained networks," *IEEE Trans. Broadcasting*, vol. 53, no 1, pp. 112-119, 2007.

[22]A. Aaron, R. Zhang, B. Girod, "Wyner-Ziv coding of motion video," in *Asilomar Conf. Signals, Systems and Computers*, pp. 240-244, 2002.

[23]J.Ascenso, C. Brites, F. Pereira, "Improving Frame Interpolation with Spatial Motion Smoothing for Pixel Domain Distributed Video Coding," in *5th EURASIP Conf. Speech and Image Processing, Multimedia* Commun. and Services, Slovak Republic, 2005.

[24]A. Aaron, S. Rane, E. Setton, B. Girod, "Transform-domain wyner-ziv codec for video," *in SPIE Proc.* - The International Society for Optical Engineering, pp. 520-528, 2004.

[25]Z. Xue, K.K.Loo, J.Cosmas, "Embedded side information refinement for pixel domain Wyner-Ziv video coding towards UMTS 3G application," *IEEE Conf. ICME07*, Beijing, P.R.China, July 2007

[26] D. Rowitch, L. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo codes," *IEEE Trans. Commun*., vol. 48, no 6, pp. 948–959, 2000.