

Error Robust Coding for the FGS Enhancement Bitstream

Rong Yan¹, Feng Wu², Shipeng Li², Ran Tao¹, Yue Wang¹ and Ya-Qin Zhang²

(1 Beijing Institute of Technology, #7 BaiShiQiao RD, Haidian, Beijing, 100081, China)

(2 Microsoft Research China, #49 ZhiChun RD, Haidian, Beijing, 100080, China)

ABSTRACT

Fine Granularity Scalable (FGS) video coding scheme provides a layered bitstream structure with different importance at different layers. This property makes FGS bitstream more suitable for transmission over the error-prone channels with unequal error protection schemes. Even though the hierarchical FGS enhancement bitstream can always resynchronize in the next video packet when there are transmission errors, each error will still cause a large amount of received enhancement bits undecodable due to the dependency among enhancement layers. In this paper, we first propose to use the Universal VLC (UVLC) technique to further limit the effects of transmission errors to within one or only a few blocks by taking advantage of the excellent self-synchronization property of the UVLC. Furthermore, the higher priority data of Coded Bit Pattern (CBP) is separated from the DCT data in the same video packet and put in the beginning of each video packet to avoid the error impact from the DCT coefficients. The experimental results show that the FGS enhancement bitstream with the proposed coding techniques has greater capability in locating and isolating errors and resynchronization. The decodable bits of FGS enhancement layer increase significantly while the bit rate increases. Meanwhile, the decoded video quality can be improved up to 0.8dB in PSNR.

1. INTRODUCTION

With the rapid growth in mobile communication, there is an increasing demand for reliable video transmission over wireless links from more and more important real applications. The FGS (Fine Granularity Scalable) coding [1][2] provides a very promising solution for robust video transmission over error-prone channels, such as wireless channels. Firstly, the FGS has an inherent error recovery feature, which can immediately recover from any transmission errors in the enhancement layer in the next frame. Secondly, the FGS provides a layered bitstream structure with different importance at different layers. This bitstream structure makes it very suitable for applying UEP (Unequal Error Protection) technique that provides stronger error protection to the more important data, such as the base layer bitstream.

In [3], we proposed a hierarchical bitstream structure with resynchronization markers for the FGS enhancement layer, which has already been adopted in MPEG-4. The enhancement layer bitstream is divided into many video packets by the resynchronization markers. When there are errors detected in the enhancement layer bitstream, the decoder only discards the video packet in which error occurs and searches for the next resynchronization point, such as VOP (Video Object Plane) start code, BP (Bit Plane) start code, and VP (Video Packet) start

code. The hierarchical bitstream structure provides a simple and yet effective solution to increase the error resilience capability of the FGS enhancement layer.

Although transmission errors can be successfully localized to within one video packet by using the hierarchical bitstream structure, the decoder still has to discard a large amount of correctly received enhancement bits due to the dependency among enhancement layers. An erroneous video packet in one layer also causes all the corresponding video packets in the higher enhancement layers undecodable, especially, when the transmission errors take places in those video packets in the lower enhancement layers. One such transmission error may cause a great amount of bits undecodable, thus we cannot ignore such its impact. The experimental results using simulated wireless channels show that the percentage of the bits correctly decoded is only about 20%~60% of the total bits transmitted. Therefore, a more challenging question is how to limit the transmission errors to an even smaller region and to make more received bits decodable.

Since the FGS enhancement layers are encoded with bit plane coding technique, there is strong dependency among bit planes. Therefore, it is more difficult to improve the error resilience capability in the enhancement layer bitstream than in the base layer bitstream. In this paper, we first try to replace the VLC table for bit plane coding with Universal VLC (UVLC) [4][5] because of the excellent self-synchronization property of UVLC. This can help the FGS enhancement decoder to limit the transmission errors to one or only a few blocks. However, the CBP data in the enhancement macroblock header may still block the UVLC resynchronization. Compared with the DCT data, the CBP data is more important for the FGS decoder. Therefore, the CBP data in each video packet is output to bitstream file before DCT data. The same UVLC table with different semantics is used to encode CBP data.

This paper is organized as follows. Section 2 redefines the semantics of UVLC to make it suitable for bit plane coding. Section 3 discusses the proposed error robust coding techniques in detail. Meanwhile, the decoding process is described according to the proposed FGS bitstream structure in this section. Section 4 presents the experimental results by the simulated wireless channel. Finally, we summarized the paper in section 5.

2. REDEFINITION OF UNIVERSAL VLC

In order to improve the capability of error detection and localization and resynchronization of the FGS enhancement bitstream, we adopt the UVLC technique [4][5][6] for coding both CBP and DCT coefficients. The rapid self-synchronization property of UVLC makes the FGS decoder limit the impact of transmission errors to a smaller region, and brings great benefit for the enhancement decoding in

error-prone environment. In addition, the simple relationship between the code number and the length of a codeword leads to fast encoding and decoding processes.

For the FGS enhancement layer coding, we redefine the semantics of each codeword in the UVLC table. The UVLC (RUN, UVLC) table is showed in Table 1, where X_n takes value one or zero. The first item denotes the case of zero run length. The second item EOB denotes the end of each block or all the coefficients of this block in this bit plane are zero. Other items denote the different run lengths. Note, there are two definitions for EOB “001” and “011”. If the number of non-zero coefficients in a block is odd, the EOB “001” is used in this block, otherwise the EOB “011” is used in this block. This is very useful for detecting the transmission errors, especially for transmission over the random error channels. The two different EOBs provide the UVLC table an additional capability to detect the parity change in the number of non-zero coefficients.

The codeword consists of the coarse code and INFO, and the two parts are interleaved bit by bit. The INFO part is defined as

$$INFO = x_n \dots x_1 x_0,$$

while the coarse code is a binary array containing a series of zeros followed by a ‘1’ in the end. L is the length of codeword in bits. The number of bits in the INFO part is $L/2$. Accordingly, The definition of the code number is

$$code\ number = 2^{L/2} + INFO - 1.$$

When INFO and L are known, the regular structure of the UVLC table makes it very easy to generate a codeword bit by bit according to the INFO. Similarly, a decoder can also directly decode the INFO bit by bit until the last ‘1’, which indicates the end of a codeword.

Table 1: RUN UVLC table.

uvlc table in compressed form	code size	absolute value of run
1	1	0
$0 x_0 1$	3	EOB & All Zero
$0 x_1 0 x_0 1$	5	1:4
$0 x_2 0 x_1 0 x_0 1$	7	5:12
$0 x_3 0 x_2 0 x_1 0 x_0 1$	9	13:28
$0 x_4 0 x_3 0 x_2 0 x_1 0 x_0 1$	11	29:60
$0 x_5 0 x_4 0 x_3 0 x_2 0 x_1 0 x_0 1$	13	61:63

If there is a one-bit transmission error in the INFO bits, only the current codeword will be incorrectly decoded, the next codeword will not be affected by this error. If there is a one-bit error at the position of zeros in the coarse code, a wrong ending ‘1’ bit is obtained. The current and the next codewords are wrongly decoded, but the third one will be synchronized again. At last, if the one-bit error occurs at the last ‘1’ bit of the codeword, the current and the followed two codewords will be wrongly decoded and then the decoder will be synchronized again. Therefore, after a random bit error, the decoder can resynchronize after at most two code words. The example decoding processes in case of random bit errors is illustrated in Figure 1.

3. ERROR ROBUST CODING FOR THE FGS ENHANCEMENT BISTREAM

As mentioned above, although the hierarchical enhancement bitstream structure illustrated in [3] can successfully isolate the errors within one video packet, the decoder has to discard the whole video packet and the corresponding video packets in higher enhancement layers when an error is detected. Thus, a large amount of transmitted data still cannot be decoded. To further limit the impact of transmission errors to one or only a few blocks, we apply the UVLC technique for the enhancement layer bit plane coding. Since the CBP data is more important and has higher priority, the CBP data are first encoded in each video packet. Moreover, separation of CBP data from DCT data also makes sure that the self-synchronization of UVLC is not affected by CBP data or DCT data.

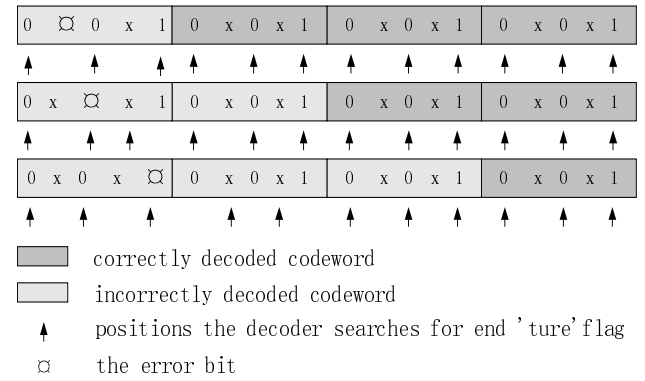


Figure 1: The decoding processes in the case of bit error occurring.

As showed in Figure 2 one video packet may consist of a VPH (Video Packet Header) and the slice data, or a BPH (Bit plane Header) and the slice data for the first video packet of the every bit plane. Each slice includes a few macroblocks in raster-scan order. Since the CBP only exists in the first two bit planes, the CBP field for each macroblock is first encoded in these bit planes. The bits followed are used to encode the run length of zero coefficients for each block. The one-bit sign is coded with the MSB (Most Significant Bit) of each coefficient together.

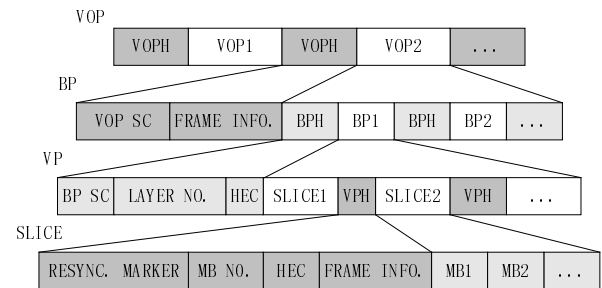


Figure 2: The hierarchical structure of the proposed enhancement layer bitstream.

The CBP field in a macroblock indicates the coding pattern of its six blocks. If CBP is incorrectly decoded, the decoder will make wrong decision on which block should be decoded, so the following macroblocks in this video packet may be affected. It is clear that the CBP is more important than the run length of DCT coefficients. As a result, we separate it from the DCT coefficients during the coding process, and place the CBP data of all MBs at the beginning of each slice. So the errors occur in run length codewords will not affect the CBP of the macroblocks followed. At the same time, the self-resynchronization of UVLC will also not be affected by the CBP data.

Figure 3 illustrated the new structure of slice we proposed in this paper. In the proposed slice structure, a unique CBP marker separates the CBP data from the DCT data in the same video packet. The CBP marker is derived from the CBP VLC table such that the codeword is not any possible emulation of the CBP UVLC tables. This marker also indicates to the decoder to stop the decoding of CBP and to start the decoding of DCT coefficient. In fact, the CBP marker provides the decoder strong capability to validate the CBP data. If an error is detected in the CBP section, the decoder has to give up the decoding of this video packet and find the next resynchronization point.

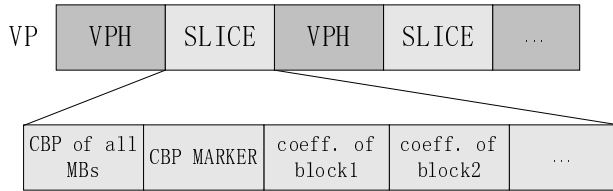


Figure 3: The proposed slice structure.

For DCT coefficient coding, we change the semantics of bit plane VLC from (RUN, END) to (RUN, EOB). As mentioned in Section 2, the UVLC coding has excellent self-synchronization property. If the transmission error doesn't corrupt the EOB codewords, the error will only affect two run length codewords followed in the worst case, and then the decoder can gain synchronization again. In other words, this kind of errors only affects the decoding results of one block. So when an error is detected, the decoder continues the decoding process until an EOB, instead of discarding the whole video packet. Only the block with errors will be set as failed and the corresponding blocks in the higher enhancement layers also will not be decodable. If the error unfortunately occurs at the EOB codewords or the error occurred in the previous codewords affects the EOB decoding, it can make several blocks undecodable. In the worst case, the decoder does not know exactly how many blocks are corrupted by this kind of errors. Consequently, once such an error is detected the decoder has to discard the rest of the current video packet.

4. EXPERIMENTS AND ANALYSIS

Simulation experiments are used to verify the proposed techniques. The test sequences, Foreman, Coastguard, and Akiyo (CIF format) are used in our experiments. In the experiments, every I frame is followed by 99 P frames, and no B frame is used. The encoded frame rate is 10Hz. The limitation

on the length of the motion vectors is set to within ± 32 pixels. The bit rate of base layer is 128 kbps with TM5 rate control. The enhancement bitstream is truncated at 128kbps/s, 256kbps/s, ..., up to 896kbps/s. The channel model generates a simulated wireless channel with random errors and its error rate is $10e-3$. The goal of the simulations is to verify the robustness of the new enhancement bitstream with the proposed techniques. Here we assume that there are no errors in the transmission of the base layer bitstream. In addition, no channel coding is applied to the enhancement bitstream for channel error protection. Each video packet contains the macroblocks of one row.

The experimental results are given in Figure 4, Figure 5 and Figure 6, which are the average values of 5 experiments because of the randomness of the simulated channel. These results are the actual decoded bits for the original and the new enhancement bitstreams with different bit rates. The actual decoded bits per frame in the enhancement bitstream are defined as the total bits for correctly decoded video packets. Note that the number of overhead bits introduced by the UVLC table is also included in it, because it is difficult to accurately calculate the number of overhead bits introduced by UVLC table compared to the original VLC table. But we can calculate the difference of the total bits used to encode two types of enhancement layer bitstreams from start of one VOP to a special video packet. Then the overhead percentage is defined as this difference versus total bits, including the base layer bits and the enhancement layer bits. The overhead percentage is shown in Table 2 for three tested sequences, where the bit rate is the sum of the base layer and the enhancement layer bitstream.

Table 2: The overhead percentage in new enhancement bitstreams.

Bit rate (kbps/s)	Foreman	Coastguard	Akiyo
256	5.6284%	6.7931%	3.5913%
384	7.7794%	8.8956%	4.9825%
512	8.2755%	9.0386%	5.6052%
640	8.1495%	8.9790%	5.7851%
768	8.0150%	9.0501%	5.7613%
896	7.8339%	9.2090%	5.7526%
1024	7.6987%	9.0514%	5.7472%

From the experimental results we can see that the actual decoded bits of the two types of enhancement layer bitstreams linearly increase as the effective channel bandwidth increases from 128 kbps/s to 896 kbps/s. The decoded bits of the new enhancement bitstream are more than that of the original enhancement bitstream at all bit rates. The difference becomes more and more significant as the bandwidth increasing, and at the 896kbps/s the actual decoded bits of the new enhancement bitstream are two times as that of the original bitstream. For Foreman sequence, the gain of PSNR using the new bitstream is about 0.61dB when the enhancement bit rate is 896kbps, for the Coastguard sequence, it is about 0.86dB and for the Akiyo sequence it is about 0.64dB. Table 3 gives the PSNR gain of the three sequences at different bit rates.

Table 3: PSNR gain when transmitting error-phone channel.

Bit rate (kbits/s)	Foreman	Coastguard	Akiyo
256	0.06381	0.085325	0.159099
384	0.230292	0.391258	0.272949
512	0.405489	0.599582	0.37682
640	0.426495	0.637222	0.463146
768	0.44641	0.695074	0.604344
896	0.542404	0.818691	0.671055
1024	0.608885	0.856736	0.639564

From the number of actual decoded bits, the new enhancement bitstream should provide more gain in the coding efficiency. The reason why the PSNR gain is not as much as we expected is that some errors in the new enhancement bitstream are not detected. Those wrong DCT coefficients are still used to reconstruct video in the decoder. This will reduce the quality of the decoded video. In other words, although more bits are decoded in the new enhancement bitstream, some gains are reduced by the incorrectly decoded data. New methods are being investigated to discard those wrongly decoded DCT coefficients and hopefully, it will boost the performance of the proposed techniques significantly further.

5. CONCLUSION

Simple and effective error resilience techniques are very important to the FGS enhancement bitstream. In this paper, we first proposed to use the Universal VLC (UVLC) technique to further limit the impact of transmission errors to one or only a few blocks because of its excellent self-synchronization property. Moreover, the data of Coded Bit Pattern (CBP) with higher priority is separated from the DCT data in the same video packet. With the help of these two techniques, the FGS enhancement bitstream is more robust to random errors. Future works involves how to take advantage of the significantly increases correctly decoded bits and reduce the effects of wrongly decoded DCT coefficients.

6. REFERENCES

- [1] F. Ling, W. Li and H.-Q. Sun, "Bitplane coding of DCT coefficients for image and video compression", *Proceedings of SPIE VCIP'99*, San Jose, Jan.25-27, 1999.
- [2] W. Li, "Fine granularity scalability in MPEG-4 for streaming video", *ISCAS 2000*, Geneva, Switzerland, May 28-31, 2000.
- [3] Rong Yan, Feng Wu, Shipeng Li, and Ran Tao, "Error resilience methods for FGS video enhancement bitstream", *IEEE-PCM2000*, Sydney, Australia, Dec.13-15, 2000.
- [4] G. Bjontegaard, "Response to call for proposals for H.26L", *ITU-T Q.15/16, Doc. Q15-F-11*, Seoul, Korea, Nov. 3-6, 1998.
- [5] G. Bjontegaard, "Definition of the new coding elements from Telenor", *ITU-T Q15/16, Doc. Q15-J-28*, Osaka, May 16-18, 2000.
- [6] Y. Itoh, N. -M. Cheung, "Universal variable length code for DCT coding", *2000 IEEE International conference on*

image processing (ICIP2000), Vancouver, British Columbia, Canada, Sep. 10-13, 2000.

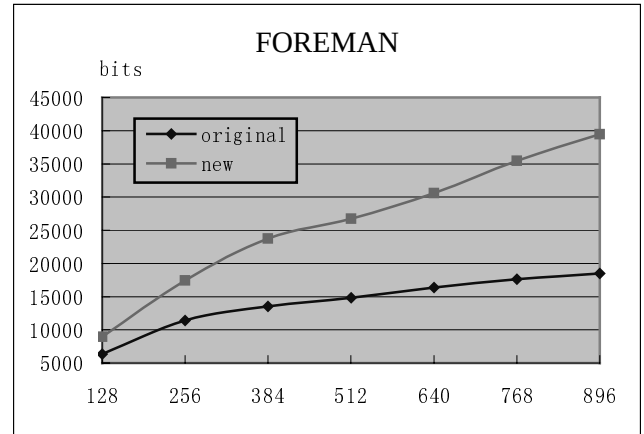


Figure 4: The actual decoded bits vs. the enhancement layer bit rate for Foreman sequence.

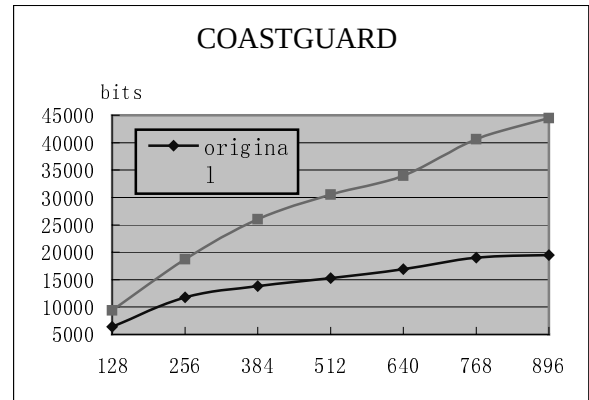


Figure 5: The actual decoded bits vs. the enhancement layer bit rate for Coastguard sequence.

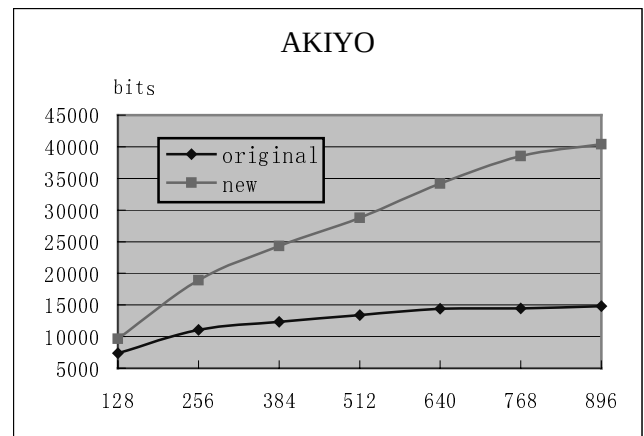


Figure 6: The actual decoded bits vs. the enhancement layer bit rate for Akiyo sequence.

[Figures and Tables](#)[Topics](#)[3 Citations](#)[9 References](#)[Related Papers](#)

Corpus ID: 15856116

Error Robust Coding for the FGS Enhancement Bitstream

R. Yan, Feng Wu, [+3 authors](#) [Ya-Qin Zhang](#) • Published 2001 • Computer Science

TLDR This paper proposes to use the Universal VLC (UVLC) technique to further limit the effects of transmission errors to within one or only a few blocks by taking advantage of the excellent self-synchronization property of the UVLC. [Expand](#)

No Paper Link Available [Save to Library](#) [Create Alert](#) [Cite](#)

Figures and Tables from this paper

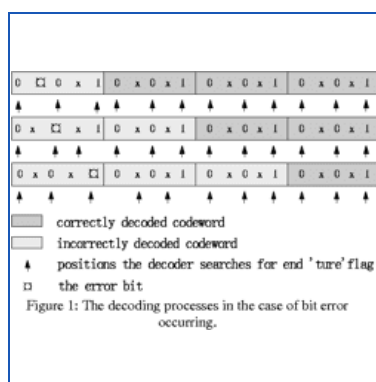


Figure 1

TABLE 1: KUN UVLC TABLE

uvlc table in compressed form	code size	absolute value of run
1	1	0
0 x ₀ 1	3	EOB & All Zero
0 x ₁ 0 x ₀ 1	5	1:4
0 x ₂ 0 x ₁ 0 x ₀ 1	7	5:12
0 x ₃ 0 x ₂ 0 x ₁ 0 x ₀ 1	9	13:28
0 x ₄ 0 x ₃ 0 x ₂ 0 x ₁ 0 x ₀ 1	11	29:60
0 x ₅ 0 x ₄ 0 x ₃ 0 x ₂ 0 x ₁ 0 x ₀ 1	13	61:63

Table 1

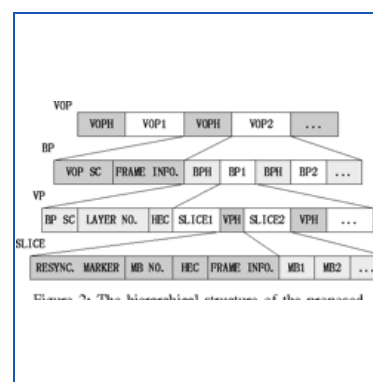


Figure 2

[View All 9 Figures & Tables](#)

Topics

 AI-Generated

Figures and Tables

Topics

3 Citations

9 References

Related Papers

Fine Granularity Scalable

Universal Variable Length Code

Transmission Errors

Dct Coefficient

Peak Signal-to-noise Ratio

Enhancement Layer

Bit Rate

3 Citations

Search authors, publica



Citation Type 

More Filters

Sort by Relevance 

Syntax friendly and syntax amendment approaches to error resiliency in the MPEG-4 Simple Scalable Profile

T. Shanableh G. Reid Computer Science · [Signal Process. Image Commun.](#) · 2005

PDF

 1 Excerpt  Save

MPEG video markup language and its applications to robust video transmission

Xiaoming Sun Chang-Su Kim C.-C. Jay Kuo Computer Science, Engineering ·

[J. Vis. Commun. Image Represent.](#) · 2005

 5

PDF

 1 Excerpt  Save

MPEG-4/XML FGS Approach for Multicast Video Synchronization

Xiaoming Sun C.-C. Jay Kuo Computer Science · [IS&T/SPIE Electronic Imaging](#) · 2003

TLDR It is demonstrated that the difficult resynchronization problem in layered video multicast can be effectively dealt with the MPEG-4/XML format. [Expand](#)

 2

 2 Excerpts  Save

9 References

[Figures and Tables](#)
[Topics](#)
[3 Citations](#)
[9 References](#)
[Related Papers](#)

Error Resilience Methods for FGS Video Enhancement Bitstream

Feng Wu R. Tao R. Yan S. Li Computer Science · 2000

TLDR The experimental results show that the proposed hierarchical enhancement layer bitstream has stronger error detection and resynchronization capabilities when transmitted over error-prone channels and provides much better decoded video.

[Expand](#)

15

[PDF](#)

2 Excerpts

Save


 Search

[Sign In](#)

Proceedings 2000 International Conference on... · 2000

TLDR The simulation results show that the proposed UVLC for DCT coefficient coding slightly outperforms several VLC tables of existing standards in terms of coding efficiency and can simplify the implementation and processing of VLC decoder.

[Expand](#)

28

Save

Bitplane coding of DCT coefficients for image and video compression

Fan Ling Weiping Li Hongqiao Sun Computer Science · [Electronic Imaging](#) · 1998

TLDR Bitplane coding of the DCT coefficients is a new coding scheme that provides a better performance than run_value coding under all conditions. [Expand](#)

43

Save

Fine granularity scalability in MPEG-4 for streaming video

Weiping Li Fan Ling Xuemin Chen Computer Science, Engineering ·

2000 IEEE International Symposium on Circuits and... · 2000

This paper describes the bitplane coding technique to achieve fine granularity scalability (FGS) in MPEG-4. It reports its coding efficiency results and discusses a few improvement methods.

68

1 Excerpt

Save

Figures and Tables

Topics

3 Citations

9 References

Related Papers

Related Papers

○○○○○○○○○

Error Resilience Methods for FGS Video Enhancement Bitstream

Feng Wu R. Tao R. Yan S. Li

Computer Science

TLDR The experimental results show that the proposed hierarchical enhancement layer bitstream has stronger error detection and resynchronization capabilities when transmitted over error-prone channels and provides much better decoded video.

15

Save Alert

Layered Wyner–Ziv Video Coding

Qian Xu Zixiang Xiong

Computer Science ·

[IEEE Transactions on Image Processing](#)

TLDR L base layer coding plus Wyner–Ziv enhancement layer coding are more robust against channel errors than H.26L FGS coding when the channel is noisy.

102

Save Alert

Error-resilient unequal protection of fine granularity scalable video bitstreams

Hua Cai B. Zeng G. Shen Shipeng Li

Computer Science · [IEEE International Conference on Communications](#)

TLDR An error-resilient unequal protection method is developed that adds redundant information optimally for loss protection and cancels completely the dependency among bitstream after loss recovery, leading to a better error resilience as well as higher robustness.

15

Save Alert

>

Stay Connected With Semantic Scholar

[Sign Up](#)[Figures and Tables](#)[Topics](#)[3 Citations](#)[9 References](#)[Related Papers](#)

What Is Semantic Scholar?

Semantic Scholar is a free, AI-powered research tool for scientific literature, based at the Allen Institute for AI.

[Learn More](#)

About

[About Us](#)[Meet the Team](#)[Publishers](#)[Blog](#)[AI2 Careers](#)

API

[API Overview](#)[API Tutorials](#)[API Documentation](#)[API Gallery](#)

Help

[FAQ](#)[Librarians](#)[Tutorials](#)[Contact](#)

Product

[Product Overview](#)[Semantic Reader](#)[Scholar's Hub](#)[Beta Program](#)[Release Notes](#)

Research

[Publications](#)[Researchers](#)[Research Careers](#)[Prototypes](#)[Resources](#)

Proudly built by AI2

[Collaborators & Attributions](#) • [Terms of Service](#) • [Privacy Policy](#) • [API License Agreement](#)

