Optical Engineering

SPIEDigitalLibrary.org/oe

Fast prediction algorithm for multiview video coding

Abdelrahman Abdelazim
Stephen James Mein
Martin Roy Varley
Djamel Ait-Boudaoud

SPIE

Fast prediction algorithm for multiview video coding

Abdelrahman Abdelazim
American University of the Middle East
Block 3, Building 1, P.O. Box 220, Egaila Area
Dasman 15453, Kuwait
E-mail: abdelrahman.abdelazim@aum.edu.kw

Stephen James Mein
Martin Roy Varley
University of Central Lancashire
School of Computing, Engineering and Physical Sciences
Preston PR1 2HE, United Kingdom

Djamel Ait-Boudaoud
University of Portsmouth
Portland Building, Portland Street
Portsmouth PO1 3AH, United Kingdom

Abstract. The H.264/multiview video coding (MVC) standard has been developed to enable efficient coding for three-dimensional and multiple viewpoint video sequences. The inter-view statistical dependencies are utilized and an inter-view prediction is employed to provide more efficient coding; however, this increases the overall encoding complexity. Motion homogeneity is exploited here to selectively enable inter-view prediction, and to reduce complexity in the motion estimation (ME) and the mode selection processes. This has been accomplished by defining situations that relate macro-blocks' motion characteristics to the mode selection and the inter-view prediction processes. When comparing the proposed algorithm to the H.264/MVC reference software and other recent work, the experimental results demonstrate a significant reduction in ME time while maintaining similar rate-distortion performance. [DOI: 10.1117/1.OE.52.3.037401]

Subject terms: multiview video coding; motion estimation; video compression; mode decision.

Paper 121610 received Nov. 4, 2012; revised manuscript received Feb. 4, 2013; accepted for publication Feb. 12, 2013; published online Mar. 4, 2013.

1 Introduction

Recent advances in digital technologies have paved the way for the development of numerous real-time applications deemed too complex in the past. A vast array of those applications requires transmission and storage of digital videos. Examples include, but are not limited to, digital TV, video streaming, multimedia communications, remote monitoring, videophones and video conferencing. Advances in digital video can be classified as one of the most influential modern technologies, due in part to the fast, widespread use of digital video applications in everyday life. Consequently, over the last three decades, high-quality digital video has been the goal of companies, researchers and standardization bodies.[1] Multiview, in particular, is a highly relevant topic in the current research by institutions attempting to advance real-time multiview applications. As a result, and due to the wide expansion of three-dimensional (3-D) and free viewpoint video applications, the H.264 multiview video coding (MVC) standard has been developed as an extension to the H.264/advanced video coding (AVC) standard to enable efficient coding of scenes captured from multiple cameras.[2] Since all cameras capture the same scene from different viewpoints, inter-view statistical dependencies can be expected. Therefore, in addition to the H.264/AVC very refined motion estimation (ME) and motion compensation (MC) processes, H.264/MVC exploits inter-view prediction for more efficient coding. However, this further increases the overall encoding complexity and makes the standard complexity unbearable for real-time encoding. Therefore, there is a tremendous need to reduce encoding complexity and to design a flexible, rate-distortion-optimized, yet computationally efficient, encoder for various applications.

Recently proposed algorithms reduce encoder complexity by locating corresponding objects in neighboring views by means of a global disparity vector and exploiting the mode distribution correlation between neighboring views.[3–9] These algorithms can only perform well for certain video sequences and camera configurations, given that the inherent scene characteristics are not taken into account.

In this paper a more efficient approach has been taken to solve the MVC standard complexity problem by utilizing the high correlation between a macroblock (MB) and its enclosed partitions to estimate motion homogeneity. Based on this result inter-view prediction is selectively enabled or disabled. Moreover, if the MVC is divided into three layers in terms of motion prediction—the full and subpixel motion search, the mode selection process and then a repetition of the first and second for inter-view prediction—the proposed algorithm significantly reduces the complexity in the three layers. This is accomplished by extending the algorithm proposed in Ref. 10 and applying it to the inter-view prediction.

This paper is organized as follows: The MVC concepts and requirements are outlined in Sec. 2. Section 3 reviews some related work on reducing computational complexity in the MVC prediction process. The proposed algorithm is presented in Sec. 4. Experimental results are presented in Sec. 5. Finally, a conclusion is provided in Sec. 6.

2 H.264 Multiview Coding

MVC is an extension to the H.264/MPEG-4 AVC video compression standard developed with joint efforts by moving picture experts group (MPEG)/video coding experts group to enable efficient encoding of sequences captured simultaneously from multiple cameras using a single video stream. Therefore the design is aimed at exploiting inter-view dependencies in addition to reducing temporal redundancies.[2,11]

The stereo high profile of the MVC standard was standardized in June 2009. MVC streams are backward-compatible with H.264/AVC, which allows for older devices and software to decode stereoscopic video streams, ignoring additional information for the second view.[12]

2.1 *MVC Targeted Applications*

The MVC main targeted applications are:

- Free viewpoint television (FTV): a system allowing the user to control the viewpoint and add new views of a scene from any three-dimensional (3-D) position.
- Multiview 3-D television: a television designed to display 3-D materials. Those materials can be formed using any 3-D production techniques, such as stereoscopic capture, multiview capture, or two-dimensional (2-D) plus depth.
- Stereoscopic: two-view video.

Two main challenges face most multiview applications. The transmission of a huge amount of data, which requires the development of highly efficient coding schemes, and any compression scheme designed specifically for multiview video streams should support random-access functionality, allowing viewers to access arbitrary views with minimum time delay. Therefore, a set of requirements has been laid out for designing the MVC, as explained in the following section.

2.2 *Requirements*

Most of the requirements, as well as test data and evaluation conditions for the multiview coding standard, are defined by the MVC project,[13] a summary of those is expressed in the following points.

- Large gain compared to independent compression of each view.
- Temporal random access and view random access.
- View scalability, meaning any part of the bitstream can be accessed by the decoder to generate a low-quality video output.
- Parallel processing to reduce delays, its implementation allows for the encoding of multiple views simultaneously.
- Camera parameters (extrinsic and intrinsic) were required to be transmitted with the bitstream to support the main view interpolation.
- Backward compatibility with the AVC.
- Consistent quality among views.

2.3 *Temporal and Inter-View Correlation*

Several analyses[14,15] have been carried out to investigate temporal and inter-view correlation by measuring the statistical dependencies that can be exploited for prediction.

Figure 1 shows the eight possible first-order spatial and temporal neighbor pictures of a picture in a multiview video (MVV) sequence with linear camera arrangement, where V indicates the views and T the time-points (the temporal position).

If $F_{0,0}$ is considered to be the current frame, and the linear camera arrangement shown in Fig. 1 is considered; there are nine possible frames that could be used as reference frames for motion prediction purposes, as shown the figure. $F_{0,-1}$ and $F_{0,1}$, represent the preceding and the succeeding frames in the same view, respectively, and are normally considered as the reference frames in the H.264/AVC.

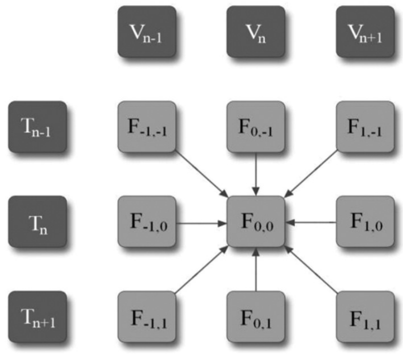


Fig. 1 Eight possible first-order spatial and temporal neighbor pictures.

In Refs. 14 and 15 the analysis results demonstrate that, frequently, for a significant number of MBs, inter-view prediction is more efficient than temporal prediction. However, for all video sequences temporal prediction is the mode chosen most often. This is due to illumination difference or imperfect calibration of cameras.[16] Inter-view prediction means that an MB in $F_{0,0}$ finds a best match in V_{n-1} or V_{n+1}, while temporal prediction means that MB in $F_{0,0}$ finds a best match in V_n.

A comparison between inter-view prediction and temporal prediction modes is shown in Table 1. The comparison reflects the percentage between the MBs that find their best match in the same views or in neighbors's views. In Table 1 standard multiview testing sequences were used.

Table 1 indicates that, on average 13.1% of the MBs of all sequences find a best match in other views. This can lead to a considerable bit-rate reduction, but at the expense of increasing the encoder complexity. The following section outlines some of the proposed prediction structures for MVC.

2.4 *Prediction Structures*

Since MVC is a direct extension of AVC, with the addition of inter-view prediction, the MVC prediction structure is based on the multiple reference picture technique in H.264/AVC. Therefore, in the design stage of the standard, different prediction structures have been proposed.[17–23] Those structures vary significantly in terms of the overall performance

Table 1 Results of temporal and inter-view correlation analysis.

Sequence Name	T [%]	V [%]
Ballroom	83.24	16.76
exit	86.42	13.58
Uli	95.65	4.35
Race1	98.26	1.74
Breakdancers	70.93	29.07
Average	**86.9**	**13.1**

and the encoder requirements for reference picture selection and memory management. The following subsections outline three of those variations that are enabled in the standard reference software:

2.4.1 *Temporal prediction using hierarchical B pictures*

The simplest way to encode a set of video streams from different cameras is to encode them separately using the H.264/AVC, as shown in Fig. 2, which shows two group of pictures (GOP) from two views, each containing eight frames. In the figure, the hierarchical B pictures[24] is employed as it is considered the most efficient temporal prediction structure.

The first picture of a video sequence is intra-coded as an instantaneous decoding refresh picture and so-called key pictures, referred to as the "I" picture in Fig. 2. Then at regular intervals, defined by the GOP size, frames are coded as "I" frames. As discussed, this method is simple, but inefficient for multiview videos.

2.4.2 *Inter-view prediction for key pictures*

A straightforward improvement of temporal prediction using hierarchical B pictures is to employ inter-view prediction for key pictures only, as shown in Fig. 3. In this approach, all "I" frames except those in the first view are encoded as P pictures. This can lead to a significant number of bits savings due to the fact that "I" frames require a greater number of bits than P frames.[24]

The downside of this method is that individual views can no longer be encoded or decoded independently, as they share reference pictures. Furthermore, the design of the encoder and the decoder becomes more complex, specifically for managing reference frames and data buffers.

2.4.3 *Inter-view prediction for key and non-key pictures*

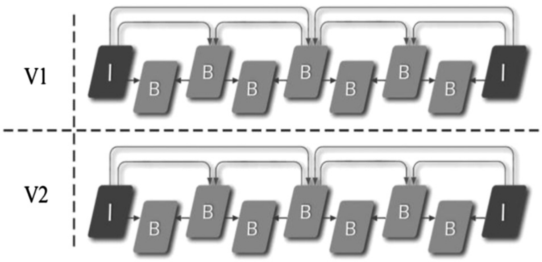
Another approach commonly used for MVC is the inter-view prediction for key and non-key pictures. Figure 4 shows that this method allows greater flexibly for motion prediction; however, it is at the expense of increasing the overall encoding complexity. Using this method, a coding gain of 1.7 dB can be achieved.[15]

The MVC standard reference software joint multiview video model (JMVM)[25] allows great flexibility in encoding multiview videos. JMVM uses hierarchical B pictures for each view and at the same time applies inter-view prediction to every second view, using previously encoded frames from adjacent camera views. This is accomplished by employing a number of user-configurable parameters in the software main configuration file. For example, if an eight-view video is encoded, the user first inputs the number of views and the coding order. Then the user has the flexibility to select which view is used as a reference and how many references are used for key and non-key frames, given that the reference view is previously encoded.

3 Efficient Multiview Prediction Algorithms

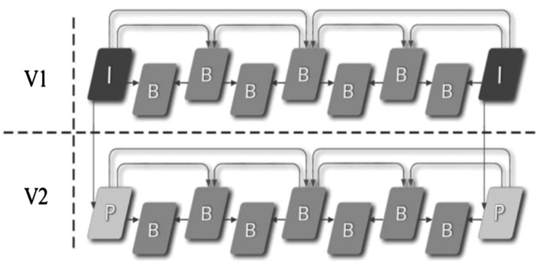
Most of the fast prediction algorithms designed for and applied to H.264/AVC can be implemented in any view of the MVC views.[26–28] However, as discussed in the previous section, due to the inter-view flexibility in the MVC, the number of possible references for any frame is far more than the AVC.

Recently, a few algorithms[3–8] have been proposed to speed up the prediction process in the view direction. For example, in Ref. 5 a fast inter-frame prediction algorithm was presented. It works by deciding whether or not the inter-view prediction is used for an MB based on co-located MBs in the temporal direction. If the co-located MBs in two reference frames in the same view find a best match using inter-view prediction, the MB to be coded uses reference



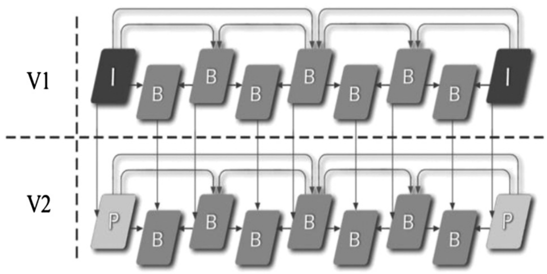
Display Order	1	2	3	4	5	6	7	8	9
Coding Order	0	5	3	6	2	7	4	8	1

Fig. 2 Temporal prediction using hierarchical B pictures.



Display Order	1	2	3	4	5	6	7	8	9
Coding Order	0	5	3	6	2	7	4	8	1

Fig. 3 Inter-view prediction for key pictures.



Display Order	1	2	3	4	5	6	7	8	9
Coding Order	0	5	3	6	2	7	4	8	1

Fig. 4 Inter-view prediction for key pictures and non-key pictures.

frames in the view directions. Otherwise the MB uses reference frames in the temporal direction. Furthermore, if the RD cost is greater than a particular threshold, references from other directions are used. Additionally, three more steps were added to speed up the motion prediction and the mode decision (MD) process by making use of the camera parameters and their effect on the object position in different views. Although the algorithm works well for a number of video sequences, such as Ballroom and Exit, the exploitation of thresholds to adaptively control the inter-view prediction has led to a significant bit-rate increase for some sequences.

In Ref. 6, an algorithm that makes use of the mode distribution correlation between neighbor views has been proposed to enhance the complexity efficiency. Mode complexity parameters for the current MB are defined from previously encoded co-located MB in neighbor views. Then, that parameter is compared to a threshold and based on the comparison results the MB can be categorized into one of three categories:

- MB with simple mode: all modes are terminated apart from the 16×16 mode.
- MB with medium mode: only the 16×16, 8×16 and 16×8 modes are examined.
- MB with complex mode: all modes are tested.

This scheme is based on statistical analysis that proved the correlation between the co-located MBs in neighbor views. However, introducing thresholds to control the RD performance has led to limiting the gain in some sequences when compared to the standard reference software. The algorithm is shown in Fig. 5 as a flowchart.

A similar algorithm to Ref. 6 has been proposed in Ref. 7, the basic idea of this method is to utilize the spatial property of the motion field to decide on when to use the inter-view prediction. Similar to Ref. 6, neighboring MBs are used to predict the motion of the current MB. In the first step motion homogeneity around the MB is determined by comparing the average MV of the neighbor MBs and the corresponding one in different views to a threshold. If a MB is found to be within a homogeneous motion region, only the inter-16×16 and intra-16×16 modes are tested, otherwise all the modes are examined. The algorithm is shown in Fig. 6 as a flowchart.

In Ref. 8, an object-based fast prediction MD method has been suggested. Segmentation is used to divide the frames into foreground and background objects. First, motion-based segmentation is applied to non-anchor frames by using information from both motion vectors and intensity value. Then, a disparity-based segmentation is carried out by considering the distribution of disparity vectors in the reference anchor picture. After the segmentation, interview prediction is only employed for MBs in the foreground regions. The algorithm is shown in Fig. 7 as a flowchart. From the figure it can be seen that the algorithm applies several complex pre-processing steps for the segmentation purpose that limits the overall gain.

4 Proposed Algorithm

Inter-view prediction coding of regions with fast movement is advantageous in the encoding aspect.[5] Reversely, we found inter-view prediction coding is not exploited in stationary

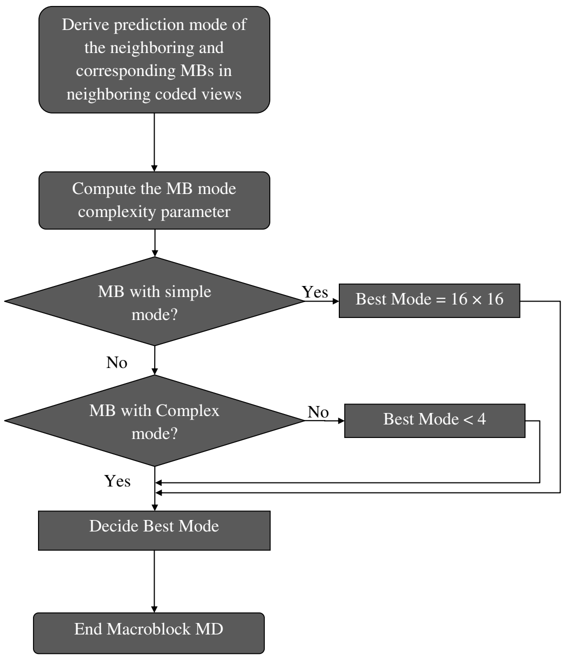


Fig. 5 Shen's fast MD algorithm.[6]

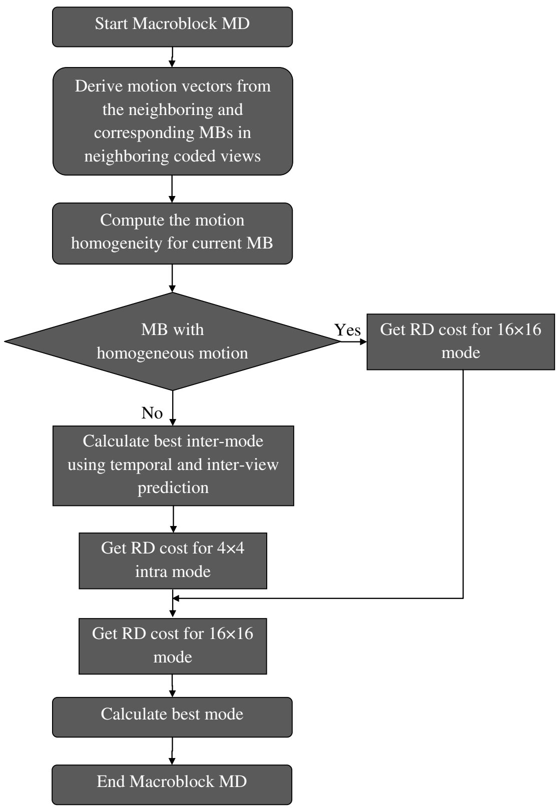


Fig. 6 Shen's selective disparity estimation.[7]

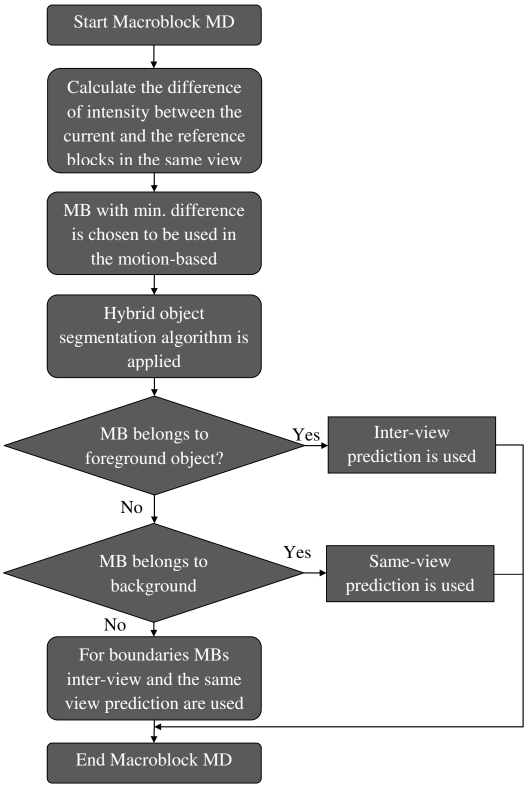


Fig. 7 Lee's fast MD algorithm.[8]

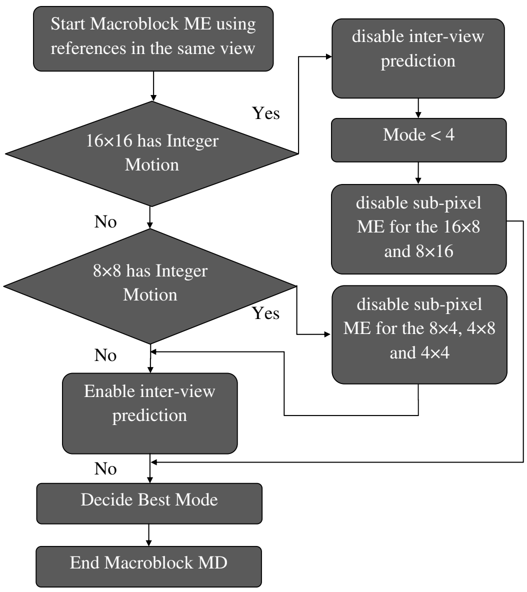


Fig. 8 The proposed multi-layered prediction algorithm.

areas. Furthermore, there is a tendency to encode stationary areas with large-size blocks. Therefore, when the best motion vector of a large-size block is an integer, the possibility that a large block is chosen as the best mode is very high. Thus, although the ME of the small-size blocks within this macroblock is not performed, there is little performance degradation and the encoding complexity can be improved significantly as in Ref. 10. In our proposed method, not only ME of smaller blocks but also the inter-view ME were not performed.

In contrast to the different algorithms discussed in the previous section, the proposed algorithm does not depend on motion prediction results of the co-located MB in the same view or in different views. Furthermore, no thresholds are used to keep the balance between the RD cost and reducing the complexity. Instead, the MB internal information is exploited to reduce the complexity.

If the motion prediction in the MVC is considered to be a three-layer process, with the first being the full- and sub-pixel motion search, the second the mode selection process and the third a repetition of the first and second for inter-view prediction, the proposed algorithm significantly reduces the complexity in the three layers. The first and second layers are the same for the AVC and MVC. The algorithm takes advantage of the proven fact[7,8] that only fast-moving objects in any view tend to find their best matches in neighbor views. It also takes advantage of the fast motion and mode selection algorithm that has been proposed in Ref. 10, which utilizes the correlation between an MB and its enclosed partitions's motion results in different layers to define areas with integer-motion in any frame. Those areas can also be classed as homogeneous areas. An additional step has been added to limit the use of inter-view prediction to fast-moving objects, thereby reducing the overall complexity. The algorithm can be summarized in the following steps:

- If the 16×16 finds a best match in the full pixel ME that does not change after performing the fractional pixel ME (integer-pixel motion), then disable inter-view prediction, limit the mode to 16×16, 16×8 and 8×16 and disable the subpixel ME for the 16×8 and 8×16.
- If the 8×8 finds a best match in the full pixel ME that does not change after performing the fractional pixel ME (integer-pixel motion), then disable the sub-pixel ME for the 8×4, 4×8 and 4×4.

The algorithm is shown in Fig. 8 as a flowchart.

The main advantage of this method in comparison to other schemes is that it makes use of some of the standard available tools to find homogeneity instead of employing additional pre-processing steps to segment the frames. Also, no additional statistical analyses need to be carried out and therefore statistical results do not need to be stored.

5 Experiments

To evaluate the proposed algorithm, a comprehensive set of experiments for various kinds of video sequences[29] with different motion characteristics have been carried out. Table 2 shows the sequences properties. These data sets vary in the number of cameras/views, the arrangement of the cameras,

Table 2 Test video sequences.

Sequences	Image property	Camera arrangment
Akko&Kayo	640 × 480, 30 fps	100 cameras with 5 cm horizontal and 20 cm vertical spacing; 2D array
Flamenco	640 × 480, 30 fps	5 cameras with 20 cm spacing; 2D/parallel (Cross)
Race	640 × 480, 30 fps	8 cameras with 20 cm spacing; 1D/parallel
Rena	640 × 480, 30 fps	100 cameras with 5 cm spacing; 1D/parallel
Uli	1024 × 768, 25 fps	8 cameras with 20 cm spacing; 1D/parallel convergent
Ballet	640 × 480, 25 fps	8 cameras with 20 cm spacing; 1D/parallel
Breakdancing	1024 × 768, 15 fps	8 cameras with 20 cm spacing; 1D/arc
Exit	640 × 480, 25 fps	8 cameras with 20 cm spacing; 1D/parallel

distance between cameras, as well as properties of the images in terms of image size and frame rate. All sequences are provided in YUV 4:2:0 planar formats.

The scheme is implemented on a JMVM 9.15 encoder.[25] The test platform uses an Intel Core(TM) i7 CPU 920 @ 2.47 GHz with 8.0 GB RAM running Windows 7. The Intel VTune performance analyzer was used to measure the number of machine cycle differences that reflects the total encoding Time Saving (TS), as shown in Eq. (1). This provides accurate information about processor utilization as the complexity differences between algorithms is calculated in terms of basic operations used in the computer, including addition, multiplication, shift and comparison.

Table 3 Experiment encoder configurations.

Parameter	Value	Parameter	Value
Resolution	640 × 480	GOP size	16
	1024 × 768	MV resolution	1/4 Pel
QP setting	14–22–30–38	No. of frames	200–300
		Motion search range	32
Frame rate in/out	15–25 and 30 HZ	Reference picture	2
		Search function	SAD

Table 4 Comparison between the proposed algorithm and the JMVM 8.0 software.

Sequence	TS (%)	BDPSNR (dB)	BDBR (%)
Akko&kayo	55.4	−0.09	1.09
Flamenco	53.6	−0.06	1.62
Race	42.8	−0.03	0.86
Rena	66.5	−0.1	1.04
Uli	52.3	−0.01	0.73
Ballet	76.12	−0.02	0.92
Breakdancing	63.55	−0.1	0.93
Exit	64.8	−0.05	1.34
Average	**59.41**	**−0.05**	**1.06**

$$\mathrm{TS} = \frac{T_{\mathrm{JMVM}} - T_{\mathrm{Proposed}}}{T_{\mathrm{JMVM}}} \times 100\%. \quad (1)$$

Additionally, Bjontegaard delta peak signal to noise ratio (BDPSNR), and Bjontegaard delta bit-rate (BDBR)[30] have been used to evaluate the proposed algorithm performance versus the JMVM encoder and recent work in the area. The test condition is shown in Table 3.

Initially the algorithm was compared to the MVC reference software.[25] The experimental results are shown in Table 4. It can be seen that the proposed scheme achieves an average of 59.38% time saving with negligible losses in PSNR and negligible increase in bit-rate.

Table 5 Comparison between the proposed algorithm and the algorithm proposed in Ref. 5.

Sequence	TS (%)	BDPSNR (dB)	BDBR (%)
Akko&kayo	6.2	0.05	−0.6
Flamenco	8.3	0.07	−1.10
Race	3.1	0.04	0.54
Rena	4.1	0.09	−1.23
Uli	16.4	0.05	0.085
Ballet	12.5	0.12	0.225
Breakdancing	11.7	0.08	−1.4
Exit	4.1	0.06	0.2
Average	**8.3**	**0.07**	**−0.41**

Table 6 Comparison between the proposed algorithm and the algorithm proposed in Ref. 6.

Sequence	TS (%)	BDPSNR (dB)	BDBR (%)
Akko&kayo	9.3	0.04	0.53
Flamenco	6.8	0.05	−1.03
Race	8.7	0.06	−0.7
Rena	14.6	−0.02	0.75
Uli	15.7	−0.03	0.55
Ballet	6.8	0.01	−0.09
Breakdancing	11.6	0.09	−0.6
Exit	9.5	−0.01	1.2
Average	**10.38**	**0.023**	**0.076**

Additionally the proposed algorithm has been compared with the recently proposed algorithm.[6–8] The comparison results are shown in Tables 5–7, respectively.

The tables show that the time saving is video-content-dependent, however the proposed scheme results in significant time savings when compared to MVC reference software and other known work while maintaining the same RD performance. In comparison to the algorithms proposed in Refs. 5 to 7, our proposed algorithm provides an average time saving of 8.3%, 10.4% and 23%, respectively. Notice that the proposed algorithm demonstrates considerable speedup for many sequences, irrespective of whether the general motion is still, slow, or fast, as the homogeneous areas are detected across all the different sequences.

Table 7 Comparison between the proposed algorithm and the algorithm proposed in Ref. 7.

Sequence	TS (%)	BDPSNR (dB)	BDBR (%)
Akko&kayo	29.4	−0.01	0.15
Flamenco	27.8	−0.04	0.21
Race	19.6	0.06	0.27
Rena	25.6	−0.03	0.12
Uli	24.2	−0.02	1.01
Ballet	20.2	−0.01	−0.01
Breakdancing	18.5	0.05	−0.73
Exit	21.4	−0.11	−0.143
Average	**23.33**	**−0.014**	**0.10**

6 Conclusion

The large amount of video data and particularly high-computational complexity makes the MVC encoder difficult to be implemented in real-time applications. This paper has presented a fast algorithm for multiview video coding. In addition to reducing the complexity of the subpixel ME and the MD for frames in the same view, a novel early reference termination has been incorporated to the inter-view prediction of MBs. The proposed algorithm depends on the property of video sequences that fast-moving objects are likely to be predicted using inter-view references, while background objects are more likely to be predicted using references from the same view. The MBs' inherited correlation is exploited as a motion-based segmentation to locate fast-moving objects.

This novel method gives the algorithm automated adaption to any video sequence with any characteristic, while most proposed fast algorithms in the area rely heavily on the spatial and the temporal correlation between MBs, which limits their time-saving to certain sequences. Thus the advantage of the resulting multiview prediction structure is achieving significant coding gains and being highly flexible regarding its adaptation to all kinds of spatial and temporal setups. Unlike most algorithms available in literature, the performance of the complexity reduction algorithm does not depend on empirically obtained thresholds. This algorithm automatically adapts to different sequence statistics without the need for tuning thresholds. Furthermore, the algorithm is unique, as it does not take advantage of the spatial mode distribution between MBs. Instead it relies on the relationship between the MB and its enclosed partitions. The advantage of this, in contrast to other schemes, is the obvious consistency of the resultant RD performance.

To assess the proposed algorithm, a comprehensive set of experiments were conducted. The results show that the proposed algorithm significantly reduces the ME time while maintaining similar rate-distortion performance, when compared to both the H.264/MVC reference software and recently reported work. This saved computation can advance the progress in the realization of the H.264 multiview extension in real-time applications and low-complexity coding systems.

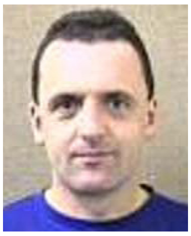
References

1. I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons, Chichester, West Sussex (2003).
2. ISO/IEC/JTC1/SC29/WG11, "Multi-view coding using AVC," Bangkok, Thailand (January 2006).
3. J. Lu et al., "An epipolar geometry-based fast disparity estimation algorithm for multi-view image and video coding," *IEEE Trans. Circ. Syst. Video Technol.* **17**(6), 737–750 (2007).
4. G. Jiang et al., "Fast multi-view disparity estimation for multi-view video systems," Springer Lecture Notes in Computer Science, Vol. 4179, in *8th International Conference on Advanced Concepts for Intelligent Vision Systems, ACIVS 2006*, Antwerp, Belgium, pp. 493–500 (2006).
5. X. Li et al., "A fast inter frame prediction algorithm for multi-view video coding," in *Proc. IEEE International Conference on Image Processing (ICIP)*, San Antonio, TX, pp. 417–742 (2007).
6. L. Shen et al., "Fast mode decision for multiview video coding," in *Proc. IEEE International Conference on Image Processing (ICIP)*, Cairo, Egypt, pp. 2953–2956 (2010).
7. L. Shen et al., "Selective disparity estimation and variable size motion estimation based on motion homogeneity for multi-view coding," *IEEE Trans. Broadcast. Technol.* **55**(4), 761–766 (2009).
8. S. Y. Lee, K. M. Shin, and K. M. Shin, "An object-based mode decision algorithm for multi-view video coding," in *ISM '08 Proceedings of the Tenth IEEE International Symposium on Multimedia*, pp. 74–81, IEEE, Washington, DC (2008).

9. P. An, C. Lu, and Z. Zhang, "Object segmentation using stereo images," in *International Conference on Communications, Circuits and Systems, ICCCAS*, Chengdou, China, Vol. 1, pp. 534–538 (2004).
10. A. Abdelazim et al., "Low complexity hierarchical prediction algorithm for H.264/SVC," in *Proceedings of Pacific-Rim Symposium Image Video Technology (PSIVT)*, Singapore, pp. 14–17 (2010).
11. Y. S. Ho and K. J. Oh, "Overview multi-view video coding, in the proceedings of systems," *EURASIP Conf. on Speech and Image Processing*, pp. 5–12 (2007).
12. A. Smolic, "Introduction to multiview video coding," ISO/IEC JTC 1/SC 29/WG 11N9580, Antalya, Turkey (January 2008).
13. ISO/IEC JTC1/SC29/WG11, Requirements on Multiview Video Coding v.4., Doc. N7282, Poznan, Poland (July 2005).
14. P. Merkle et al., "Efficient prediction structures for multi-view video coding," *IEEE Trans. Circ. Syst. Video Technol.* **17**(11), 1461–1473 (2007).
15. P. Merkle et al., "Efficient compression of multi-view video exploiting inter-view dependencies based on H.264/MPEG4-AVC," in *Proc. IEEE International conference on Multimedia and Expo, ICME*, Toronto, Ontario, Canada, pp. 1717–1720 (2006).
16. U. Fecker and A. Kaup, "Statistical analysis of temporal and spatial block matching algorithm for multi-view video sequences," ISO/IEC JTC1/SC29/WG11, MPEG2005/M11546, Hong Kong, China (January 2005).
17. ISO/JTCIEC1/SC29/WG11, "Survey of Algorithms used for Multi-view Video Coding (MVC)," Doc. N6909, Hong Kong, China (January 2005).
18. A. Smolic and P. Kauff, "Interactive 3D video representation and coding technologies," in *Proc. IEEE*, **93**(1), 98–110 (2005).
19. K.-J. Oh and Y.-S. Ho, "Multi-view video coding based on the lattice-like pyramid GOP structure," in *Proc. Picture Coding Symposium (PCS) 2006, Picture Coding Symposium*, Beijing, China (2006).
20. X. Cheng, L. Sun, and S. Yang, "A multi-view video coding scheme using shared key frames for high interactive application," in *Proc. Picture Coding Symposium (PCS)*, Beijing, China (2006).
21. Y. Yang et al., "Hyper-space based multiview video coding scheme for free viewpoint television," in *Proc. Picture Coding Symposium(PCS) Symposium*, Beijing, China (2006).
22. F. Shao et al., "A new image correction method for multiview video system," in *IEEE Int. Conf. on Multimedia and Expo*, Toronto, Canada (2006).
23. A. Kaup and U. Fecker, "Analysis of multi-reference block matching for multi-view video coding," in *Proc. 7th Workshop Digital Broadcasting*, Erlangen, Germany (2006).
24. H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *IEEE Int. Conf. on Multimedia and Expo*, Toronto, Canada (2006).
25. ISO/IEC JTC1/SC29/WG11, Joint Multiview Video Model (JMVM) 7.0, JVT Document, JVT-Z207 (January 2008).
26. L. Shen et al., "Fast inter mode decision using spatial property of motion field," *IEEE Trans. Multimedia* **10**(6) 1208–1214, (2008).
27. B. G. Kim, "Novel inter-mode decision algorithm based on macroblock tracking for the P-slice in H.264/AVC video coding," *IEEE Trans. Circ. Syst. Video Technol.* **18**(2), 273–279 (2008).
28. A. Abdelazim et al., "Fast mode decision for the H.264/AVC based on frequency domain motion estimation," *Opt. Eng.* **50**(7), 1–3 (2011).
29. Y. Su, A. Vetro, and A. Smolic, "Common test conditions for multiview video coding," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6), Document: JVT-T207, pp. 15–21 (2006).
30. G. Bjøntegaard, "Calculation of average PSNR differences between RD-Curves," ITU-T SG16 Q.6 Document, VCEG-M33, Austin, TX (April 2001).

Abdelrahman Abdelazim is an assistant professor of computer engineering at the American University of the Middle East (AUM) in Kuwait. He holds a BEng in digital communication and a PhD degree in engineering, both from the University of Central Lancashire, Preston, United Kingdom. He is a Fellow of the Higher Education Academy. Prior to joining AUM, he was a lecturer in engineering at the University of Central Lancashire. He previously worked at Blackpool College, Lancashire Business School, and in industry on a number of digital communication projects. His research interests are in the area of reducing the complexity of video coding encoders in real-time scalable and multiview applications. He has published several conference and journal papers on video compression and transmission. He is a Member of the Institution of Engineering and Technology (MIET) and SPIE.

Stephen Mein holds a BEng (2002) and a PhD (2009) in electronic engineering. Since 2007, he has worked as a lecturer in electronics within the School of Computing, Engineering and Physical Sciences at the University of Central Lancashire. His research interests focus on the application of image-processing techniques and probabilistic methods to problems in the field of ultrasonic nondestructive testing.



Martin R. Varley holds a BSc (1985) and a PhD (1990) in electrical and electronic engineering. He has worked in United Kingdom higher education for over 25 years and is currently a principal lecturer in electronic and computer engineering and academic lead for engineering in the School of Computing, Engineering and Physical Sciences at the University of Central Lancashire. He is a member of the Institution of Engineering and Technology and has authored and coauthored over 60 journal and conference papers. His research interests include applications of digital signal and image processing including compression, immersive visualisation and robotics, artificial neural networks, and engineering education.

Djamel Ait-Boudaoud is currently the dean of faculty of technology at the University of Portsmouth. With an academic career spanning over 20 years, he has held several senior academic appointments in the UK and overseas. Prior to joining the University of Portsmouth, he was the head of School of Computing, Engineering and Physical Sciences at the University of Central Lancashire. He is a chartered engineer (CEng) and a Fellow of the Institution of Engineering and Technology. He holds a PhD (1991) from the University of Nottingham, where he conducted research in system and circuit integration for digital signal processing applications. He has authored and coauthored several research papers; his research interests include VLSI architectures, image processing and computer vision, and evolutionary algorithms.