Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 613714, 12 pages
http://dx.doi.org/10.1155/2015/613714

Research Article

A Novel High Efficiency Fractal Multiview Video Codec

Shiping Zhu, Dongyu Zhao, and Ling Zhang

Department of Measurement Control and Information Technology, School of Instrumentation Science and Optoelectronics Engineering, Beihang University, Beijing 100191, China

Correspondence should be addressed to Shiping Zhu; spzhu@163.com

Received 17 July 2014; Accepted 15 September 2014

Academic Editor: Guido Maione

Multiview video which is one of the main types of three-dimensional (3D) video signals, captured by a set of video cameras from various viewpoints, has attracted much interest recently. Data compression for multiview video has become a major issue. In this paper, a novel high efficiency fractal multiview video codec is proposed. Firstly, intraframe algorithm based on the H.264/AVC intraprediction modes and combining fractal and motion compensation (CFMC) algorithm in which range blocks are predicted by domain blocks in the previously decoded frame using translational motion with gray value transformation is proposed for compressing the anchor viewpoint video. Then temporal-spatial prediction structure and fast disparity estimation algorithm exploiting parallax distribution constraints are designed to compress the multiview video data. The proposed fractal multiview video codec can exploit temporal and spatial correlations adequately. Experimental results show that it can obtain about 0.36 dB increase in the decoding quality and 36.21% decrease in encoding bitrate compared with JMVC8.5, and the encoding time is saved by 95.71%. The rate-distortion comparisons with other multiview video coding methods also demonstrate the superiority of the proposed scheme.

1. Introduction

In recent years, multiview video (MVV) is attracting considerable attention. This is because MVV can provide consumers with depth sense to the observed scene as if it really exists in front of consumers, allow consumers to freely change views, and interactively modify the properties of a scene. This type of video may be offered in the future home electronics devices, such as immersive teleconference, 3DTV [1], 3 D mobile phone, and home video camcorder. For example, in immersive teleconference, there is an interaction between consumers. Participants at different geographical sites meet virtually and see each other in either free viewpoint or 3DTV style. The immersiveness provides a more natural way of communications. However, in such a low bit rate communication channel such as the wireless mobile network, owing to the insufficient bit budget, MVV compression will cause the heavy loss of visual detail information.

MVV contains a large amount of statistical dependencies since it is a collection of multiple videos simultaneously captured in a scene at different camera locations. Therefore, efficient compression techniques are required for the above consumer electronic applications [2].

Fractal compression, which has the advantages of high compression ratio, resolution independence, and fast decoding speed, is considered as one of the most promising compression methods. The basic idea of fractal image coding is to find a contractive mapping whose unique attractor approximates the original image. For the decoding, an arbitrary image with the same size of the original image is input into the decoder, and, after several times of iteratively applying the recorded contractive mappings to the input image, the reconstructed image will be obtained. Much effort [3–5] has been made to the fractal still image compression after Jacquin's fractal block coding algorithm [6]. However, a little work has been reported on the fractal video compression, let alone the fractal multiview video compression [7]. For fractal video compression, there are two extensions of still image compression, which are cube-based compression [8] and frame-based compression [9]. In the former method, video sequences are partitioned into nonoverlapping 3D range blocks and overlapping 3D domain blocks with larger size than range blocks. Then the key issue turns to find the best matching domain cuboid and the corresponding contractive mapping for every range cuboid, which is very complicated. In the latter method, each frame is encoded using the previous frame as a domain pool except the first frame which is encoded using a still image fractal scheme or some other methods. The main advantage of the frame-based algorithm is that decoding a frame consists of just one application of mapping so that iteration is not required at the decoder. However, the temporal correlation between the frames may not be effectively exploited, since the size of the domain block is larger than that of the range block [10].

In this paper, a novel highly efficient fractal multiview video codec by combined temporal/interview prediction is proposed. The anchor viewpoint video is encoded by improved frame-based fractal video compression approach, which combines the fractal coder with the well-known motion compensation (MC) technique. The other viewpoint videos are not only predicted from temporally neighboring images but also from the corresponding images in adjacent views.

This paper is organized as follows. The fractal compression theory and mathematical background is summarized in Section 2. The anchor viewpoint video compression algorithm is presented in Section 3 and then the proposed high efficiency fractal multiview video codec is presented in Section 4. The experimental results are shown in Section 5. Finally the conclusions are outlined in Section 6.

2. The Fractal Compression Theory and Mathematical Background

2.1. Mathematical Background. The mathematical background of fractal coding technique is the contraction mapping theorem and the collage theorem [11].

For the complete metric space (X, d), where X is a set and d is a metric on X, the mapping, $\omega : X \rightarrow X$ is said to be contractive if and only if

$$d(\omega(x), \omega(y)) \leq \alpha \cdot d(x, y), \quad 0 \leq \alpha < 1, \ \forall x, y \in X, \tag{1}$$

where α is called the contractivity factor of the contractive mapping.

For a contractive mapping $\omega : X \rightarrow X$ on (X, d); then there exists a unique point, $x_f \in X$, such that, for any point $x \in X$,

$$x_f = \omega(x_f) = \lim_{n \to \infty} \omega^n(x). \tag{2}$$

Such a point x_f is called a fixed point or the attractor of the mapping $\omega : X \rightarrow X$, where $\omega^n(x)$ represents the nth iteration application of ω to x. This is the famous Banach's fixed point theorem or contractive mapping theorem [11].

For the fractal image coding, if the encoder can find a contractive mapping whose attractor is the original image, then we only need to store the mapping with less bits instead of the original pixel values. But in the practical implementation, it is impossible to find a contractive mapping whose attractor is exactly the original image. Instead, the fractal encoder attempts to find the contractive mapping whose collage is close to the original image.

The collage theorem is as follows.

For the complete metric space (X, d), α is the contractivity factor of contractive mapping $\omega : X \rightarrow X$; then the fixed point x_f of the contractive mapping ω satisfies

$$d(x, x_f) \leq \frac{1}{1-\alpha} d(x, w(x)). \tag{3}$$

This means that the decoded attractor x_f is close to the original image x, if the collage $w(x)$ is close to the original image x. Therefore it converts to the minimization problem of the collage error.

2.2. Fractal Image Coding. In the practical implementation of the fractal image encoding process, the original image is firstly partitioned into nonoverlapping range blocks, covering the whole image, and overlapping domain blocks, usually twice the size of the range blocks in both width and height. For each range block the goal is to find a domain block and a contractive mapping that jointly minimize a dissimilarity (distortion) criterion. Usually the RMS (root mean square) metric is used. The contractive mapping applied to the domain block classically consists of the following parts [12]:

(i) geometrical contraction (usually by downsampling the domain block to the same size of the range block),

(ii) affine transformation (modeled by the 8 isometric transformations which contain the identity, rotation by 90°, 180°, and 270° and reflection about the midhorizontal axis, the midvertical axis, the first diagonal, and the second diagonal),

(iii) gray value transformation (a least square optimization is performed in order to compute the best values for the parameters s and o which are scaling factor and offset factor, resp.).

Here, s and o can be computed by minimizing the following equation:

$$E(s, o) = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(s \cdot d_{ij} + o - r_{ij}\right)^2, \tag{4}$$

where $\{d_{ij}\ (i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m)\}$ is the pixel value of the domain block after geometrical contraction and affine transformation and $\{r_{ij}\ (i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m)\}$ is the pixel value of the range block.

Then s and o can be obtained by making $\partial E(s, o)/\partial s$ and $\partial E(s, o)/\partial s$ equal to zero. So that

$$s = \frac{nm \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} r_{ij} - \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij}}{nm \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}^2 - \left(\sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}\right)^2},$$
$$o = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} r_{ij} - s \cdot \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij}}{nm}. \tag{5}$$

The fractal encoding process can be finished by storing all the data necessary for each range block including the location of the corresponding domain block, the index of the applied isometric transformation, and the values s and o. The decoding process is to iteratively apply the stored transformations to an arbitrary initial image.

3. The Anchor Viewpoint Video Compression

The most well-known fractal video codec is a hybrid fractal coder of circular prediction mapping (CPM) and noncontractive interframe mapping (NCIM) [13], in which the first four frames are encoded by CPM and the remaining frames are encoded by NCIM. In both the CPM and the NCIM, each range block is motion compensated by a domain block in the adjacent frame, which is of the same size as the range block. The main difference between the CPM and the NCIM is that the CPM should be contractive and the decoding process needs iteration, while the NCIM need not be contractive. The simulation results show better performance for the NCIM-coded frames than that for the CPM-coded frames.

Different from the abovementioned approach, in our proposed scheme, we first partition the video sequences into groups of frames (GOFs) to avoid error propagation. Every first frame in each GOF is encoded by intraframe prediction without depending on the previous frames, and the remaining frames in each GOF are encoded by combining fractal with the motion compensation (CFMC) as shown in Figure 1. In CFMC, each range block is motion compensated by a domain block in the adjacent previously predicted frame rather than the previous source frame, which is of the same size as the range block.

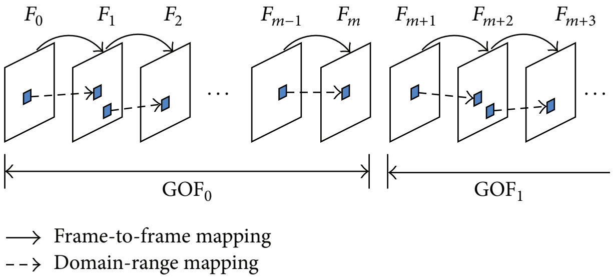


Figure 1: The structure of anchor viewpoint video coding.

3.1. Intraframe Prediction. The intraframe prediction in our scheme uses for reference the intraprediction modes of international video coding standard H.264/AVC [14]. We make some modifications to make it appropriate to the overall fractal video compression scheme. Firstly, we partition each frame into range blocks of maximum size 16 × 16 and minimum size 4 × 4, using the quadtree structure [15], while H.264 only has two kinds of sizes of blocks which are 4×4 and 16×16, respectively, and does not use the quadtree structure. Secondly, H.264 specifies 9 intraprediction modes for 4 × 4 luma blocks and 4 intraprediction modes for 16 × 16 luma blocks as shown in Figures 2 and 3, respectively, while we use the 9 modes in Figure 2 for range blocks in the bottom level of the quadtree and the 4 modes in Figure 3 for range blocks in other levels of the quadtree. Thirdly, H.264 chooses the best mode using Lagrangian rate distortion optimization (RDO) technique [16] by traversing all the possible block sizes and prediction modes which is very time consuming, while for simplification, in our scheme, subdivision of the range block is performed only when the prediction error is still above a prescribed threshold and the minimum allowable partition is not reached.

A prediction frame is generated and subtracted from the original frame to form an error-frame, which is further transformed, quantized, and entropy encoded. In parallel, the quantized data are rescaled and inverse transformed and added to the prediction frame to reconstruct a coded version of the frame which is stored for later predictions. In the decoder, the bitstreams are entropy decoded, rescaled, and inverse transformed to form a decoded error-frame. The decoder generates the same prediction that is created at the encoder and adds this to the decoded error-frame to produce a reconstructed frame.

3.2. Combining Fractal with Motion Compensation. Traditional motion compensation technique making use of block matching is based on the assumptions of locally translational motion and conservation of image intensity, whereas we may relax these assumptions when applying motion compensation technique used in our fractal video compression. The assumption of translation is replaced by an affine motion and the assumption of conservation of image intensity is replaced by the concept of linearly changing image intensity.

We investigate and compare the following four schemes:

S0: traditional (translational motion and conservation of image intensity) block matching,

S1: translational motion with gray value transformation,

S2: affine block matching using 8 isometric transformations,

S3: affine block matching using 8 isometric transformations with gray value transformation.

Exhaustive search in a search window which is formed by extending the collocated domain block by ±7 pixels in four directions is used for a range block as shown in Figure 4, where F_m represents the current frame to be encoded and F'_{m-1} represents the adjacent coded version.

Figure 5(a) compares the PSNR performance of the different schemes on the “Ballroom” video. Clearly the gray value transformation leads to significant PSNR gain, whereas the gain for affine transformation is only moderate. Figure 5(b) compares the encoding time of the different schemes. Obviously, the schemes without affine transformation (S0 and S1) are much faster compared to schemes using the affine motion model (S2 and S3), while gray value transformation is compu-tationally not very expensive. Test results on other video sequences such as “Race” and “Flamenco” show the same trends.

Considering the tradeoff between computational complexity and quality gain, we finally adopt S1 in our CFMC

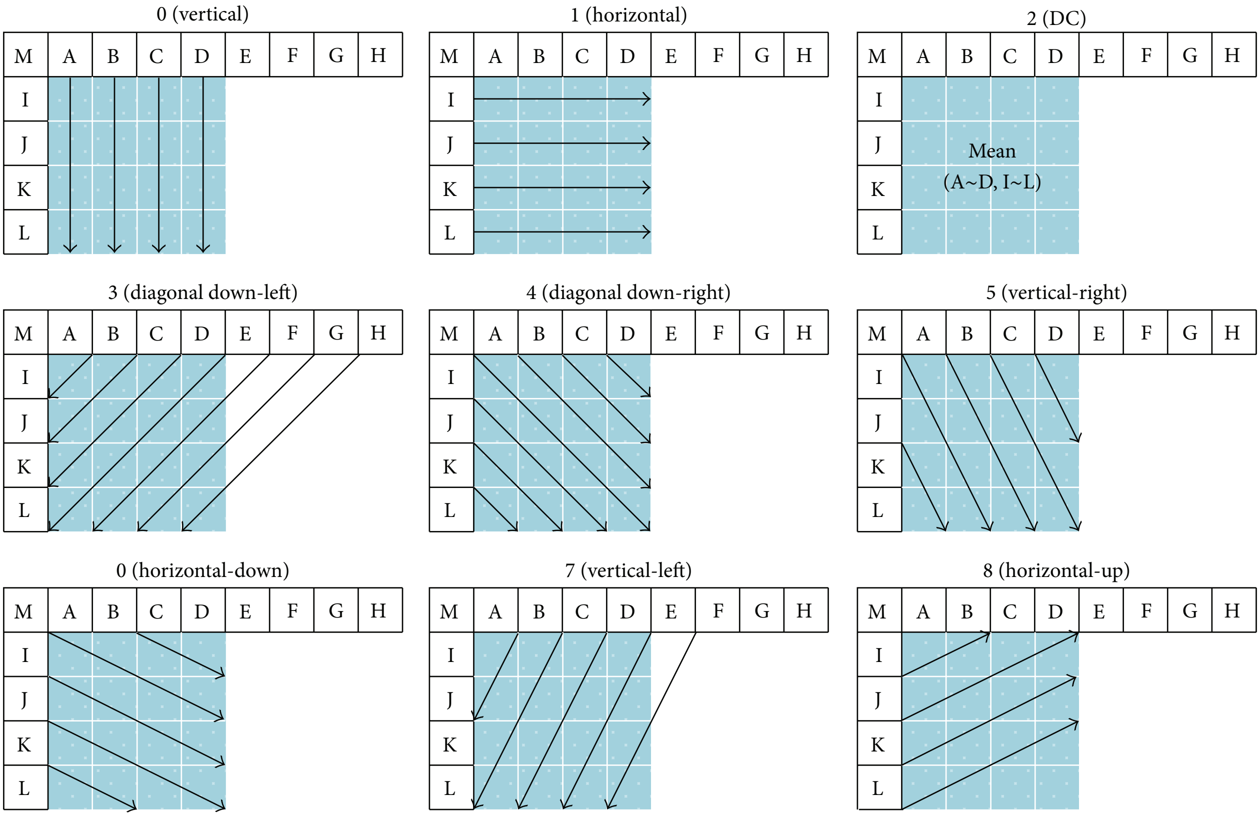


Figure 2: Nine intraprediction modes.

algorithm. Namely, range blocks in the current frame are predicted by domain blocks of the reference-frame using translational motion with gray value transformation.

To achieve better quality, the prediction frame is subtracted from the original frame to form an error-frame, which is transformed, quantized, and entropy encoded. In parallel, the quantized data are rescaled and inverse transformed and added to the prediction frame to reconstruct a coded version of the frame which is stored for later predictions, and the CFMC data including the positions of corresponding domain blocks, scale factors, and offset factors are entropy encoded. In the decoder, the bitstreams of error-frame are entropy decoded, rescaled, and inverse transformed to form a decoded error-frame. The bitstreams of CFMC data are entropy decoded and generate the prediction frame by applying corresponding translational motion with gray value transformation only once without iteration. Then a reconstructed frame is produced by adding the prediction frame to the decoded error-frame. This contrasts to 2D fractal video coding schemes which do not consider the coding of error-frames at all.

4. Fractal Multiview Video Compression Scheme

The coding structure between different views needs to be considered when extending single view fractal video coding to multiview video coding. Therefore, we propose the temporal-spatial prediction structure and fast disparity estimation algorithm.

4.1. Temporal-Spatial Prediction Structure. Multiview video sequences are captured by several cameras at the same time and there exists a high degree of correlation between interviews and intraviews. So we propose a temporal-spatial prediction structure based on view center and view distance. When processing the multiview video signal, disparity compensation and CFMC are combined to reduce the number of intraframe coded frames and improve the view compensation efficiency.

The proposed new prediction structure is shown in Figure 6 which contains 3 views. The center viewpoint video is the anchor video coded with the intraframe prediction and CFMC algorithm in Section 3 and the other viewpoint videos are coded based on disparity compensation and CFMC.

Figure 7 is the geometric schematic diagram of temporal-spatial prediction. Disparity estimation and CFMC are used for prediction to reduce data redundancy adequately.

4.2. Fast Disparity Estimation Algorithm. Geometric constraints between neighboring frames in multiview video sequences can be used to eliminate spatial redundancies. Parallax distribution constraints, which contain epipolar constraint, directional constraint, and spatial-time domain correlation, are used in the proposed fast disparity estimation algorithm.

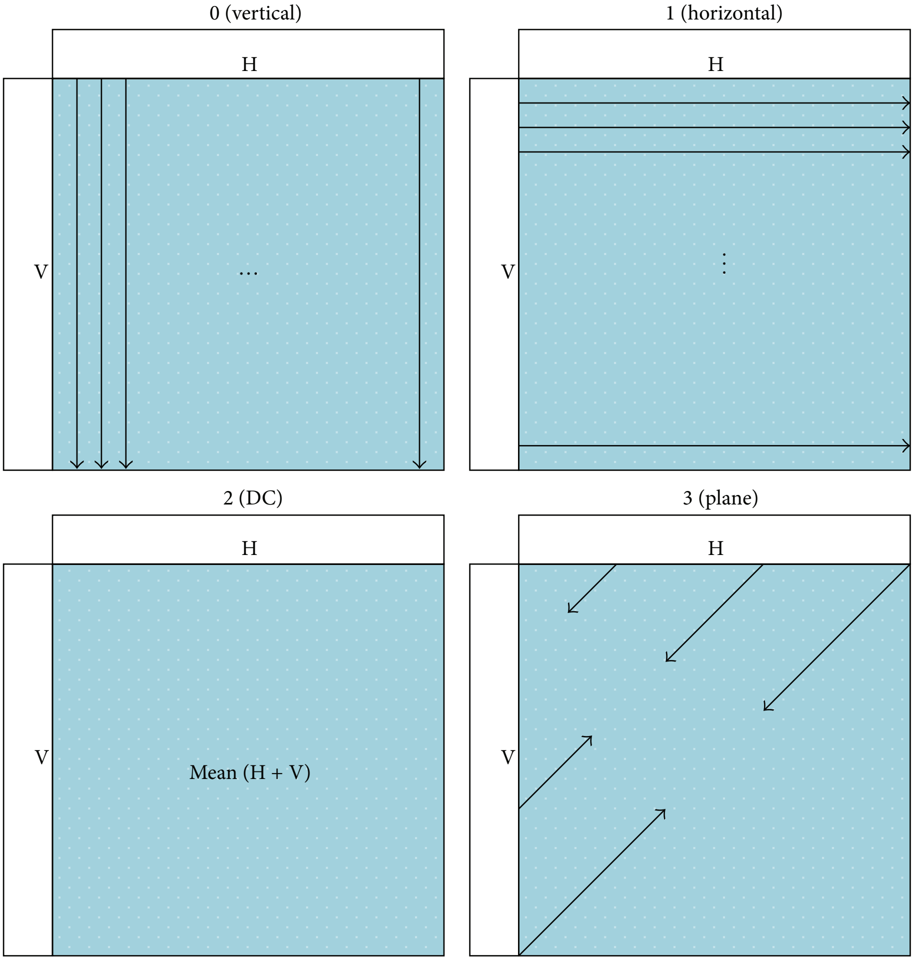


FIGURE 3: Four intraprediction modes.

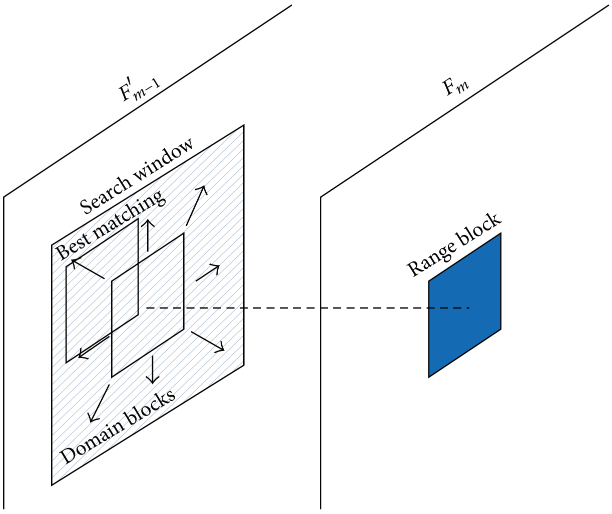


FIGURE 4: Search window for range block.

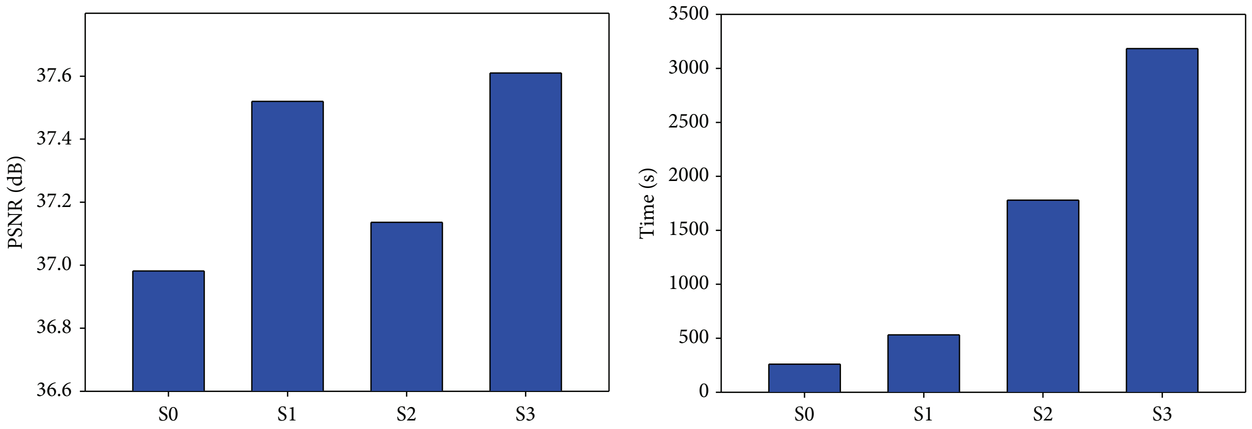
Epipolar constraint: the epipolar geometry is the intrinsic projective geometry between two views. Epipolar line can be found in the right image for a point of left image, on which the corresponding point can be searched. For parallel systems, only the search of x direction is needed just along the scan line. In other words, traditional two-dimensional search can be simplified to one-dimensional search along the scan line.

Directional constraint: in parallel systems, the vertical component of disparity vector is always equal to zero. The horizontal component is determined by the following formula:

$$d_x(x, y, z) = \frac{FB}{z}, \tag{6}$$

where $d_x(x, y, z)$ represents the horizontal component of disparity vector corresponding to the point (x, y, z). F represents the camera's focal length. B represents the baseline distance between the two cameras. We can see from formula (6) that the horizontal component of disparity vector is always positive; that is, for the same scene, the left perspective projection image always moves slightly left relative to the right image. Therefore, searching only in one direction instead of two directions is enough.

The spatial-time domain correlation: the disparity vector in the same frame has a strong correlation. For two adjacent frames, only a few pixels move and the positions of most



(a) PSNR comparisons

(b) Encoding time comparisons

FIGURE 5: Performance comparisons of the four schemes.

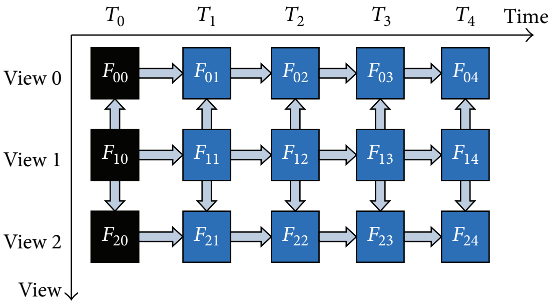


FIGURE 6: The proposed temporal-spatial prediction structure.

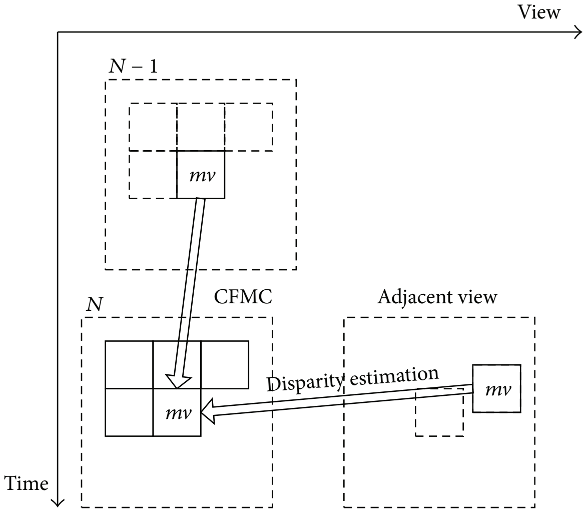


FIGURE 7: Disparity estimation and CFMC.

pixels have not changed. The disparities of these pixels whose positions have not changed keep basically unchanged. Therefore, corresponding disparity vectors of the neighboring range blocks can be considered as the starting point for a small area search to find the actual disparity vector quickly.

From the three constraints above, we can summarize our fast disparity estimation algorithm as follows. Firstly, search between the disparity vectors of the previously encoded neighboring range blocks to find one with the minimum RMS error as the starting search point. Then search in the horizontal to the right direction within several pixels distance to find the final disparity vector with the minimum RMS error.

Fast disparity estimation algorithm is helpful to reduce the number of search candidates and further to decrease the disparity estimation time. It can greatly improve the coding efficiency, which makes full use of the correlation between left and right views and can find the best matching block more quickly.

4.3. Fractal Multiview Video Compression Process. The overall coding process of the proposed fractal multiview video codec is shown in Figure 8.

Step 1. Partition the video sequences into GOFs.

Step 2. Encode the first frame of GOF in the anchor view using intraframe prediction and process the error-frame.

Step 3. Encode the first frame of GOF in the nonanchor views using fast disparity estimation from collocated frame in the adjacent view and process the error-frame.

Step 4. Encode the remaining frames of GOF in the anchor view using CFMC from the previous decoded frame and process the error-frames.

Step 5. Encode the remaining frames of GOF in the nonanchor views using CFMC from the previous decoded frame and then encode them using fast disparity estimation from collocated frames in the adjacent view. Choose the one with the minimum RMS and process the error-frames.

Step 6. Encode the next GOF using Step 2 to Step 5 until all the frames have been finished.

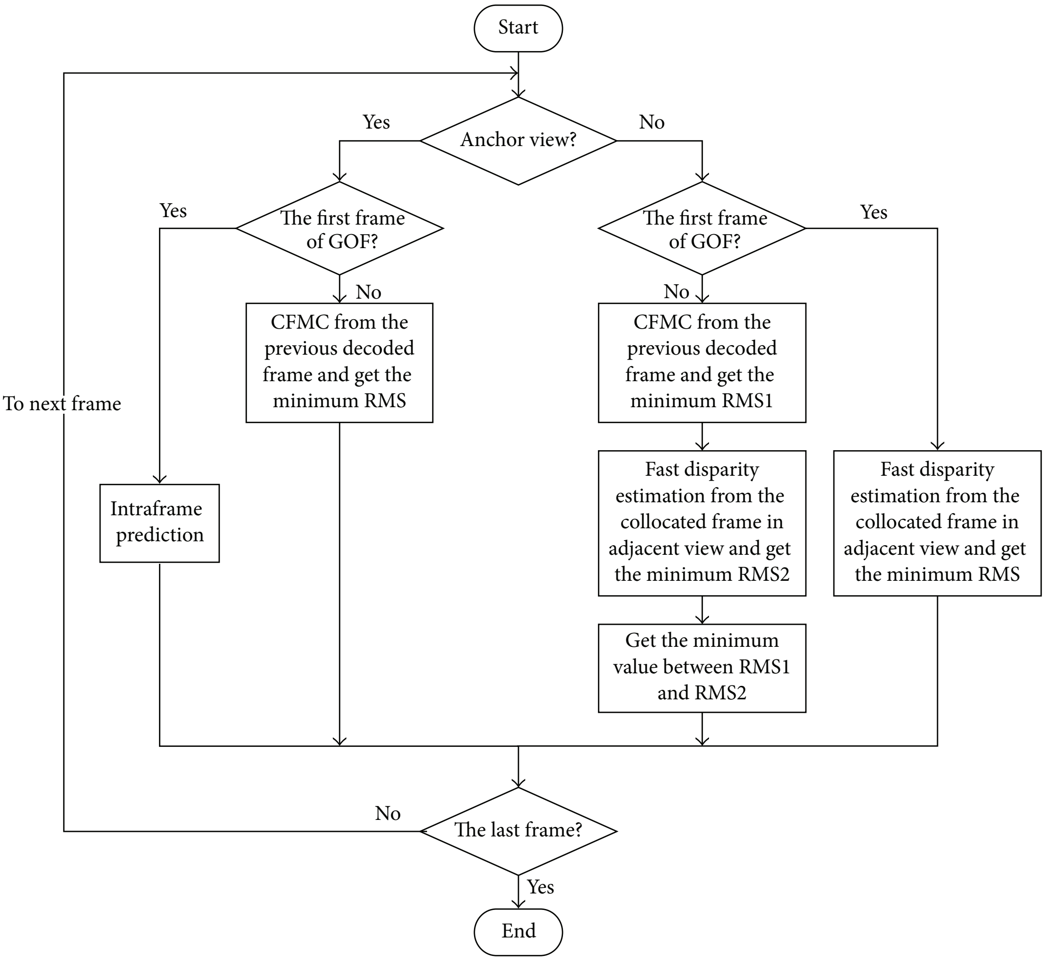

Start
Anchor view?
Yes
No
The first frame of GOF?
Yes
No
The first frame of GOF?
No
Yes
CFMC from the previous decoded frame and get the minimum RMS
CFMC from the previous decoded frame and get the minimum RMS1
To next frame
Intraframe prediction
Fast disparity estimation from the collocated frame in adjacent view and get the minimum RMS2
Fast disparity estimation from the collocated frame in adjacent view and get the minimum RMS
Get the minimum value between RMS1 and RMS2
No
The last frame?
Yes
End

Figure 8: The encoding process block diagram of the proposed system.

5. Experimental Results

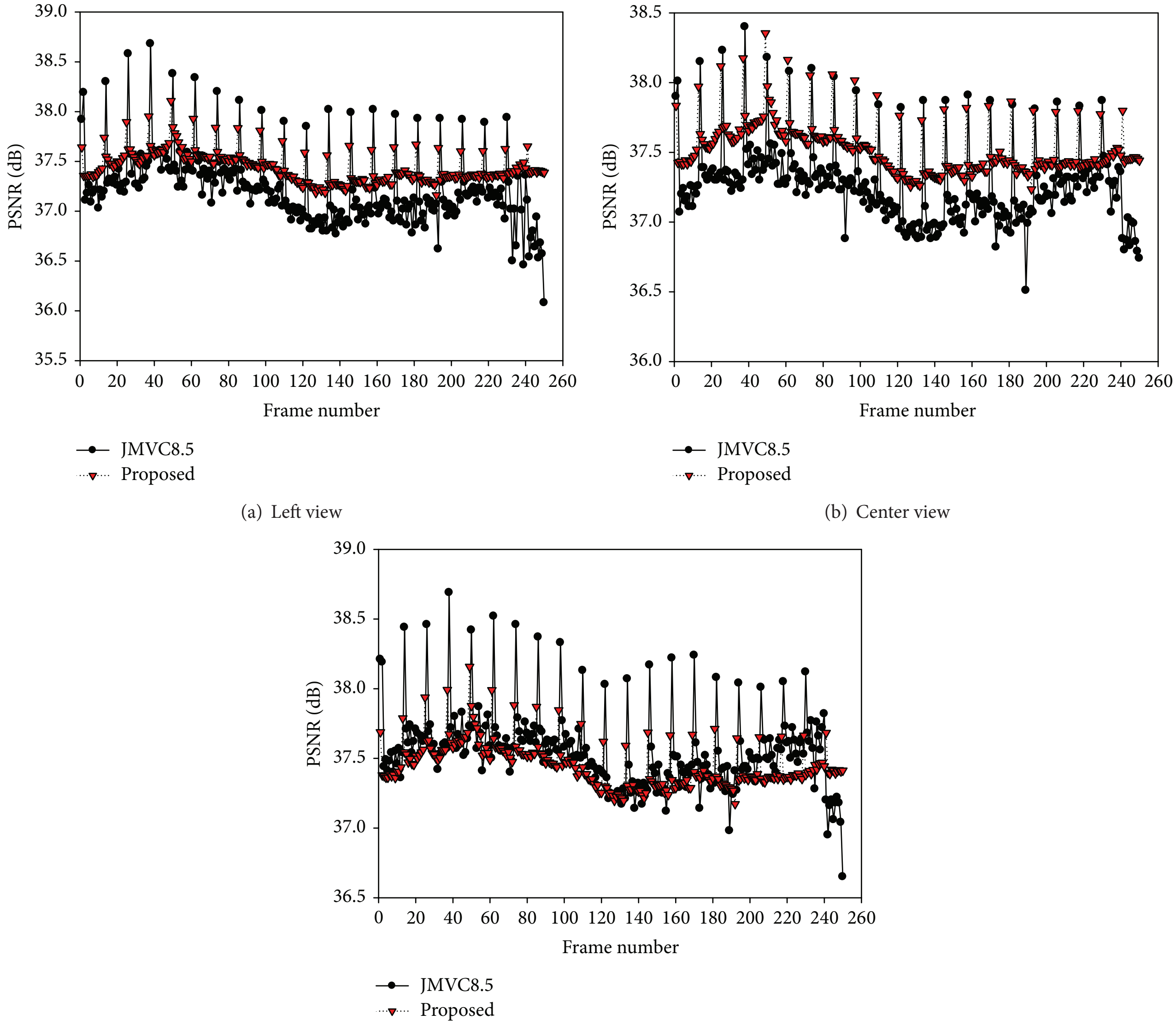
To verify the performance of the proposed fractal multiview video codec, three multiview video sequences "Ballroom" (250 frames), "Race" (300 frames), and "Flamenco" (300 frames) are tested. The quadtree structure we used contains three levels and the sizes of range blocks are 16 × 16, 8 × 8, and 4 × 4, respectively. Frame filtering is also applied. The simulations are run on a PC with Intel Core i7 quad 3.4 GHz CPU and 8 GB DDR RAM. The proposed method is implemented and compared with the state-of-the-art multiview video coding (MVC) reference software JMVC8.5 [20]. The configuration settings of the JMVC8.5 simulation environment are shown in Table 1.

Table 2 shows the PSNR, bit number, and encoding time comparisons between the proposed codec and JMVC8.5 using the simulation conditions given above. The numerical results are obtained by computing the average value of all the encoded frames. Overall, average time savings of about 95.71%, bit number savings of about 36.21%, and PSNR gains of 0.36 dB can be observed. Apparently, there is a great improvement on encoding time, bitrate, and PSNR compared to the JMVC8.5.

Table 1: Configuration settings of the JMVC8.5 simulation environment.

NumViewsMinusOne	2
ViewOrder	0-2-1
NumberReferenceFrames	2
GOP size	12
IntraPeriod	12
SearchMode	0: BlockSearch
Frames to be encoded	all frames
QP of all frames	28
SearchRange	±7 pixels
Motion estimation	Full search
SymbolMode	CAVLC

Figures 9(a)–9(c) show the PSNR comparisons of the "Ballroom" left, center, and right view, respectively, between the proposed fractal multiview video codec and JMVC8.5. Figures 10(a)–10(c) show the coding bit number comparisons of the "Ballroom" left, center, and right view, respectively.



(a) Left view

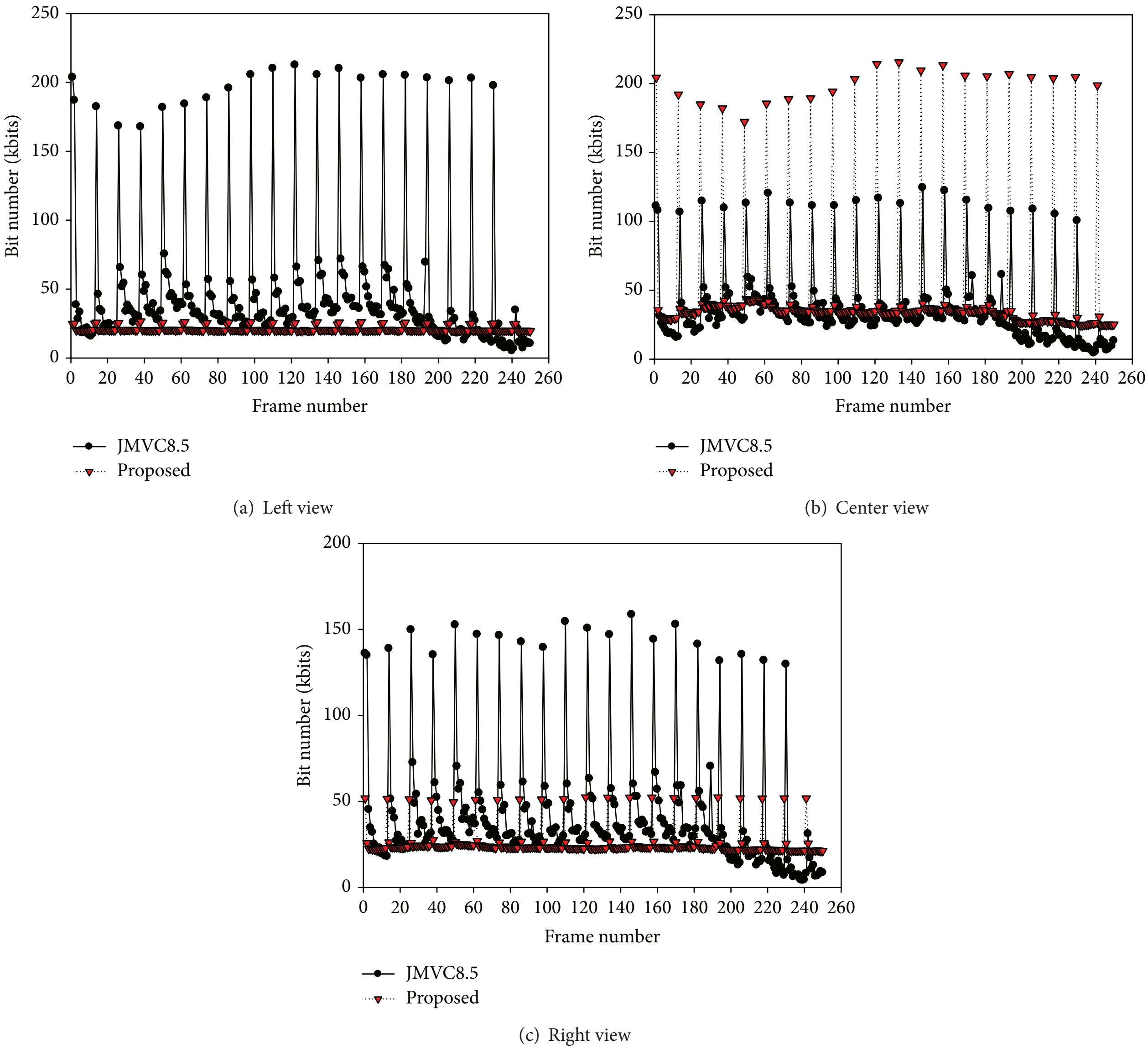
(b) Center view

(c) Right view

Figure 9: PSNR comparisons of "Ballroom."

Table 2: Performance comparisons between the proposed codec with JMVC8.5.

Video sequences		PSNR (dB)		Bit number (bits/frame)		Time (ms/frame)	
		JMVC8.5	Proposed	JMVC8.5	Proposed	JMVC8.5	Proposed
Ballroom	Left	37.22	**37.44**	46275	**20924**	22976	**632**
	Center	37.25	**37.52**	35635	**47503**	36595	**2127**
	Right	37.55	**37.45**	41230	**25627**	23384	**632**
Race	Left	37.35	**37.98**	91580	**22545**	21055	**661**
	Center	37.63	**38.12**	57711	**85537**	35543	**2226**
	Right	37.46	**37.99**	89584	**31300**	21438	**661**
Flamenco	Left	39.54	**40.06**	59219	**21883**	20665	**606**
	Center	40.31	**40.25**	39659	**51859**	34552	**2048**
	Right	39.34	**40.07**	63750	**27508**	21949	**610**
Average		38.18	**38.54**	58294	**37187**	26462	**1134**



(a) Left view

(b) Center view

(c) Right view

Figure 10: Bit number comparisons of "Ballroom."

Figures 11(a)–11(c) show the encoding time comparisons of the "Ballroom" left, center, and right view.

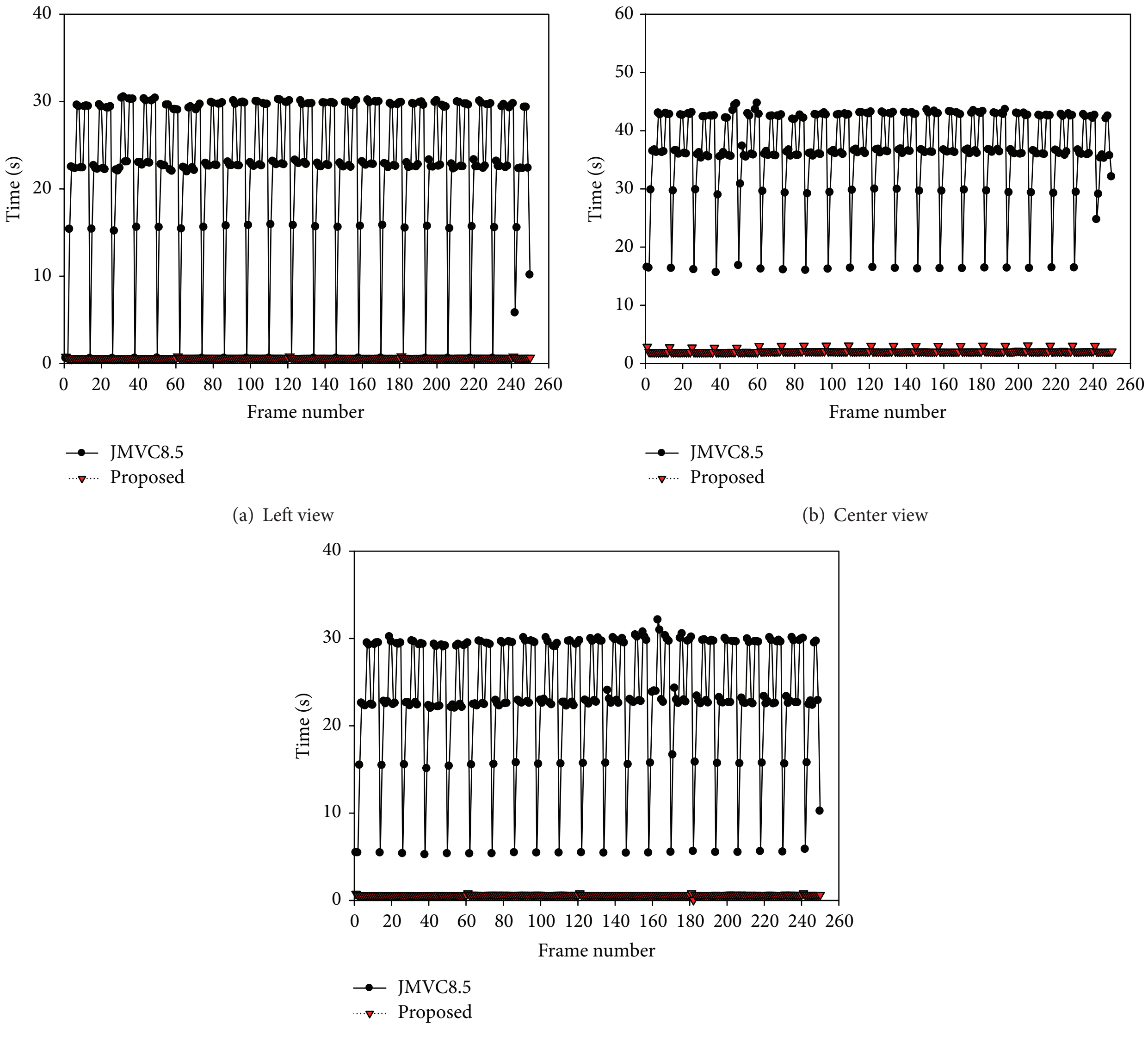
From Figures 9–11, we can conclude that the proposed fractal multiview video codec reduces the encoding time and improves coding efficiency greatly, which leads to more real-time applications. Figure 12 shows the 18th frame original and decoded images of "Race" resulted from JMVC8.5 and the proposed method.

To verify the good performance of the proposed method, Figure 13 illustrates the rate-distortion comparisons of "Flamenco" among the proposed method, color correction preprocessing method [17], histogram-matching method [18], illumination compensation method [19], no correction method [17], fractal method proposed in [7], and JMVC8.5 [20]. In order to facilitate the comparison, the average performance comparisons of "Flamenco" between the proposed codec and the other schemes computed with the Bjontegaard metric [21] are shown in Table 3.

From Figure 13 and Table 3, we can see that, with the bitrate increasing, the proposed fractal multiview video codec performs better, and the overall encoding performance of the proposed codec is superior to that of algorithms in the references.

6. Conclusion

In this paper, a novel high efficiency fractal multiview video codec is presented to improve the encoding performance.



(a) Left view

(b) Center view

(c) Right view

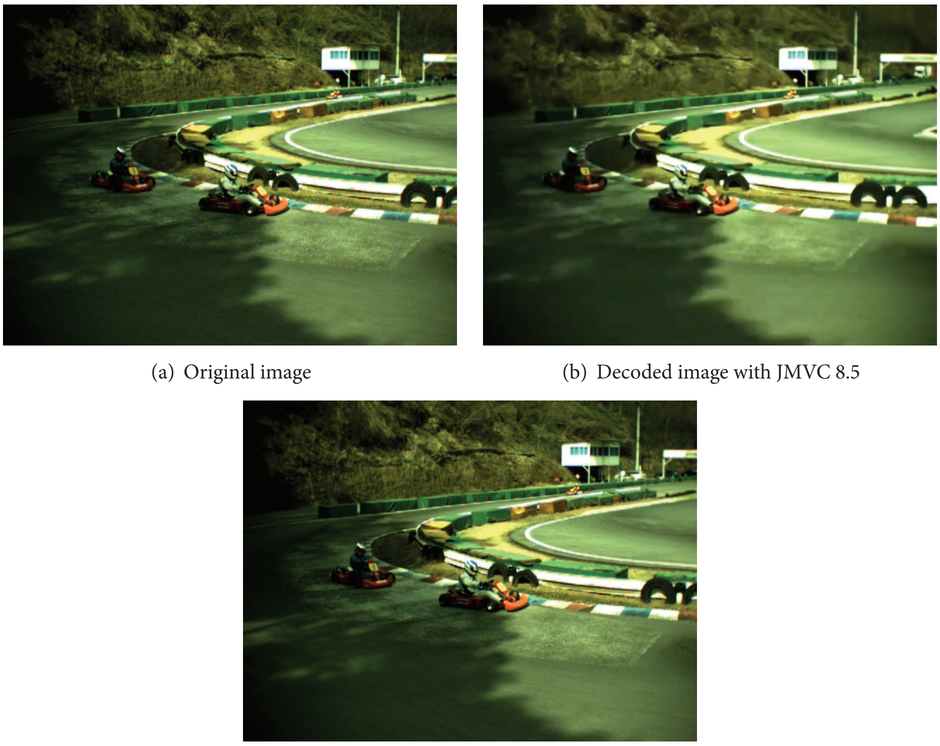
Figure 11: Encoding time comparisons of "Ballroom."

Table 3: Average performance comparisons of "Flamenco" between the proposed codec and the other schemes computed with the Bjontegaard metric.

Schemes	BDPSNR (dB)	BDBR (%)
Color correction preprocessing [17]	1.06	−7.8
Histogram-matching [18]	1.29	−4.3
Illumination compensation [19]	1.43	−5.7
No correction method [17]	1.64	−7.5
Fractal [7]	0.06	−0.4
JMVC8.5	2.35	−17.8

The video sequences are firstly partitioned into GOFs to avoid error propagation. Then, we improve the intraframe prediction to make it suitable for fractal encoding and propose the CFMC algorithm to get better performance of the anchor view. In addition, temporal-spatial prediction structure and fast disparity estimation algorithm are applied to further raise the compression efficiency.

Experimental results show that the proposed fractal multiview video codec spends less encoding time and achieves higher compression ratio with better decoding image quality. An average encoding time saving of about 95.71%, a bitrate saving of about 36.21%, and a PSNR gain of 0.36 dB can be achieved compared with JMVC8.5. In addition, it also shows superiority compared with color correction preprocessing method, histogram-matching method, illumination



(a) Original image

(b) Decoded image with JMVC 8.5

(c) Decoded image with the proposed method

FIGURE 12: The decoded image results of 18th frame of "Race."

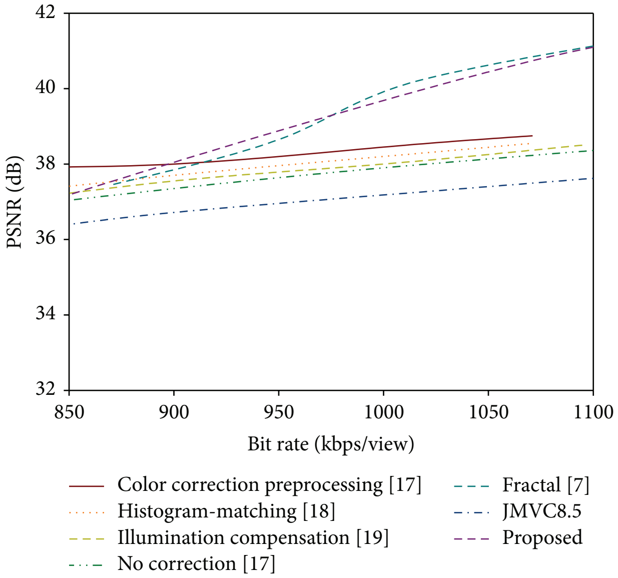


FIGURE 13: Rate-distortion comparisons of "Flamenco."

compensation method, and no correction method. It has built a good foundation for the further research of multiview fractal video coding and other related coding methods.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This project is funded by the National Natural Science Foundation of China (NSFC) under Grants nos. 61375025, 61075011, and 60675018 and also the Scientific Research Foundation for the Returned Overseas Chinese Scholars from the State Education Ministry of China. The authors would also like to express their appreciations to the reviewers for their thorough review and very helpful comments, which help improve this paper.

References

[1] H. M. Ozaktas and L. Onural, *Three-Dimensional Television: Capture, Transmission, Display*, Springer, New York, NY, USA, 2007.

[2] P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1461–1473, 2007.

[3] T. K. Truong, C. M. Kung, J. H. Jeng, and M. L. Hsieh, "Fast fractal image compression using spatial correlation," *Chaos, Solitons and Fractals*, vol. 22, no. 5, pp. 1071–1076, 2004.

[4] C.-C. Wang and C.-H. Hsieh, “An efficient fractal image-coding method using interblock correlation search,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 2, pp. 257–261, 2001.

[5] Y. Iano, F. S. da Silva, and A. L. M. Cruz, “A fast and efficient hybrid fractal-wavelet image coder,” *IEEE Transactions on Image Processing*, vol. 15, no. 1, pp. 98–105, 2006.

[6] A. E. Jacquin, “Image coding based on a fractal theory of iterated contractive image transformations,” *IEEE Transactions of Image Processing*, vol. 1, no. 1, pp. 18–30, 1992.

[7] S. Zhu, L. Li, and Z. Wang, “A novel fractal monocular and stereo video codec with object-based functionality,” *EURASIP Journal on Advances in Signal Processing*, vol. 2012, no. 1, article 227, 14 pages, 2012.

[8] M. S. Lazar and L. T. Bruton, “Fractal block coding of digital video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 3, pp. 297–308, 1994.

[9] Y. Fisher, D. Rogovin, and T. P. Shen, “Fractal (self-VQ) encoding of video sequences,” in *Visual Communications and Image Processing (VCIP ’94)*, vol. 2308 of *Proceedings of SPIE*, pp. 1359–1370, Chicago, Ill, USA, September 1994.

[10] C.-S. Kim and S.-U. Lee, “Fractal coding of video sequence by circular prediction mapping,” *Fractals*, vol. 5, no. 1, pp. 75–88, 1997.

[11] M. Barnsley, *Fractals Everywhere*, Academic Press, New York, NY, USA, 1988.

[12] A. Pommer, C. Hufnagl, and A. Uhl, “Fractal motion compensation,” in *Proceedings of the Visual Communications and Image Processing*, pp. 1050–1061, San Jose, Calif, USA, January 1998.

[13] C.-S. Kim, R.-C. Kim, and S.-U. Lee, “Fractal coding of video sequence using circular prediction mapping and noncontractive interframe mapping,” *IEEE Transactions on Image Processing*, vol. 7, no. 4, pp. 601–605, 1998.

[14] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[15] Y. Fisher, *Fractal Image Compression-Theory and Application*, Springer, New York, NY, USA, 1995.

[16] G. J. Sullivan and T. Wiegand, “Rate-distortion optimization for video compression,” *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[17] C. Doutre and P. Nasiopoulos, “Color correction preprocessing for multiview video coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 9, pp. 1400–1405, 2009.

[18] U. Fecker, M. Barkowsky, and A. Kaup, “Histogram-based prefiltering for luminance and chrominance compensation of multiview video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1258–1267, 2008.

[19] A. Vetro, P. Pandit, H. Kimata, A. Smolic, and Y. K. Wang, “Joint Multiview Video Model (JMVM) 8.0,” ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16, Doc. JVT-AA207, April 2008.

[20] Joint Video Team of ITU-T VCEG and ISOIEC MPEG, “WD 1 Reference software for MVC (JMVC8.5),” March 2011.

[21] G. Bjontegaard, “Calculation of average PSNR differences between RD curves,” in *Proceedings of the 13th VCEG-M33 Meeting*, Austin, Tex, USA, April 2001.

Advances in
Operations Research
Advances in
Decision Sciences
Journal of
Applied Mathematics
Algebra
Journal of
Probability and Statistics
The Scientific
World Journal
International Journal of
Differential Equations
International Journal of
Combinatorics
Advances in
Mathematical Physics
Journal of
Complex Analysis
Journal of
Mathematics
Mathematical Problems
in Engineering
Abstract and
Applied Analysis
Discrete Dynamics in
Nature and Society
International
Journal of
Mathematics and
Mathematical
Sciences
Journal of
Discrete Mathematics
Journal of
Function Spaces
International Journal of
Stochastic Analysis
Journal of
Optimization