

Waseda University Doctoral Dissertation

**Research on Quality-Optimized Algorithms for Low
Computational Complexity Techniques in Video Processing**

Wenxin YU

Graduate School of Information, Production and Systems
Waseda University

April 2014

Abstract

In recent years, with the wide application of mobile devices in people's lives, the rapid development of high-definition video coding and decoding technology, the rapid progress of the new generation of communication technology, as well as the growing demand for cultural and entertainment activities, the three-dimensional (3-D) products have become increasingly popular in daily life. In most traditional 3-D multimedia systems, only one pre-determined viewpoint of an image or video can be seen by an observer. If the viewpoint is changed, the realistic 3-D impression will become much weaker and the quality of the 3-D video will worsen. To increase the number of viewpoints for the observers and image more comfortable for viewers, the free viewpoint television (FTV) was introduced. Auto-stereoscopic displays provide a 3-D impression to an observer without the need to wear additional glasses, and the observers can enjoy a realistic 3-D impression from certain different viewpoints. Such a display shows a number of slightly different views at the same time. People can watch the stereo-view 3D movie not only in the cinema, but also at home or via internet, or even by the mobile devices without any glasses. However, to simultaneously deliver so many views at a same time, an extremely large amount of bandwidth is required.

On the terminal devices, the requirements of the computational ability and the video quality are very high, especially for the mobile devices. Although the storage and computational ability is growing rapidly, they still cannot meet the unlimited users' needs. Based on this kind of background, the novel algorithms with low computational complexity in video processing are proposed for the mobile applications. Our first proposal focuses on a low power and low computational complexity video decoding with adaptive granularity in temporal scalability for H.264/AVC decoder. Compared with the previous works (Mohammed E. Al-Mualla (ISCAS2003) and etc.), my proposed algorithm can realize frame rate down

conversion with low computational complexity. The second proposal focuses on the Frame Compatible Format Fast Encoder with Stereo Matching. Frame compatible format fast encoder is used to reduce computational complexity in the encoding side for the multi-view encoding cases. The third proposal focuses on the Novel Hole-filling Algorithm in View Synthesis. It can be used to generate some more virtual views in the decoding side to meet the requirements and improve the video quality. One real view and one depth are used in the proposed methods. Compared with the previous works (Zeng (VCIP 2012), Lai-Man Po (ICIP2011) and etc.), my proposed algorithm can improve the video quality a lot through PSNR and SSIM evaluation.

This dissertation consists of five chapters which are as follows:

Chapter 1 [Introduction] introduced the conventional coding standards first. For the 2-D cases, H.264/AVC standard and the frame rate conversion are introduced. For the 3-D cases, Multiview Video Coding (MVC), Depth Image Based Rendering (DIBR), Stereo Matching and Frame-compatible Format (FCF) are introduced. Finally the major contributions are listed.

Chapter 2 [Adaptive Low Computational Complexity Algorithms in Video Decoding Process] introduced the low power and low computational complexity algorithms for the 2-D cases.

This work is reduced the decoding time with frame rate down conversion and keeps the video quality in an acceptable degree.

The previous works focused on the frame rate up conversion in the transcoding process, by Mohammed E. Al-Mualla (ISCAS2003), Nicolas Beucher (SIPS2006), Truong Quang Vinh (ICCES2009) and etc.. There is also one work that proposed for the frame rate down conversion in the encoder by Yukihiro Bando (ICIP2009). These works can't be used to reduce the computational complexity for the terminal devices in the decoding side.

At first, temporal scalable decoding process with certain frame rate conversion is introduced. It realized the frame rate conversion by skipping the B or P frames. There are three main algorithms in the p frame skipping process which are used to improve the video quality. The first one is the reference frame index decision algorithm (RFD). Based on this algorithm, it will choose the nearest forward frame which is not skipped as the refined reference frame. The second one is the motion vector composition algorithm (MVC). It calculates the Δ block (the offset use block as unit) based on the motion vector value of the 4×4 block in current frame at first, finds the four overlap blocks in the skipped frame and ensure the exact position of the block by using the difference between the motion vector value of the 4×4 block and the Δ block (turn to quarter pixel precision). Then chose the maximum overlap block as the candidate block, and use the motion vector of that block to do the motion vector composition and get the refined motion vector which point to the refined reference frame. The third one is the block-partition mode decision algorithm (BPD). In this part, it refined the unsuitable block-partition and sub block-partition modes after finished RFD and MVC to guarantee the video quality. The RFD is used to reduce the frame rate and the MVC, BPD are used to keep the video quality.

Based on the content in different frames, we proposed the adaptive frame skipping method. Calculate the sum of the motion vector value of the current frame at first, then compared with the threshold to decide that skip this frame or keep it.

Through the experimental results, the PSNR loss is very small (0.1 to 0.2 dB) for B frame skipping. The PSNR loss is 0.4 to 1.8 dB for 2/3 P frame skipping and reduce the computational complexity about 60%. The PSNR can be more improved 0.2 to 0.9 dB by using the adaptive frame skipping method, but the reduction of the computational complexity will reduce about 5%.

Chapter 3 [Frame Compatible Format Fast Encoder with Stereo Matching] introduced an improved stereo matching algorithm based on the work which proposed by Wang (ISCAS2012) and compared with the work by Zeng (VCIP 2012).

Some previous works have been down for the Frame Compatible Format, by

Anthony Vetro (ICIP2010), Ying Chen (ICME2011), Siao-Wei Chen (ICASSP2013), Taoran Lu (ICME2013) and etc.. These works mainly focus on how to partition the views, how to reduce the quality loss in the down-sampling process and how to improve the video quality in the up-sampling process. There is also one previous work focuses on how to utilize the relation between the content similarities of different views by Zeng (VCIP 2012). But the depth information prediction is too rough to get the best match block in Zeng's work. Our work combined the improved fast stereo matching algorithm which can provide much more accurate depth information into the FCF.

The reconstruction of a stereo scene from a pair of images taken from different directions is called stereo matching. The traditional stereo matching algorithm can't be combined into the FCF, because of the huge computational complexity. To meet with the requirements of fast algorithm for FCF, an improved fast stereo matching algorithm is proposed. An AD-Census cost is used here as the matching cost which combines the absolute difference and Census cost. Choose the reliable disparity points based on three detection rules and use nearest three points with reliable disparities to predict the unknown points. The BD-PSNR is improved 0.01 to 0.2 dB, the BD-BR is reduced 0.04% to 4%. Our proposed fast stereo matching algorithm can provide much more accurate depth information for the FCF and just cost $1/20 \sim 1/10$ computational time than the traditional stereo matching algorithm.

Chapter 4 [The Novel Hole-filling Algorithms for View Synthesis] introduced the novel hole-filling algorithms in view synthesis for the 3-D cases.

This proposed method works in the 3D decoding process. It uses one real view and one depth map to generate several virtual views with the novel hole-filling method in the view synthesis process.

The image information occluded in the real view may become visible in the "virtual" image. Some holes will then appear in the virtual image after the 3-D warping in the view synthesis process. The video quality loss is very big, so the hole-filling algorithm is required to improve the virtual views' quality.

Some previous works have already been proposed. Lai-Man Po (ICIP2011) introduced a mask in-painting algorithm, Kwan-Jung Oh (PCS2009) presented a background priority in-painting algorithm and Ismael Daribo (MMSP2010) proposed a gradient priority in-painting algorithm. These works only use some neighbor information to predict the unknown region, but our proposed algorithm uses some long distance information in the prediction process and it can obtain much better objective and subjective results than the previous works.

The proposed algorithm includes two parts. The first one is an integrated hole-filling algorithm for view synthesis. This is a spatial prediction algorithm, it ensures the boundary of the foreground objects by distinguishing the different layers in the texture image with depth information, predicts the textural and structural information in the zero-region by using the geometry principle, and paints the zero-region with a priority order based on the gradient information. The proposed algorithm uses a geometry solution to keep some long distance information in the prediction process. Therefore, the proposed method will improve not only the objective quality of a synthesized virtual view but also its subjective quality and the 3-D performance for human vision. The second one is an extension work of the first work. It combines the temporal prediction and the spatial prediction together. It fetches the temporal information from the previous frames, distinguishes the absolute background and relative background for the temporal hole-filling and updates the boundary information after temporal hole-filling for the next spatial prediction process. In some of the cases, the camera nearly doesn't move at all. Then the prediction based on the temporal information is much more reliable and stable than the spatial prediction in this kind of situations. However, in the camera moving cases, the spatial prediction is much better than the temporal prediction. Through the proposed spatial prediction result which Compare with Lai-Man Po (ICIP2011), the PSNR is improved 0.10 to 1.95 dB, the SSIM is improved 0.0006 to 0.0043. The combined realization can obtain the advantages from both of the spatial and temporal prediction method. Also compared with Lai-Man Po (ICIP2011), the PSNR is improved 0.10 to 3.02 dB, the SSIM is improved 0.0006 to 0.0122.

Chapter 5 [Conclusion] summarizes this dissertation.

Acknowledgement

I owe a great many thanks to a great many people who helped and supported me during the writing of this dissertation.

All works would not have been possible without the guidance and the help of my supervisor, Professor Satoshi Goto, whose sincerity and encouragement I will never forget. His foresight and sagacity give me unbounded confidence in the research work. His patient and understanding personality makes me enjoy the time to work with him. With his guidance, I got not only the knowledge, but also the faith to defeat every predictable or unpredictable difficulty.

I am highly indebted to Professor Shinji Kimura, Professor Takeshi Yoshimura and Professor Jiro Katto of Waseda University for their guidance and concise instructions through my research. Thanks for their advices and comments to my dissertation, which is professional and helpful.

My thanks and appreciations also go to my colleague in the past years. Thanks Dr. Xin Jin, Dr. Minghui Wang, Dr. Gang He, Dr. Ning Jiang, Dr. Jiu Xu and Ms. Xinwei Xue for the sincerely advices. Discussing with them I can always find the solutions of the problems. Thanks to all the colleges in Goto Lab, who enriched my study life in the past five years and gave me their support consistently. I cannot have a fruitful life here without them.

I also extend my heartfelt thanks to my family, who love me and support me in every hour during my life.

Contents

Abstract	I
Acknowledgement	VII
Index of Figures	XI
Index of Tables	XIV
1. Introduction.....	1
1.1 Conventional video coding standards.....	1
1.2 H.264/AVC standard	1
1.2.1 H.264's key features.....	3
1.3 The Frame Rate Conversion.....	6
1.4 Multiview Video Coding (MVC)	7
1.5 Depth Image Based Rendering (DIBR).....	9
1.6 Stereo Matching	11
1.7 Frame-compatible Format (FCF)	11
1.8 Our contribution	12
1.8.1 Adaptive Low Computational Complexity Algorithms in Video Decoding Process 12	
1.8.2 Frame Compatible Format Fast Encoder with Stereo Matching	14
1.8.3 The Novel Hole-filling Algorithms for View Synthesis	15
1.9 Dissertation Organization.....	18
2. Adaptive Low Computational Complexity Algorithms in Video Decoding Process	19
2.1 Introduction	19
2.2 Previous works.....	20
2.3 Temporal scalable decoding process with frame rate conversion method for surveillance video	22
2.3.1 Frame rate conversion method in the proposed process.....	22
2.3.2 Temporal scalable decoding process	24
2.3.3 Experimental result	32
2.3.4 Conclusion	36
2.4 Adaptive solution of temporal scalable decoding process with frame rate conversion method for surveillance video	38
2.4.1 Adaptive solution for surveillance video	38
2.4.2 Experimental result	44
2.4.3 Conclusion	45

2.5	Adaptive decoding process with temporal prediction method for common video	47
2.5.1	Details of the adaptive decoding process	47
2.5.2	Experimental result	49
2.5.3	Conclusion	53
3.	Frame Compatible Format Fast Encoder with Stereo Matching.....	55
3.1	Introduction	55
3.2	Previous Work	56
3.3	The combined architecture of the fast algorithm of FCF with stereo matching	61
3.4	The proposed algorithm	62
3.4.1	The relationship between stereo matching and FCF	62
3.4.2	The modified stereo matching algorithm	62
3.4.3	Fast encoding algorithm for FCF using stereo matching	72
3.4.4	Simulation results.....	74
3.4.5	Conclusion	75
4.	The Novel Hole-filling Algorithms for View Synthesis	77
4.1	Introduction	77
4.2	Previous works.....	78
4.2.1	Mask Used In-painting.....	78
4.2.2	The Background Region In-painting.....	79
4.2.3	The In-painting Based on Some Special Priority	81
4.3	An integrated hole-filling algorithm for view synthesis.....	82
4.3.1	Architecture of the integrated hole-filling algorithm for view synthesis	82
4.3.2	Details of the hole-filling algorithm.....	84
4.3.3	Experimental Results	98
4.3.4	Conclusion	102
4.4	Combined hole-filling with spatial and temporal prediction.....	103
4.4.1	The architecture of the proposal.....	103
4.4.2	Details of the combined hole-filling algorithm	104
4.4.3	Experimental results.....	109
4.4.4	Conclusion	111
5.	Conclusion of the dissertation	112
	Reference.....	112
	Publication List.....	124

Index of Figures

Fig.1. H.264 encoding/decoding processing.....	2
Fig.2. Intra prediction in H.264. (a) Prediction samples for a 4 x 4 block. (b) Directions for prediction modes for 4 x 4 and 8 x 8 blocks	5
Fig.3. Sub blocks of a macroblock in inter prediction.....	6
Fig.4. Depth image-based virtual view synthesis	9
Fig.5. General concept of 3-D warping [84].....	10
Fig.6. The relationship between the Frame-compatible format fast encoder and View synthesis	15
Fig.7. The difference between the spatial and temporal prediction.....	17
Fig.8. The traditional solution for the multi-layers bit-stream transmission	23
Fig.9. The proposed frame rate down conversion algorithm.....	23
Fig.10. Temporal scalable decoding process	25
Fig.11. B frame skipping illustration.....	26
Fig.12. P frame skipping illustration	27
Fig.13. Motion vector composition (Illustration)	29
Fig.14. Multi-step motion vector composition (Illustration).....	29
Fig.15. Block-partition and sub block-partition modes	30
Fig.16. Sub block-partition mode decision flow	31
Fig.17. Block-partition mode decision flow	32
Fig.18. Intersection.yuv with 1/2 P frame skipping (Illustration) [86].....	36
Fig.19. The comparison between before and after skipping [86]	38
Fig.20. The illustration of the skipped frame selecting	39
Fig.21. The correlation coefficient between video quality loss and the motion vector information (y-axis: video quality loss (dB); x-axis: motion vector information energy).....	41
Fig.22. The correlation coefficient between video quality loss and the residual information (y-axis: video quality loss (dB); x-axis: residual	

information energy)	42
Fig.23. The correlation coefficient between the residual information and the motion vector information (y-axis: motion vector information energy; x-axis: residual information energy)	42
Fig.24. The deciding process of the adaptive scheme	43
Fig.25. The flowchart of the adaptive scheme.....	44
Fig.26. The temporal prediction illustration	47
Fig.27. The temporal prediction flowchart	48
Fig.28. The relationship between the threshold and the number of skipped frames in container_cif.yuv	50
Fig.29. The relationship between the threshold and the number of skipped frames in mobile_cif.yuv.....	50
Fig.30. The relationship between the threshold and the number of skipped frames in coastguard_cif.yuv	51
Fig.31. The relationship between the threshold and the number of skipped frames in foreman_cif.yuv	52
Fig.32. Matching scheme in top-bottom arrangement.....	57
Fig.33. Illustration of the depth map fetching process [85].....	58
Fig.34. The flowchart of the previous fast algorithm.	60
Fig.35. The architecture of using stereo matching in fast FCF encoder.....	61
Fig.36. The actual horizontal shift between the reference and target [85]	62
Fig.37. The result from stereo matching algorithm	63
Fig.38. The marked shift value for 4*4 blocks.....	63
Fig.39. The marked shift value for 8*8 blocks.....	64
Fig.40. The marked shift value for 16*16 blocks.....	64
Fig.41. Mode 0 in the arrangement modes of block corresponding	65
Fig.42. Mode 1 in the arrangement modes of block corresponding	66
Fig.43. Mode 2 in the arrangement modes of block corresponding	67
Fig.44. Mode 3 in the arrangement modes of block corresponding	67
Fig.45. Probabilities of partition type using simple shift obtaining	68

Fig.46. Probabilities of partition type using stereo matching.....	68
Fig.47. The process of reliable disparity propagation	71
Fig.48. Flowchart of the new fast FCF encoding algorithm.....	73
Fig.49. Simple mask in-painting.....	79
Fig.50. General in-painting circumstance.....	80
Fig.51. Manipulation of hole to have neighborhood only come from background	81
Fig.52. Notation diagram	81
Fig.53. Architecture of the proposed method	83
Fig.54. Illustration of the distinguishing process [84].....	85
Fig.55. Flowchart of the distinguishing algorithm	86
Fig.56. Result of the boundary detection.....	87
Fig.57. Flowchart of the detection process	89
Fig.58. Steps and the results of textural and structural detection [84]	90
Fig.59. Illustration of curve fitting and prediction	91
Fig.60. The flowchart of the isophote prediction	91
Fig.61. The flowchart of the predicting process	93
Fig.62. Illustration of the intersection determination	94
Fig.63. The flowchart of the intersection determination	94
Fig.64. Illustration of the isophote prediction process	95
Fig.65. Illustration of the gradient information	96
Fig.66. Illustration of the filling directions.....	97
Fig.67. Detailed comparison of the results [84]	100
Fig.68. The architecture of the proposed algorithm	103
Fig.69. The detailed flowchart of the proposed algorithm	105
Fig.70. Some results of temporal information fetching [89]	106
Fig.71. The errors occur in the temporal prediction process without distinguishing the absolute and relative background layers [87]	107
Fig.72. The illustration of the updated boundary information [88].....	108
Fig.73. The subjective comparison result [88]	110

Index of Tables

Table 1	Experimental sources and parameters:	33
Table 2	Experimental result: (I-B-B-P-B-B-P...)	33
Table 3	Experimental result: (I-P-P-P-P...)	34
Table 4	The comparison result between I-P-P-P... and I-B-B-P-B... with the same frame rate	35
Table 5	Experimental sources and parameters:	45
Table 6	Experimental result:	46
Table 7	The experimental sources and parameters:	49
Table 8	The experimental result:	53
Table 9	The encoding results of previous FCF fast encoding [67]	74
Table 10	The encoding results of FCF fast encoding by using stereo matching	74
Table 11	The comparison result with the previous work	75
Table 12	Some additional experimental results	101
Table 13	PSNR and SSIM results by the proposed algorithm, [57] and the view synthesis reference software	109

1. Introduction

1.1 Conventional video coding standards

Video coding is a very important technology in our daily life. In the recent year, it developed faster and faster, because of the growth of the mobile applications and the requirements of the high definition video.

People can watch high-definition video (1080p) on the smartphones, cinemas, 3D movies, at home or via the Internet. With the wide application of the new generation of communication technology, there are big demands of the high definition video content in our daily life. Computer storage and communication bandwidth can not meet the needs of people at present and future. So the data compression techniques are more and more important. To compress the redundant content in maximum, a lot of coding techniques are proposed, such as “Block segment”, “Homogenous matching” and “Transform domain compression” methods.

However, high compression ratio always corresponds to higher computation complexity and higher power consumption. The power consumption is quite limited in the mobile cases, so "Low power consumption" has become an important indicator. A low power and fast algorithm is proposed to balance the coding efficiency and the computational complexity.

1.2 H.264/AVC standard

H.264 is a new generation of digital video compression format after MPEG4, which is jointly proposed by the International Organization for Standardization (ISO) and the International Telecommunication Union (ITU). H.264 is one of the technical standards of video encoding and decoding of ITU-T named H.26x series. H.264 is a digital video coding standard developed by ITU-T's VCEG (video coding expert group) and ISO/IEC's MPEG (active image coding expert group) in the joint video

team (JVT). The standard came first from the development of ITU-T's project called H.26L. Although the name of H.26L is not very common, it has been used. H.264 is one of the standards that ITU-T is named after H.26x series. AVC is the name of ISO/IEC MPEG.

H.264 is built on the basis of MPEG-4 technology, and its coding and decoding process mainly consists of 5 parts: inter frame and intra prediction (Estimation), transform (Transform) and inverse transform, quantization (Quantization) and inverse quantization, Loop Filter, and entropy coding (Entropy Coding).

The main goal of the H.264 standard is to provide better image quality at the same bandwidth compared with other existing video coding standards. By this standard, the compression efficiency of the same image quality is increased by about 2 times than that of the previous standard (MPEG2).

The H.264 encoding and decoding processing are shown in Fig.1.

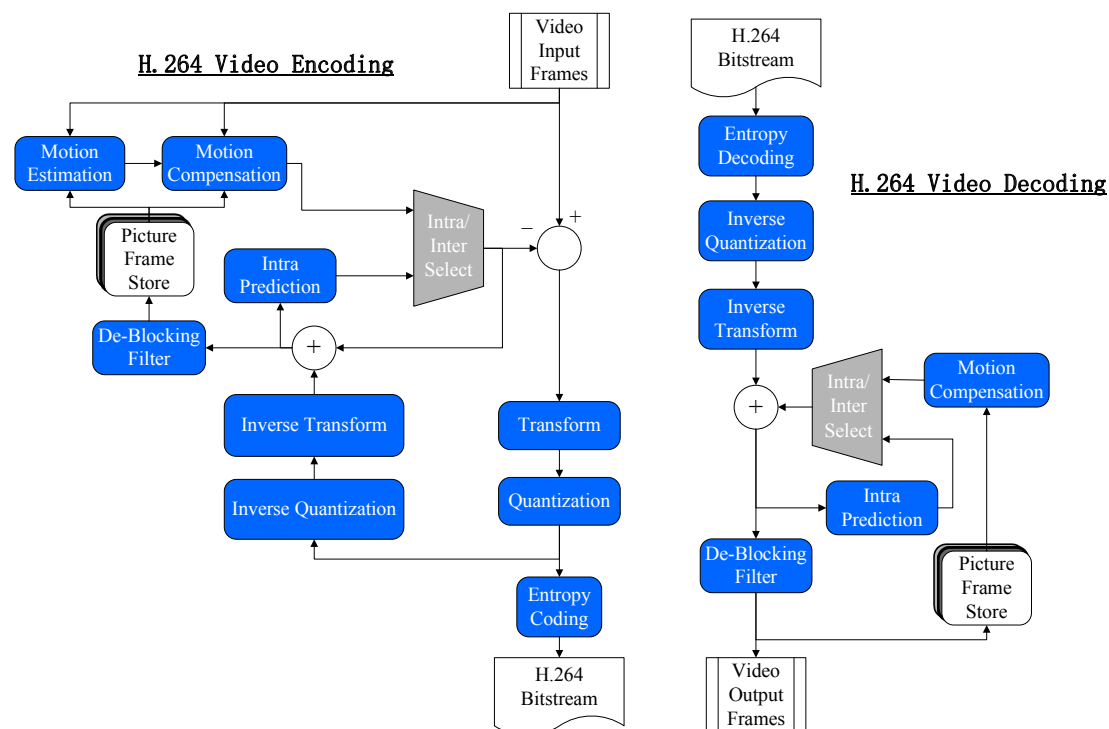


Fig.1. H.264 encoding/decoding processing

1.2.1 H.264's key features

H.264, like previous standards, is a hybrid coding mode of DPCM plus transform coding. But it uses "regression basic" simple design, without many options, to obtain much better compression performance than H.263++; strengthen the adaptability to various channels, the use of "network friendly" structure and grammar, is conducive to the processing of bit error and packet loss; the application target range is wide, to meet different rates, different Resolution and requirements for different transmission (storage) occasions.

Technically, it concentrates on the merits of past standards and absorbs the experience accumulated in standard setting. Compared with H.263 V2 (H.263+) or MPEG-4 simple classes (Simple Profile), H.264 can save up to 50% of the code rate at most digital rates when using the best encoder similar to the above coding method. H.264 can continuously provide higher video quality at all rates. H.264 can work in low delay mode to adapt to real-time communication applications (such as video conferencing), and can work well in applications without delay constraints, such as video storage and server based video streaming applications. H.264 provides the tools needed to handle packet loss in packet transport network, as well as tools for dealing with bit errors in error prone wireless networks.

At the system level, H.264 has proposed a new concept for conceptual segmentation between the Video Coding Layer (VCL) and the network extraction layer (Network Abstraction Layer, NAL). The former is the expression of the core content of the video content, and the latter is the delivery of a specific type of network. The structure facilitates information encapsulation and better priority control of information.

1.2.1.1 Entropy decoder

In the H.264, two different entropy coding methods are adopted: Universal Variable Length Coding (UVLC) and Context-based Adaptive Binary Arithmetic

Coding (CABAC).

In H.263 and other standards, different VLC codes are used according to the type of data to be coded, such as transform coefficient and motion vector. The UVLC code table in H.264 provides a simple way to use the unified variable length encoding table, no matter what type of data the symbol represents. Its advantages are simple; the disadvantage is that a single code table is derived from the probability statistical distribution model, without considering the correlation between the coded symbols, and the effect is not very good at medium high bit rate.

Therefore, an optional CABAC method is also provided in H.264. Arithmetic coding enables both sides of coding and decoding to use all probability models of syntactic elements (transform coefficients and motion vectors). In order to improve the efficiency of arithmetic coding, the basic probability model can adapt to the statistical characteristics changed with the video frames through the process of content modeling. Content modeling provides conditional probability estimation of coded symbols. Using appropriate content models, the correlation between symbols can be removed by selecting the corresponding probability models of coded symbols adjacent to the coded symbols, and different syntactic elements usually maintain different models.

1.2.1.2 Intraframe prediction

Intra coding is used to reduce spatial redundancy of images. In order to improve the intra coding efficiency of H.264, the adjacent macroblocks usually contain similar attributes in a given frame by making full use of the spatial correlation of neighboring macroblocks. Therefore, when a given macroblock is coded, it can be predicted first by the surrounding macroblock, and then the difference between the predicted value and the actual value is encoded so that the bit rate can be greatly reduced relative to the direct encoding of the frame.

H.264 provides 9 modes for 4×4 pixel macroblock prediction, including 1 DC prediction and 8 direction prediction. H.264 also supports 16×16 intra frame coding for flat areas with little spatial information in the image. Fig.2a shows the neighboring

pixels used to predict a 4×4 block. Fig.2b shows the eight prediction directions.

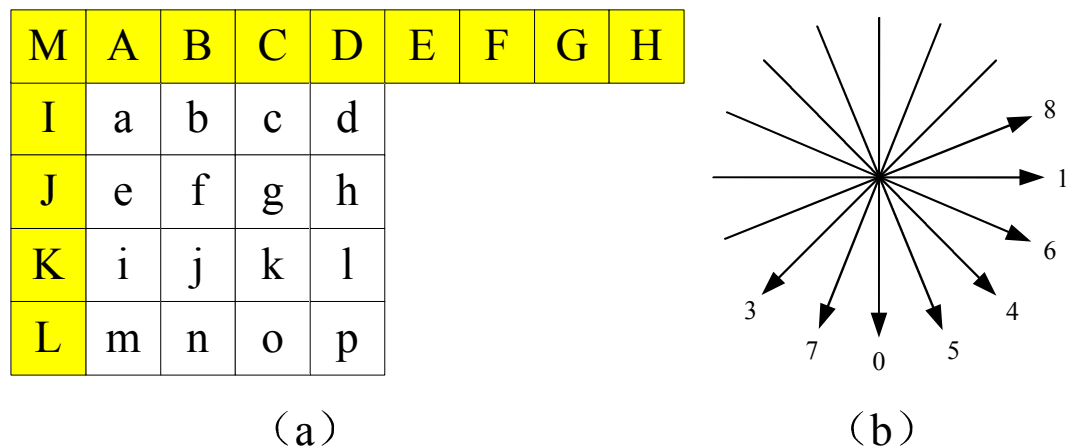


Fig.2. Intra prediction in H.264. (a) Prediction samples for a 4×4 block. (b) Directions for prediction modes for 4×4 and 8×8 blocks

1.2.1.3 Interframe prediction

Inter prediction coding uses motion redundancy and motion estimation and compensation in continuous frames. The motion compensation of H.264 supports most of the key features of the previous video coding standard, and adds more functions flexibly. In addition to supporting P frames and B frames, H.264 also supports a new inter flow transmission frame, SP frame. The bitstream contains SP frames, which can quickly switch between streams with similar contents but different code rates, and support random access and fast playback mode. The motion estimation of H.264 has the following 4 characteristics:

- 1> Macroblock segmentation of different sizes and shapes
- 2> High precision subpixel motion compensation
- 3> Multi frame prediction
- 4> Block filter

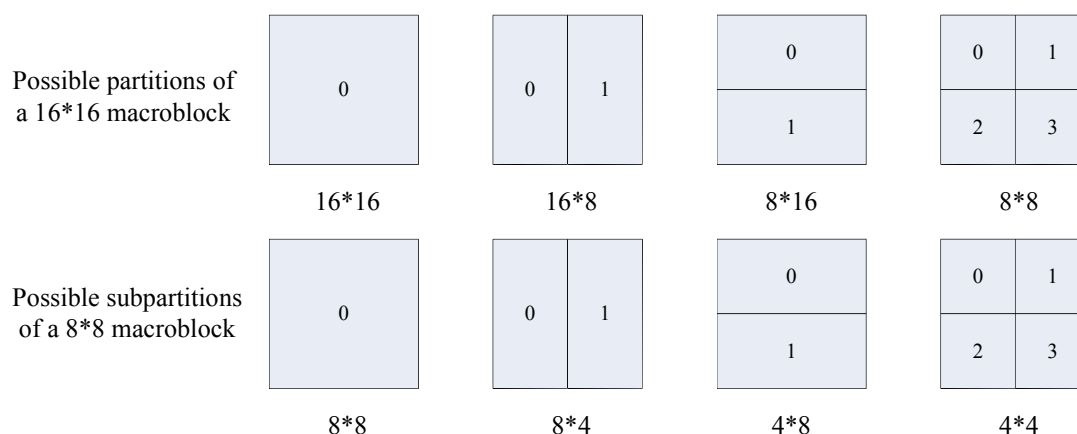


Fig.3. Sub blocks of a macroblock in inter prediction

1.2.1.4 Transform coding

In the aspect of transformation, H.264 uses a DCT based transformation based on 4×4 pixel blocks, but uses an integer based spatial transformation, and there is no inverse transformation because of the error of the trade-off. The transformation matrix is shown in Figure 5. Compared with the floating-point operation, the integer DCT transformation can cause some additional errors, but because the quantization error after the DCT transform also exists, the quantization error caused by the integer DCT transform is not much less than that. In addition, the integer DCT transform also has the advantage of reducing computation complexity and complexity, and is advantageous to the transplantation of fixed-point DSP.

1.2.1.5 Deblocking filter

H.264 defines an adaptive block removal filter, which can handle the horizontal and vertical block edges in the prediction loop, greatly reducing the cube effect.

1.3 The Frame Rate Conversion

Frame rate is used to measure the number of display frames. The so-called

measurement unit is Frames per Second (FPS) or "Hertz" (Hz) per second.

Because of the special physiological structure of human eyes, if the frame rate of the picture is above 16, it will be considered to be coherent. This phenomenon is called visual persistence. That's why movie films are captured one frame by one frame and then broadcast quickly.

Frame rate conversion is a very important technology, when we need to translate a video stream with a specific frame rate to a different frame rate. It is necessary for conversion among various display formats with different frame rates.

1.4 Multiview Video Coding (MVC)

Multiview video Coding is a new type of video technology with stereoscopic perception and interactive operation function. It is a video signal obtained by a group of parallel and converging camera arrays.

In 2001, MPEG set up a 3DTV [7] working group, whose primary task was to define the scope and application scenarios of the 3D audio and video field [8], and to set standards for the key technologies. Multi view video is a rapidly rising field of research in recent years, under the framework of 3DTV. In the application of video surveillance, multi view video coding technology helps to realize multi view stereo monitoring, multi camera linkage and other applications.

The implementation of multi-view video coding (MVC) [9] can be based on a traditional hybrid coding framework (such as H.264), or a new generation of video coding tools, such as wavelet coding and distributed coding. The correlation between points of view is an important feature of multi-view video sequence. It has a great relationship with the form of the camera array, the distance of the camera, the distance between the camera and the photographed object, and it directly reflects the parallax of the two images of the adjacent view at the same time.

Because of the large amount of data redundancy in the system, how to organize and compress data becomes an important research topic. So the current MVC focuses

on how to improve the efficiency of compression and the ability to read random, and these studies can be classified from two main aspects, one is the prediction structure, and the two is the prediction tool. The prediction tool refers to the spatial prediction methods between the multi stream view, including luminance compensation, parallax / motion compensation, 2D direct prediction, and view interpolation. Because the correlation utilization between the perspectives is the main factor that determines the efficiency of MVC compression, the further improvement of the MVC [10] [11] compression efficiency depends on the design of the new prediction tool. The prediction structure refers to the mutual prediction reference relationship between multi-view video spatio-temporal frames. It represents which frames are processed to eliminate the spatiotemporal redundancy of the data, so whether the traditional mixed coding, wavelet, or distributed coding can't be separated from the design of the prediction relationship. In addition, prediction structure is an important index for determining random read performance, fast decoding performance and network transmission cost, and has attracted wide attention in MVC research. The application of MVC code stream to the traditional flow transmission framework will produce visual angle switching, so how to design a new type of switching frame and how to analyze the influence of handoff on the prediction structure are also of great significance [12].

The main problem of the current multi-view video compression is the contradiction between the data compression efficiency and the random reading ability [13]. The multi-view video itself has a large amount of data. In the transmission application or local fast decoding, the user does not need all the data information. Therefore, the dependence of the data is small, but it is just as well as compression. It is a contradiction. Multiview video itself is a very strong correlation source, and the theory of multi-source coding based on distributed coding may be theoretically analyzed and discussed. Multi source coding can theoretically decode the trade-off between cost, random read ability and coding efficiency of MVC, so it has great potential in the field of MVC.

1.5 Depth Image Based Rendering (DIBR)

With the enhancement of processor capability and the gradual maturity of 3D display technology, the application of multi-view video will be more and more extensive. Unlike traditional two dimensional video, multi-view 3D video can allow viewers to view video content from various angles to make it more realistic and immersive. Because the number of cameras is limited, the generation of any view determined by the view point is one of the key problems in the multi-view video application, which is called the dense or rendering technology of the view point.

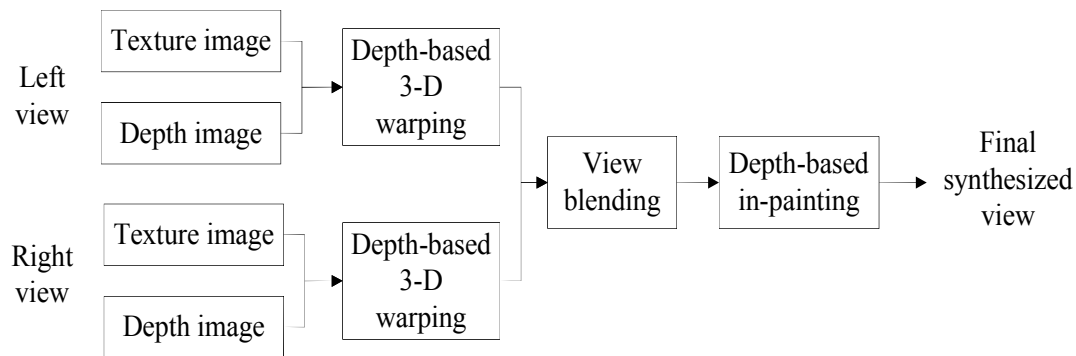


Fig.4. Depth image-based virtual view synthesis

The depth image based rendering technology developed in recent years has aroused great interest and has become a hot topic in the field of computer graphics and virtual reality. In terms of image realism and rendering speed, image rendering technology has the advantage that the traditional geometry based rendering technology is incomparable. Depth based image rendering (DIBR) is a method of image rendering which introduces depth information of the image. At present, MPEG/JVT uses "two-dimensional video + depth" 3D video representation method, and does not need to transmit multiple views, but by using the rendering technology based on depth image (DIBR) at the decoder to generate a 3D scene of one or more virtual viewpoints in real time, producing a three-dimensional visual effect. Therefore, "virtual" views are needed to render to support future displays [15]. A simple example

of depth image-based virtual view synthesis is shown in Fig.4. One main problem is that the regions occluded in the original view may be visible in the “virtual” view [16] [17]. The disadvantage of existing methods is that the high quality rendering can be done only for small gaps between two viewpoints [18] - [22].

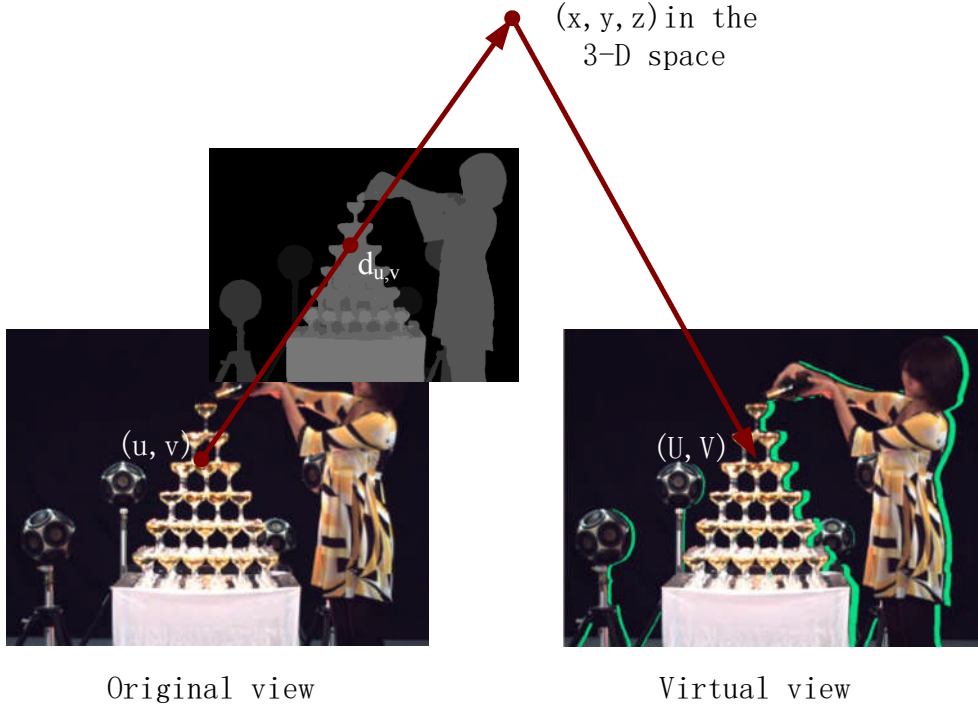


Fig.5. General concept of 3-D warping [84]

3-D warping is a key technique used in DIBR, the general concept of which is shown in Fig.5. In 3-D warping, pixels in a reference image are back-projected to the 3-D spaces, and then re-projected onto the target viewpoint. A problem is that the information in the virtual view occluded in the original view may be needed to be visible. Some holes will then appear in the virtual image, which can be also called zero-regions. The information of an occluded region in the original image is lost in the virtual image and needs to be concealed. Therefore, a hole-filling technique is necessary and in-painting is the most popular method for solving such hole-filling problems.

1.6 Stereo Matching

This topic has already developed for a long period, but developed fast and fast in the recent years, because of the demands from the development of 3D video market and computer science. A lot of relative algorithms have been proposed and it will perform an important role for many applications as virtual reality and robotic.

1.7 Frame-compatible Format (FCF)

Frame-compatible format is a spatial multiplex of two neighboring views into one single frame. Usually, it refers to sub-sampling technology for the two views and then packed the two sub-sampled views together into one frame. It is considered as one of the most promising solutions of 3D distribution on the existing system, as it is completely compatible for the existing video codec, such as H.264/MPEG-4 AVC and MPEG-2.

Frame-compatible format enables the stereo video compressed with existing encoders, transmitted through existing channels and decoded by existing receivers and players, with minimal changes. Such advantages will benefit the stereo video being quickly deployed to the already 3D market.

Consequently, this format has achieved a wide approval. For example, it has been included by H.264/MPEG-4 AVC as Supplemental Enhancement Information (SEI) [74], and High Definition Media Interface (HDMI) specification (v1.4) [75] has announced their services will entirely support this format. As the SEI message has been specified in the latest version of the AVC standard support, this format is expected to be used throughout the delivery chain from production to distribution, through the receiving devices, and all the way to display.

1.8 Our contribution

1.8.1 Adaptive Low Computational Complexity Algorithms in Video Decoding Process

The surveillance system becomes a more and more important tool in our daily life to protect residents from crime and to improve the public safety. They are already widely used in public access facilities such as airports, transport stations and banks to provide security. By considering the space for storing the video sequence and the communication bandwidth, the single layer format is used for the surveillance system generally. Nowadays the surveillance video can be received by the different terminal devices based on the good network environment. And the different terminal devices have different computational ability and different type of power supplying. By considering the various situations of these devices, it requires a hierarchical decoding process to make the single layer bit stream sources much more flexible, provide some choices of the bit stream format for the different situation of the terminals and realize the low complexity and low power decoding. Comparing with the original multi layer bit stream transmission, it can also reduce the transmission bandwidth.

The first proposed method is a temporal scalable decoding process with frame rate conversion method for surveillance video. This method can be used to reduce the computational complexity in the decoding process and keep the video quality at the same time, and make the single layer bit stream sources much more flexible for various terminal devices. It is realized based on frame-skipping conception with the proposed reference frame index decision algorithm, motion vector composition algorithm and block-partition mode decision algorithm. Compare with the frame rate conversion in transcoding process, it is much lower complexity and more flexible. Through the experimental results, the reduction of computational complexity

(decoding time) depends on the number of skipped frames, the more frames was skipped the more reduction of the computational complexity will be got. The PSNR loss is very small (about 0.1 ~ 0.2 (dB)) for B frame skipping. And the PSNR loss is about 0.7 ~ 2 (dB) (the loss of SSIM is only 0.002~0.007) for 2/3 P frame skipping and reduce the computational complexity about 60%.

The second proposed idea is the adaptive frame rate conversion in the temporal scalable decoding process for surveillance video. In the first proposed ideas, the certain frame skipping scheme is used. It is easy to control the frame rate and compare the result, and the PSNR loss is a little high. But sometimes the content of the surveillance video is still or the objects in the video only have some small motion. If can avoid to skip the frames which have a lot of moving objects and skip the still frames or the frames which only have slow or small moving objects, the PSNR will be improved a lot. This idea will be introduced and the work is not finished, the result will be estimated.

As an extension work, the third proposed idea is the adaptive decoding process with temporal prediction method for common video. It uses the temporal prediction to choose the lowest cost frame to skip. It is much more flexible than the previous method, and can deal with more complexity situations. So it not only can be used for the surveillance video, but also can be used in the common video cases. Through the experimental result, the proposed adaptive low power decoding process with temporal prediction method for common video improves the video quality a lot that compares with the previous temporal scalable decoding process with frame rate conversion method. The rate of the decoding time reduction (power consumption reduction) is always related to the number of the skipped frames. The more frames are skipped the more decoding time reduction will be got. In the sequences which only have slight movements, the PSNR is only improved a little about 0.01 – 0.08 dB. But it is improved obviously about 1.9 – 2.4 dB in the sequences which have strong movements.

1.8.2 Frame Compatible Format Fast Encoder with Stereo

Matching

The reconstruction of a stereo scene from a pair of images taken from different directions is one of the classical problems in computer vision area. Nowadays, along with the huge development of computer performance and 3D video market, stereo matching plays an important role in the depth generation. Besides, Frame-compatible format (FCF) is considered as one of the most promising solutions of 3D distribution on the existing system, as it is completely compatible for the existing video codec, such as H.264/MPEG-4 AVC and MPEG-2. Researches have showed that content similarity in two packed views in FCF can lead to prediction correlation in encoding. This feature is used when researchers design the fast encoder for Frame Compatible Format.

The previous work of fast encoder for FCF uses a simple block based shift obtaining method, which cannot figure out the accurate content correlation. The stereo matching algorithm is definitely the best choice to reveal the content matching in the packed view, which is quite suitable to be used in this research. The difficulty is that stereo matching algorithm usually costs lot of time and computational resource. If it needs too much time to calculate the depth map, the fast algorithm becomes useless.

Traditionally, the process of local stereo matching is divided as cost computation, support aggregation and disparity refinement. In the proposed method, the process of traditional matching algorithm is rearranged. The proposed algorithm reduces computational complexity and ensures the accuracy of the result at the same time. Through the experimental results, a better FCF fast encoder is proposed by implementing the stereo marching algorithm into it. The proposed novel frame compatible format fast encoder can achieve both less PSNR loss and the lower bit-rate increment. It can also reduce the data transmission cost a lot. Besides, no one has combined stereo matching into FCF fast encoder research currently. This research shows the promising possibility of this field.

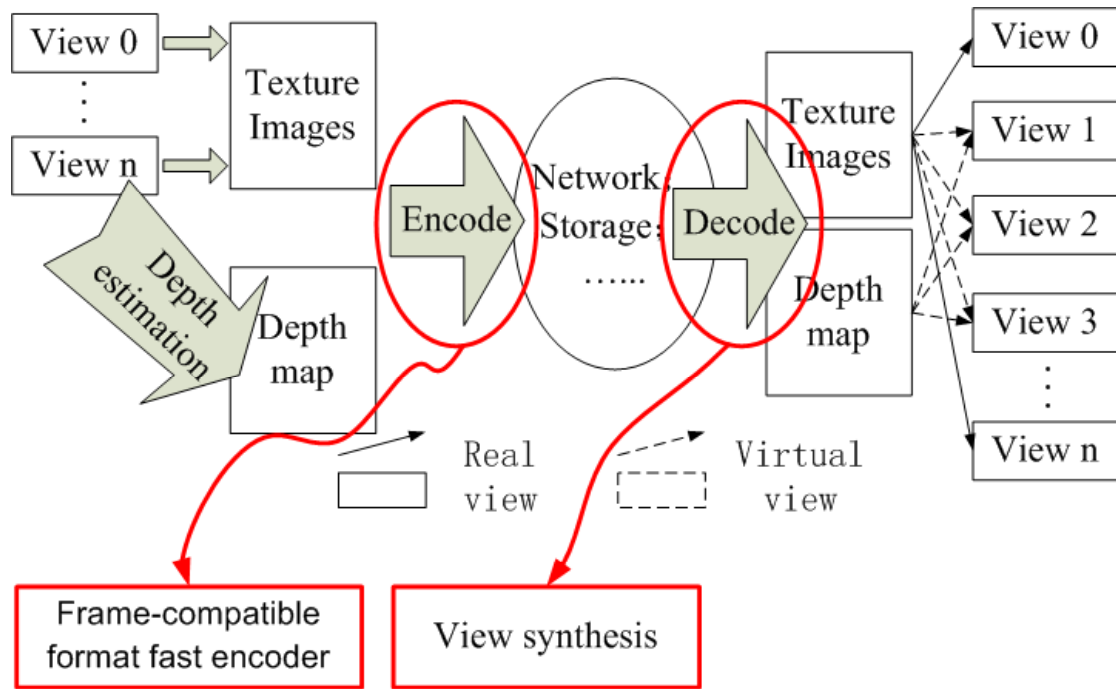


Fig.6. The relationship between the Frame-compatible format fast encoder and View synthesis

The Fig.6 shows the relationship between my two proposals in the 3-D field. The Frame-compatible format fast encoder with stereo matching works in the encoding side and the novel hole-filling algorithms for view synthesis which will introduced in the next part works in the decoding side.

1.8.3 The Novel Hole-filling Algorithms for View Synthesis

In recent years, three-dimensional (3-D) products have become increasingly popular in daily life. In most traditional 3-D multimedia systems, only one pre-determined viewpoint of an image or video can be seen by an observer. If the viewpoint is changed, the realistic 3-D impression will become much weaker and the quality of the 3-D video will worsen. To increase the number of viewpoints for the observers and image more comfortable for viewers, the free viewpoint television (FTV) [23] – [27] was introduced. The interest in FTV has been continuously increasing in recent years. Auto-stereoscopic displays provide a 3-D impression to an

observer without the need to wear additional glasses [28], and the observers can enjoy a realistic 3-D impression from certain different viewpoints. Such a display shows a number of slightly different views at the same time. To simultaneously deliver so many views, an extremely large amount of bandwidth is required.

Therefore, view synthesis [29] was introduced to solve this problem. Depth-image-based rendering (DIBR) [30] is a technology for synthesizing novel realistic images at a slightly different view perspective using a textured image and its associated depth values. DIBR is used to generate additional virtual views of a real-world scene from an image or video, as well as the associated per-pixel depth information. 3-D warping is a key technique used in DIBR. In 3-D warping, pixels in a reference image are back-projected to 3-D spaces, and then re-projected onto the target viewpoint. An inherent problem of the view synthesis concept is that image information occluded in the original view may become visible in the “virtual” image. Some holes will then appear in the virtual image, which can be also called zero-regions. The information of an occluded region in the original image is lost in the virtual image and needs to be concealed. Therefore, a hole-filling technique is necessary and in-painting is the most popular method for solving such hole-filling problems.

The first proposed algorithm is an integrated hole-filling algorithm for view synthesis. The proposed algorithm includes five parts: an algorithm for distinguishing different regions, foreground and background boundary detection, textural and structural isophote detection, a texture image isophote prediction algorithm, and an in-painting algorithm with gradient priority order. An isophote is the boundary between two regions in different layers or two regions with much different luma and chroma information, and intersects with the zero-region from the background. The proposed method ensures the boundary of the foreground objects by distinguishing the different layers in the texture image with depth information, predicts the textural and structural information in the zero-region using the geometry principle, and paints in the zero-region in a priority order based on the gradient information. Textural and structural information is very important in the 3-D cases, and has a significant effect

on the 3-D impression in human vision. Therefore, the proposed method will improve not only the objective quality of a synthesized virtual view but also its subjective quality and the 3-D performance for human vision. In addition to multi-viewpoint cases, the proposed method can also be used in the conversion of 2-D video into 3-D video. Based on the experimental results, the objective and subjective qualities of an image are improved considerably using the proposed algorithm. Through a detailed comparison, the proposed algorithm was shown to ensure the boundary contours of foreground objects as well as the structural information in the zero-region of a virtual image much clearly than the previous methods.

The second proposed algorithm is an extension work of the first one. It combined the temporal information together to increase the prediction accuracy. It combines the temporal prediction and the spatial prediction together. In some of the cases, the camera nearly doesn't move at all. The prediction based on the temporal information is much more reliable and stable than the spatial prediction in this kind of situations. However, in the camera moving cases, the spatial prediction is much better than the temporal prediction. Therefore, the combined realization can obtain the advantages from both of the spatial and temporal prediction method. It obtained the advantages from both of the prediction methods. Therefore, the performance is much stable than the previous works not only in the camera fixed cases, but also in the camera moving cases. The experimental results showed that both of the objective quality and the subjective quality are much better than the other methods in most of the cases.

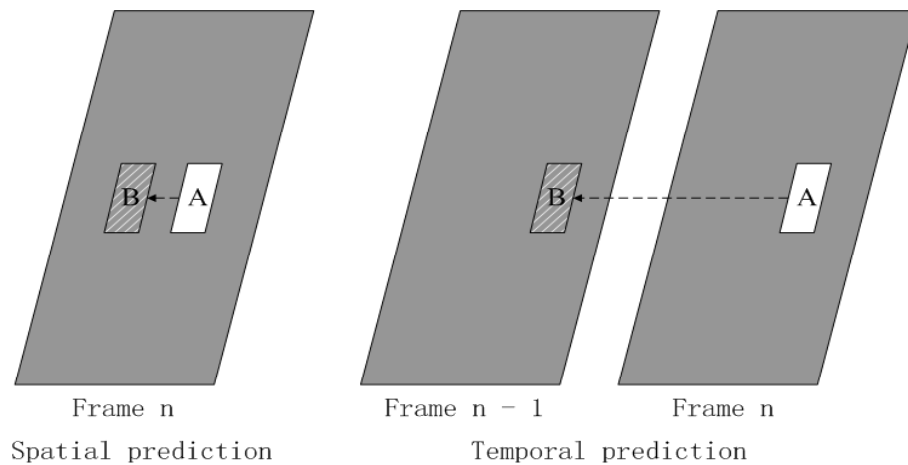


Fig.7. The difference between the spatial and temporal prediction

Fig.7 shows the difference between the spatial and temporal prediction. The spatial prediction refers the position which is in the same frame, and the temporal prediction refers the position which is in the previous frames.

1.9 Dissertation Organization

The rest of this dissertation is organized as follows. Chapter 2 presents adaptive low computational complexity algorithms in video decoding process. Chapter 3 introduces the frame compatible format fast encoder with stereo matching. Chapter 4 presents the novel algorithms with low transmission bandwidth in view synthesis. Chapter 5 concludes the whole dissertation.

2. Adaptive Low Computational Complexity

Algorithms in Video Decoding Process

2.1 Introduction

The surveillance system becomes a more and more important tool in our daily life to protect residents from crime and to improve the public safety [31] – [33]. They are already widely used in public access facilities such as airports, transport stations and banks to provide security. Considering the storage space and communication bandwidth, the single layer video format is always used in surveillance systems. The surveillance video can be received by the different terminal devices based on the good network environment. The different terminal devices have different computational ability and different power supply. Therefore, a hierarchical decoding process is required to make the single layer bit stream sources much more flexible to realize the low complexity and low power decoding, and to provide some choices of the bit stream format for the different situation of the terminals.

Scalable video coding [34] – [37] can realize hierarchical encoding and decoding. However, it is hard to implement in surveillance systems due to its much slower encoding speed and more than 30% extra storage space cost. This is a fatal demerit in some storage limited cases.

Lots of works have been published about the frame rate conversion in the encoder [38] – [41] and transcoder [42] – [44], while few are proposed in decoder. The encoding process just provides a certain frame rate for the terminal devices, especially in the surveillance system. There is only a single layer bit stream which can be received by the terminal devices without any other choices. If the computation ability of the terminal devices is not enough, the single layer bit stream may not be decoded in real time, and it is a big problem for the terminals. The transcoding process should decode some parts of the single layer bit stream at least, and then encode some parts

of it again in accordance with the new requirements. Obviously it is not convenient for the terminals, and it performs both some parts of the decoding process and some parts of the encoding process for one more time. Therefore, it requires longer computation time, more power consumption and more coding equipments.

2.2 Previous works

In [1], a set of specific application instructions is introduced to accelerate the motion compensation frame rate conversion (MC-FRC) algorithm based on block motion estimation (BME). The proposed instruction set is universal and can support many MC-FRC algorithms. The instructions are described more precisely how to perform acceleration. The acceleration can achieve real-time performance of video stream.

In [2], a frame rate interpolation algorithm for multimedia is proposed. Motion vector estimation is based on joint motion estimation and space time smoothing and refinement. The error MV is corrected based on the consistency of the MV field in the spatial domain and the temporal domain. An adaptive motion compensation interpolation method is proposed to reduce the shadow block in the middle frame, especially in the boundary region.

In [3] [4], frame rate conversion is an effective method to solve the quality degradation of moving images in liquid crystal display. It is also suitable for low frame rate video signals such as movies. Basic motion compensation results in gaps between coverage area and uncovered area. In order to avoid the gap, a motion compensation method based on target location is proposed. Through it, full motion compensation is possible and motion estimation is performed through block activity normalization. They also propose to use four frames adaptive interpolation. Finally, the original image is compared with the converted image after the image is extracted.

In [5], it indicates that frame rate conversion is essential for video data exchange between different industries and applications. Motion compensation interpolation

(MCI) has been proven to provide good frame rate conversion results. The main drawback of this method is the blocking artifact at the boundary of the interpolation block. A low complexity method (MFI) is proposed to convert block level motion field into pixel level motion field.

In [6], a frame rate up conversion algorithm based on weighted adaptive motion compensation interpolation (WAMCI) is proposed, which reduces blocking and facts based on block processing. The proposed method is based on the weighting of interpolation schemes and the results of multiple interpolation filters. In addition, by applying overlapping block motion compensation (OBMC) technology, the block effect on the block boundary is reduced. In order to reduce the shortcoming of overlapping processing, the motion analysis method is used to determine the motion type and the WAMCI is applied adaptively.

But up to now, most of the previous works are the frame rate up conversion. Our work is a frame rate down conversion in the decoding process. It doesn't need any additional devices and can provide some different decoding modes to the terminal users. It can reduce the transmission bandwidth comparing with the original multi layer bit stream transmission.

I can also show something to prove that the proposed low complexity algorithm is necessary. Some people done the experiments and show the results on the internet:

Experimental environment:

They use pad with four cores (with 1.2Ghz, 1GB DDR3) to do the experiments.

The experimental result:

- a) Most of the 720p video can be played in real time
- b) Some of the 1080p video can't be played in real time
- c) All of the 2160p video can't be played in real time

We can find that the low complexity decoding process is necessary for this kind of cases.

2.3 Temporal scalable decoding process with frame rate conversion method for surveillance video

2.3.1 Frame rate conversion method in the proposed process

In the proposed process, the frame rate conversion is based on the frame-skipping conception which is a idea to skip the decoding process or encoding process of some frames to achieve the goals such as reduce the data for transmission, reduce the computation complexity, and etc.

There are two main kinds of frame-skipping in the decoding process are proposed in this paper:

1> The B frame-skipping (main and extended profile)

The B frame is a kind of bi-directional predict-frame. It is predicted by the reference frames in the forward and the backward direction, but it is not as the reference frame for any other frames. So it can be skipped without any decreasing to the quality of other frames.

2> The P frame-skipping (baseline, main and extended profile)

The P frame may be the reference frame for other P frames or B frames. So the P frame-skipping may cause the quality loss of other frames.

The Fig.8 shows the traditional solution for the multi-layers bit-stream transmission. In the traditional way, if the users want multi-layers bit-stream to use, then multi-layers bit-stream has to be transmitted. The Fig.9 shows the proposed frame rate down conversion algorithm. It is easy to find that the users can obtain multi-layers bit-stream by just transmitting single layer bit-stream with proposed frame rate down conversion algorithm.

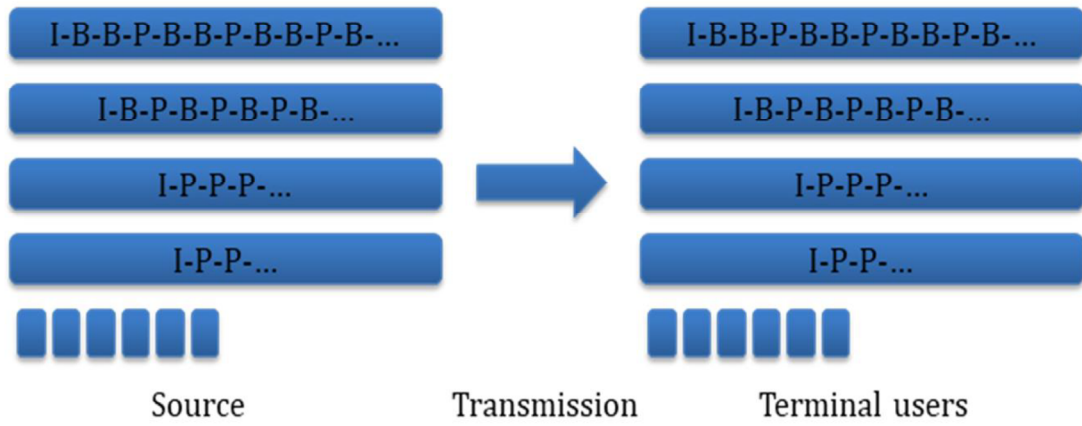


Fig.8. The traditional solution for the multi-layers bit-stream transmission

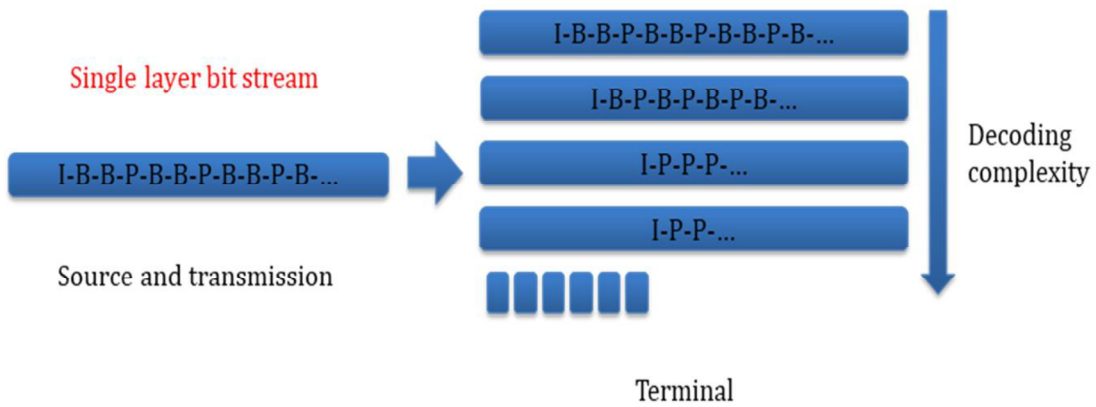


Fig.9. The proposed frame rate down conversion algorithm

The Traditional solution needs multi layers bit-stream source and large transmission bandwidth. In recent years, the temporal scalable video coding is introduced to solve this problem. It uses the single layer bit-stream source, but about 10% to 30% more transmission bandwidth is required. The proposed method uses single layer bit-stream source, no extra transmission bandwidth overhead, can reduce the decoding consumption and can also be combined with the SVC

2.3.2 Temporal scalable decoding process

2.3.2.1 The temporal scalable decoding process flow and system architecture

This proposed process combines the B frame skipping process and the P frame skipping process in one system. (As shown in Fig.10)

The B frame skipping and P frame skipping can also convert the frame rate and reduce the decoding complexity. But as the previous description, the B frame is not as the reference frame and is skipped with less video quality loss. So the combined process considers the B frame skipping at first.

In this process, when the decoder read a new picture, check the skipping scheme module at first. This module is used to decide which frames need be skipped and the skipping scheme is depended on the situation of the terminal devices or the needs of the terminal.

Then decide the frame type. If it is a B frame, enter the B frame skipping module. Check the skipping scheme. If it is in the skipping scheme, implement the B frame skipping (The result is shown in Fig.11), end the picture initialization, exit current picture and read the next frame. If it is a P frame in the previous step, enter the P frame skipping module. It is also need check the skipping scheme. If this P frame need be skipped, go into P frame skipping part (The result is shown in Fig.12). The macroblock is the basic unit to do the decoding, so read a macroblock at first in this part and only read the motion vector information from the bit stream without the residual texture information and the next decoding process, and decide the block-partition mode. If it is not in the skipping scheme, then decide whether the reference frame was skipped. If not, decode it directly. If yes, save the motion vector information for the motion vector predictor (In order to avoid the following refinement affect the motion vector predictor for other macroblocks), refine the reference frame index for current macroblock and do the motion vector composition

between the motion vector of current block and the motion vector of the block that is the nearest one which the motion vector of current macroblock points to in the skipped reference frame, then decode it (The details of the P frame skipping part will be described in the next Section.). If it is IDR or I frame, do the decoding process as usual. Obviously in this proposed method, the residual texture information of the skipped frames was lost. But in the same reason, the computational complexity, the time consumption and the frame rate was reduced at the same time.

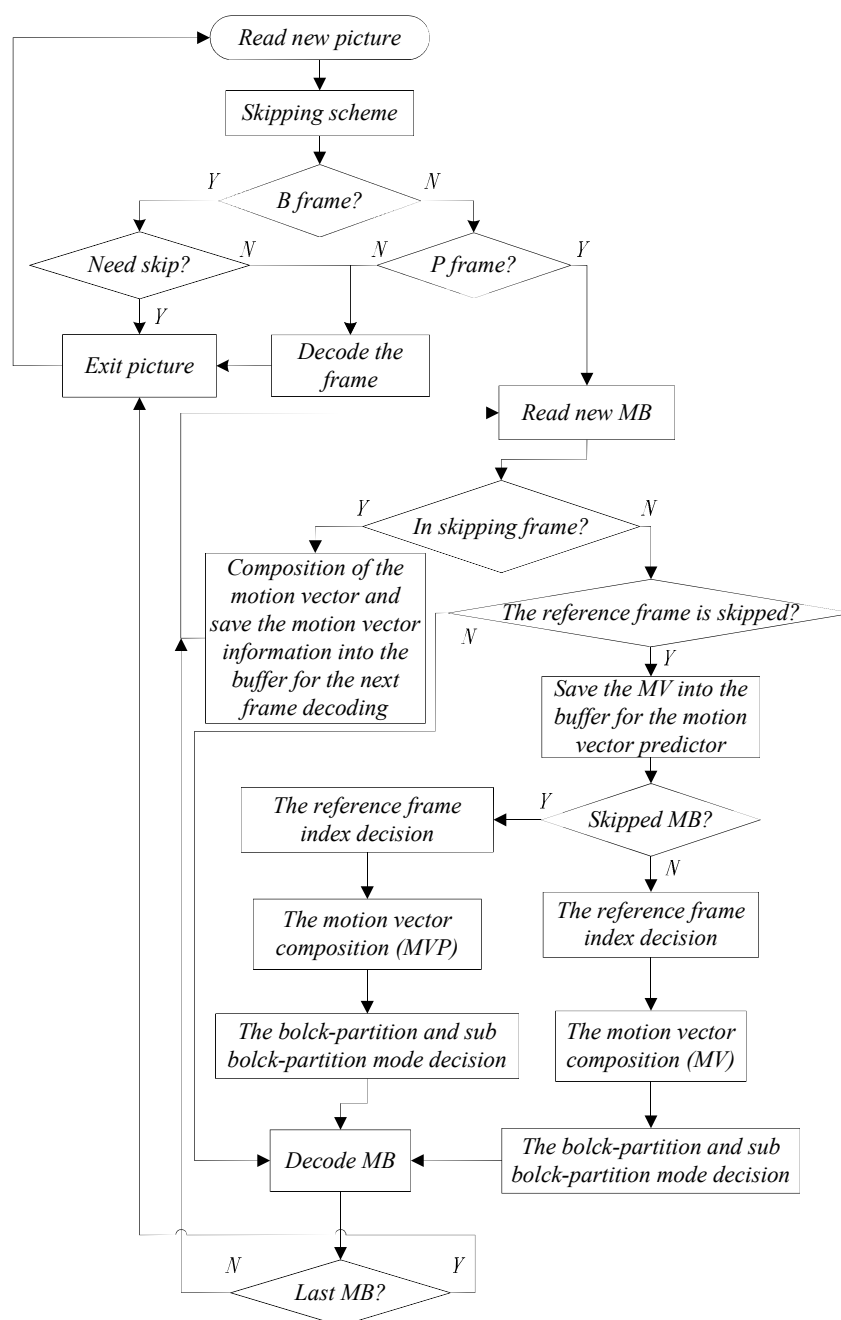


Fig.10. Temporal scalable decoding process

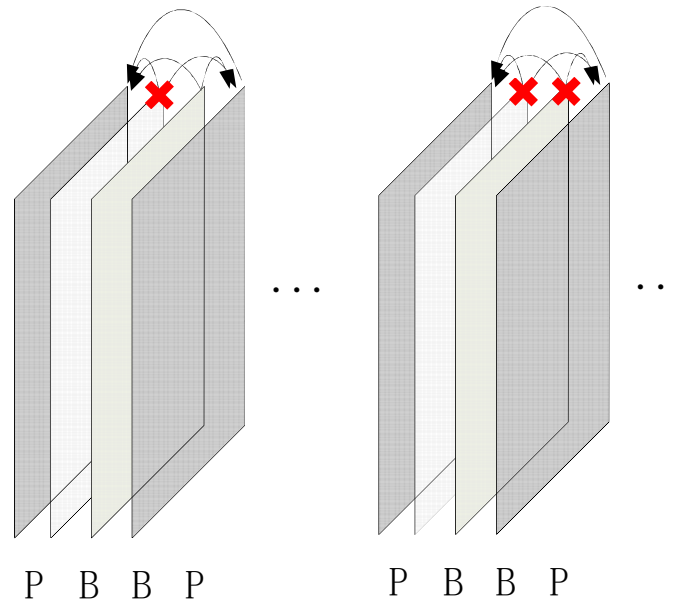


Fig.11. B frame skipping illustration

2.3.2.2 The proposed method in the P frame-skipping process

The P frame-skipping process can be divided into three main parts: the reference frame index decision part, the motion vector composition part and the block-partition decision part. There are three algorithms in these parts are proposed in the P frame-skipping process:

- 1> The reference frame index decision algorithm.
- 2> The motion vector composition algorithm.
- 3> The block-partition mode decision algorithm.

2.3.2.3 The reference frame index decision algorithm

When one of P frame which may be reference frame of next frames was skipped, it will not be stored into the storable picture buffer and will not in the reference frame list. But the motion vector of the macroblocks in the next frame still point to that position which has already been illegal. So the reference frame index of these macroblocks should be refined.

In my proposed method, it will choose the nearest forward frame which is not skipped as the reference frame. A reference frame buffer is added into the process for

the reference index decision. And the decision process obeys the rule, which is as following:

$$\text{Reference frame} = \sum_{i=0}^n S_i \text{Frame}_i$$

When

$$\begin{cases} S_i = 1 & \text{MIN}\{E_i(\text{abs}(\text{Frame}_{\text{num}_i} - \text{Frame}_{\text{num}_n}))\} \quad (i = 0, 1, 2, 3 \dots n) \\ S_i = 0 & \text{Other} \end{cases}$$

When

$$\begin{cases} E_i = 1 & \text{Frame}_i \text{ is existent} \\ E_i = \infty & \text{Other} \end{cases}$$

The Frame_num is the serial number of I frame or P frames. And the MIN function is to get the minimum one of the formula in it. The E is to ensure the checked frame is existent, and S is to check which frame is the nearest existent frame from the current frame.

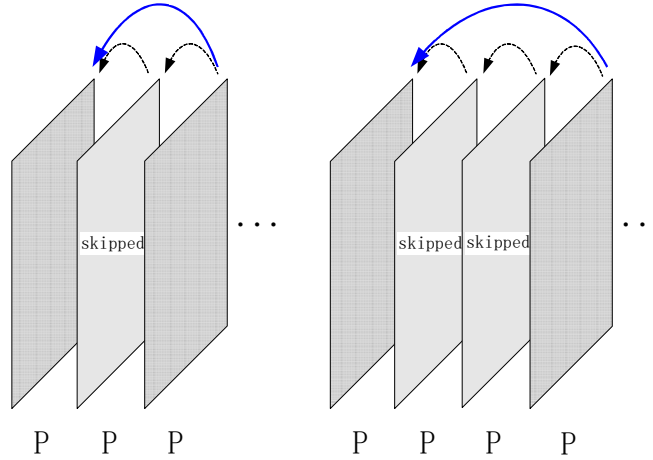


Fig.12. P frame skipping illustration

The reference frame index for deblocking decisions need be refined at the same time. Sometimes it can get satisfying result after this refinement without any motion vector composition and the block-partition decision. It often occurs in the case which is skip-macroblock both in the current frame and the skipped frame.

2.3.2.4 The motion vector composition algorithm (for refined reference frame)

The precision of the motion vector value is quarter pixel, but the quarter pixel interpolation can't be done in the skipped frames, so the approximate motion vector composition is needed between the motion vector of the macroblock in current frame and the motion vector of the macroblock in the skipped frame.

(As shown in Fig.13) The proposed composition method is similar with some was proposed in the video transcoding process[45], but the 4×4 blocks are used as the basic units in my proposed composition process.

It calculates the Δ block (the offset use block as unit) based on the motion vector value of the 4×4 block in current frame at first, finds the four overlap blocks in the skipped frame and ensure the exact position of the block by using the difference between the motion vector value of the 4×4 block and the Δ block(turn to quarter pixel precision). Then chose the maximum overlap block as the candidate block , and use the motion vector of that block to do the motion vector composition and get the refined motion vector which point to the refined reference frame.

In Fig.4, the 4×4 BLOCK 5, 6, 8, 9 are the four overlap blocks of the $BLOCK'_1$ and the $BLOCK_2$ is the maximum overlap block of the $BLOCK'_1$. The black real lines are the original motion vectors and the break lines are the refined motion vectors. After the motion vector composition, the $BLOCK'_2$ is got as the reference position instead the original one $BLOCK''_1$ in the refined reference frame.

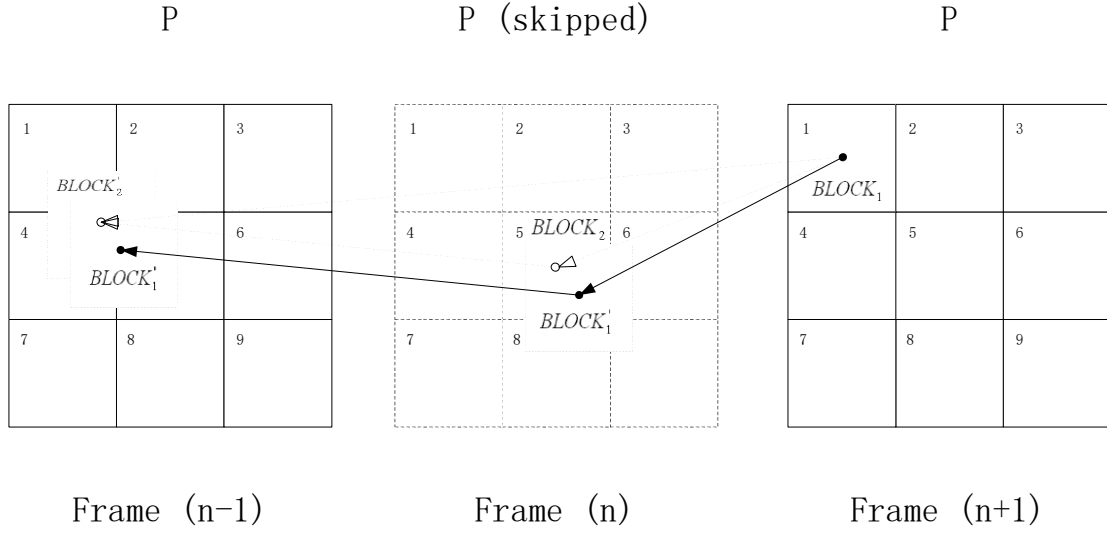


Fig.13. Motion vector composition (Illustration)

The motion vector composition process can be described as the following functions:

$$MV_X = \begin{cases} MV_X(BLOCK_1 \rightarrow BLOCK_1') + MV_X(BLOCK_1' \rightarrow BLOCK_1'') & \text{Original} \\ MV_X(BLOCK_1 \rightarrow BLOCK_2) + MV_X(BLOCK_2 \rightarrow BLOCK_2') & \text{Refined} \end{cases}$$

$$MV_Y = \begin{cases} MV_Y(BLOCK_1 \rightarrow BLOCK_1') + MV_Y(BLOCK_1' \rightarrow BLOCK_1'') & \text{Original} \\ MV_Y(BLOCK_1 \rightarrow BLOCK_2) + MV_Y(BLOCK_2 \rightarrow BLOCK_2') & \text{Refined} \end{cases}$$

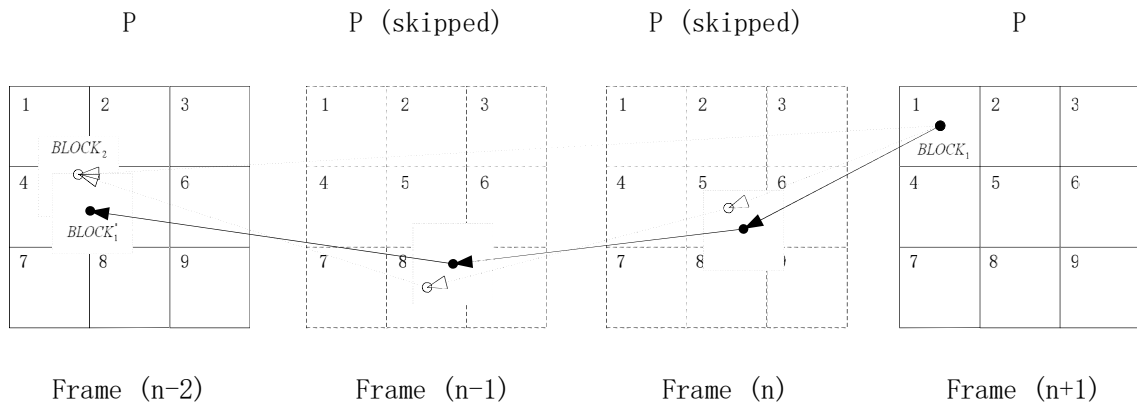


Fig.14. Multi-step motion vector composition (Illustration)

If there are more than one frame that are skipped between two frames. The basic motion vector composition method is the same as the previous one, but include more steps (multi-step motion vector composition) (As shown in Fig.14) and do the motion vector composition before it is saved into the motion vector buffer for the next frame decoding (As shown in Fig.10).

2.3.2.5 The block-partition mode decision algorithm

After finished the previous two parts, there are different motion vector composition processes for different block-partition and sub block-partition modes in the rest decoding process. If the block-partition or sub block-partition mode is not suitable for the current macroblock, some of the motion vector information may be ignored, and the quality of the decoded picture will decrease. So decide the block-partition exactly is the guarantee to decode the picture correctly and keep the video quality.

There are four types of the block-partition modes and four types of the sub block-partition modes. (As shown in Fig.15)

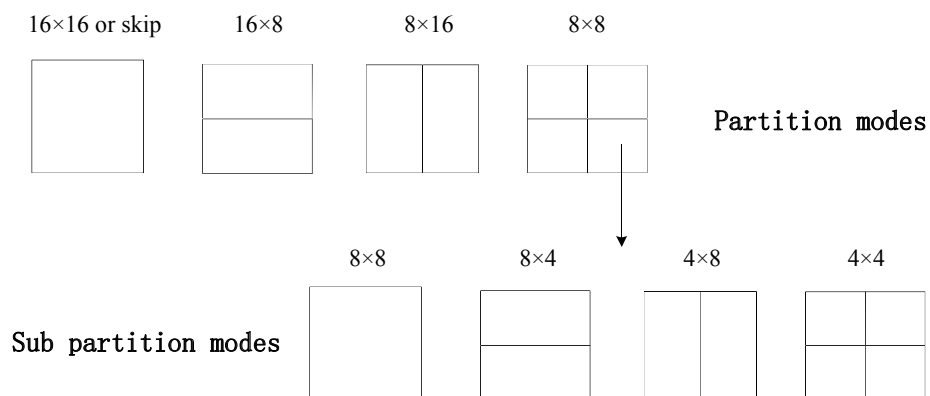


Fig.15. Block-partition and sub block-partition modes

In the proposed process, the 4×4 block is used as the unit in the motion

composition part. And the block-partition mode decision algorithm is based on the relation decision among the motion vectors of the neighbor blocks in one macroblock. So the 4×4 block is also considered as the unit in the block-partition and sub block-partition part.

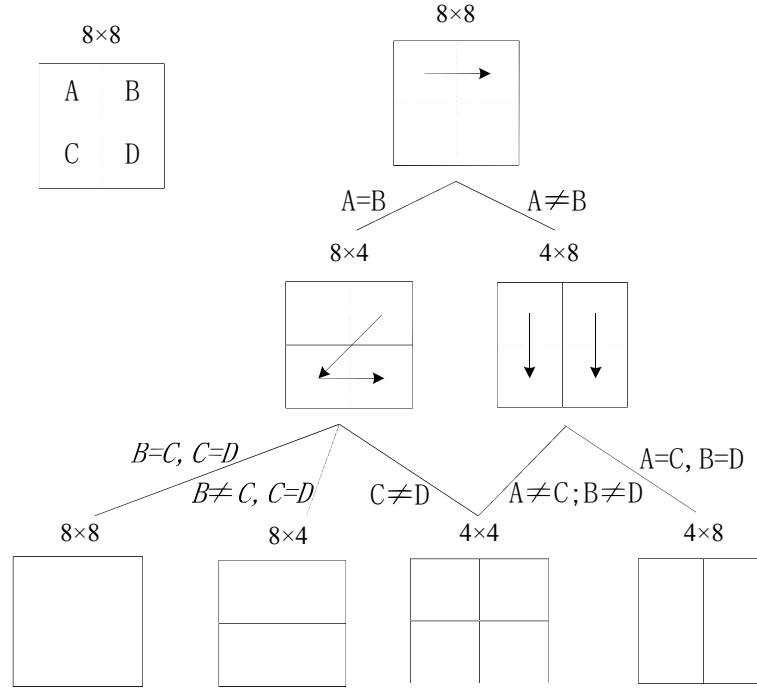


Fig.16. Sub block-partition mode decision flow

The sub block-partition mode decision flow and block-partition mode decision flow are shown in the Fig.16 and Fig.17. The arrows in the figures describe the order of the motion vector comparison among the 4×4 blocks and among the 8×8 blocks. And the A, B, C, D is the motion vector value of each 4×4 block or 8×8 block in different position.

The sub block-partition decision has to be done at first in the proposed mode decision algorithm. Compare the motion vector value of the 4×4 blocks, and decide the sub block-partition mode step by step. (As shown in Fig.16) If there is at least one sub block which is not 8×8 sub block-partition mode in one macroblock, the block-partition mode of this macroblock must be the 8×8 block partition mode. (As shown in Fig.17)

If all of the sub blocks are 8×8 blocks after the sub block-partition mode decision, then go on the block-partition process that integrate the 8×8 blocks which have the same value of the motion vector to be together and decide the block-partition mode of the macroblock step by step by comparing the motion vector value of the 8×8 block following the arrows. (As shown in Fig.17)

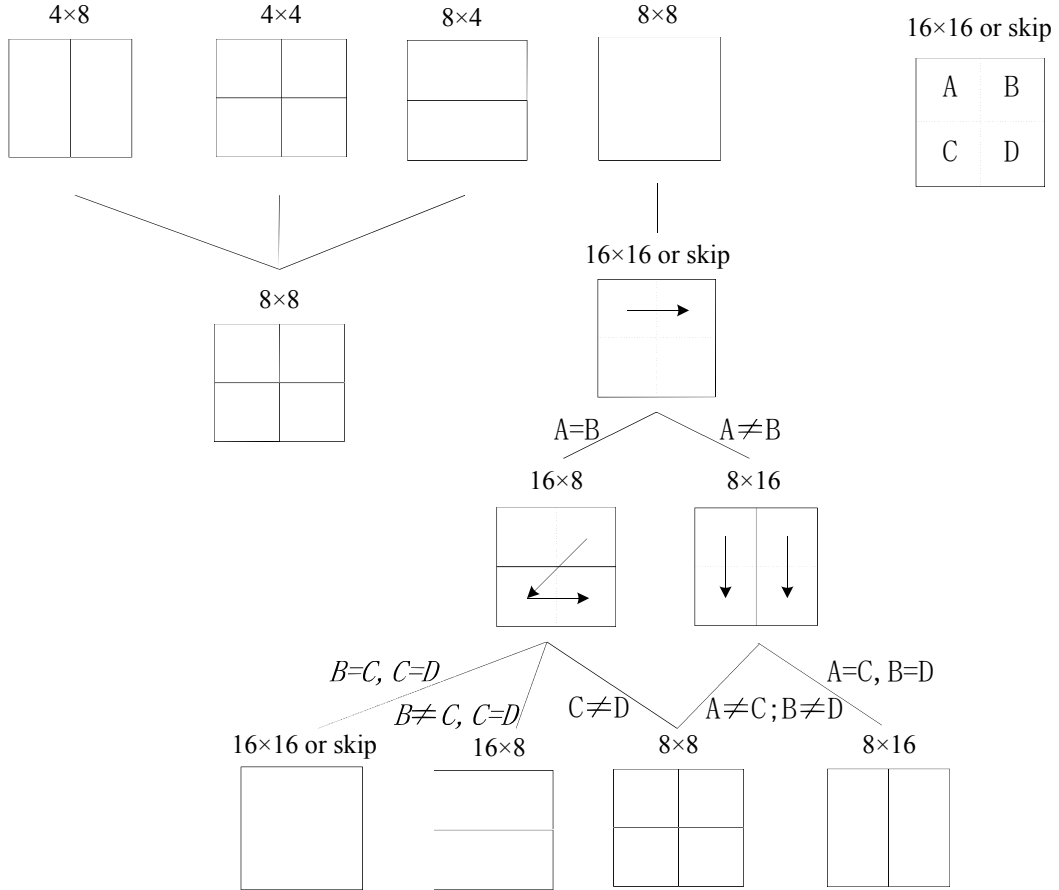


Fig.17. Block-partition mode decision flow

2.3.3 Experimental result

The experimental sources and parameters are shown in Table 1. And the result of proposed method which is tested by using JM 15.1 reference software is shown in the Table 2 (With B frame), Table 3 (Without B frame) and Table 4 (Comparison result between I-P-P-P... and I-B-B-P-B... with the same frame rate). [46] (QVGA is

320×256, CIF is 352×288)

Table 1 Experimental sources and parameters:

Sequence name	Format	FramesToBe Encoded	Intra Period	Qp
StoppedVehicle.yuv	QVGA	200	25	28
Intersection.yuv	CIF	200	25	28
Sit_handover_bag.yuv	CIF	200	25	28
Sit_leave_bag.yuv	CIF	200	25	28

Table 2 Experimental result: (I-B-B-P-B-B-P...)

Sequence name	Skipped Frames	Frame rate	PSNR	Time reduction
StoppedVehicle.yuv	No skipping	25	38.42(dB)	0
	Half B frame skipping	17	38.37(dB)	33.5%
	All B frame skipping	9	38.34(dB)	66.2%
	Half P frame skipping	5	37.94(dB)	82.0%
Intersection.yuv	No skipping	25	37.98(dB)	0
	Half B frame skipping	17	37.88(dB)	34.6%
	All B frame skipping	9	37.82(dB)	69.1%
	Half P frame skipping	5	36.97(dB)	77.9%
Sit_handover_bag.yuv	No skipping	25	38.79(dB)	0
	Half B frame skipping	17	38.69(dB)	35.8%
	All B frame skipping	9	38.61(dB)	71.5%
	Half P frame skipping	5	37.85(dB)	80.5%
Sit_leave_bag.yuv	No skipping	25	38.55(dB)	0
	Half B frame skipping	17	38.54(dB)	32.1%
	All B frame skipping	9	38.52(dB)	65.0%
	Half P frame skipping	5	37.67(dB)	72.2%

Table 3 Experimental result: (I-P-P-P-P...)

Sequence name	Skipped Frames	Frame rate	PSNR	Δ PSNR	SSIM	Δ SSIM	Time reduction
Stopped Vehicle.yuv	No skipping	25	37.90 (dB)	0	0.9625	0	0
	1/3 P frame skipping	17	37.49 (dB)	-0.41	0.9612	-0.0013	24.5%
	1/2 P frame skipping	13	37.36 (dB)	-0.54	0.9607	-0.0018	40.3%
	2/3 P frame skipping	9	37.23 (dB)	-0.67	0.9603	-0.0022	60.7%
Intersecti on.yuv	No skipping	25	37.35 (dB)	0	0.9592	0	0
	1/3 P frame skipping	17	36.23 (dB)	-1.12	0.9559	-0.0033	25.1%
	1/2 P frame skipping	13	35.64 (dB)	-1.71	0.9538	-0.0054	41.1%
	2/3 P frame skipping	9	35.23 (dB)	-2.12	0.9525	-0.0067	60.5%
Sit_hand over_bag .yuv	No skipping	25	38.12 (dB)	0	0.9786	0	0
	1/3 P frame skipping	17	37.23 (dB)	-0.89	0.9774	-0.0012	24.9%
	1/2 P frame skipping	13	36.89 (dB)	-1.23	0.9767	-0.0019	38.2%
	2/3 P frame skipping	9	36.56 (dB)	-1.56	0.9761	-0.0025	56.1%
Sit_leave _bag.yuv	No skipping	25	37.98 (dB)	0	0.9783	0	0
	1/3 P frame skipping	17	36.72 (dB)	-1.26	0.9746	-0.0037	26.8%
	1/2 P frame skipping	13	36.35 (dB)	-1.63	0.9724	-0.0059	42.8%
	2/3 P frame skipping	9	35.96 (dB)	-2.02	0.9708	-0.0075	61.4%

In Table 3, two different image quality assessment methods are used to evaluate the loss of the video which have done the frame rate conversion with P frame skipping conception. The SSIM is based on fetching the structure information of the scene adaptively by human vision. It reflects the deformation of the structure information of the scene, and it can also reflect the video quality in the human vision to a certain extent.

Table 4 The comparison result between I-P-P-P... and I-B-B-P-B... with the same frame rate

Sequence name	Frame rate (f/s)	GOP	ΔPSNR_1 (dB)	ΔSSIM_1	ΔPSNR_2 (dB)	ΔSSIM_2
StoppedVehicle	25	I-B-B-P-B...	0	0	0	0
		I-P-P-P...	0	0	0	0
	17	I-B-B-P-B...	-0.26	-0.0004	0	0
		I-P-P-P...	-0.60	-0.0019	-0.41	-0.0013
	9	I-B-B-P-B...	-0.67	-0.0018	0	0
		I-P-P-P...	-1.15	-0.0031	-0.67	-0.0022
Intersection	25	I-B-B-P-B...	0	0	0	0
		I-P-P-P...	0	0	0	0
	17	I-B-B-P-B...	-0.61	-0.0021	0	0
		I-P-P-P...	-1.58	-0.0042	-1.12	-0.0033
	9	I-B-B-P-B...	-1.74	-0.0047	0	0
		I-P-P-P...	-3.40	-0.0094	-2.12	-0.0067
Sit_handover_bag	25	I-B-B-P-B...	0	0	0	0
		I-P-P-P...	0	0	0	0
	17	I-B-B-P-B...	-0.77	-0.0009	0	0
		I-P-P-P...	-1.39	-0.0013	-0.89	-0.0012
	9	I-B-B-P-B...	-1.95	-0.0034	0	0
		I-P-P-P...	-2.75	-0.0073	-1.56	-0.0025
Sit_leave_bag	25	I-B-B-P-B...	0	0	0	0
		I-P-P-P...	0	0	0	0
	17	I-B-B-P-B...	-1.15	-0.0032	0	0
		I-P-P-P...	-1.99	-0.0049	-1.26	-0.0037
	9	I-B-B-P-B...	-2.70	-0.0063	0	0
		I-P-P-P...	-3.60	-0.0103	-2.02	-0.0075

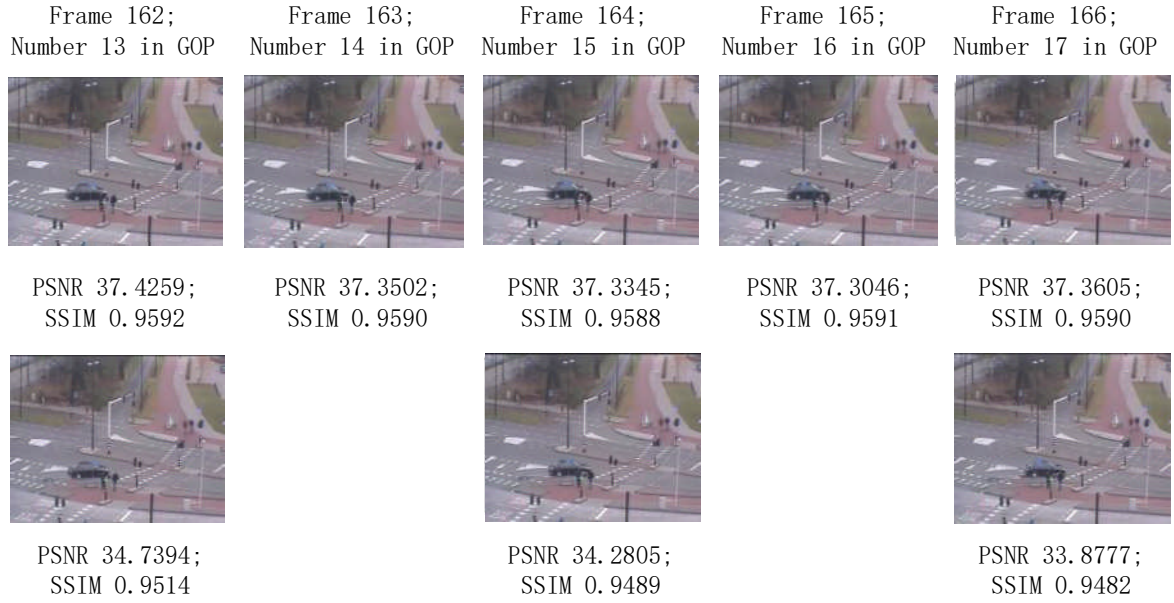


Fig.18. Intersection.yuv with 1/2 P frame skipping (Illustration) [86]

2.3.4 Conclusion

In this section, a temporal scalable decoding process with frame rate conversion method for surveillance video is introduced.

Through the experimental results, the proposed process and methods are effective to realize the low complexity decoding obviously and can provide multi-types of the frame rate to the terminal devices for playing the surveillance video.

It can be found that the reduction of computational complexity (decoding time) depends on the number of skipped frames, the more frames was skipped the more reduction of the computational complexity (decoding time) will be got. The PSNR loss is very small (about 0.1 ~ 0.2 (dB)) for B frame skipping. And the PSNR loss is about 0.7 ~ 2 (dB) for 2/3 P frame skipping and reduce the computational complexity about 60%. Although the PSNR loss is a little high, the loss of SSIM is only 0.002~0.007 for 2/3 P frame skipping. So the structure information of the scene is affected very little through the proposed temporal scalable decoding process, and the

video quality is also affected very little in the human vision. (As shown in Fig.18)

2.4 Adaptive solution of temporal scalable decoding

process with frame rate conversion method for

surveillance video

2.4.1 Adaptive solution for surveillance video

2.4.1.1 The basic idea of the adaptive scheme:

After comparing the quality of the picture between before and after frame skipping operation (based on the certain frame skipping scheme), it can be found that the stronger movements cause bigger video quality loss subjectively. The result is shown in Fig.19. The red circles point out the unclear parts, and they are the parts which have strong movements. The yellow circles point out the slow moving objects, and they are affected little by the frame skipping operation.

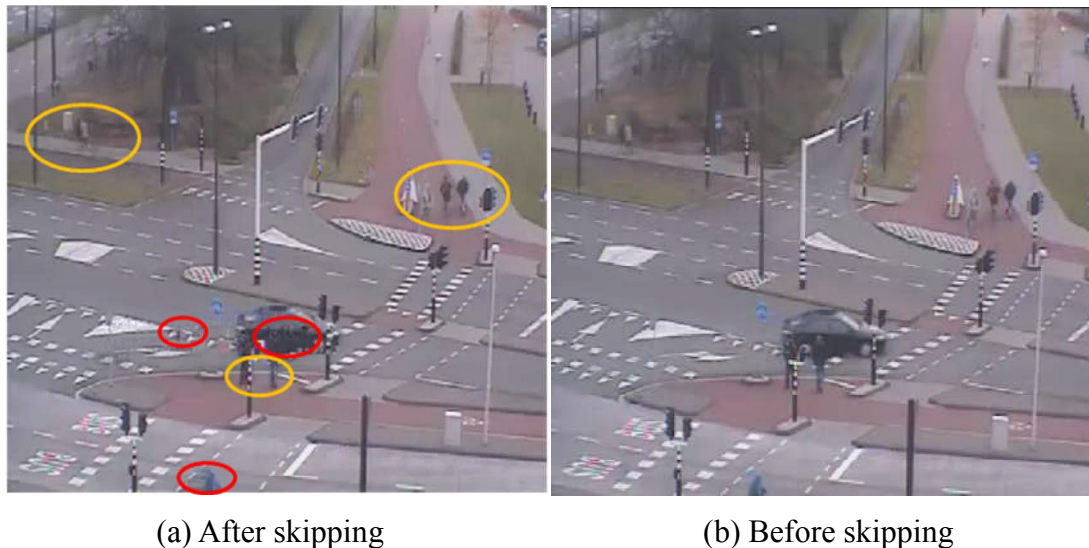


Fig.19. The comparison between before and after skipping [86]

We can also analyze the features of the text sequences:

StoppedVehicle.yuv: There is only a small fast moving objects and some slow moving objects in this sequence.

Intersection.yuv; Sit_handover_bag.yuv; Sit_leave_bag.yuv: There are some fast moving objects in these sequences.

Then we can find that the sequences which only have small and slow moving objects can get better experimental results than the sequences which have big and fast moving objects.

So the basic idea of the proposed adaptive frame rate-down conversion method is to avoid skipping the frames which have strong movements.

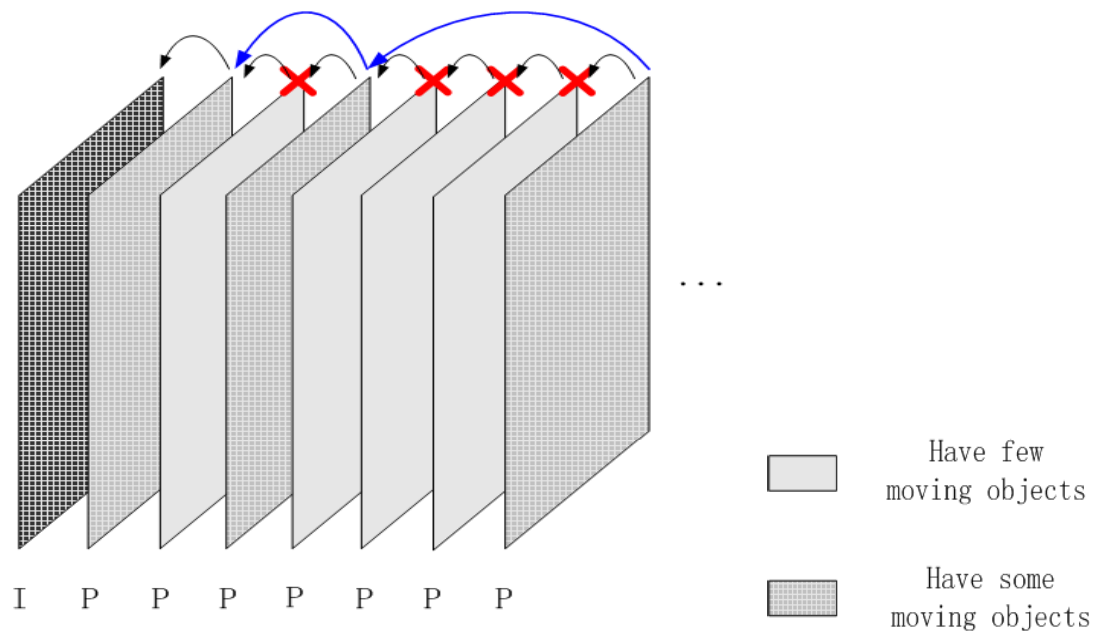


Fig.20. The illustration of the skipped frame selecting

It chooses the frames which only have slight movements to skip, and keep the frames which have strong movements (As shown in Fig.20). If there are a lot of moving objects in the current frame, then keep it and decode it as usual. And if there

is few moving objects in the frame, then skip it. Based on this principle, the loss of the video quality can be reduced a lot in the same case of the frame rate conversion.

But how to evaluate the degree of the movement in one frame is the most important problem in this method. It is necessary to find the factors which affect the video quality a lot.

2.4.1.2 The factors cause the video quality loss:

In the temporal scalable decoding process with certain frame skipping scheme, there were two main factors cause the video quality loss:

- 1> The first one was from the motion vector composition algorithm. The skipped frames were not decoded completely in the process, so couldn't do the one forth pixel interpolation and couldn't get the predicted position by the motion vectors in the skipped frames. By using the motion vector composition algorithm, the motion vectors could be refined to the refined reference frames. But the motion vector composition was a kind of approximate composition, and it would cause some errors.
- 2> The second one was the loss of the residual information in the skipping process. It is the more important factor than the first one. In order to reduce the complexity of the decoding process, the temporal scalable decoding process only completed the motion vector composition without the residual information composition. It would reduce the decoding complexity and the decoding time effectively, but also reduced the video quality in some degree at the same time, especially based on the certain frame skipping scheme.

2.4.1.3 The assumption and proving:

The residual information is the most important factor which causes the video quality loss, so it is the suitable value to evaluate the degree of the movement and the

degree of the possible video quality loss in one frame. But the residual information of the skipped frames can't be got in the proposed process.

Then assume that if one frame has bigger motion vector energy, it will have bigger residual energy at the same time. So based on this assumption, instead of the residual information, the motion vector information can be used to evaluate the degree of the movement and the degree of the possible video quality loss in one frame.

The residual information has strong randomness, so it can't be predicted accurately. Then next step is that try to prove the relationship between the residual information and the motion vector information in probability.

The statistic data sampling from the Intersection.yuv sequence. And frame is used as the unit to calculate the value of video quality loss, the motion vector energy and the residual information energy. The correlation coefficients among the video quality loss, motion vector energy and residual energy are used to prove the relationship between the motion vector and the residual information in probability. The linear regression is done by the least squares method.

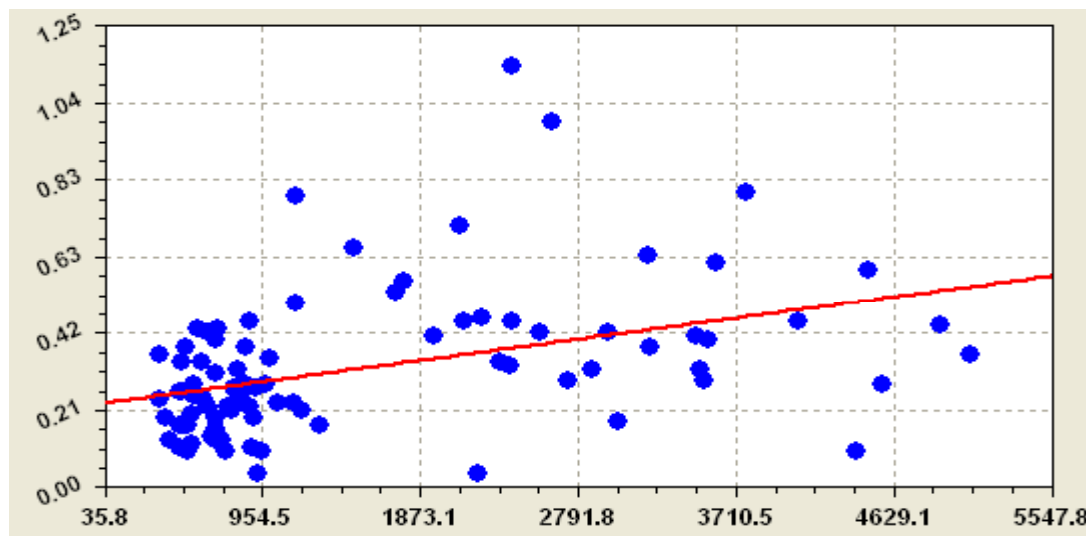


Fig.21. The correlation coefficient between video quality loss and the motion vector information
(y-axis: video quality loss (dB); x-axis: motion vector information energy)

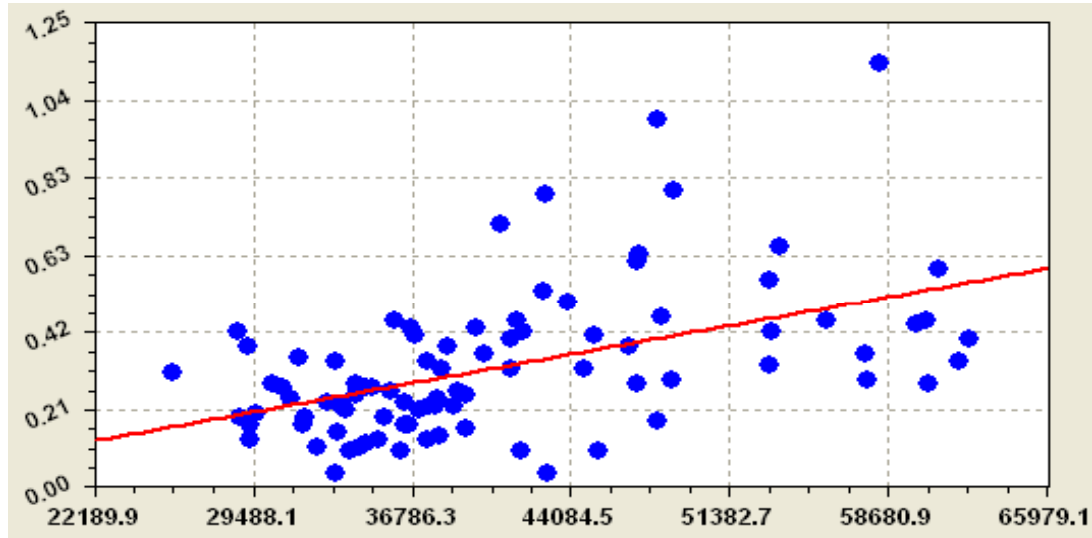


Fig.22. The correlation coefficient between video quality loss and the residual information (y-axis: video quality loss (dB); x-axis: residual information energy)

As shown in Fig.21 and Fig.22, the correlation coefficient of the residual information is 0.4924 bigger than the value of the motion vector 0.4029. So the loss of the residual information is the most important factor to affect the video quality loss.

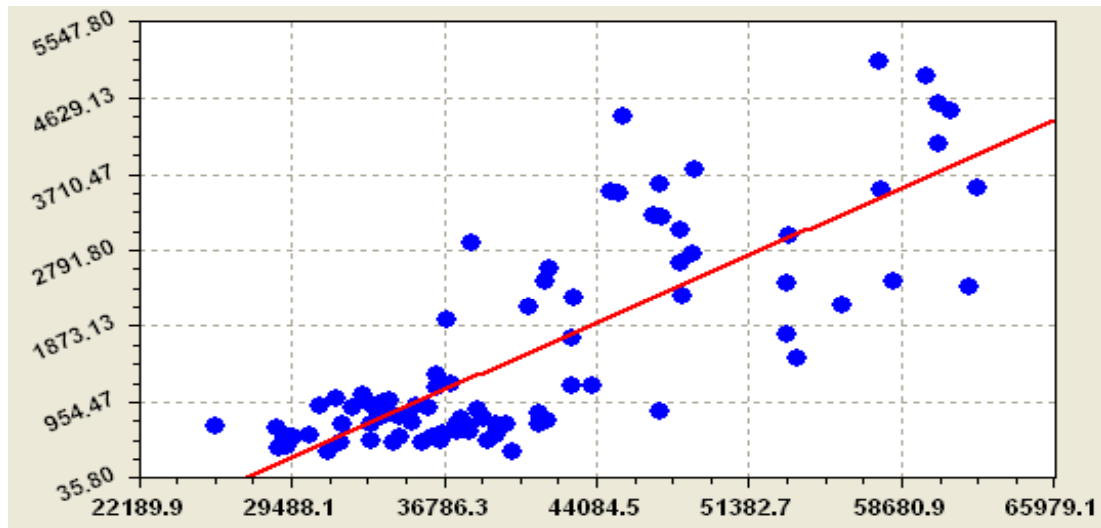


Fig.23. The correlation coefficient between the residual information and the motion vector information (y-axis: motion vector information energy; x-axis: residual information energy)

As shown in Fig.23, the correlation coefficient is 0.8021 between the residual information and the motion vector information. It is a strong correlation, and ensures

that the energy of the motion vector can be used to evaluate the energy of the residual information. Then it can be used to evaluate the degree of the possible video quality loss.

2.4.1.4 The realization of the adaptive scheme:

The basic thought for realizing this idea is to quantify the energy of the residual information in one frame and use the correlation of the movement in the continuous frames to compare and predict the energy of the residual information in the next frame, then decide skip the next frame or keep it. We use the square of motion vector and the square of the residual information to evaluate the energy of the movements.

After the proving of the relationship between the energy of the motion vector and the energy of the residual information, the realization of the adaptive scheme uses motion vector energy to instead the energy of the residual information.

The realization of the adaptive scheme is shown in Fig.24 and the flowchart is shown in Fig.25

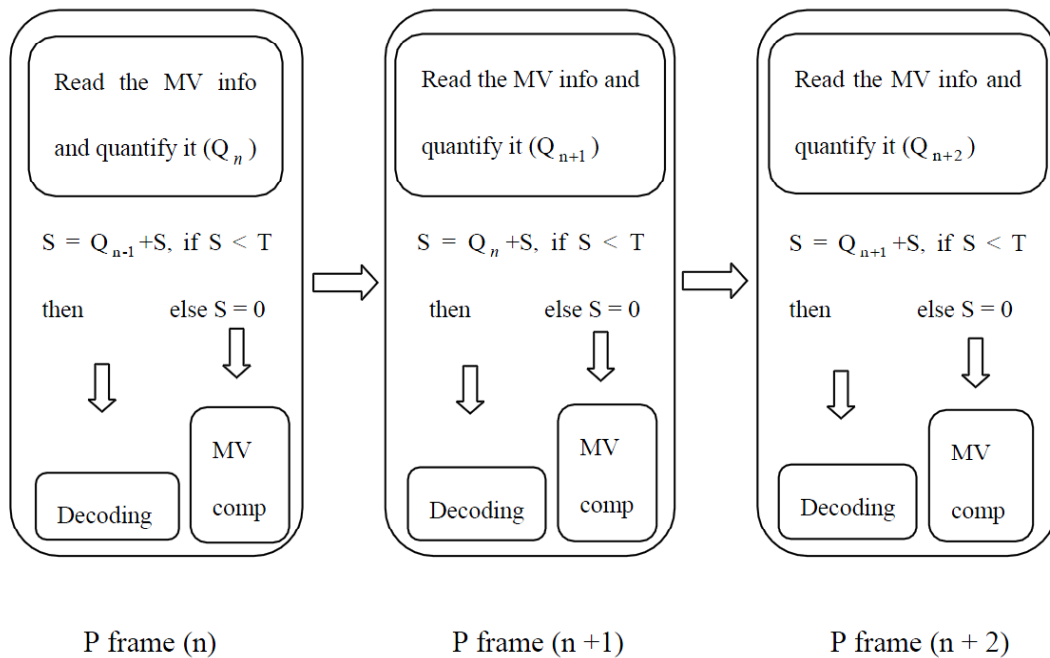


Fig.24. The deciding process of the adaptive scheme

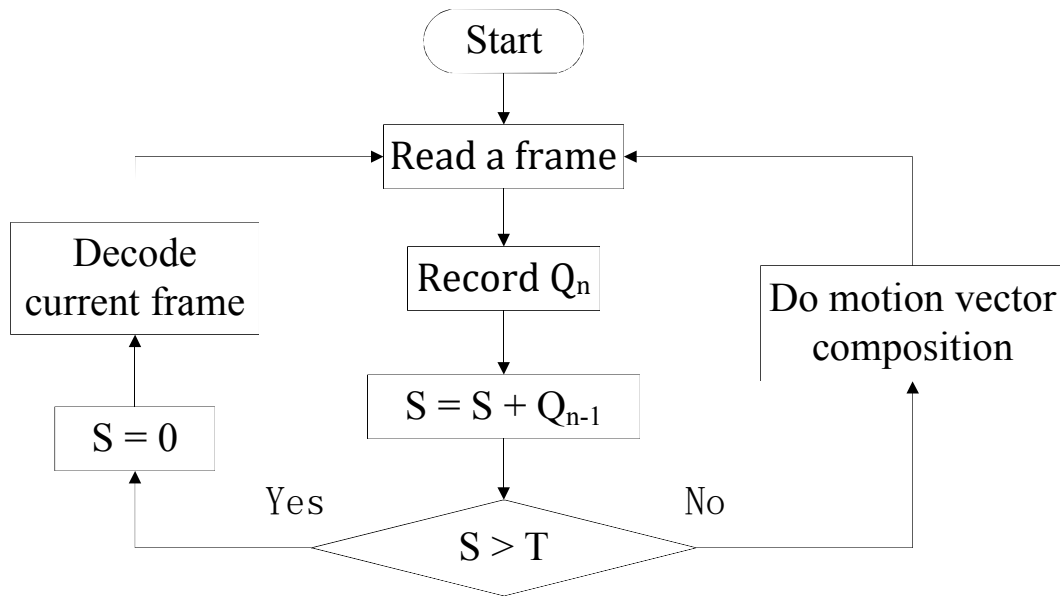


Fig.25. The flowchart of the adaptive scheme

In the P frame (n), read the motion vector information and quantify it as Q^n at first. Add the quantified value Q^{n-1} of the previous frame and the sum of the quantified value S before the previous frame. Then uses the updated S to compare with the threshold which is used to control the frame rate, and decide decoding this frame or skipping it. If S is bigger than the threshold, set it to zero and decode this frame. If S is smaller than the threshold, keep the value of S and only do the motion vector composition in the current frame and record the Q^n for the decision factor in the p frame ($n+1$). S is used to record the degree of the motion vector energy accumulation. It avoids that the movement energy in some frames which only have slight movements will be ignored in the adaptive scheme decision process.

2.4.2 Experimental result

The experimental sources and parameters are shown in Table 5. The result of

proposed method which is tested by using JM 15.1 reference software and compared with the result in [47] is shown in the Table 6. (QVGA is 320×256 , CIF is 352×288)

Table 5 Experimental sources and parameters:

Sequence name	Format	FramesToBeEncoded	Intra Period	Qp
StoppedVehicle.yuv	QVGA	200	25	28
Intersection.yuv	CIF	200	25	28
Sit_handover_bag.yuv	CIF	200	25	28
Sit_leave_bag.yuv	CIF	200	25	28

2.4.3 Conclusion

In this section, an adaptive solution of the temporal scalable decoding process with frame rate conversion method for surveillance video is introduced. Through the experimental results, the proposed method is effective to realize the adaptive frame rate-down conversion based on the content of the pictures.

By using this proposed method, the PSNR loss is about 0.01~1.85 dB in the experimental cases. And compares with the result which is based on the certain frame skipping scheme, the PSNR is improved about 0.2~1.4 dB in corresponding cases. The loss of the decoding time reduction is only less than 5% in the worst case, and in the most of the cases it is only 0 ~ 2%.

Table 6 Experimental result:

Sequence name	Skipped Frames	Certain skipping			Adaptive skipping		
		PSNR (dB)	Δ PSNR (dB)	Time reduction	PSNR (dB)	Δ PSNR (dB)	Time reduction
Stopped Vehicle.yuv	No skipping	37.90	0	0	37.90	0	0
	30%P frame skipping	37.49	-0.41	24.5%	37.84	-0.06	23.4%
	50%P frame skipping	37.36	-0.54	40.3%	37.73	-0.17	40.0%
	70%P frame skipping	37.23	-0.67	60.7%	37.43	-0.47	55.2%
Intersecti on.yuv	No skipping	37.35	0	0	37.35	0	0
	30%P frame skipping	36.23	-1.12	25.1%	36.68	-0.67	24.7%
	50%P frame skipping	35.64	-1.71	41.1%	36.07	-1.28	41.8%
	70%P frame skipping	35.23	-2.12	60.5%	35.50	-1.85	55.7%
Sit_hand over_bag .yuv	No skipping	38.12	0	0	38.12	0	0
	30%P frame skipping	37.23	-0.89	24.9%	38.13	0.01	22.5%
	50%P frame skipping	36.89	-1.23	38.2%	37.48	-0.64	36.6%
	70%P frame skipping	36.56	-1.56	56.1%	36.73	-1.39	52.7%
Sit_leave _bag.yuv	No skipping	37.98	0	0	37.98	0	0
	30%P frame skipping	36.72	-1.26	26.8%	37.91	-0.07	21.2%
	50%P frame skipping	36.35	-1.63	42.8%	37.74	-0.24	33.2%
	70%P frame skipping	35.96	-2.02	61.4%	36.84	-1.14	52.0%

2.5 Adaptive decoding process with temporal prediction method for common video

2.5.1 Details of the adaptive decoding process

2.5.1.1 The basic thought of the temporal prediction

In the proposed temporal prediction process, the situations of the previous frames are used to predict the state of the current frames based on the continuity of the movements in the one camera cases (as shown in Fig.26).

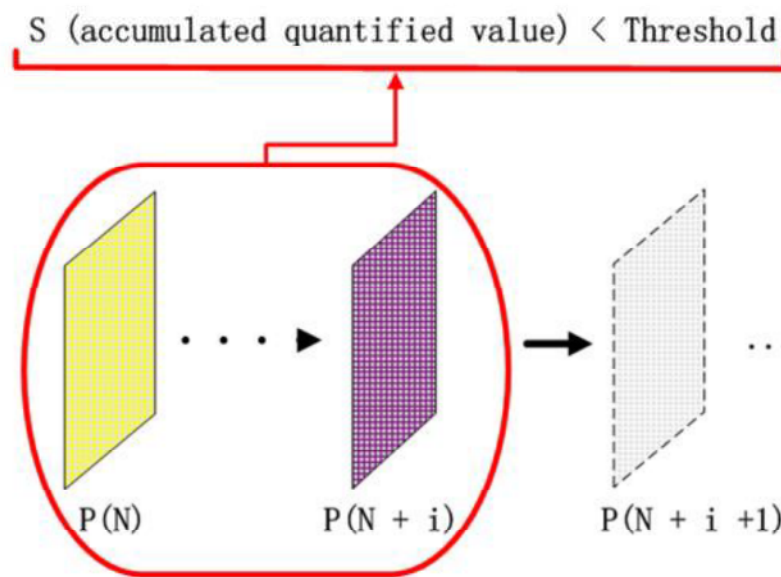


Fig.26. The temporal prediction illustration

It uses some quantified value to evaluate the degree of the movements in the previous frames. If the accumulated quantified value smaller than the threshold, consider that the movements are also slight in the current frame and skip this frame. If the accumulated quantified value exceeds the threshold, then consider that the

movements are strong in the current frame and if skip this frame, the cost of the video quality is unacceptable. So this process will choose decoding this frame as usual in such cases.

If the current frame is kept, the motion vector information and the residual information both can be recorded to evaluate its state. But if the current frame is skipped, only the motion vector information can be got. So the motion vector information is used as the accumulated quantified value in the proposed temporal prediction process.

2.5.1.2 The flowchart of the temporal prediction method

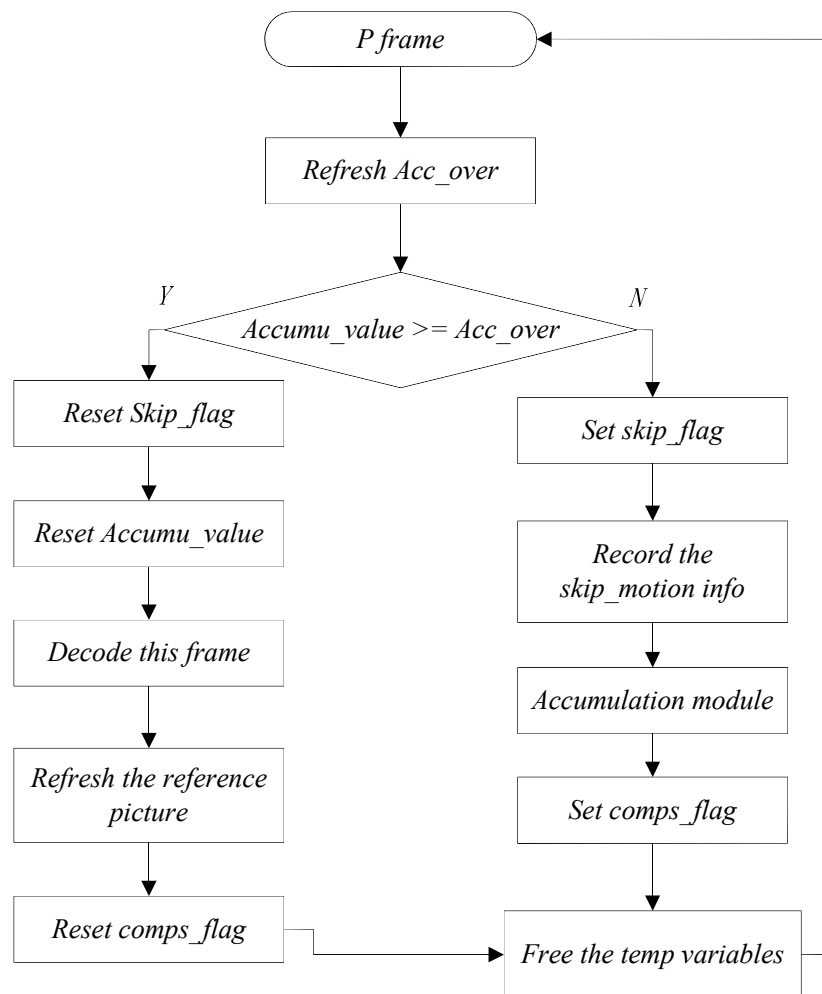


Fig.27. The temporal prediction flowchart

Table 7 The experimental sources and parameters:

Sequence name	Format	FramesToBeEncoded	Intra Period	Qp
mobile_cif.yuv	CIF	200	25	28
container_cif.yuv	CIF	200	25	28
coastguard_cif.yuv	CIF	200	25	28
foreman_cif.yuv	CIF	200	25	28

The flowchart of the temporal prediction method is shown in Fig.27. The Acc_over is the threshold value, and the Accumu_value is the accumulated quantified value. The state of the decoded frame is decided by the relationship of these two values.

If the Accumu_value is bigger, decode the current frame as usual and refresh the reference picture index. If not, skip this frame, record the motion vector information and quantified it. Then enter the accumulation module to avoid the slight movements which will be ignored without the temporal accumulation.

2.5.2 Experimental result

The experimental sources and parameters are shown in Table 7. The result of proposed method which is tested by using JM 15.1 [46] reference software and compared with the method which is proposed in [47] is shown in the Table 8. (CIF is 352×288)

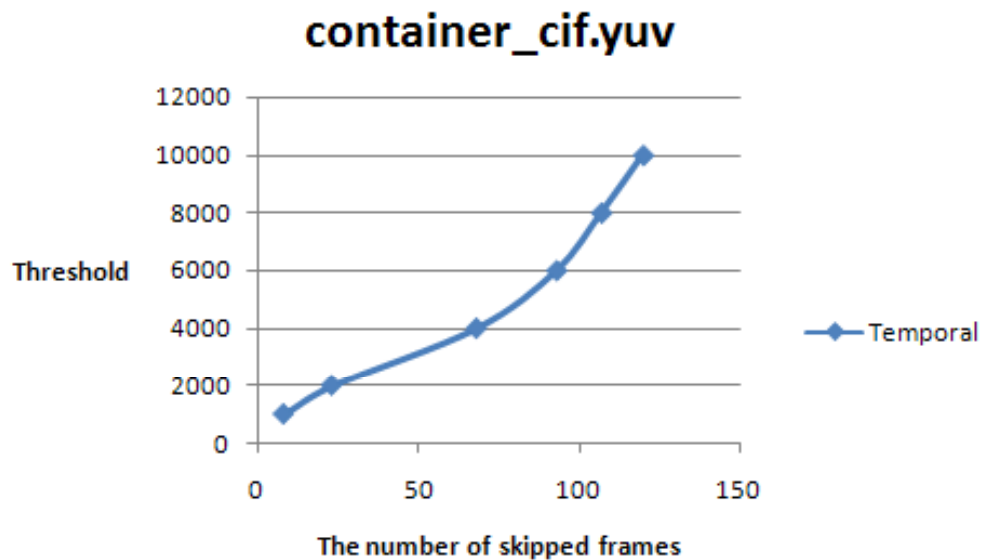


Fig.28. The relationship between the threshold and the number of skipped frames in container_cif.yuv

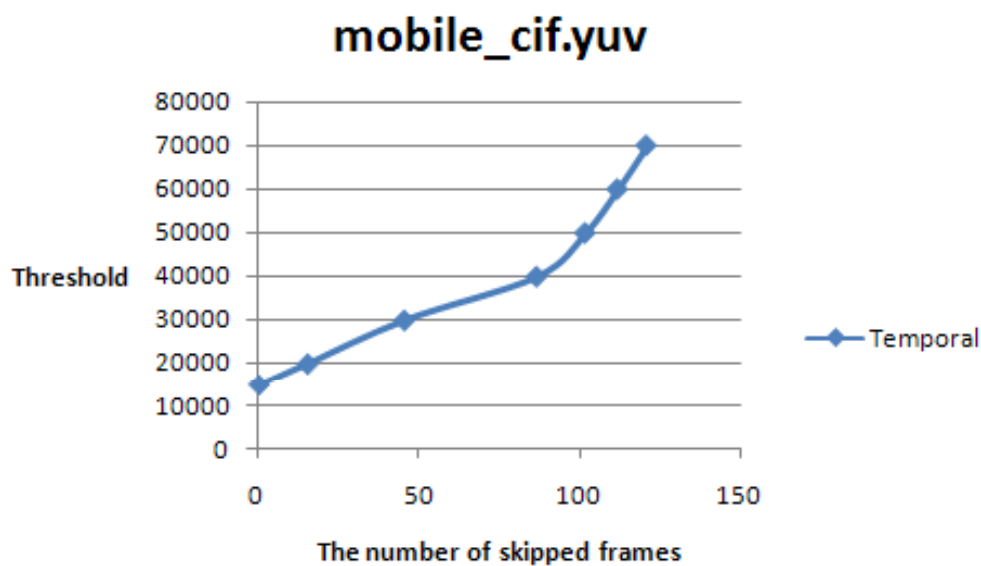


Fig.29. The relationship between the threshold and the number of skipped frames in mobile_cif.yuv

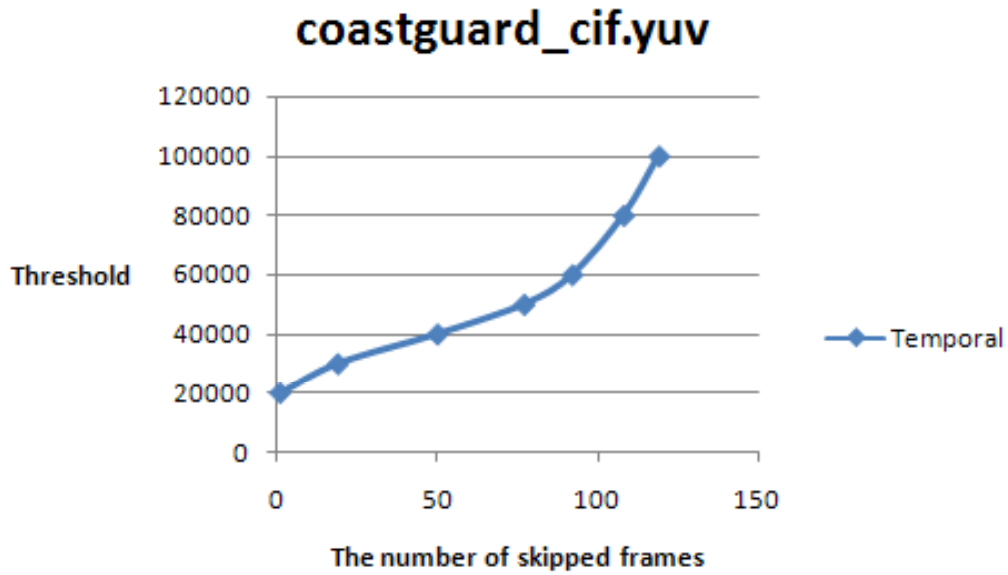


Fig.30. The relationship between the threshold and the number of skipped frames in
coastguard_cif.yuv

Through the experiment, the relationship between temporal threshold which is used in the temporal prediction and the number of skipped frames are shown in Fig.28, Fig.29, Fig.30 and Fig.31.

The direction of the curves shows the trend of the relationship and shows that the number of skipped frames changes sensitively or not when the threshold changed. Based on these curves, we can decide the threshold for a certain number of skipping frames.

Through the Fig.28, Fig.29, Fig.30 and Fig.31, the curve of container_cif.yuv can be found much more sensitive than the other three sequences. So it means that there are only some slight movements in this sequence.

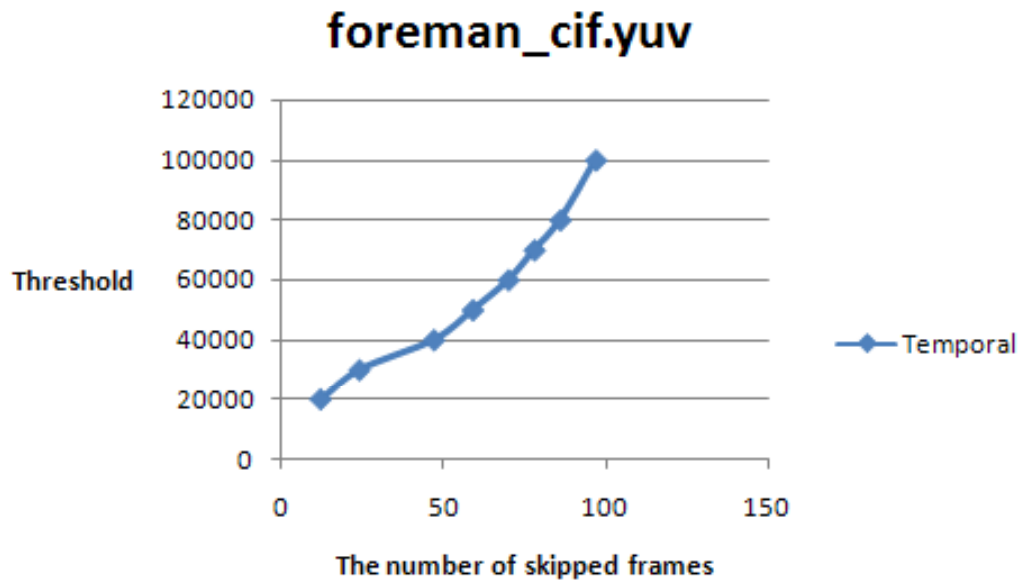


Fig.31. The relationship between the threshold and the number of skipped frames in foreman_cif.yuv

Through the result in Table 8, by using the temporal scalable decoding process, the PSNR loss is only 0.71 – 2.16 dB with the container_cif.yuv, but it is about 3.34 – 8.79 dB in the rest three sequences. It is because that there are only slight movements in the container_cif.yuv, so the motion vector information and the residual information is small. Then after the frame skipping process, the video quality loss is very small. But the improvement of the video quality is also very small in this kind of the cases

Table 8 The experimental result:

Sequence name	Skipped Frames	TSDP		TSDP + Temporal Prediction		
		PSNR (dB)	Δ PSNR (dB)	PSNR (dB)	Δ PSNR (dB)	Time reduction
mobile_ci f.yuv	No skipping	35.23	0	35.23	0	0
	30 P frames	29.63	-5.60	32.20	-3.23	13.4%
	60 P frames	27.76	-7.47	29.72	-5.51	24.1%
	90 P frames	26.44	-8.79	26.48	-8.75	34.1%
container _cif.yuv	No skipping	36.44	0	36.44	0	0
	30 P frames	35.73	-0.71	35.75	-0.69	12.1%
	60 P frames	35.09	-1.35	35.10	-1.34	19.4%
	90 P frames	34.28	-2.16	34.30	-2.14	28.5%
coastguar d_cif.yuv	No skipping	35.69	0	35.69	0	0
	30 P frames	31.49	-4.20	31.52	-4.17	13.7%
	60 P frames	30.03	-5.66	30.08	-5.61	24.2%
	90 P frames	28.77	-6.92	28.85	-6.84	34.6%
foreman_ cif.yuv	No skipping	37.44	0	37.44	0	0
	30 P frames	34.10	-3.34	36.26	-1.18	13.9%
	60 P frames	32.75	-4.69	34.87	-2.57	23.9%
	90 P frames	32.04	-5.40	33.12	-4.32	35.9%

2.5.3 Conclusion

Through the experimental result, the proposed adaptive low power decoding process with temporal prediction method for common video improves the video quality a lot that compares with the previous temporal scalable decoding process with frame rate conversion method [47].

The rate of the decoding time reduction (power consumption reduction) is always related to the number of the skipped frames. The more frames are skipped the

more decoding time reduction will be got.

In the sequences which only have slight movements, the PSNR is only improved a little about 0.01 – 0.08 dB. But it is improved obviously about 1.9 – 2.4 dB in the sequences which have strong movements.

3. Frame Compatible Format Fast Encoder with Stereo Matching

3.1 Introduction

The reconstruction of a stereo scene from a pair of images taken from different directions is one of the classical problems in computer vision area. Although this topic has been researched in a long history, it is only in recent years that numerous algorithms have been introduced [58] – [62], along with the huge development of computer performance and 3D video market. Stereo matching helps to generate the depth map through images taken from different angles. By comparing and find the difference in position of a same object in two images, the distance from the object to the baseline of camera can be calculated through similar triangle geometric knowledge.

Frame-compatible format [63] is a spatial multiplex of two neighboring views into one single frame. Usually, it refers to sub-sampling technology for the two views and then packed the two sub-sampled views together into one frame. It is considered as one of the most promising solutions of 3D distribution on the existing system, as it is completely compatible for the existing video codec, such as H.264/MPEG-4 AVC and MPEG-2. It enables the stereo video compressed with existing encoders, transmitted through existing channels and decoded by existing receivers and players, with minimal changes. Such advantages will benefit the stereo video being quickly deployed to the already 3D market. Consequently, this format has achieved a wide approval. For example, it has been included in H.264/AVC [64] as SEI [65] and HDMI v1.4 [66].

Because of the short baseline between the two packed views, it is considered that there exists content similarity in the two packed views in frame-compatible format. With the statistical analysis of the prediction correlation of the two corresponding MBs, the relative high prediction correlation between the two packed views, which results from the content similarity, has been proved. In the process of addressing the two

corresponding MBs, the shift obtaining method has been designed by searching the best matched block in one packed view for the target MB in the second packed view. In addition, with the statistical analysis of the prediction in the two packed views, the two prediction types with higher occurrence probability have been selected as the candidate prediction according to the reference block prediction types.

3.2 Previous Work

Some works have already been done about the frame compatible format encoding process. In the previous works [76] – [81], mainly focus on how to partition the views, how to reduce the quality loss in the down-sampling process and how to improve the video quality in the up-sampling process.

The previous works [82] [83] are improved method based on the MVC structure. They obtain the global disparity vector between two frames from the different views. This kind of methods can't be used in the frame compatible format cases because of the big computational complexity requirements.

Only the target of zeng's work [67] focuses on how to utilize the relation between the content similarities of different views. In this work, a simple block based shift obtaining method was used. Theoretically, as long as camera parameters and depth views are kept, displacement between two filling planes can be obtained. However, real depth images on the ground do not apply to the source of each 3D sequence. In addition, generating depth maps by stereoscopic video sequences is very challenging and unreliable. Therefore, in our proposal, we design a shift acquisition method. The goal is to get an approximate but definite shift to ensure the correlation accuracy between the corresponding MBs in the two filled planes.

The main object usually appears in the central line of each plane, so the shift acquisition method searches for only the MBs in the central row of each MBs in the plane₁ (described as MB_i) in the central line of each MBs (described as MB_t). Therefore, finding the target of MB_i shift is balanced to find the MB with minimum

SAD value (as Eq.1) in MB_i , which is set to MB_i' (Eq.2). Fig.32 illustrates the search scheme with the top-bottom FCF as an example. The shift of MB_i is determined to be Δx_i according to formula Eq.3, (x_i', y_i') is the most left upper pixel in MB_i . According to Eq.4, the average displacement $\overline{\Delta x}$ is the so-called relative displacement in our proposal. It is retrieved from each block in plane_1 to process reference blocks.

For video captured by vertical and 2D cameras, the relative displacement is correspondingly changed to $\overline{\Delta y}$ and $(\overline{\Delta x}, \overline{\Delta y})$. It should be noted that as the goal is to reduce the computational complexity, the two capture cameras between the baseline are fixed. The shift acquisition method is performed only in the first frame encoding.

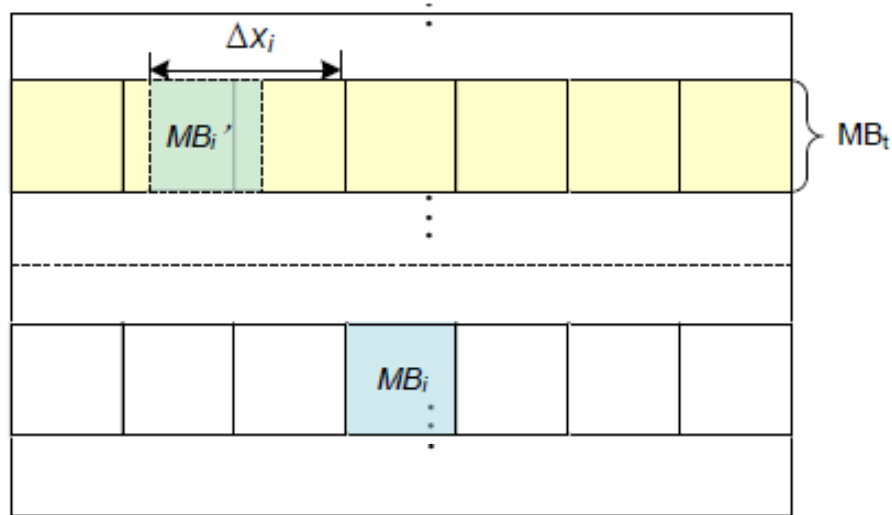


Fig.32. Matching scheme in top-bottom arrangement

$$SAD(MB_i | MB_t) = \sum_{j,k=0}^{15} |s(x_{ij}, y_{ik}) - s(x_{tj}, y_{tk})| \quad (1)$$

$$MB_i' = \arg \min(SAD(MB_i | MB_t)) \quad (2)$$

$$\Delta x_i = x_i' - x_i \quad (3)$$

$$\overline{\Delta x} = \frac{1}{\text{imagewidth}/16} \sum_{i=0}^{\text{imagewidth}/16-1} \Delta x_i \quad (4)$$

The shift obtaining method only searches the MBs in the central row of plane_0 to find the MB with the minimum SAD value as the shift position and get the shift value

for each MB.

Use the average shift value of the MBs in the central row of plane_0 as the shift value for all of the MBs in the plane_1. It is as shown in Fig.33.

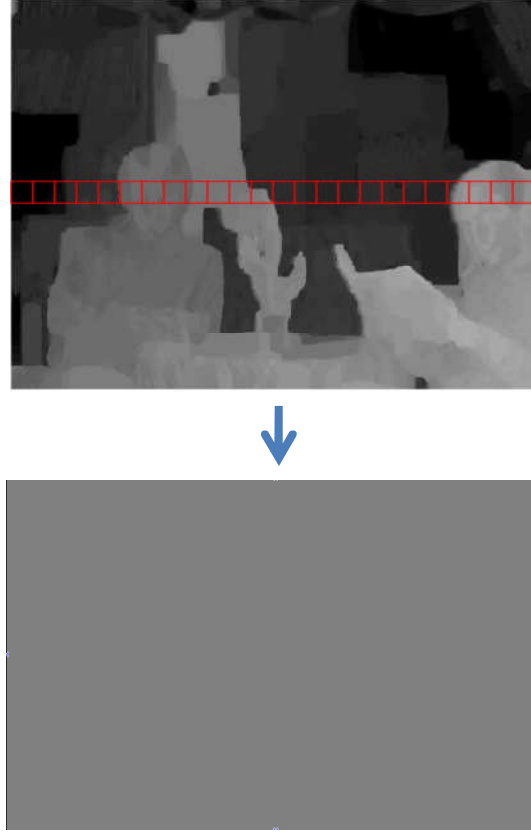


Fig.33. Illustration of the depth map fetching process [85]

Because intra and inter frame MB partitions are different from each other, for each MB in plane_1, it can handle reference MB in plane_0 by means of relative shift. In the intra coding and inter frame coding, five top-bottom FCF sequences are analyzed. All possibilities of each pattern are the average values of 5 sequence encoding results. When the configuration file is high and RDO is turned on, there are more than 100 frames on 4 QP values.

According to the prediction correlation analysis between two filling planes in FCF, three candidate sets are retrieved, including the partition candidate set in Intra and Inter frames and the direction candidate set within Intra. According to the H.264/AVC standard, the algorithm carries out the original prediction strategy on plane_0 and records the prediction types of each block. When it began encoding the first MB in plane_1, a fast algorithm was enabled. It retrieves the encoded MB partition by means of the relative shift obtained before the first frame encoding (in Intra coding, it also retrieves the predicted direction of the reference block). The candidate prediction is by mapping candidate sets according to the reference values. Fig.34 shows the flow chart of the fast algorithm in front. Considering the balance between recording accuracy and overhead consumption as statistical analysis, the intra unit coding and inter frame coding recording units are allocated to 4x4 and 8x8 respectively.

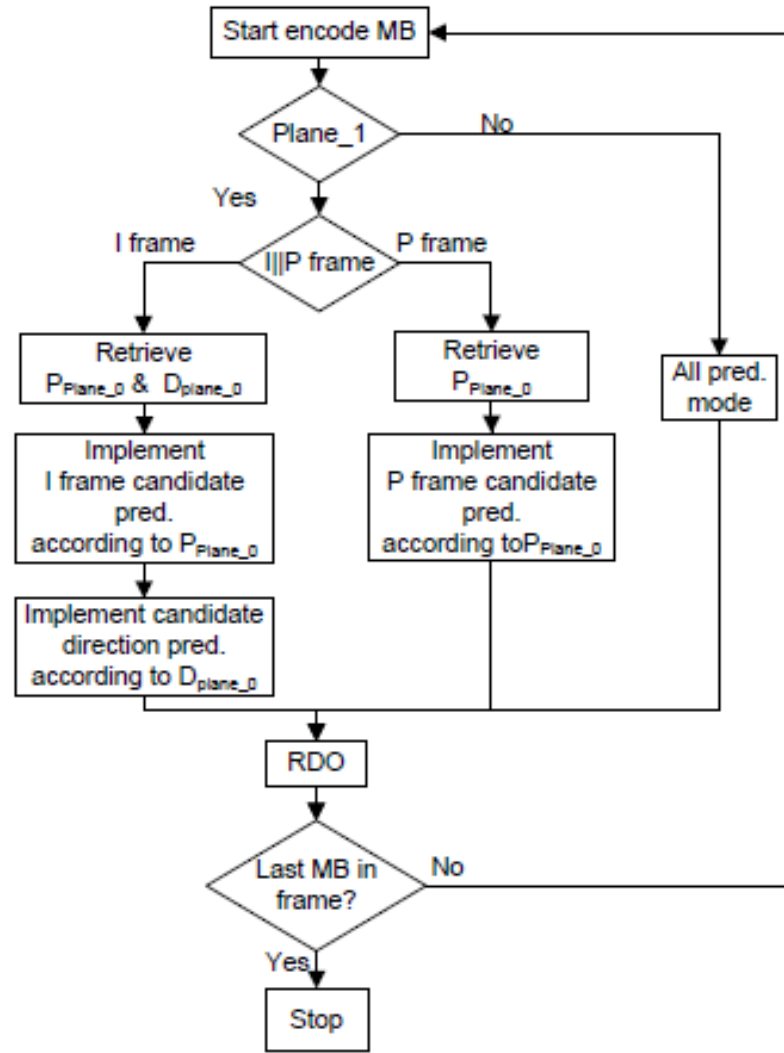


Fig.34. The flowchart of the previous fast algorithm.

It is only a simple shift operation without any accurate depth information and obviously it can't figure out the accurate details. Therefore, an improved stereo matching algorithm based on [62] is integrated into the frame-compatible format fast encoder to make it more stable, reasonable and accurate.

3.3 The combined architecture of the fast algorithm of FCF with stereo matching

The principle of fast encoder for FCF is to accept that content similarity in two packed views can lead to prediction correlation in encoding. The previous work [67] of fast encoder for FCF uses a simple block based shift obtaining method, which cannot figure out the accurate content correlation. The stereo matching algorithm is definitely the best choice to reveal the content matching in the packed view, which is quite suitable to be used in this research. The architecture of using stereo matching in fast FCF encoder is shown in Fig.35.

The difficulty is that stereo matching algorithm usually costs lot of time and computational resource. If it needs too much time to calculate the depth map, the fast algorithm becomes useless. So the research is divided into two parts, first to find out a novel fast stereo matching algorithm which can maintain good performance at the same time, second to implement this idea in to the FCF fast encoding.

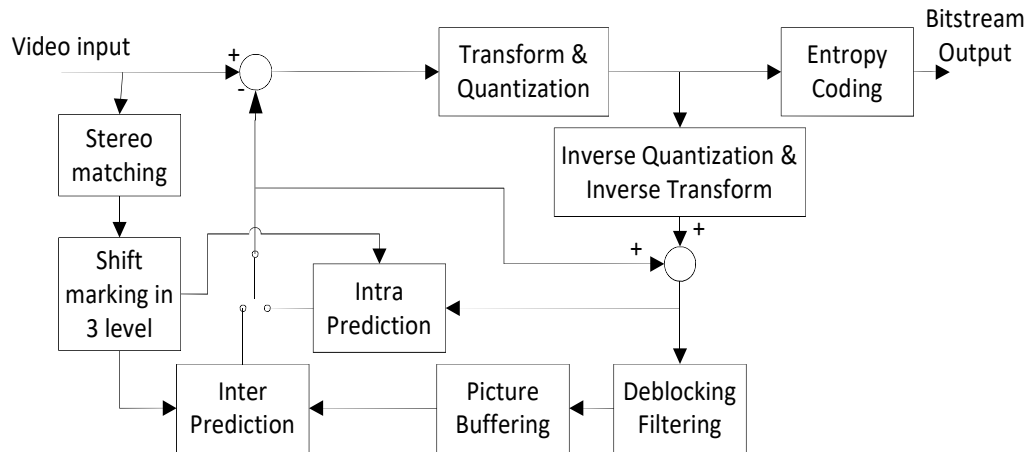


Fig.35. The architecture of using stereo matching in fast FCF encoder

3.4 The proposed algorithm

3.4.1 The relationship between stereo matching and FCF

The basis of fast encoding algorithm for FCF is the huge correlation between the content similarity and prediction similarity in packed views. According to the H.264/AVC standard, Intra_4x4 and Intra_8x8 have 9 directions respectively, and Intra_16x16 have 4 directions respectively. Because the smallest block of intra coding is 4x4, in our prediction correlation analysis, this size is designated as the recording unit within the frame. Intra prediction correlation analysis can be divided into two categories: classification type and prediction direction. On the other hand, the MB partition in the P frame includes Intra_4x4, Intra_16x16, Intra_8x8, PSKIP, P_16x16, P_16x8, P_8x16, P_8x8. In addition, there are sub_8x8, sub_8x4, sub_4x8 and sub_4x4 belonging to P_8x8 respectively.

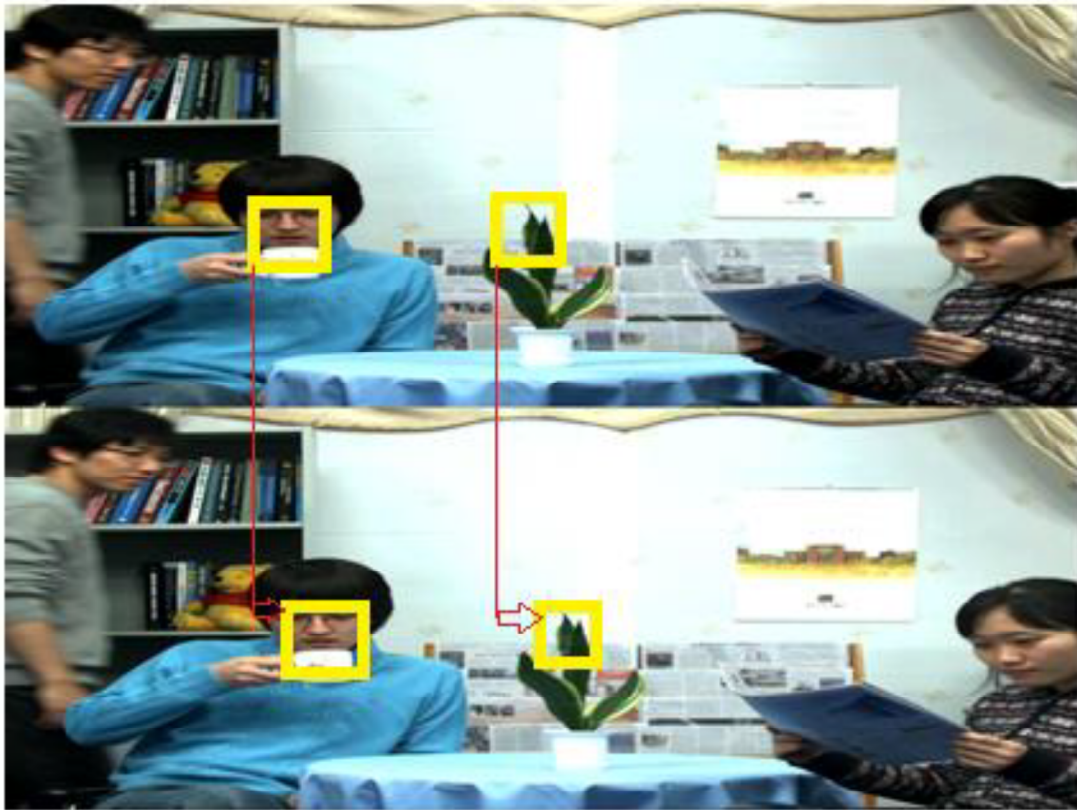


Fig.36. The actual horizontal shift between the reference and target [85]

It must be noticed that the simple shift obtaining method is quite a rough and cannot really reveal the content correspondence. As it can be found from Fig.36, the actual horizontal shift between the reference and target varies in different blocks. It is decided by the depth of the concerned object. So the traditional shift described in [67] is not the best choice.

On the other hand, the result of depth map generated by stereo matching algorithm (shown in Fig.37) is very suitable for obtaining the shift. Because the depth map is pixel based, a shift marking in blocks is need.

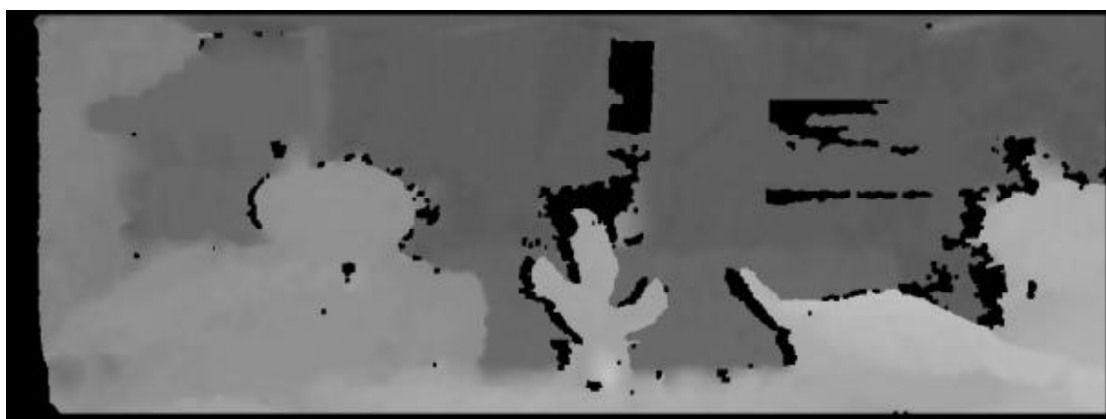


Fig.37. The result from stereo matching algorithm

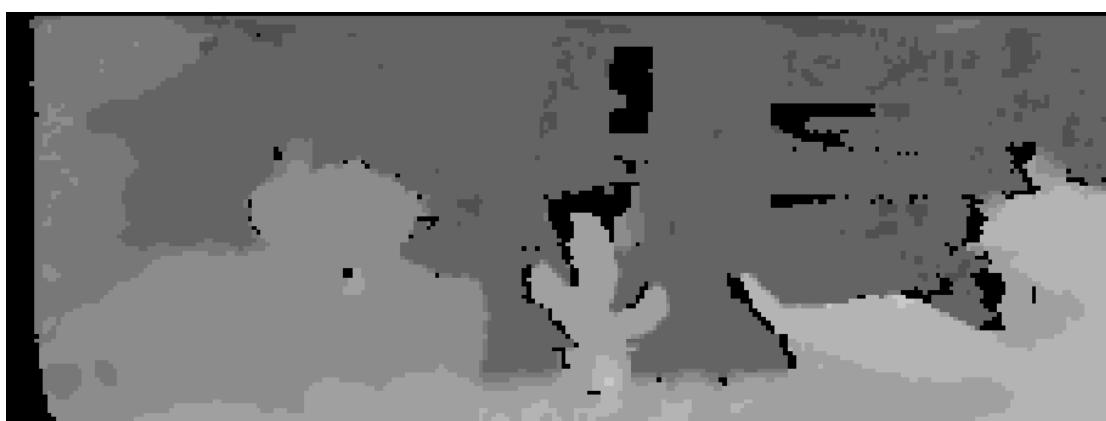


Fig.38. The marked shift value for 4*4 blocks

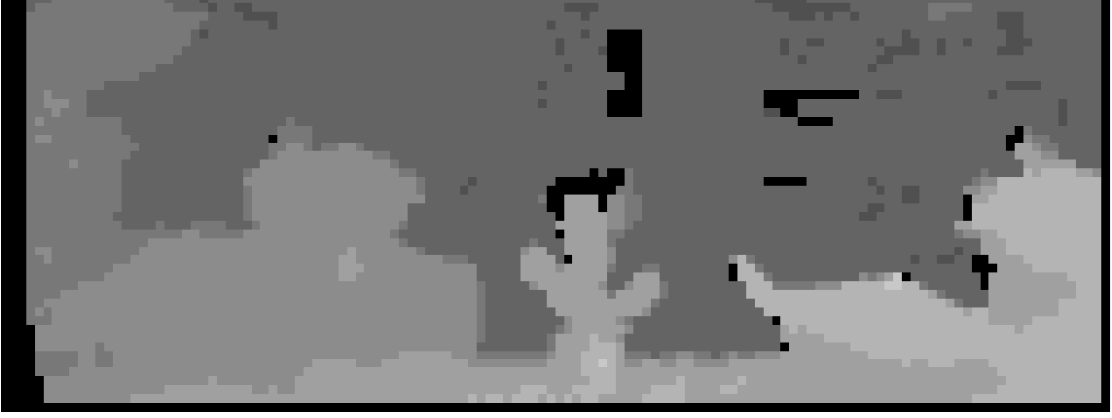


Fig.39. The marked shift value for 8*8 blocks



Fig.40. The marked shift value for 16*16 blocks

As is shown in Fig.38, Fig.39, Fig.40, the shift in a block is marked by

$$\text{shift}_B = \frac{\sum d_p}{n} \quad (p \in B) \quad (5)$$

B means the 4*4, 8*8, 16*16 block, d_p means the disparity of pixel p retrieved from stereo matching result. N here is the number of pixels with non-zero disparity in B. The shift marking step can not only transform the pixel based disparity into block based correspondence but also remove the noise of some mismatching points in a certain block. In this process, for 4*4 blocks, we keep the shift interval unit as a pixel, same as

the disparity from stereo matching. For 8×8 and 16×16 blocks, we set the shift multiple of 4 pixels. Thus, when addressing the reference MB for the current processing MB, there are 3 cases that the remainders of the shift between the two MBs are 0, 4, 8 and 12 pixels. We here group these cases into 3 modes to see the improvements of using stereo matching rather than simple shift obtaining.

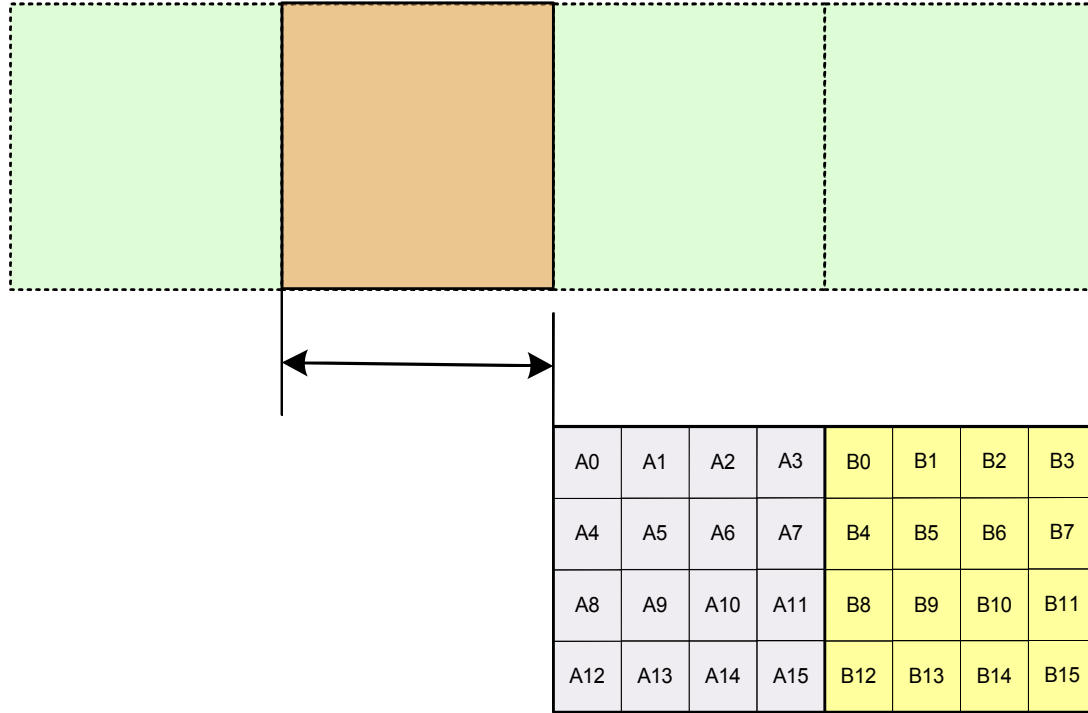


Fig.41. Mode 0 in the arrangement modes of block corresponding

In Fig.41, Ref(A) is the corresponding 16×16 block using the marked shift previously. Mode 0 is the luckiest situation that the marked shift is multiple of 16, so that we can use the encoding information of the referenced MB directly.

In the mode 1 shown in Fig.42, the shift's remainder is 4 or 12 pixels. For partition type prediction, it is used that the encoding information of the MB which is most close to the Ref(A).

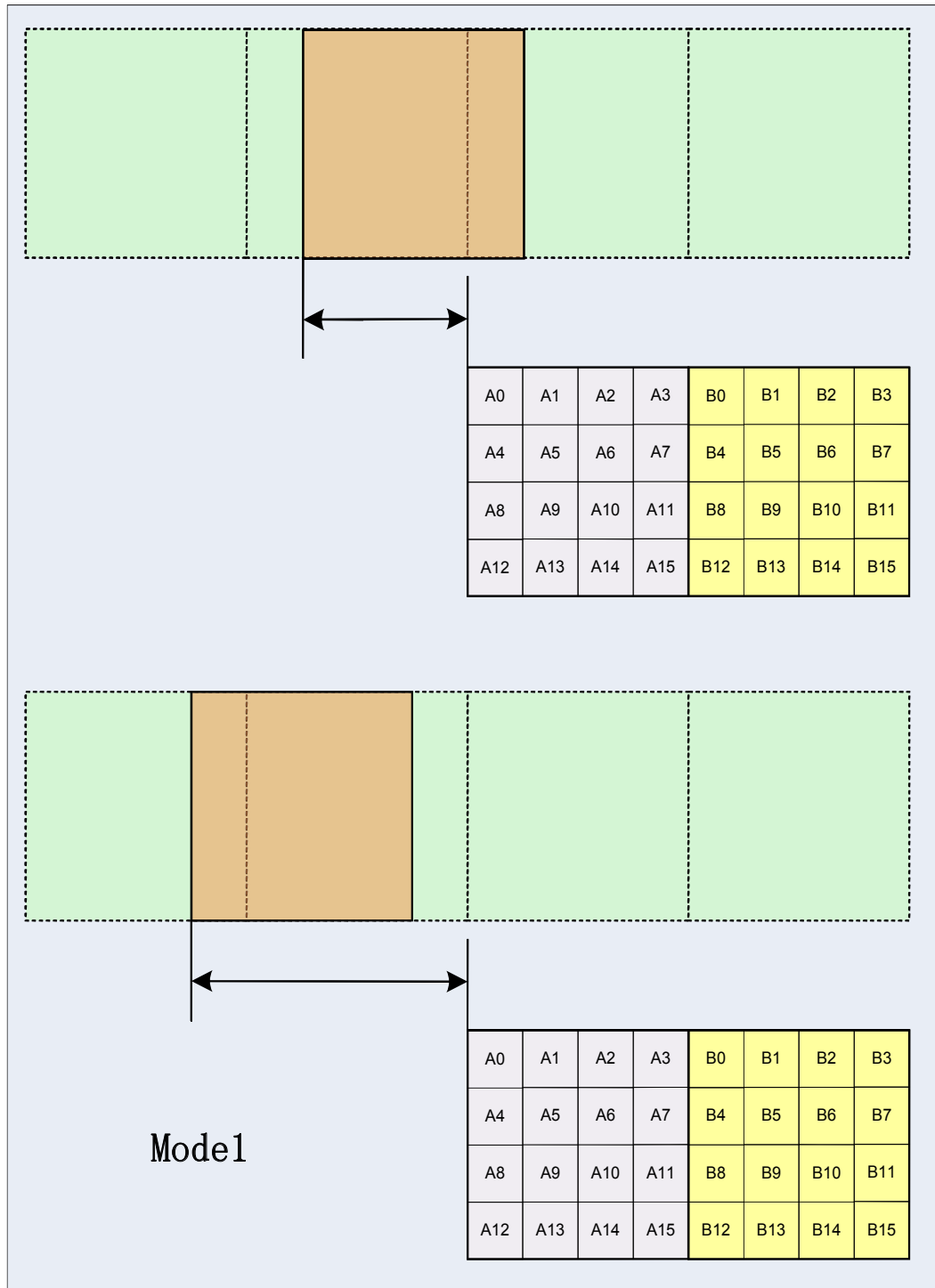


Fig.42. Mode 1 in the arrangement modes of block corresponding

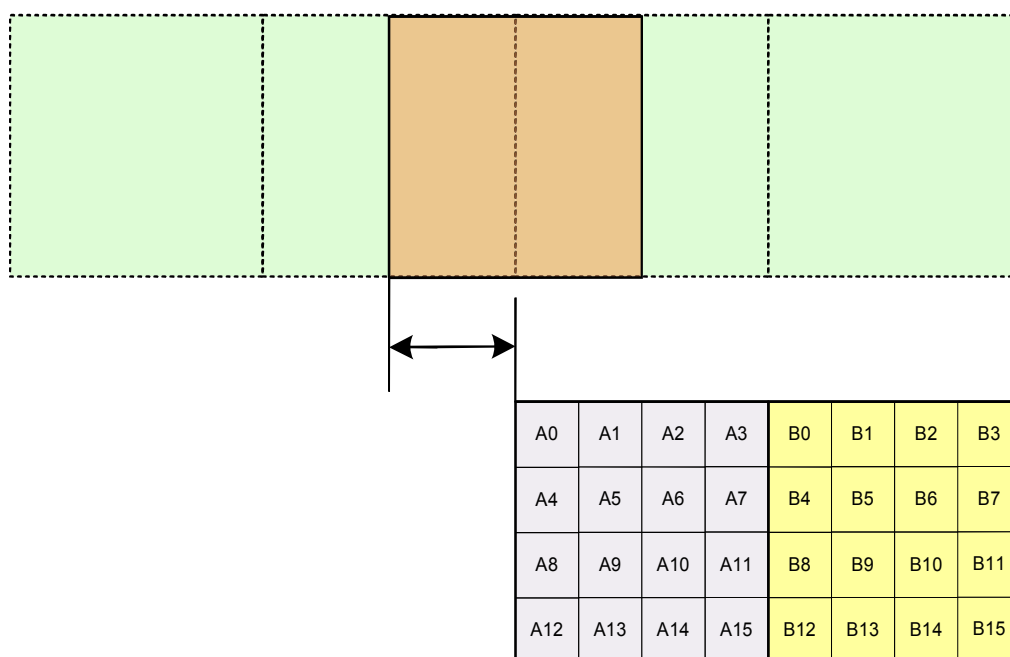


Fig.43. Mode 2 in the arrangement modes of block corresponding

Mode 2 is the worst case because the Ref(A) is half divided by two different encoded MBs, which is explained in Fig.43. 0.5 of occurrence of both partition types is counted when calculating the probabilities.

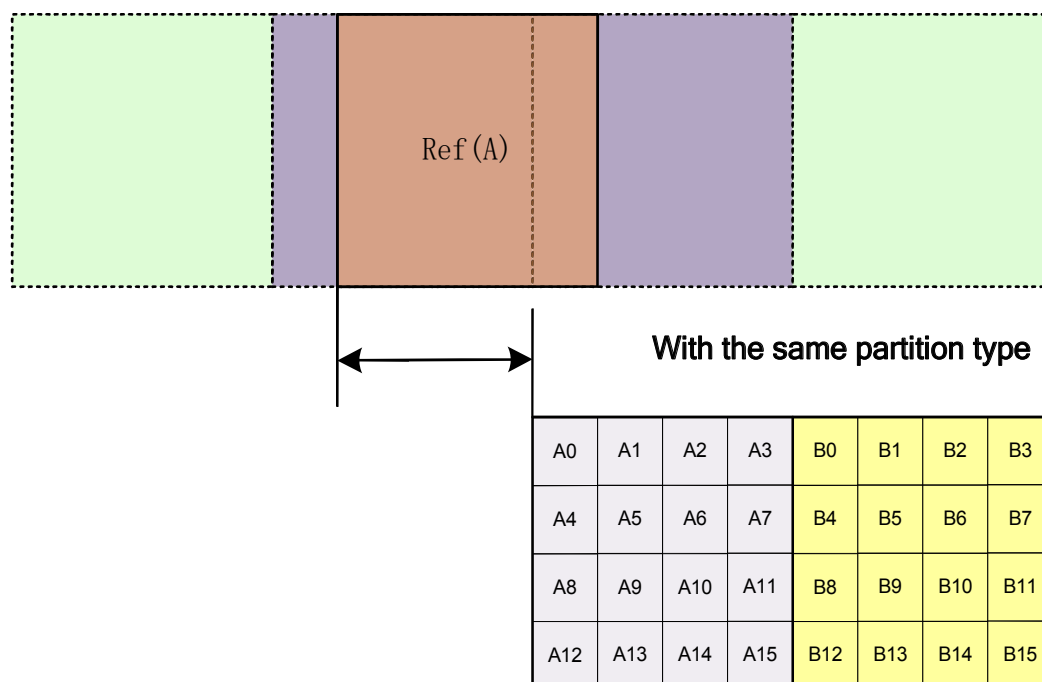


Fig.44. Mode 3 in the arrangement modes of block corresponding

If the neighbouring two MBs share the same partition type, it is Mode 3 and is shown in Fig.44. This situation needs to be cared about because if the Mode 3 can bring the highest hit rate, thus the different acceleration strategies can be performed on it.

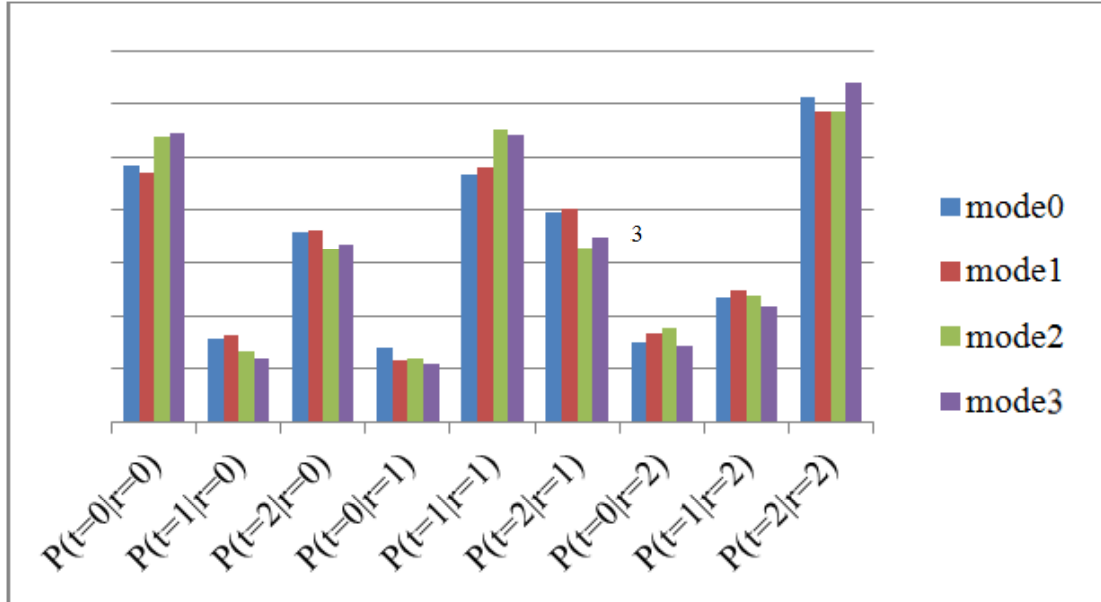


Fig.45. Probabilities of partition type using simple shift obtaining

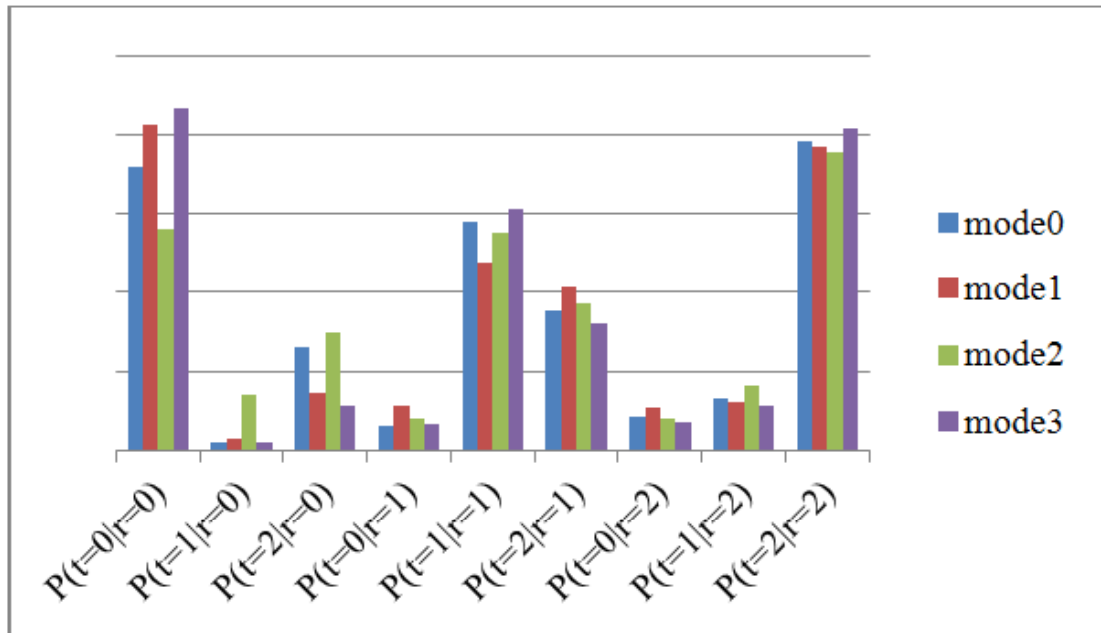


Fig.46. Probabilities of partition type using stereo matching

The probabilities of partition type are shown in Fig.45 and Fig.46, in which r means the reference mode and t means the target mode. 0, 1, 2 indicate the partition type I16MB, I4MB and I8MB respectively. It can be found that when the shift derived from stereo matching is used, the probability of get the correct prediction is much higher than using simple shift obtaining, especially in mode 3. That means the prediction of partition mode will be much more accurate in the stereo matching contained method. The experiment of prediction direction is similar and also shows this feature.

3.4.2 The modified stereo matching algorithm

To meet with the requirements of fast algorithm for FCF, the stereo matching should be as fast as possible. It can be noticed that in contradiction with the aiming of stereo matching, we do not need to pursue 100% accuracy in the middle depth results for shift marking. That means some points where the disparity is difficult to calculate can be jumped.

It is hard to obtain correct disparity in the areas of low texture and repetitive patterns, while it is much easier in the plane and texture areas. In sequences, the color of object is likely to be interfered by exposure or noise, which makes it difficult to calculate the disparity. We can leave it as 0 in the disparity value.

Based on this assumption, the modified stereo matching algorithm is proposed as follow.

3.4.2.1 Initial disparity calculation

An AD-Census cost is used here as the matching cost which combines the absolute difference and Census cost. This cost is firstly proposed in [68]. In order to make the matching fast, the cost aggregation isn't done before reliable disparity detection. The AD-Census can help to contain the 2D information in the initial cost, thus decrease the error of AD based matching. For Census, a 9×7 window is used to encode each pixel's local structure in a 64-bit string. C Census (p, d) is defined as the Hamming distance of

the two bit strings [69]. The AD-Census Cost is calculated by:

$$C(p, d) = \rho(C_{census}(p, d), \varphi_{census}) + \rho(C_{AD}(p, d), \varphi_{AD}) \quad (6)$$

Where

$$\rho(c, \varphi) = 1 - \exp(-c/\varphi) \quad (7)$$

After the matching cost of each pixel is obtained, the reliable disparity detection will be executed.

3.4.2.2 Reliable disparity detection and propagation

If the disparity of pixel satisfied the three detection rules, the pixel is accepted as a seed. Assuming that the matching pair is pixel P and P' and the seed set is S which contains all the seed points in a small 20*20 pixel neighborhood around P. We regard the matching pair P and P' and the seed set S are conditionally independent given a disparity d, then the joint distribution is:

$$p(d, P, P', S) \propto p(d|S, P) * p(P'|P, d) \quad (8)$$

Where $p(d|S, P)$ is the prior which means the possibility that P has the disparity d based on the seed set S, and $p(P'|P, d)$ is the image likelihood which implies the probability that P' is the corresponding pixel of P on another image providing a disparity d.

For the prior, at most 4 seeds are chosen surrounding P in to S (Such as S1, S2, S3, S4 in Fig.47). For every 3 pixels, a plane can be constructed by:

$$D_o = au_o + bv_o + c \quad (9)$$

Where $o(u_o, v_o)$ is a pixel on this plane. Once this plane is obtained, the $d(P)$ can be

easily interpolated based on the location of P. Then, the value of D_p is calculated on this plane based on the location of P.

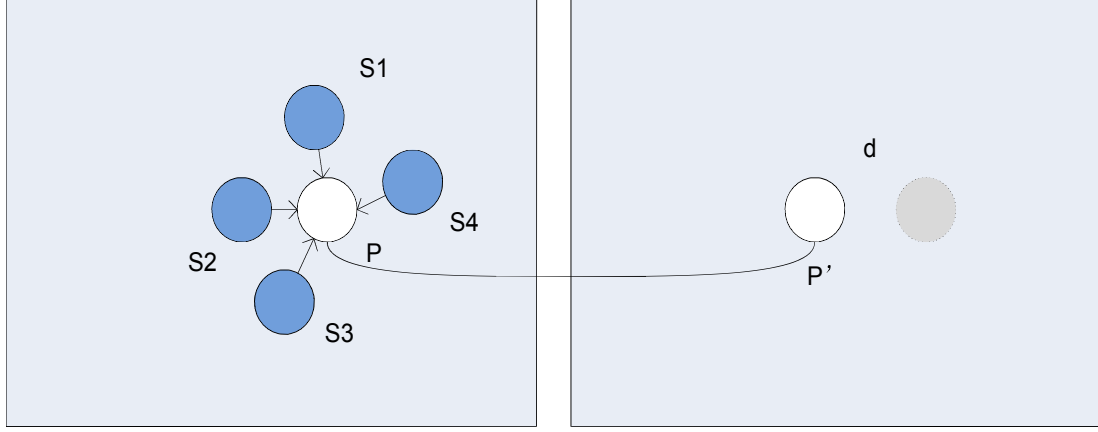


Fig.47. The process of reliable disparity propagation

Because at most 4 seeds are chosen, finally an average value is used to obtain the mean disparity μ as a recommended disparity of P. After the μ is calculated, the disparity estimation method in [70] will be used to define the prior:

$$p(d|S, P) \propto \gamma + \exp\left(-\frac{(d-\mu)^2}{2\sigma^2}\right) \quad \text{if } |d - \mu| < 3\sigma, d \in S \quad (10)$$

Then the image likelihood is expressed as:

$$p(P'|P, d) \propto \exp(-\beta ||f_P - f_{P'}||) \quad (11)$$

Where f_P is taken as the concatenation of image derivatives in a 5*5 pixel neighborhood around P, computed from Sobel filter responses. Based on the work [70], the parameters are set as $\beta = 0.03$, $\sigma = 3$. Finally, the d which leads to the highest probability as the disparity for P will be chosen.

3.4.3 Fast encoding algorithm for FCF using stereo matching

The flowchart of the proposed fast algorithm implements is shown in the Fig.48. The blue part is the added stereo matching related module. The bottom right part is the traditional FCF fast encoding algorithm. In our work, for each frame, the depth map for the bottom plane in the packed view is generated by proposed stereo matching algorithm. Once the result of stereo matching was obtained from the two packed views, the shift marking step described in Section 3.4.1 can be conducted. In the encoding process, the partition type and prediction modes are shrunk according to the encoded information in the referenced micro blocks. For different encoding modes of current MB, the leveled shift derived from the original depth map is used to find the reference block. It can be noticed that there are some black areas in the depth map, which indicate the pixels which are skipped in the stereo matching. In these areas, just the shift of the block on the left is used.

The fast algorithm is conducted for every block in plane_1. The encoder first analyses the type (I or P frame) of current frame. If it is I frame, the algorithm enters the left brunch. The marked shift in 16*16 micro block level helps to determine the reference MB's partition (Pplane_0 in Fig.48). According to the reference MB's partition, we decide the candidate partitions for current processing MB, and then use the marked shift in corresponded level to obtain the reference block's direction (Dplane_0 in Fig.48). Based on the prediction direction candidate set, we pass all the modes which are not in the candidate set so as to achieve fast encoding. If it is I frame, it only use marked shift in 16*16 micro block level to retrieve the reference MB's partition and partition prediction. The candidate set is the same with the traditional fast encoding algorithm for FCF.

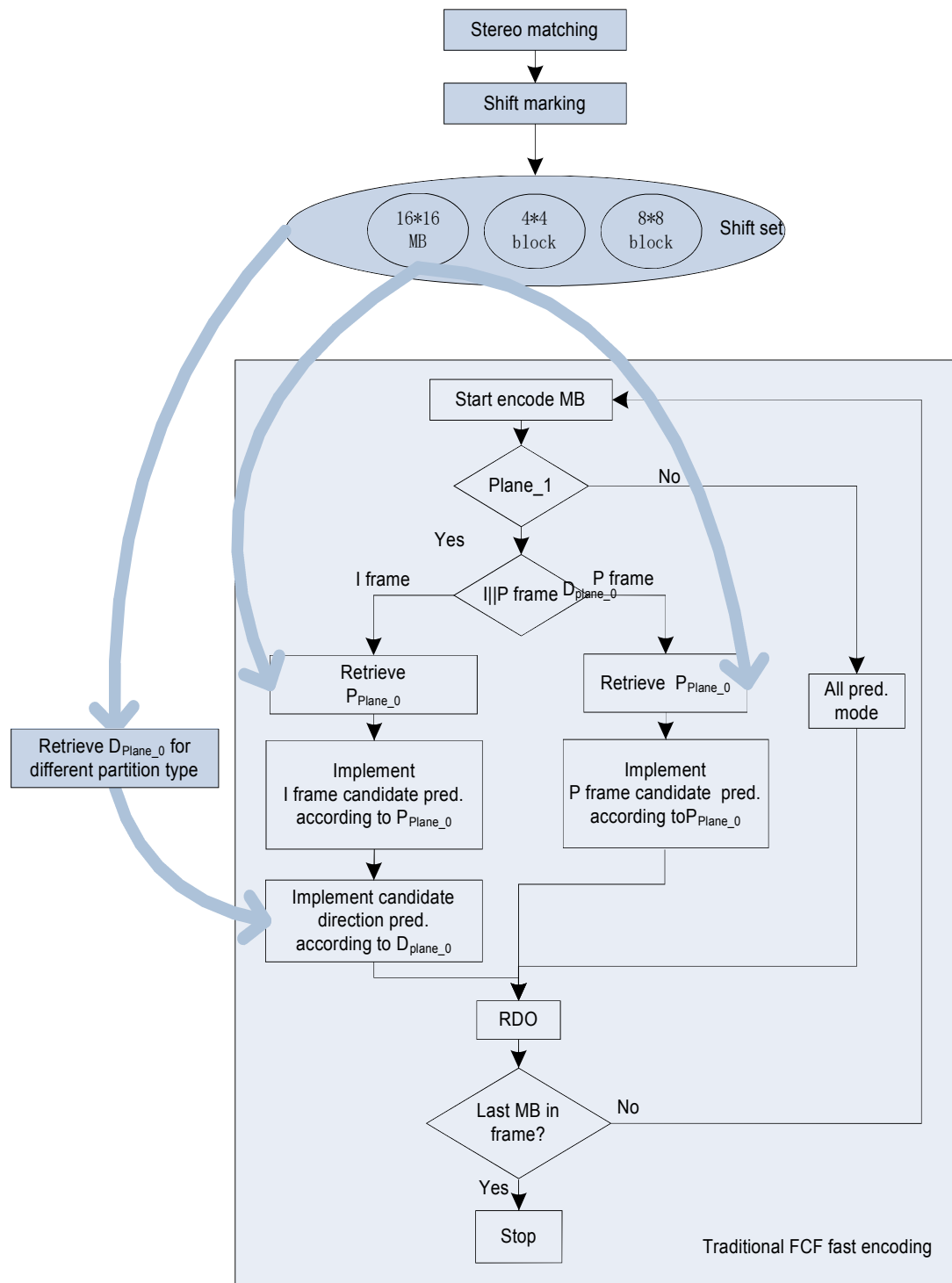


Fig.48. Flowchart of the new fast FCF encoding algorithm

3.4.4 Simulation results

The proposed fast algorithm has been integrated into the H.264/AVC reference software JM16.2 for the performance evaluation. Several top-bottom formats have been tested. In all of them, the left and right views are sub-sampled by the default down sampling tools in the software JSVM and merged together with C program before encoding with JM16.2 software.

Table 9 The encoding results of previous FCF fast encoding [67]

Sequence	BD-PSNR (dB)	BD-BR (%)	fps (%)
News34v	-0.29	5.18	67
Flower78	-0.39	8.75	64
Exit34	-0.22	5.17	63
vassar	-0.15	3.43	64

Table 10 The encoding results of FCF fast encoding by using stereo matching

Sequence	BD-PSNR (dB)	BD-BR (%)	fps (%)
News34v	-0.22	3.95	64
Flower78	-0.22	4.90	63
Exit34	-0.31	7.30	51
vassar	-0.14	3.40	57

It can be found that basically, the stereo matching can achieve a better result than simply using a shift obtaining method. It can achieve both less PSNR loss and the lower bit-rate increment. The result is not good in some sequence like Exit 34, in which the quick moving object is too closed to the camera so that it is impossible to generate correct depth information.

Table 11 The comparison result with the previous work

Sequence	Views	Without shifting method			The previous method			The proposed method		
		BD-PSNR (dB)	BD-BR (%)	ΔT (%)	BD-PSNR (dB)	BD-BR (%)	ΔT (%)	BD-PSNR (dB)	BD-BR (%)	ΔT (%)
News	3,4	-0.30	5.25	33.47	-0.29	5.18	32.61	-0.22	3.95	31.15
Flower	7,8	-0.41	8.96	36.94	-0.39	8.75	36.15	-0.22	4.90	35.59
Vassar	3,4	-0.15	3.44	36.67	-0.15	3.43	36.25	-0.14	3.40	32.29
Mobile	3,4	-0.42	8.65	35.76	-0.42	8.60	35.40	-0.25	5.13	34.54
Book	8,9	-0.29	5.10	33.20	-0.28	4.98	32.47	-0.20	3.78	31.23

Through the comparison results with the previous work, we can find that the proposed method can reduce the BD-Bandwidth and improve the BD-PSNR, but increase the coding time a little.

3.4.5 Conclusion

The reconstruction of a stereo scene from a pair of images taken from different directions is one of the classical problems in computer vision area. Nowadays, along with the huge development of computer performance and 3D video market, stereo matching plays an important role in the depth generation. Besides, Frame-compatible format (FCF) is considered as one of the most promising solutions of 3D distribution on the existing system, as it is completely compatible for the existing video codec, such as H.264/MPEG-4 AVC and MPEG-2. Researches have showed that content similarity in two packed views in FCF can lead to prediction correlation in encoding. This feature is used when researchers design the fast encoder for Frame Compatible Format.

The previous work of fast encoder for FCF uses a simple block based shift obtaining method, which cannot figure out the accurate content correlation. The stereo matching algorithm is definitely the best choice to reveal the content matching in the

packed view, which is quite suitable to be used in this research. The difficulty is that stereo matching algorithm usually costs lot of time and computational resource. If it needs too much time to calculate the depth map, the fast algorithm becomes useless.

Traditionally, the process of local stereo matching is divided as cost computation, support aggregation and disparity refinement. In the proposed method, the process of traditional matching algorithm is rearranged. The proposed algorithm reduces computational complexity and ensures the accuracy of the result at the same time. Through the experimental results, a better FCF fast encoder is proposed by implementing the stereo marching algorithm into it. The proposed algorithm can achieve both less PSNR loss and the lower bit-rate increment. Besides, no one has combined stereo matching into FCF fast encoder research currently. This research shows the promising possibility of this field.

4. The Novel Hole-filling Algorithms for View Synthesis

4.1 Introduction

In recent years, three-dimensional (3-D) products have become increasingly popular in daily life. In most traditional 3-D multimedia systems, only one pre-determined viewpoint of an image or video can be seen by an observer. If the viewpoint is changed, the realistic 3-D impression will become much weaker and the quality of the 3-D video will worsen. To increase the number of viewpoints for the observers and image more comfortable for viewers, the free viewpoint television (FTV) was introduced. The interest in FTV has been continuously increasing in recent years. Auto-stereoscopic displays provide a 3-D impression to an observer without the need to wear additional glasses, and the observers can enjoy a realistic 3-D impression from certain different viewpoints. Such a display shows a number of slightly different views at the same time. To simultaneously deliver so many views, an extremely large amount of bandwidth is required.

Therefore, view synthesis was introduced to solve this problem. Depth-image-based rendering (DIBR) [48] is a technology for synthesizing novel realistic images at a slightly different view perspective using a textured image and its associated depth values. DIBR is used to generate additional virtual views of a real-world scene from an image or video, as well as the associated per-pixel depth information. 3-D warping is a key technique used in DIBR. In 3-D warping, pixels in a reference image are back-projected to 3-D spaces, and then re-projected onto the target viewpoint. An inherent problem of the view synthesis concept is that image information occluded in the original view may become visible in the “virtual” image. Some holes will then appear in the virtual image, which can be also called zero-regions. The information of an occluded region in the original image is lost in

the virtual image and needs to be concealed. Therefore, a hole-filling technique is necessary and in-painting is the most popular method for solving such hole-filling problems.

Some researches [49] [50] have already been done in this field. A simple approach repeats the last valid background sample line-wise into the zero-region [51]. View Synthesis Reference Software (VSRS) [52] also provides a solution by blurring the boundary of the foreground objects. However, structural information is very important in 3-D cases. Criminisi et al. noted that exemplar-based texture synthesis contains an essential process required to replicate both texture and structure. The Criminisi's in-painting algorithm [53] conceals a zero-region with a priority order based on the texture and structure information. However, this kind of priority order is not sufficient for the prediction of such information over the long-term.

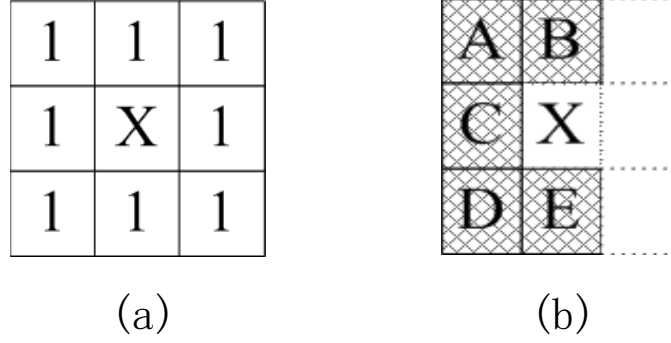
4.2 Previous works

Some works are already done for the image in-painting. I will introduce them and simply classify them in the following section:

4.2.1 Mask Used In-painting

This kind of in-painting method [71] uses the neighbor pixels to predict the value of the unknown pixel by giving some different weight values to the different neighbor pixels.

Fig.49 shows a simple example of the in-painting method with mask. In (a), A, B, C, D, E are the known neighbor pixels, X is the unknown pixel. In (b), it shows the weight of the neighbor pixels.



$$X = (A + B + C + D + E) / 5$$

Fig.49. Simple mask in-painting

Undefined pixels X can be interpolated as:

$$V'_X = \frac{1}{k} \sum_{q \in W_X} V_q$$

where k is the number of defined pixels within W_X .

- V_X is the pixel color at the X position which one at the boundary of the holes
- W_X is a neighborhood around X in the non-zero region.

4.2.2 The Background Region In-painting

The general in-painting problem is as follow [50]: the region to be in-painted Ω and its boundary $\partial\Omega$ are defined and the pixel p belonging to Ω would be in-painted by its neighboring region $B \in (p)$ as shown in Fig.50.

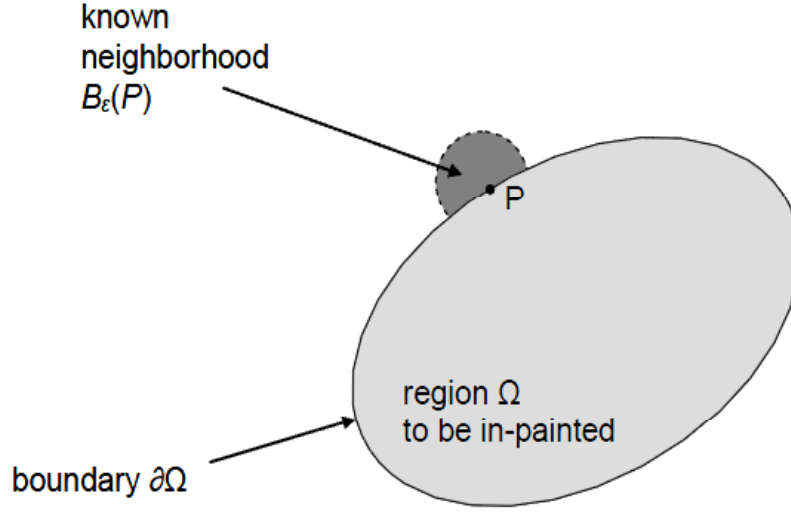


Fig.50. General in-painting circumstance

This concept is quite reasonable for an ordinary picture in-painting, but because a certain hole $\partial\Omega$ can be located in the foreground and background, it should be changed to apply to the hole filling in the view synthesis. In this case, they replace the boundary area adjacent to the foreground, and depict the background area on the opposite side in the following equation:

$$p_{fg} \in \partial\Omega_{fg} \rightarrow p_{bg} \in \partial\Omega_{bg}$$

$$B_{\varepsilon}(p_{fg}) \rightarrow B_{\varepsilon}(p_{bg})$$

where fg and bg mean the foreground and background.

That is to say, they deliberately manipulate the hole so that neighbors can only come from the background, as shown in Fig.51. Now, because the hole has only background pixels, it is more natural in the in-painted images than the previous ones.

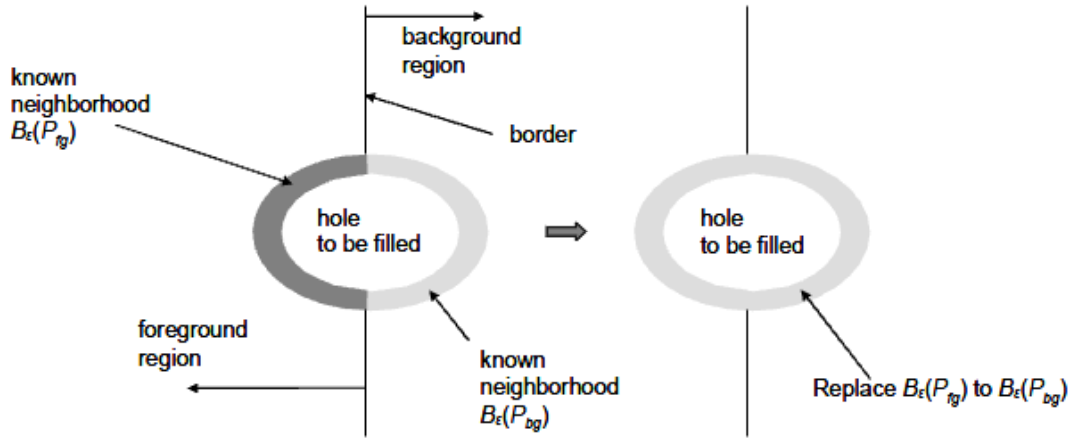


Fig.51. Manipulation of hole to have neighborhood only come from background

In order to distinguish foreground from background, they use corresponding depth data to mix images. For the two depth pixels that are horizontally opposite on the hole, they use pixels with larger depth values as the foreground.

4.2.3 The In-painting Based on Some Special Priority

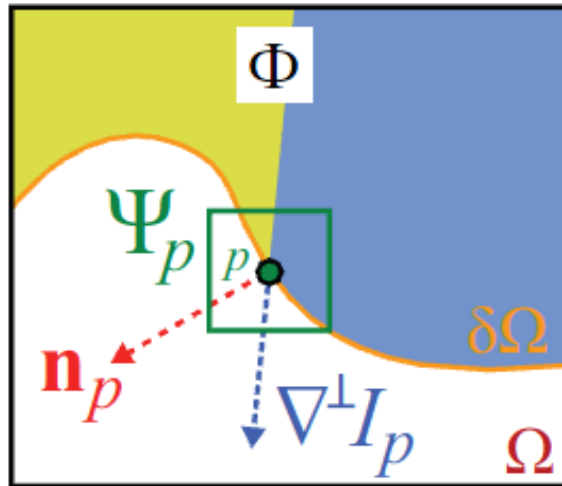


Fig.52. Notation diagram

The most classical algorithm in this kind of in-painting is Criminisi in-painting method. Criminisi et al. [53] first noticed that sample based texture synthesis involves the basic processes needed to replicate texture and structure, and uses the sampling concept [73] of Efros and Leung's methods.

[72] is an improved work of the Criminisi in-painting method in the recent year. These works only use the short term isophote tendency to do the prediction, and it is not enough. Therefore, our proposed algorithm is introduced.

4.3 An integrated hole-filling algorithm for view synthesis

4.3.1 Architecture of the integrated hole-filling algorithm for view synthesis

The architecture of the proposed process is shown in Fig.53. It starts with the algorithm for distinguishing between the different layers. This algorithm divides a texture image into several layers based on the depth information. It can distinguish foreground information from the corresponding background information.

The next step is the foreground and background boundary detection algorithm. This algorithm is used to clarify the boundary information between the foreground objects and the zero-region clearly and to determine the primary in-painting order of the background.

Textural and structural isophote detection is the third step. This step is mainly used to determine the visible textural and structural information, which have a stable and long-term trend in the background side and next to the zero-region. Textural and structural lines are called isophotes, and are used to predict the clear textural and structural information in the zero-region.

The fourth step is the isophote prediction algorithm. The prediction process is based on a geometry principle. The detected textural and structural information in the third step will be fitted by different curves, the function of which is to predict the tendency of the textural and structural information. Thus far, the zero-region has been divided into several parts based on this type of information.

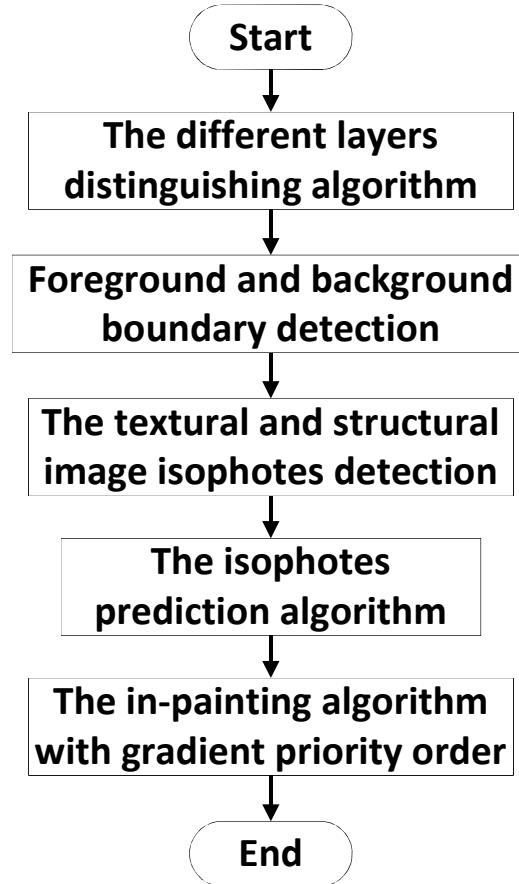


Fig.53. Architecture of the proposed method

The last step is the in-painting algorithm with gradient priority order. Based on the primary in-painting order, which is determined in the second step, this algorithm introduces an advanced priority order using the gradient information. It searches and compares the information of the boundary between the zero-region and the background, paints the position with the highest priority, and updates the boundary information. This process is iterated until there are no suitable positions that satisfy the priority condition. After these steps have been completed, some small holes may

remain in the virtual image. Average in-painting, which uses the average value of the neighboring pixels, is then used to fill in these holes.

4.3.2 Details of the hole-filling algorithm

4.3.2.1 Algorithm for distinguishing different layers

This algorithm distinguishes the different layers based on the depth information. It determines the relative foreground and background layers in the texture image and is an important base for the next steps. Its accuracy will directly affect the hole-filling results.

Some of the previous methods, such as VSRS [14], also check the foreground and background layers initially. However, these methods only check one pair of points on both sides of the nonzero-region and zero-region at the same horizontal level to determine the relative foreground and background. Most of the depth map is generated by a depth camera, which is not very accurate. Therefore, if only one pair of points is checked, some errors will occur when the depth information has artifacts.

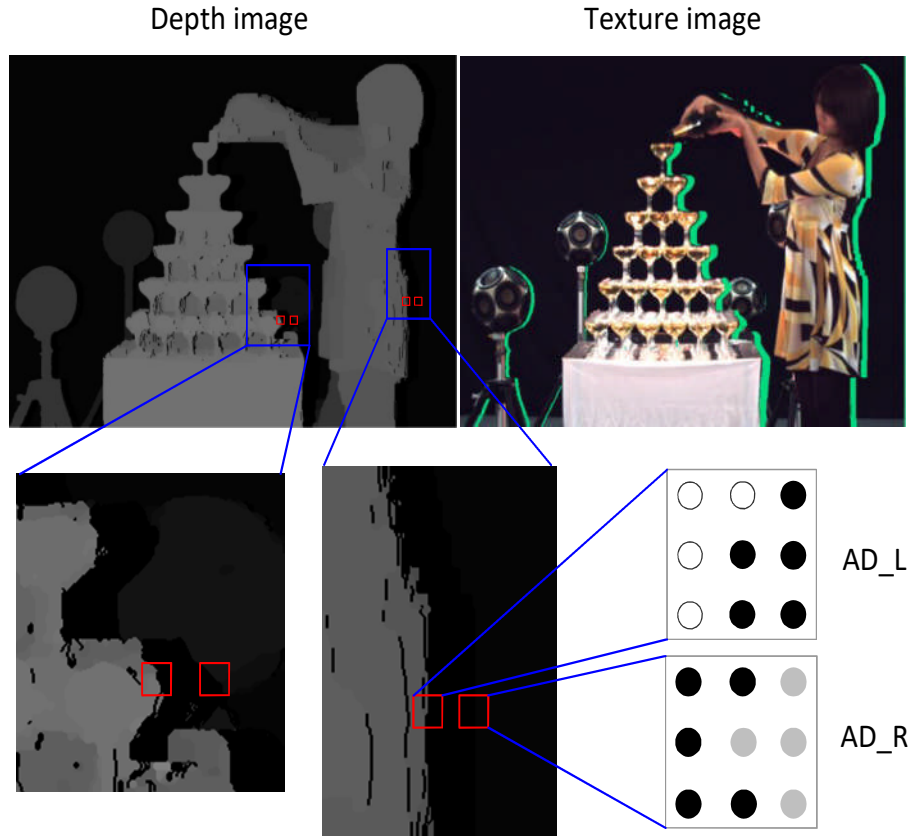


Fig.54. Illustration of the distinguishing process [84]

An illustration of the proposed distinguishing algorithm is shown in Fig.54. One pair of patches, which includes both the nonzero-region and zero-region at the boundary of the holes at the same horizontal level in the depth image, is first obtained. One point means one pixel, and one patch means one 3×3 block which consists of 9 pixels in all of the experiment in this paper. The average depth value of one patch which means the average depth value of the 9 pixels should then be obtained for a comparison.

In Fig.54, it can be found that the depth information become much more inaccurate after 3-D warping. It can be also found that If only one pair of points is used for the layer decision, the left side will be considered as the relative background layer and the right side will be considered as the relative foreground layer in the case which is shown in Fig.54. Obviously such result is totally wrong. If one pair of patches is used for the layer decision, this kind of errors can be avoided. Therefore,

the two-patch comparison process is considerably more stable than the two-point comparison process, and can provide a much more accurate determination of the different layers.

There are two kinds of situations shown in the comparison results. For the first, if one of the depth values is much bigger than the other one, the layer that contains the larger value will be determined as the relative foreground layer in the texture image, while the layer that contains the smaller value will be determined as the relative background layer in the texture image. For the second situation, if the average depth values of the two patches are almost identical, it indicates that a hole has occurred through the 3-D warping process, resulting in inaccurate depth information. In this type of situation, both sides of the holes will be determined as the background layer.

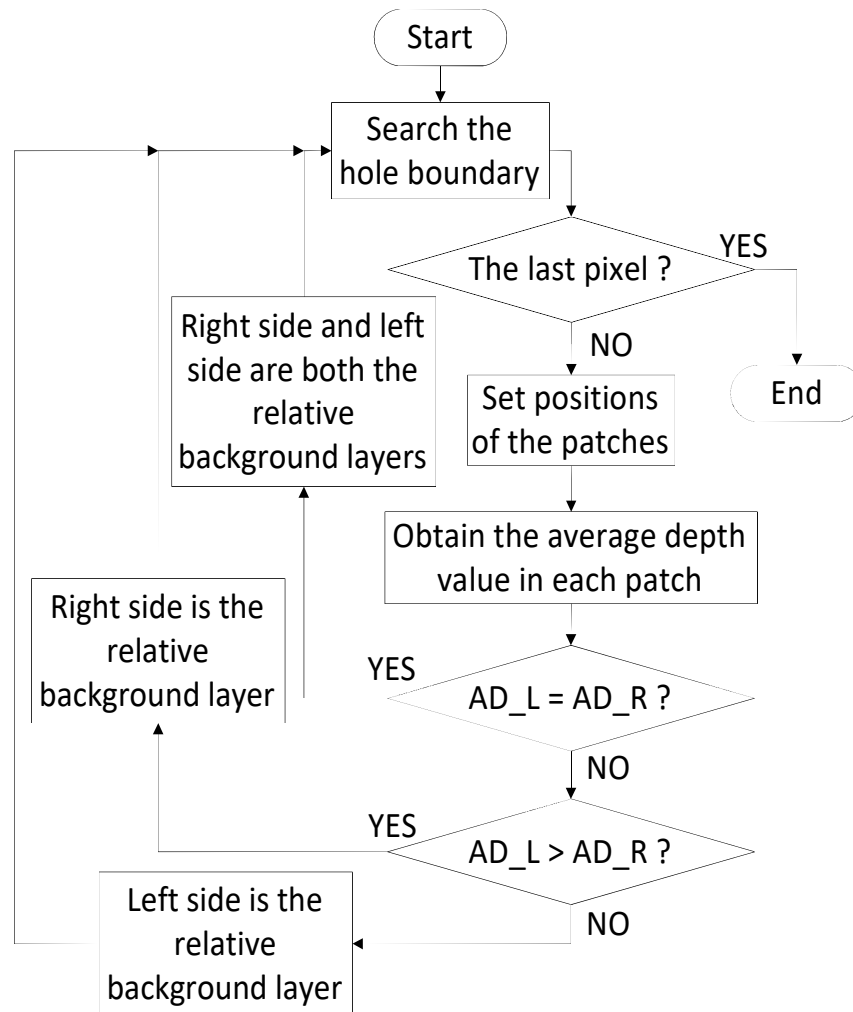


Fig.55. Flowchart of the distinguishing algorithm

The flowchart of the distinguishing algorithm is shown in Fig.55. AD_L is short for average depth value of the left patch and AD_R is short for average depth value of the right patch. If AD_L equals to AD_R , the right and left side will be both considered as the relative background layers. If AD_L bigger than AD_R , the right side will be considered as the relative background layer. If AD_L smaller than AD_R , the left side will be considered as the relative background layer.

4.3.2.2 Foreground and background boundary detection

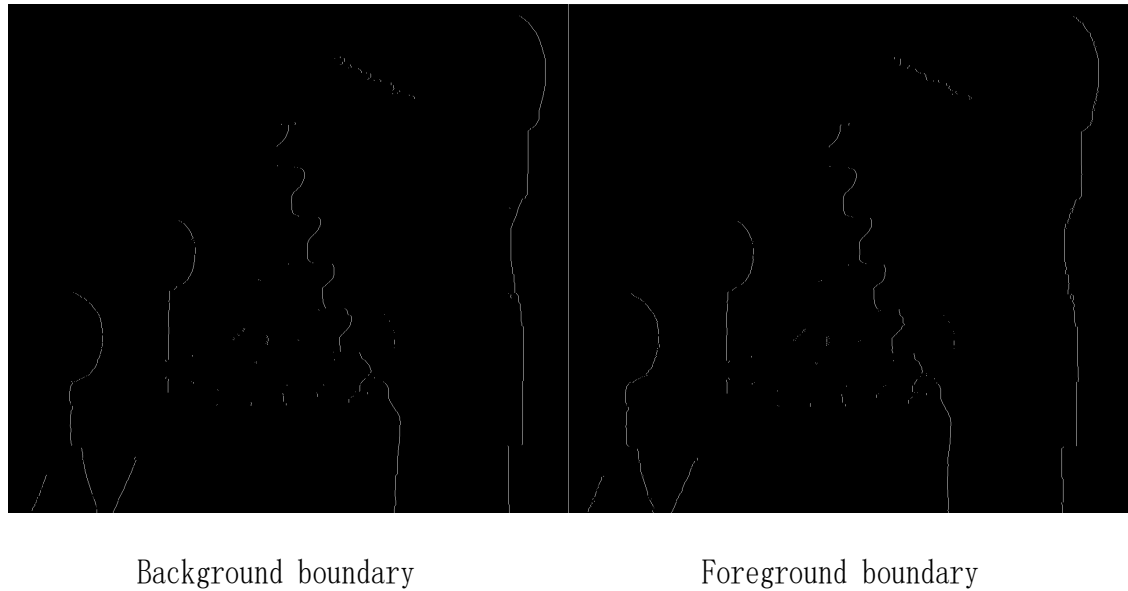


Fig.56. Result of the boundary detection

Based on the first step, the boundary of the different layers can be detected and recorded, and this information between the zero-region and the different layers should be separated into two types. The first type is the relative foreground boundary information, and the other is the relative background boundary information. These two types of the information are detected and recorded as shown in Fig.56. The relative foreground and relative background boundary information are recorded in two binary maps pixel by pixel respectively. Searching the boundaries around the zero-regions, if it is next to the relative background layers, this point should be

recorded as the relative background boundary information in the relative background boundary binary map. In the similar way, the relative foreground boundary binary map can be obtained.

The relative foreground boundary information is used to ensure that the in-painting process will not blur the boundary of the foreground objects, which is a problem that exists in several previous methods. In the previous in-painting process, the zero-region is painted, which blurs the boundary of the foreground objects at the same time. Textural and structural information is very important in this type of in-painting cases for virtual view. If the boundary blurs, the final 3-D impression will be affected significantly, even if the objective quality is improved. The relative background boundary information is used as a primary in-painting priority order, and is important basic information for the next step in the textural and structural isophote detection.

4.3.2.3 The textural and structural isophote detection

In this step, a Canny edge detection algorithm is used to obtain all of the edge information from gray-scale version of the texture image. The gray scale image is first obtained. Based on this gray-scale information, a binary edge map is calculated using the Canny edge detection algorithm.

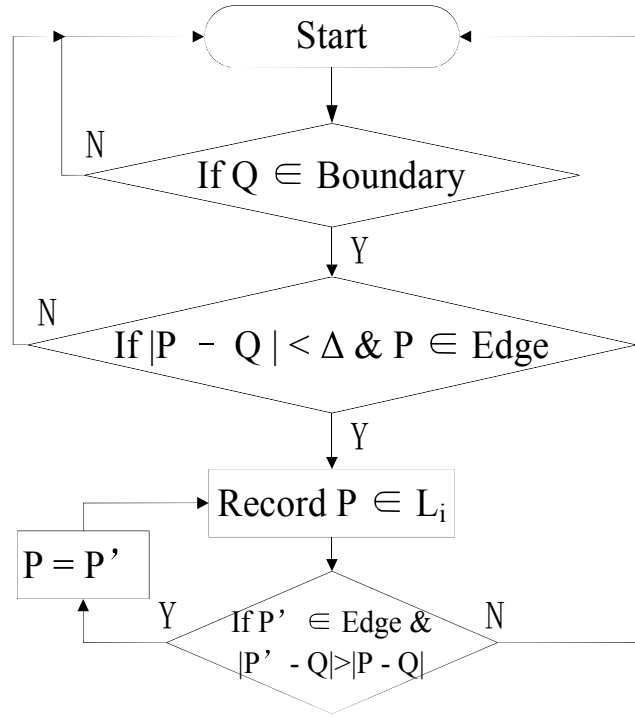


Fig.57. Flowchart of the detection process

The flowchart of the detection process is shown in the Fig.57. The process works on the basis of the boundary information of the relative background and the binary edge map. Each point on the binary boundary map is first checked as point Q . If the point Q is found on the boundary map and P is on the edge map, the distance between Q and P is checked. If the distance is smaller than Δ (where Δ is the threshold used to control the continuity of the detected isophotes), point P is recorded as the first point of the detected isophote, L_i ($i = 0, 1, 2 \dots, n$). The neighbor points P' are then checked. The threshold is set as 2 by manual in all of the experiments in this paper. That means only when the point P is a neighbor point of the point Q , the point P' is on the edge map and the distance between Q and P' is bigger than the distance between Q and P , P' is recorded into the detected isophote, L_i . The detection process should be iterated repeatedly until all of the isophotes have been discovered. The textural and structural isophote detection results can then be obtained as shown in Fig.58.

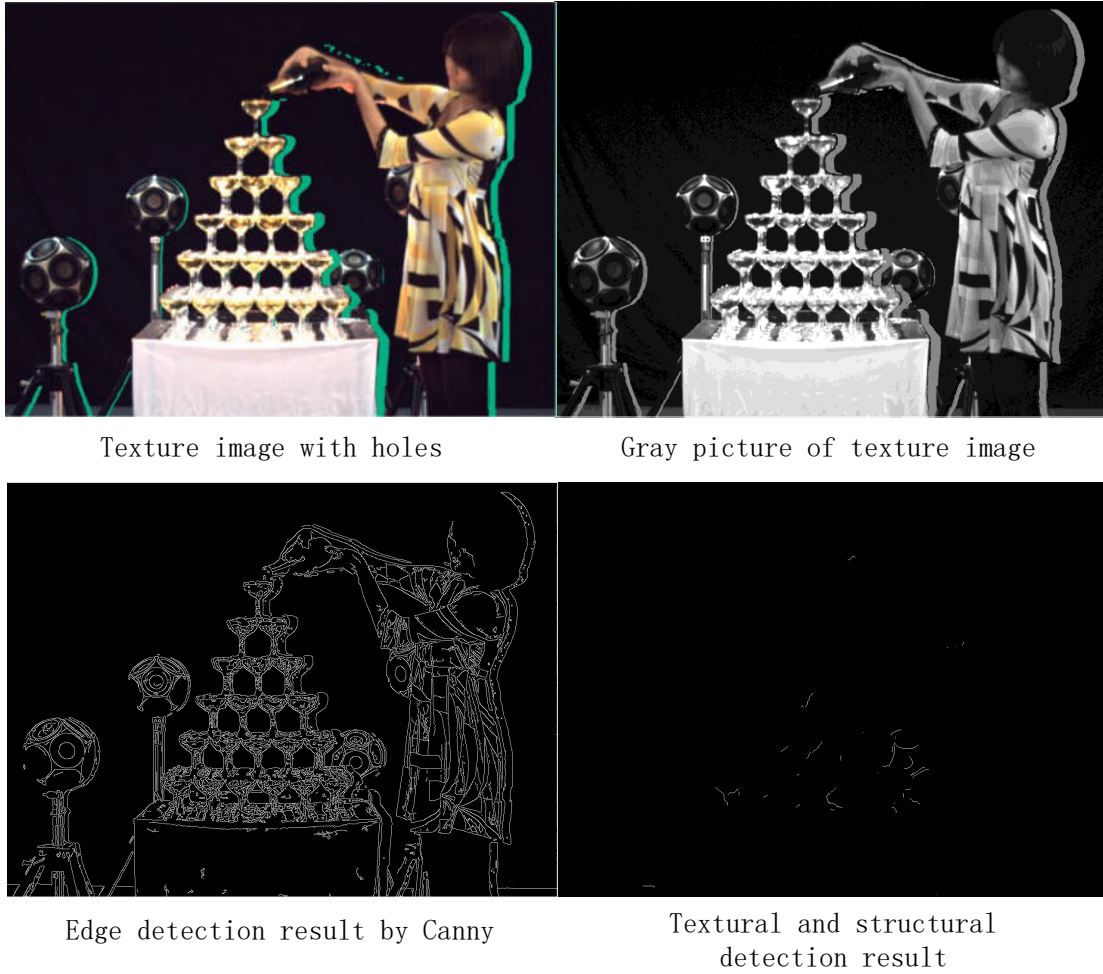


Fig.58. Steps and the results of textural and structural detection [84]

4.3.2.4 Isophote prediction algorithm

Based on the results of the previous step, the textural and structural isophotes can be predicted in this step. The proposed algorithm uses curve functions to fit the existing isophotes, and predict the unknown isophotes in the zero-region are predicted using these curve functions.

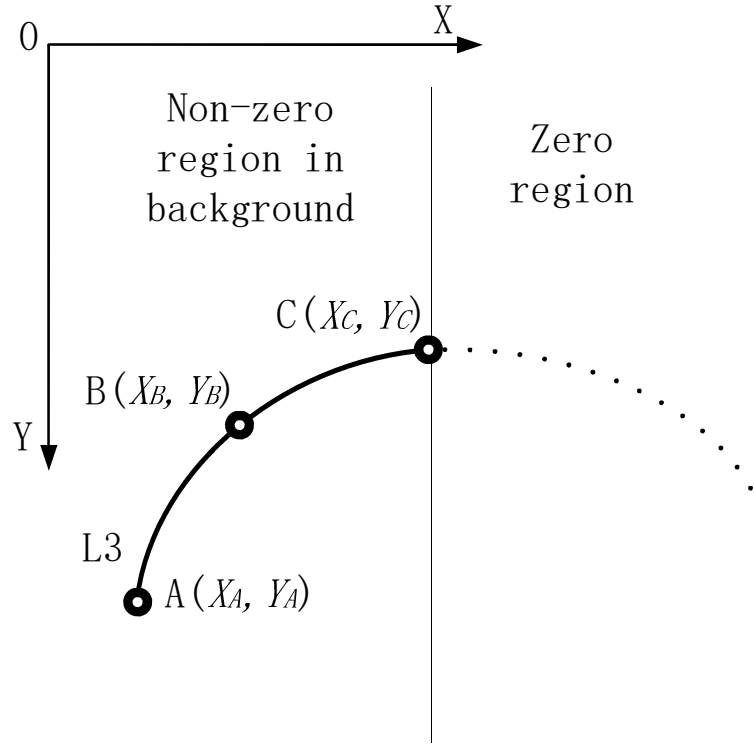


Fig.59. Illustration of curve fitting and prediction

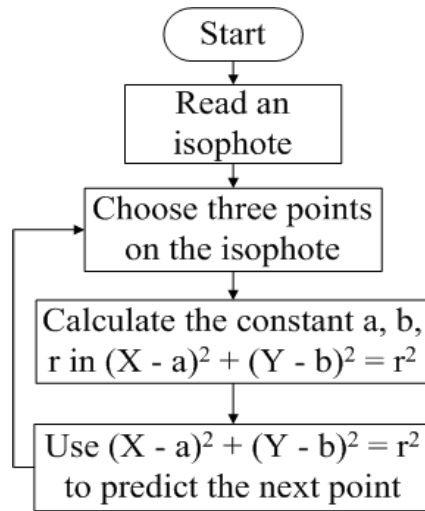


Fig.60. The flowchart of the isophote prediction

The curve fitting and prediction are shown in Fig.59 and the flowchart of the isophote prediction is shown in Fig.60. In the proposed method, the following equations are used to fit the different isophotes.

$$\begin{cases} (X - a)^2 + (Y - b)^2 = r^2 & (1) \\ Y = aX + b & (2) \end{cases}$$

Three points on the existing isophote are used to solve the quadratic equation (1) and obtain the constant values of a , b , and r , and to use the equation for predicting the next point on the isophote in the zero-region. Point A is always the first point on the isophote line, while points C and B are the newest-added point and the midpoint of the isophote line, respectively. Point B changes when point C is updated. This process should be repeated after one point is added into the existing isophote.

The basic idea of this step is to predict the textural and structural isophotes that have a stable long-term trend. If the curves that need to be fitted are very complex, they will be considered unstable and unpredictable. Therefore, only a quadratic equation is sufficient to describe the tendency of existing isophotes.

Threshold T is set for certain special cases. If radius $r > T$, then r will be considered infinite and the isophote will be considered as a beeline. The threshold T is used to check if the isophote is a beeline, so it should be a large value and it is set as 5000 by manual in all of the experiments in this paper. In this type of a situation, the equation (2) is used to predict all points of this isophote in the zero-region.

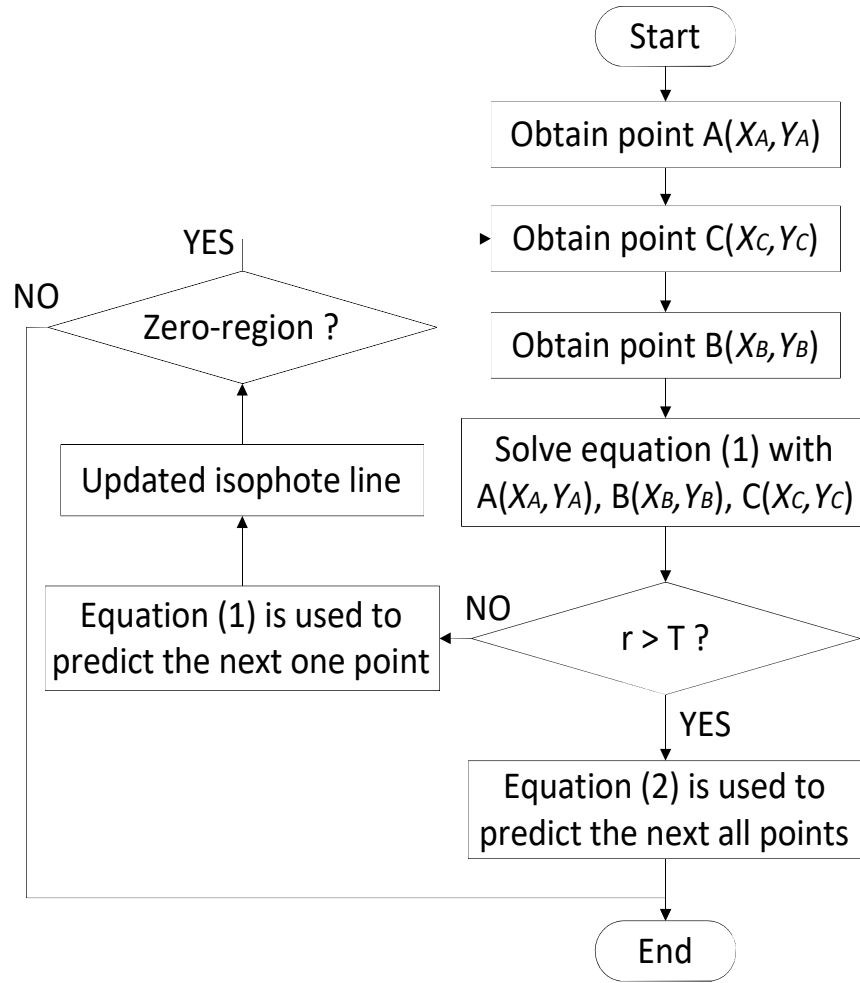


Fig.61. The flowchart of the predicting process

Fig.61 shows the flowchart of the predicting process. Point A, B and C are obtained at first. Solve equation (1) with (X_A, Y_A) , (X_B, Y_B) , (X_C, Y_C) and obtain the constant values a , b , r for the equation (1). The a and b are the parameters about the centre of the circle which contains the predicted curve and r is the curvature radius of this curve. If r is bigger than threshold T , this curve will be considered as a beeline, and the equation (2) will be used to predict the rest unknown pixels in the zero-region. If r is smaller than threshold T , the next point will be predict with the equation (1), and updated the new predicted point into the existing isophote line. This process should be repeated until the next point is not in the zero-region.

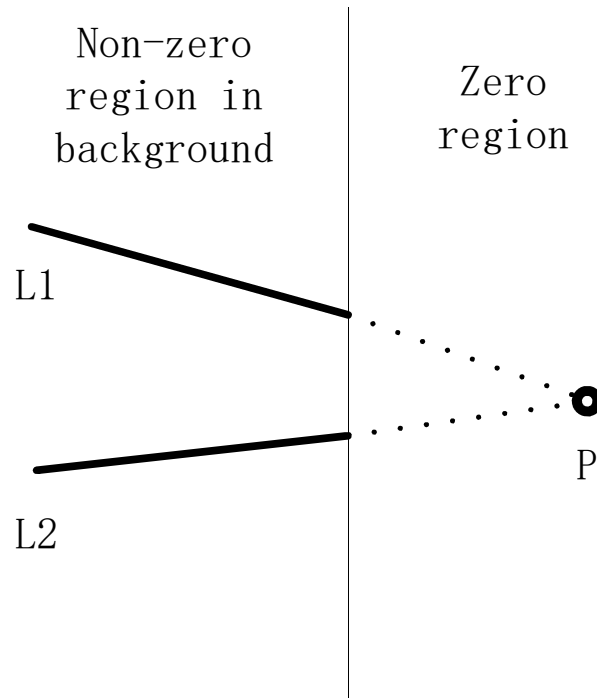


Fig.62. Illustration of the intersection determination

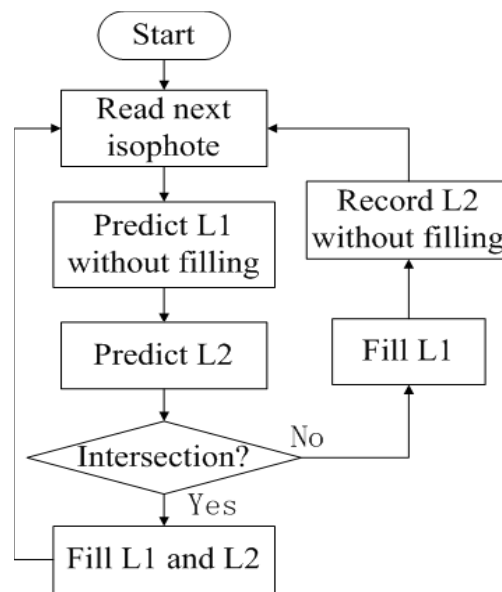


Fig.63. The flowchart of the intersection determination

There is another special situation in which two predicted isophotes intersect in the zero-region. The solution for this is shown in Fig.62 and the flowchart of this process is shown in Fig.63. When the prediction of isophote L1 is completed, this isophote is recorded without filling it with texture pixels. Until the prediction of the next isophote, L2, is finished, L1 and L2 have a point of intersection, P, which is then

set as the end of these two isophotes. If L1 and L2 have no point of intersection, then L1 is filled with the texture pixels, and L2 is recorded as the new L1; the previous process is then repeated with the next new L2.

4.3.2.5 In-painting algorithm with gradient priority order

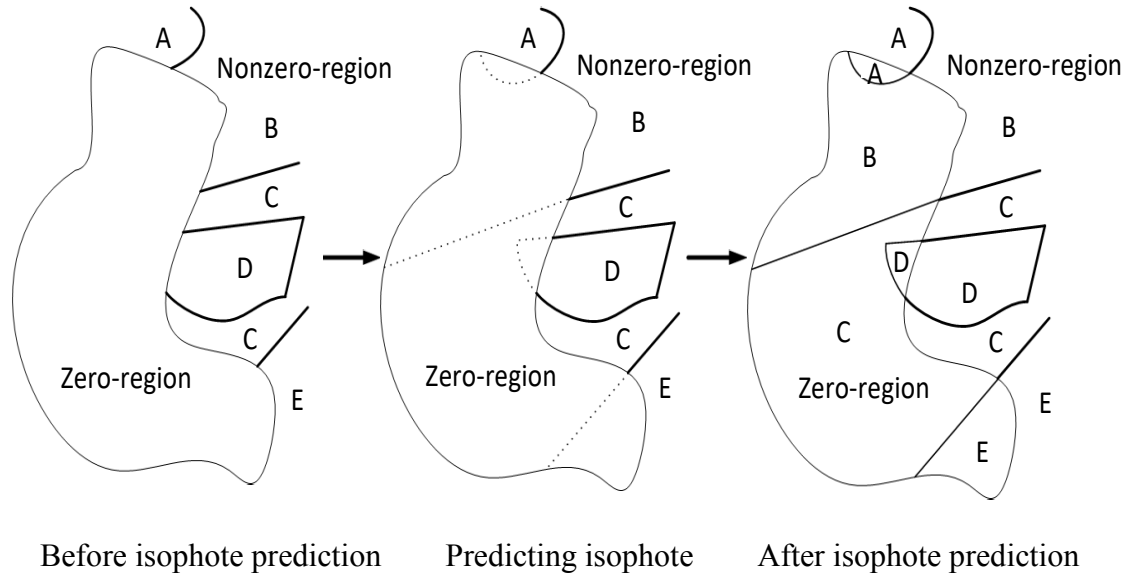


Fig.64. Illustration of the isophote prediction process

Up to this point, by following the prediction process, the zero-region has already been divided into several sections. The isophote prediction process and results of this are shown in the Fig.64. The in-painting algorithm with gradient priority order is then introduced.

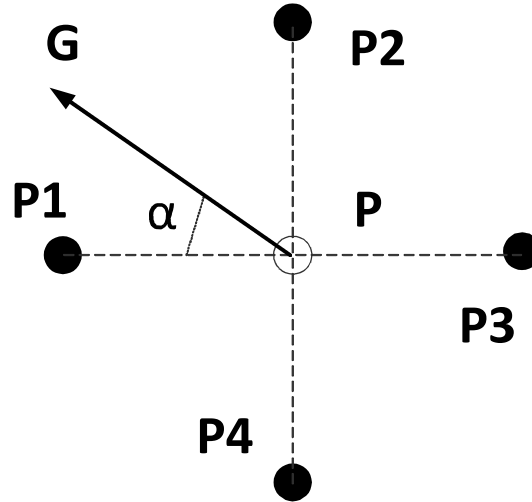


Fig.65. Illustration of the gradient information

David G. Lowe [54] had already proved that the nearest neighbor points can describe the direction features of the current point much more stable than the other points. Therefore the nearest 4 points have been chosen to calculate the length of the gradient and the direction of the gradient for the current point.

As shown in Fig.65, P is the current pixel: P1, P2, P3 and P4 are the four neighboring pixels; and G and α are the length and direction of the gradient, respectively. If the neighboring pixel is in the zero-region, the value of the neighboring point is set to zero. The boundary map of the relative background layer should be first searched, and the length and direction of the gradient of each pixel on the boundary map should then be calculated. The point that has the longest gradient length and a neighboring point in the zero-region in the direction of the gradient will be chosen as the prediction source. After filling this predicted point in the zero-region, the point should be added to the chosen prediction candidates. This process is iterated until there are no suitable pixels remaining. During the selection process, eight neighboring pixels can be filling candidates for the current pixel. Therefore, the direction of the gradient should be separated into eight different directions. After eight directions prediction, there are still some zero-regions remaining. Because some points in the direction of the longest gradient are not in the zero-region in the eight directions prediction. Two types of four directions prediction are then performed. In

order to restore the textual and structural information in the zero-region to the most degree, four types of two directions prediction are performed after the four directions prediction. The illustration of the filling directions is shown in Fig.66.

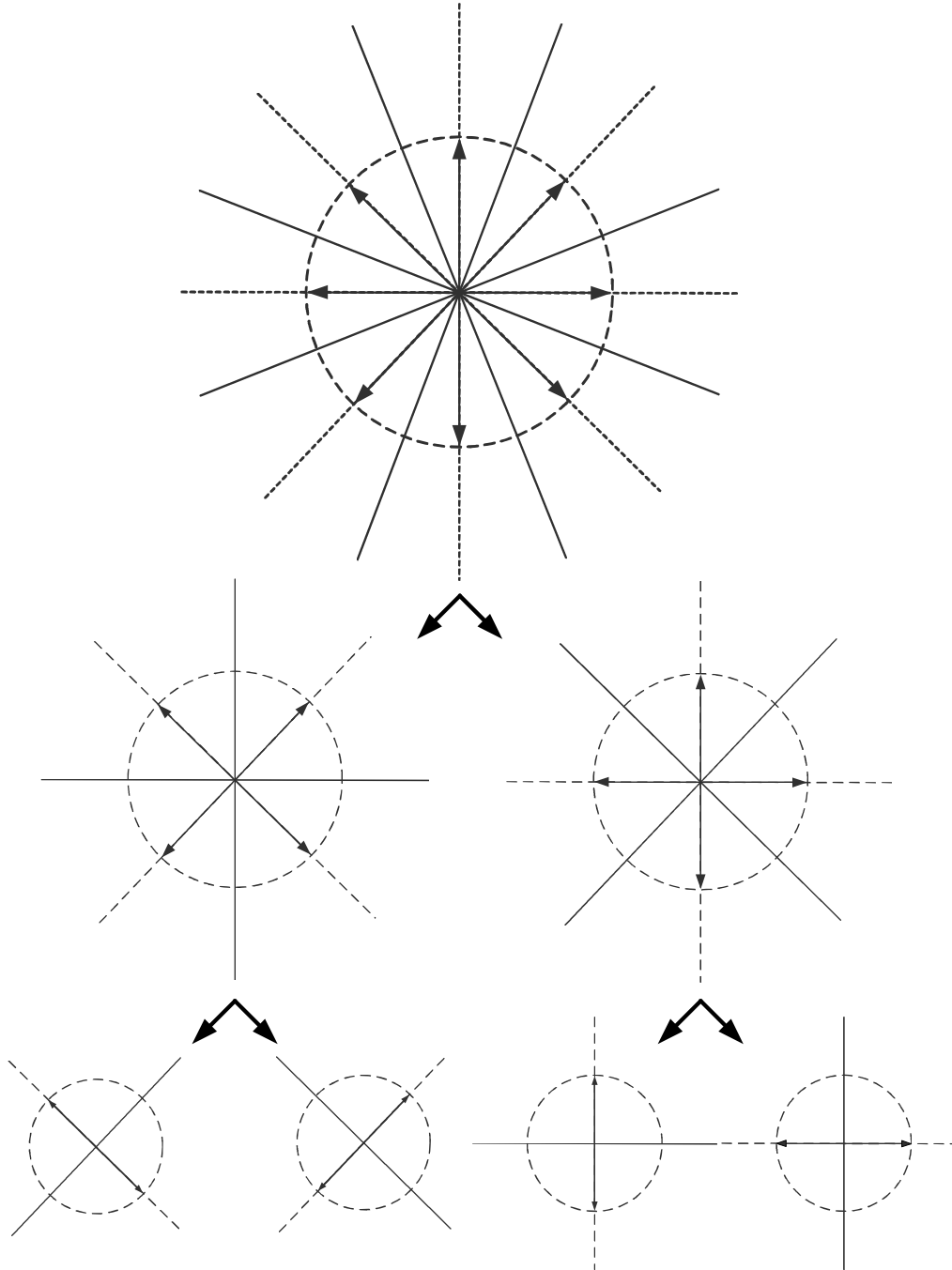


Fig.66. Illustration of the filling directions

The length and direction of the gradient are calculated on the basis of the gray-scale version of the texture image and by using following equations:

$$\begin{cases} dx = V_{P3} - V_{P1} & (3) \\ dy = V_{P2} - V_{P4} & (4) \\ G = \sqrt{dx^2 + dy^2} & (5) \\ \alpha = \tan^{-1}(dy/dx) & (6) \end{cases}$$

V is the value of the pixels in the gray-scale version of the texture image.

4.3.3 Experimental Results



1. VSRS [14]



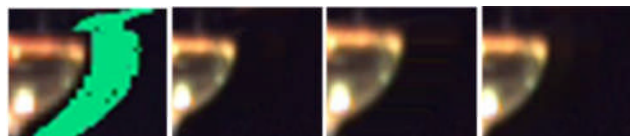
2. Line-wise approach with boundary-ensuring principle



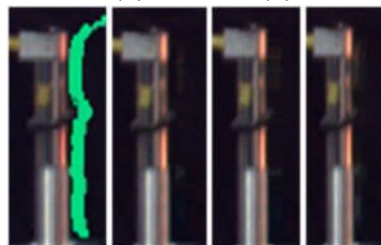
3. Proposed algorithm



4. Original virtual view



a(4) a(3) a(2) a(1)



b(4) b(3) b(2) b(1)

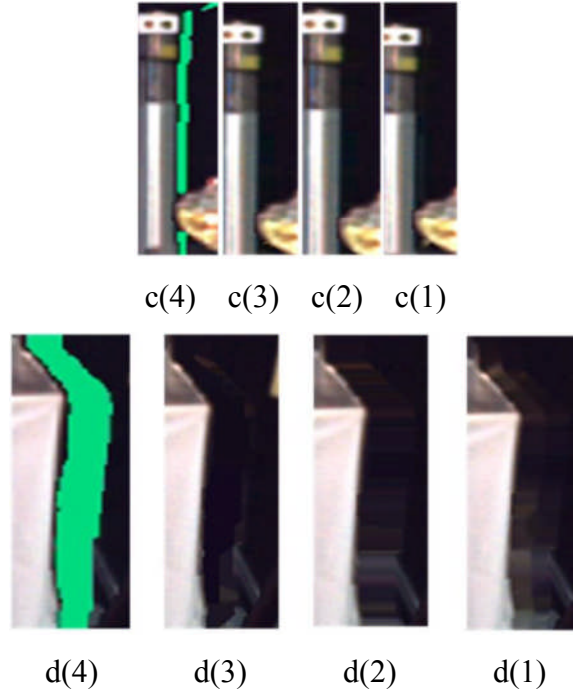


Fig.67. Detailed comparison of the results [84]

Experiments were conducted using the Champagne_tower sequence based on the VSRS 3.5. The quality evaluation criterion for the proposed and other methods used PSNR [55] and SSIM [56]. The view 39 and 40 are two adjacent cameras with slightly different camera perspectives. The baseline between two adjacent cameras is approximately 65 mm for all test sequences. View 39 is the real view, whereas view 40 is a synthesized virtual view; a comparison between them is shown in Fig.64 错误! 未找到引用源。 .

PSNR is computed locally, that is only for the defective area in the image and can be considered an objective quality evaluation criterion, while SSIM is determined for the entire image and can be considered a subjective quality evaluation criterion. In the comparison results between the VSRS and the proposed algorithm, the PSNR is improved by 0.2425 dB and the SSIM is improved by 0.005, whereas in the comparison results between line-wise approach and the proposed algorithm, the PSNR is improved by 0.1664 dB and the SSIM is almost the same. Through this comparison, the proposed algorithm is shown to improve not only the objective

quality of the virtual view but also the subjective quality.

Some additional results are provided in Table 12. In these cases, the PSNR is improved by 0.0788 ~ 2.3883 dB and the SSIM is improved by 0.0003 ~ 0.0065.

The details and human vision comparison results are shown in Fig.67. The first result is obtained using the VSRS, the second by a line-wise approach with a boundary ensuring principle, and the third by the proposed algorithm; and the fourth image is the original virtual view for a comparison. Through a, b, c, d four details comparison for human vision, it is clear that the VSRS makes the boundary between the foreground and background blurred, the line-wise approach amplifies the artifacts around the zero-regions, and the proposed algorithm can obtain better boundary contours of the foreground objects and more accurate background information for the in-painting result. In the 3-D cases, the textural and structural information is very important. Therefore, the proposed algorithm performs significantly better than the previous methods.

If multiple viewpoints are to be synthesized, all of them will be referred to the known viewpoint. Because the synthesized results are the approximate results and are not reliable enough as the reference view. If the previous synthesized viewpoints are used as the reference views, the errors which caused by the approximate process will be propagandized to other virtual views.

Table 12 Some additional experimental results

Sequence name	Position	VSRS		Line-wise approach		Proposed algorithm	
		PSNR(dB)	SSIM	PSNR(dB)	SSIM	PSNR(dB)	SSIM
Champagne_tower	39 to 41	23.5867	0.7686	23.7574	0.7712	23.8362	0.7726
	41 to 39	23.1932	0.7653	23.3555	0.7657	23.4358	0.7669
Mobile	3 to 4	32.5907	0.9790	33.0770	0.9819	33.1972	0.9822
	4 to 3	32.2911	0.9837	34.0639	0.9860	34.2578	0.9867
Beer garden	5 to 6	22.1965	0.7085	21.2598	0.7063	23.6481	0.7128
	6 to 5	24.4131	0.7213	24.2863	0.7218	24.8427	0.7225

4.3.4 Conclusion

The proposed integrated hole-filling algorithm for view synthesis includes five steps: an algorithm for distinguishing different regions, foreground and background boundary detection, texture image isophote detection, a textural and structural isophote prediction algorithm, and an in-painting algorithm with gradient priority order.

Based on the experimental results, the objective and subjective qualities of an image are improved considerably using the proposed algorithm. Through a detailed comparison, the proposed algorithm was shown to ensure the boundary contours of foreground objects as well as the structural information in the zero-region of a virtual image much clearly than the previous methods.

4.4 Combined hole-filling with spatial and temporal prediction

4.4.1 The architecture of the proposal

4.4.1.1 Architecture of the proposed algorithm

This proposed algorithm combines the spatial prediction (SP) and temporal prediction (TP) in one system, and obtains the advantages from both of the methods by effective integrated architecture, selecting principles and suitable proposals in details. The Fig.68 shows the brief architecture of the proposed combined algorithm.

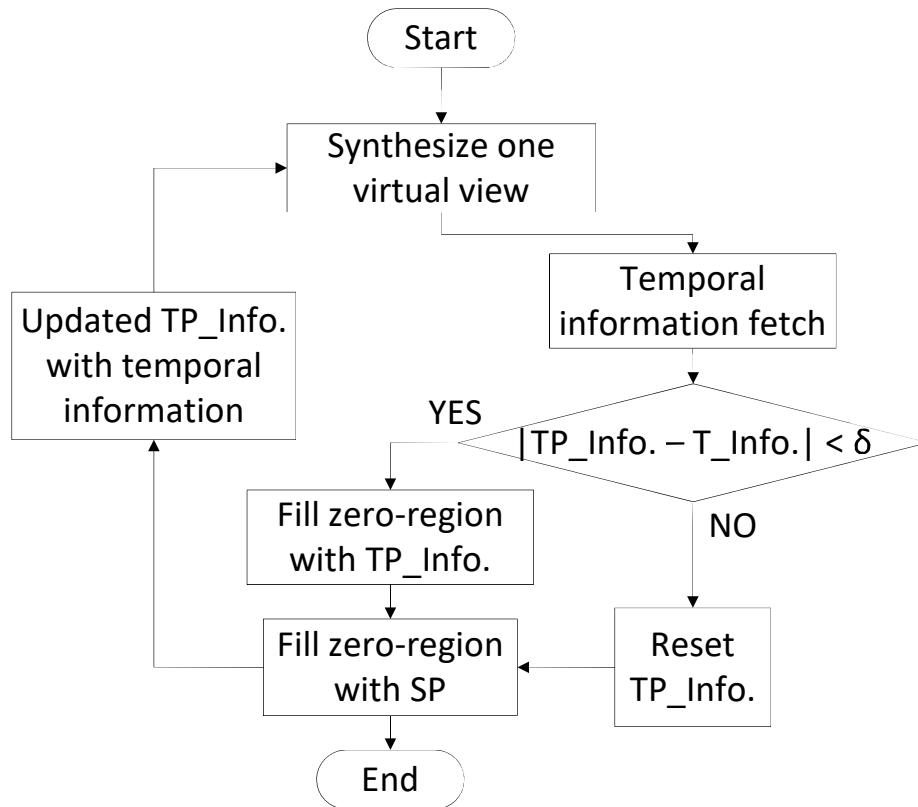


Fig.68. The architecture of the proposed algorithm

4.4.1.2 Temporal prediction

In Fig.68, TP_Info. is shorten for temporal prediction information, and T_Info. is shorten for temporal information. T_Info. is the background information of the current frame. When one virtual view is synthesized, the background information of this frame will be fetched at first. TP_Info. is the updated temporal information, and it is used as the reference information in the temporal prediction. After the temporal information fetching, TP_Info. and T_Info. are compared with each other in the low depth value region. The low depth value region is set as the region where depth value is less than 20 in this paper. The δ means that more than half of the pixels are different. If the difference between TP_Info. and T_Info. is smaller than δ , that means the camera is almost fixed and the updated TP_Info. is reliable. Then the zero-region should be filled with the TP_Info. at first and the SP process will start. If the difference between TP_Info. and T_Info. is larger than δ , that means the camera is moving or the environmental luma information is changing. The updated TP_Info. is not reliable at all in this kind of situation. The TP_Info. should be reset and the SP process will start.

4.4.1.3 Spatial prediction

SP is shorten for spatial prediction, and this part refers to [57] in this paper. There are five main parts in this process: the different regions distinguishing algorithm, foreground and background boundary detection, the texture image isophotes detection, the textural and structural isophotes prediction algorithm, the in-painting algorithm with gradient priority order.

4.4.2 Details of the combined hole-filling algorithm

4.4.2.1 Detailed flowchart of the combined algorithm

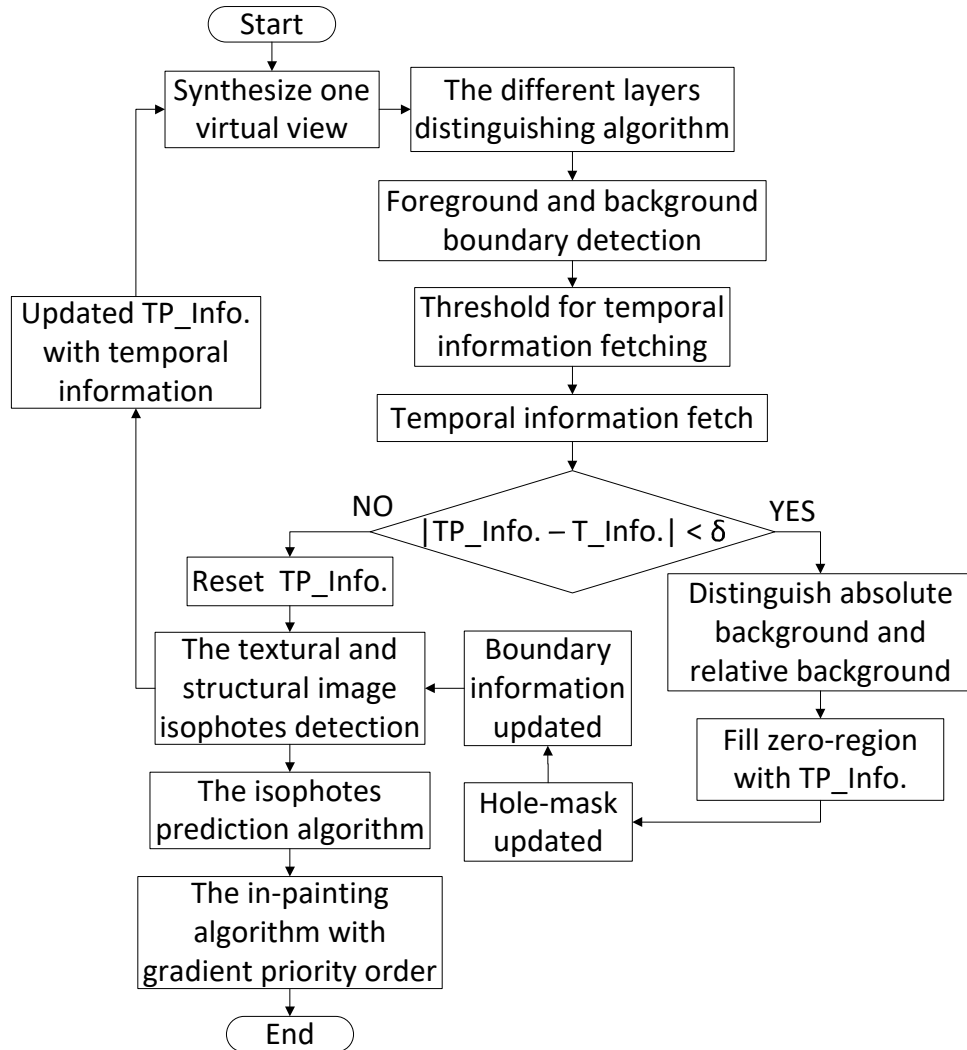


Fig.69. The detailed flowchart of the proposed algorithm

In Fig.69, the details of the combined method are shown. Some new problems appear in the combined prediction process. In order to solve this kind of problems, some new proposed methods are introduced in the next several subsections.

4.4.2.2 Threshold for temporal information fetching

To fetch the temporal information, the maximum depth value of the background

layers should be obtained. However, it is not easy to get the appropriate depth value directly. After the different regions distinguishing and the foreground and background boundary detecting, the boundaries of the relative foreground and boundaries of the relative background can be obtained at first. Then the depth value of the relative foreground boundaries and the depth value of the relative background boundaries can be obtained. The average depth value of the relative foreground boundaries and the relative background boundaries can be considered as the appropriate depth value for the temporal information fetching.

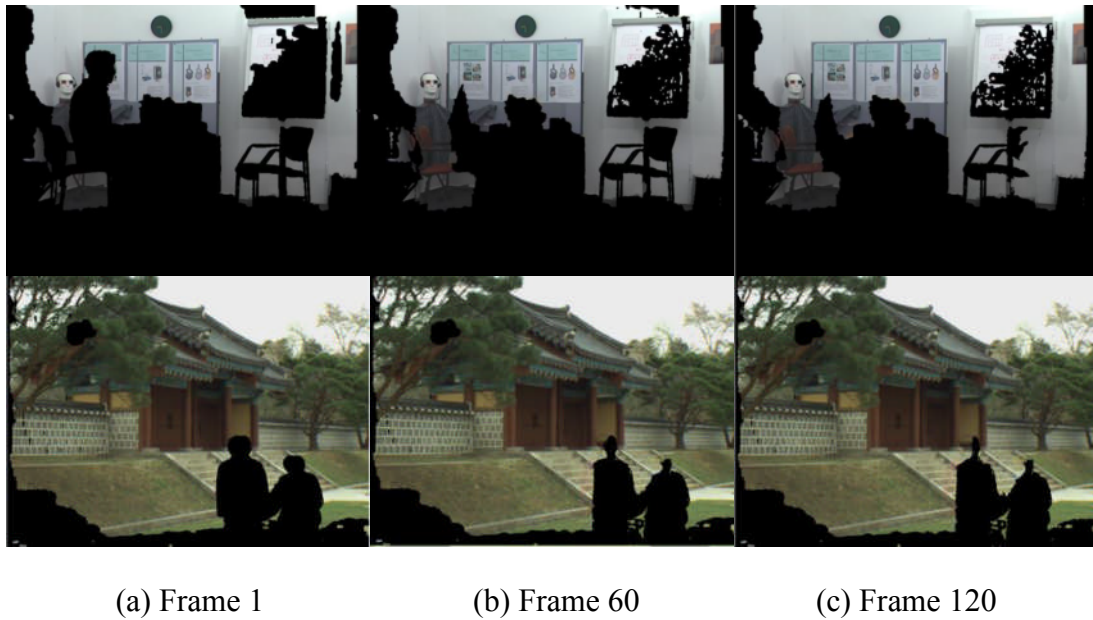


Fig.70. Some results of temporal information fetching [89]

Some results show in Fig.70. (a) is the temporal information fetching results after frame 1, (b) is the temporal information fetching results after frame 60 and (c) is the temporal information fetching results after frame 120. Through these experimental results, it is obvious that the method is very effective.

4.4.2.3 Distinguish absolute background and relative background

After the different regions distinguishing and the foreground and background

boundary detecting, the relative foreground and the relative background layers have already been detected. Some holes are between two foreground objects, they are also between a relative foreground layer and a relative background layer, and have relative foreground boundary and the relative background boundary. But the TP_info. is the temporal information of the absolute background, so only distinguishing the relative background is not enough. There are some errors shown in Fig.71, and red circles point out this kind of errors.



Fig.71. The errors occur in the temporal prediction process without distinguishing the absolute and relative background layers [87]

Therefore, before the temporal prediction process, check the depth value around the relative background side of the hole-regions at first. In the previous step, a threshold value has been obtained for the temporal information fetching. It can be also considered as the threshold for distinguishing the absolute background layers from the relative background layers. Check the depth value around relative background side of

the holes and compare them with the threshold value. If it is smaller than threshold, then the relative background layer next to this hole should be considered as the absolute background layer.

4.4.2.4 Hole-mask and boundary information updated

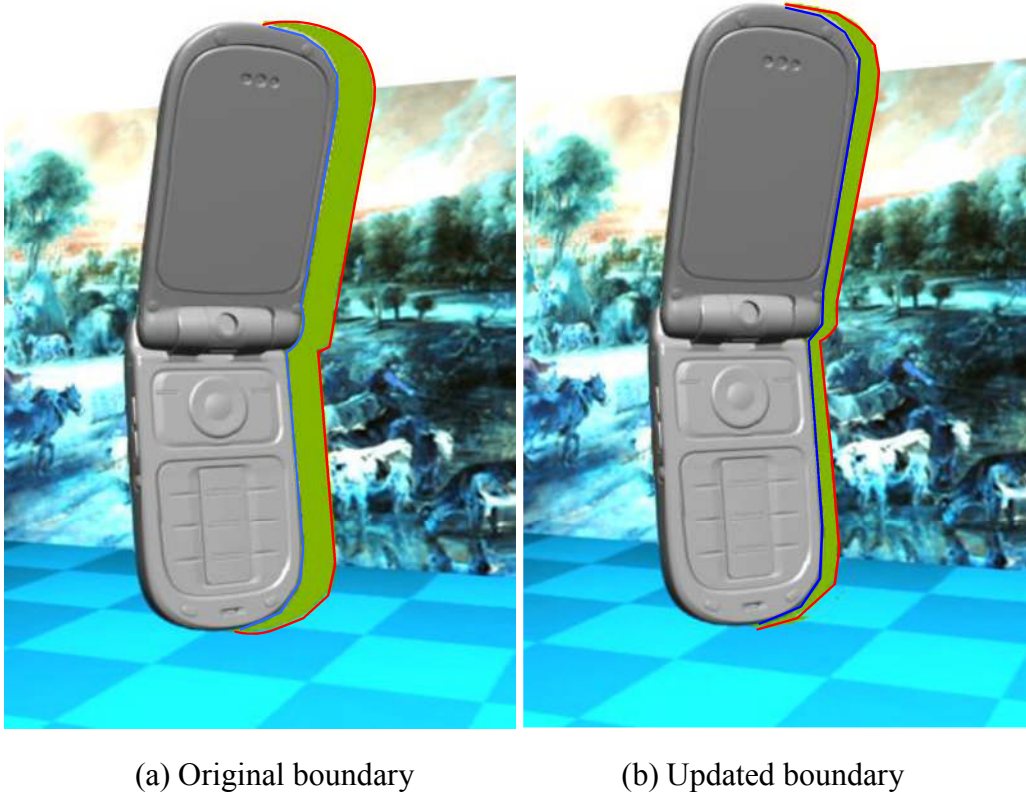


Fig.72. The illustration of the updated boundary information [88]

The boundary information is very important information for the spatial prediction method in this paper. After temporal prediction, some zero-regions have already been filled by the temporal information and the boundary information needs to be updated for the next texture image isophotes detection process.

Fig.72 shows the result of the boundary information updated process. Red lines are the relative background information and the blue lines are the relative foreground information. (a) is the original boundary information and (b) is the updated result for the boundary information. The hole-mask information is also updated in the same time for the spatial filling.

4.4.3 Experimental results

We compared our results with those of the MPEG view synthesis reference software (VSRS) version 3.0. For evaluating the proposed algorithm, five 3D video sequences are used: “Book arrival”, “Lovebird1”, “Newspaper” and “Mobile”. The “Book arrival”, “Lovebird1” and “Newspaper” have a resolution of 1024 x 768 pixels. “Mobile” has a resolution of 720 x 540 pixels. For every sequence, the rectified videos of several views with slightly different camera perspectives are available. The baseline between two adjacent cameras is approximately 65 mm for all test sequences.

Table 13 PSNR and SSIM results by the proposed algorithm, [57] and the view synthesis reference software.

Sequence	Resolution	Camera	PSNR (dB)			SSIM		
			VSRS	[57]	Proposed	VSRS	[57]	Proposed
Book arrival	1024 x 768	8 -> 6	30.3756	30.4128	30.5233	0.8992	0.8995	0.8998
Book arrival	1024 x 768	8 -> 10	28.8872	31.1354	31.4368	0.8995	0.9012	0.9019
Lovebird1	1024 x 768	6 -> 4	22.9546	22.9672	23.6723	0.6653	0.6652	0.6733
Lovebird1	1024 x 768	6 -> 8	22.6164	22.7222	22.7315	0.6257	0.6268	0.6269
Newspaper	1024 x 768	4 -> 2	20.4637	20.4958	21.2876	0.7063	0.7066	0.7115
Newspaper	1024 x 768	4 -> 6	19.8349	19.9517	19.9720	0.6603	0.6613	0.6613
Mobile	720 x 540	5 -> 3	28.1611	28.2534	31.1170	0.9675	0.9679	0.9751
Mobile	720 x 540	5 -> 7	28.3479	28.5864	31.4102	0.9624	0.9631	0.9746

One original view is chosen to generate two additional virtual views which are on the left and right sides of the original view in the experiment. The position of each

“virtual” camera is 2 cameras away from the original camera location, and it gives a baseline of approximately 130 mm.

Two quality evaluation criteria are used in the experimental results comparison. PSNR is computed locally, that is only for the defective area in the image, while SSIM is determined for the entire image. Therefore, the PSNR can be considered an objective quality evaluation criterion, while the SSIM can be considered a subjective quality evaluation criterion.

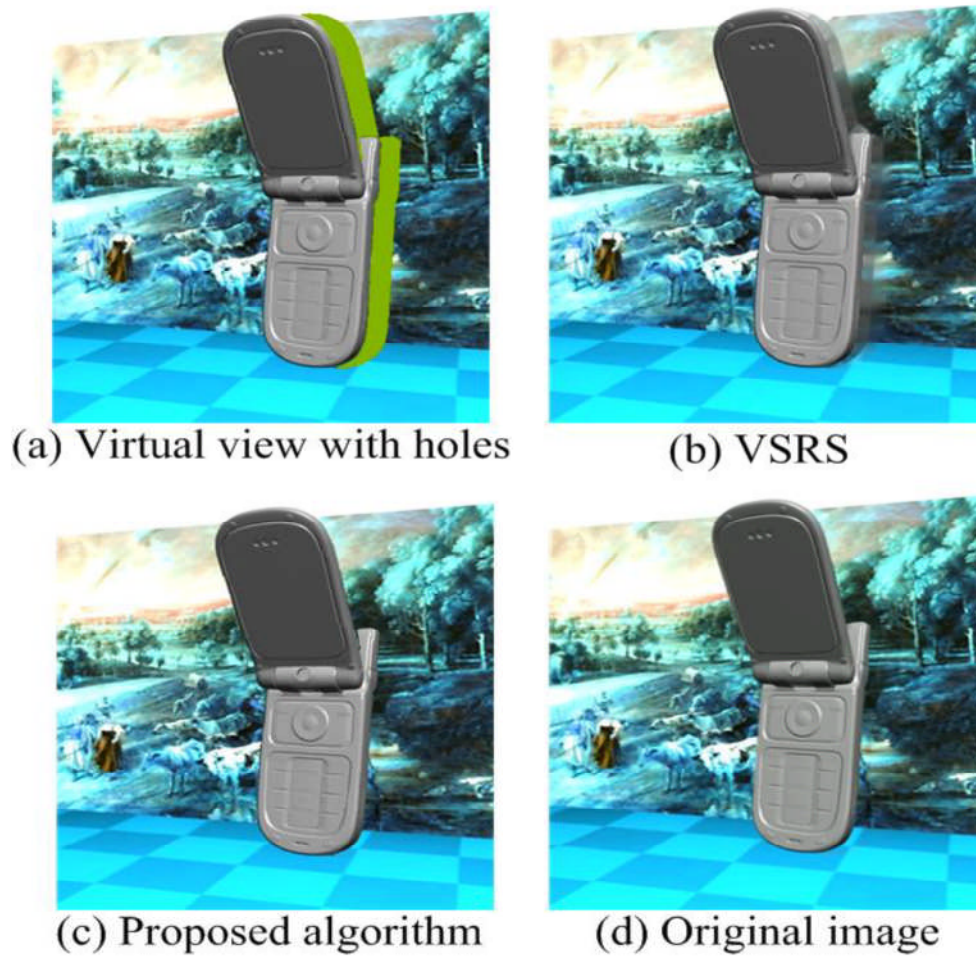


Fig.73. The subjective comparison result [88]

Table 13 shows some objective and subjective results and Fig.73 shows some details of the subjective results. In Fig.73, (a) is the virtual image with hole, (b) is the hole filling result by VSRS, (c) is the hole filling result by the proposed algorithm and (d) is the original image. Through this experimental result, it is obvious that the

proposed method can obtain much better result than the previous method not only in the objective comparison, but also in the subjective comparison.

4.4.4 Conclusion

In this section, a combined hole-filling algorithm has been introduced. It integrated the spatial prediction and the temporal prediction together, and obtained the advantages from both of the prediction methods. Therefore, the performance is much stable than the previous works not only in the camera fixed cases, but also in the camera moving cases.

The experimental results showed that both of the objective quality and the subjective quality are much better than the other methods in most of the cases.

5. Conclusion of the dissertation

In this dissertation, we proposed some novel algorithms with low computational complexity and low transmission bandwidth in video processing. In detail, our contributions are as follows:

First, in the 2-D cases, we proposed a low power and low computational complexity video decoding with adaptive granularity in temporal scalability for H.264/AVC decoder. By reducing the frame rate, proposed algorithm provides much more display formats for the single layer bit stream. Besides, the decoding complexity is adaptively reduced to ensure videos with different resolutions being real-time decoded by low speed terminal devices in power restrained situations. The proposed process contains four parts: the reference frame index decision algorithm, motion vector composition algorithm, block-partition decision algorithm and the adaptive selecting algorithm. According to these methods, the frame rate can be converted down and the video quality can be kept on an acceptable level.

Second, for the stereo matching, we proposed a frame compatible format fast encoder with stereo matching. Traditionally, the process of local stereo matching is divided as cost computation, support aggregation and disparity refinement. In the proposed method, the process of traditional matching algorithm is rearranged. The proposed algorithm reduces computational complexity and ensures the accuracy of the result. The proposed novel FCF fast encoder is proposed by implementing the stereo marching algorithm into it. It can achieve both less PSNR loss, the lower bit-rate increment and reduce the data transmission cost a lot at the same time. Besides, no one has combined stereo matching into FCF fast encoder research currently. This research shows the promising possibility of this field.

Third, for the view synthesis, we proposed a novel combined hole-filling algorithm for view synthesis. It combines the temporal prediction and the spatial prediction together. In some of the cases, the camera nearly doesn't move at all. Then the

prediction based on the temporal information is much more reliable and stable than the spatial prediction in this kind of situations. However, in the camera moving cases, the spatial prediction is much better than the temporal prediction. Therefore, the combined realization can obtain the advantages from both of the spatial and temporal prediction method. The spatial prediction part of the proposed algorithm includes five parts: an algorithm for distinguishing different regions, foreground and background boundary detection, textural and structural isophote detection, a texture image isophote prediction algorithm, and an in-painting algorithm with gradient priority order. An isophote is the boundary between two regions in different layers or two regions with much different luma and chroma information, and intersects with the zero-region from the background. The proposed method ensures the boundary of the foreground objects by distinguishing the different layers in the texture image with depth information, predicts the textural and structural information in the zero-region using the geometry principle, and paints in the zero-region in a priority order based on the gradient information. Textural and structural information is very important in the 3-D cases, and has a significant effect on the 3-D impression in human vision. Therefore, the proposed method will improve not only the objective quality of a synthesized virtual view but also its subjective quality and the 3-D performance for human vision. In addition to multi-viewpoint cases, the proposed method can also be used in the conversion of 2-D video into 3-D video.

Finally, through the experimental results, we can find that the performance of the proposed algorithms is very good. In the 2-D cases, it provides a lot of temporal scalable mode for the users, and can actually reduce the decoding computational complexity a lot. In the 3-D cases (view synthesis and stereo matching), it can reduce the transmission bandwidth cost a lot and ensure the video quality in some acceptable degree.

Reference

- [1] Beucher, N.; Belanger, N.; Savaria, Y.; Bois, G., “Motion Compensated Frame Rate Conversion Using a Specialized Instruction Set Processor”, SIPS, Page(s): 130 - 135, 2006
- [2] Truong Quang Vinh; Young-Chul Kim; Sung-Hoon Hong, “Frame Rate Up-Conversion Using Forward-Backward Jointing Motion Estimation And Spatio-Temporal Motion Vector Smoothing”, ICCES, Page(s): 605 – 609, 2009
- [3] Sugiyama, K.; Aoki, T.; Hangai, S., “Motion Compensated Frame Rate Conversion Using Normarized Motion Estimation”, SIPS, Page(s): 663 – 668, 2005
- [4] Al-Mualla, M.E., “Motion field interpolation for frame rate conversion”, ISCAS, Page(s): II-652 - II-655 vol.2, 2003
- [5] Fujiwara, S.; Taguchi, A., “Motion-Compensated Frame Rate Up-Conversion Based on Block Matching Algorithm with Multi-Size Blocks”, ISPACS, Page(s): 353 - 356, 27-30 June 2005
- [6] Sung-Hee Lee, Ohjae Kwon, and Rae-Hong Park, “Weighted-adaptive motion-compensated frame rate up-conversion”, IEEE Transactions on Consumer Electronics, 486 Vol. 49, No. 3, AUGUST 2003
- [7] L. Onural, “Television in 3-D: What are the prospects?” Proc. IEEE, vol. 95, no. 6, pp. 1143–1145, Jun. 2007.
- [8] M. Tanimoto, “Overview of free viewpoint television,” Signal Process.:Image Commun., vol. 21, no. 6, pp. 454–461, Jul. 2006.
- [9] A. Smolic and P. Kauff, “Interactive 3-D video representation and coding technologies,” Proc. IEEE, Special Issue on Advances in Video Coding and Delivery, vol. 93, no. 1, pp. 98 – 110, Jan. 2005.

- [10] A. Smolic and D. McCutchen, “3-DAV exploration of video-based rendering technology in MPEG,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 3, pp. 348 – 356, Mar. 2004.
- [11] “Updated Call for Proposals on Multi-View Video Coding, ” ISO/IEC JTC1/SC29/WG11 Doc. N7567, Nice, France, Oct. 2005.
- [12] “ Requirements on Multi-View Video Coding, v. 6, ” ISO/IEC JTC1/SC29/WG11 Doc. N8064, Montreux, Switzerland, Apr. 2006.
- [13] “Draft ITU-T recommendation and final draft international standard of joint video specification”, ITU-T rec. H.264/ISO/IEC 14496-10 AVC, Mar. 2003
- [14] M.Tanimoto, T. Fujii, and K. Suzuki, “View Synthesis Algorithm in View Synthesis Reference Software 2.0 (VSRS2.0)”, ISO/IEC JTC1/SC29/WG11 M16090, Lausanne, Switzerland, February 2008.
- [15] A. Smolic, K. Müller, and A. Vetro, “Development of a New MPEG Standard for Advanced 3D Video Applications”, In *Proc. of IEEE Int. Symp. on Image Signal Processing and Analysis*, Salzburg, Austria, September 2009.
- [16] C.-M. Cheng, S.-J. Lin, S.-H- Lai, and J.-C. Yang, “Improved Novel View Synthesis from Depth Image with Large Baseline”, In *Proc. of Int. Conf. on Pattern Recognition*, Tampa, USA, December 2008.
- [17] C. Fehn, “Depth Image Based Rendering (DIBR), compression and transmission for a new approach on 3D-TV”, In *SPIE Stereoscopic Displays and Virtual Reality Systems XI*, pp. 93-104, January 2004.
- [18] A. Criminisi, P. Perez, and K. Toyama, “Region Filling and Object Removal by Exemplar-based Inpainting”, In *IEEE Trans. on Image Proc.*, vol. 13, no. 9, pp. 1200-1212, January 2004.
- [19] J. Hayes, A. Efros, “Scene Completion Using Millions of Photographs”, In *Proc.*

ACM SIGGRAPH, San Diego, USA, August 2007.

[20] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, “Dynamic Textures”, In *Int. Journal of Com. Vision*, pp. 91-109, February 2004.

[21] P. Ndjiki-Nya, M. Köppel, D. Doshkov, and T. Wiegand, “Automatic Structure-Aware Inpainting for Complex Image Content”, In *Proc. of Int. Sym. on Visual Computing*, Las Vegas, USA, December 2009.

[22] L.-Y. Wei, S. Lefebvre, V. Kwatra, and G. Turk, “State of the Art in Example-based Texture Synthesis” *EUROGRAPHICS 2009, State of the Art Report*, EG-Star, Munich, Germany, 2009.

[23] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert and T. Wiegand, “3D video and free viewpoint video – technologies, applications and MPEG standards”, *Proc. ICME*, pp. 2161-2164, July 2006.

[24] M. Tanimoto, “Overview of FVT (free viewpoint television),” *IEEE International Conference on Multimedia and Expo(ICME)*, vol. 21, pp. 1552-1553, July 2009.

[25] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, “Multiview imaging and 3DTV,” *IEEE Signal Processing Magazine*, vol. 24, No. 6, pp. 10-21, Nov. 2007.

[26] S. Zinger, L. Do, and P. H. N. de With, “Free-viewpoint depth image based rendering,” *J. V . Commun. Image Representation*, vol. 21, no. 5-6, pp. 533-541, 2010.

[27] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, “View generation with 3-D warping using depth information for FTV,” *IEEE J. Signal Process*, vol. 24, no. 1-2, pp. 65-72, Jan.-Feb. 2009.

[28] J. Konrad and M. Halle, “3-D displays and signal processing,” *IEEE Signal*

Processing Magazine, vol. 24, no.6, Nov. 2007.

[29] P. Kauff, N. Atzpadin, C. Fehn, M. Müller, O. Schreer, A. Smolic, and R. Tanger, “Depth map creation and image based rendering for advanced 3DTV services providing interoperability and scalability,” *Signal Processing: Image Communication. Special Issue on 3DTV*, Feb. 2007.

[30] C. Fehn, “Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV,” *Proceedings of SPIE*, vol. 5291, pp. 93-104, 2004.

[31] H. Dias, J. Rocha, P. Silva, C. Ledo, L.P. Reis: “Distributed Surveillance System” , *Proc. of Portuguese Conference on Artificial Intelligence (EPIA)*, pp. 257–261 (2005).

[32] E. Norouznezhad, A. Bigdeli, A. Postula, B. C. Lovell: “A High Resolution Smart Camera with Gige Vision Extension for Surveillance Applications” , *Proc. of ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC)*, pp. 1–8 (2008).

[33] T. Raty: “High-Level Architecture for a Single Location Surveillance Point” , *Proc. of International Conference on Wireless and Mobile Communications (ICWMC)*, p. 82 (2007).

[34] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Recommendation H.264/ISO/IEC 14496–10 AVC) (2003).

[35] 5) J. G. Apostolopoulos: “Architectural Principles for Secure Streaming & Secure Adaptation in the Developing Scalable Video Coding (SVC) Standard” , *Proc. of International Conference on Image Processing (ICIP)*, pp. 729–732 (2006).

[36] Y. Zheng, X. Ji, F. Wu, D. Zhao, W. Gao: “An Efficient Fgs Coding Scheme for Interlaced Scalable Video Coding” , *Proc. of International Conference on Image*

- Processing (ICIP), pp. 2485–2488 (2006).
- [37] C. A. Segall, G. J. Sullivan: “Spatial Scalability within the H.264/AVC Scalable Video Coding Extension” , IEEE Trans. on Circuits and Systems for Video Technology, Vol. 17, No. 9, pp. 1121–1135 (2007).
- [38] A. Thammineni, Arvind Raman, Sarat Chandra Vadapalli, Sriram Sethuraman: “Dynamic Frame-Rate Selection for Live LBR Video Encoders Using Trial Frames” , Proc. of IEEE International Conference on Multimedia and Expo. (ICME), pp. 817–820 (2008).
- [39] F. Pan, X. Lin, S. Ruhardju, K. P. Lim, Z. G. Li, D. J. Wu, W. Si, L. J. Jiang: “Variable Frame Rate Encoding via Active Frame-Skipping” , International Conference on Information Science, Signal Processing and their Applications (ISSPA), Vol. 1, pp. 89–92 (2003).
- [40] M. Cetin, I. Hamzaoglu: “An Adaptive True Motion Estimation Algorithm for Frame Rate Conversion of High Definition Video” , International Conference on Pattern Recognition (ICPR), pp. 4109–4112 (2010).
- [41] G. Dane, K. El-Maleh, Y.-C. Lee: “Encoder-Assisted Adaptive Video Frame Interpolation” , International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 349–352 (2005).
- [42] K.-T. Fung, Y.-L. Chan, W.-C. Siu: “Dynamic Frame Skipping for High-Performance Transcoding” , Proc. of International Conference on Image Processing, Vol. 1, pp. 425–428 (2001).
- [43] H. Shu, L.-P. Chau: “Variable Frame Rate Transcoding Considering Motion Information” , Proc. of IEEE International Symposium on Circuits and Systems (ISCAS), Vol. 3, pp. 2144–2147 (2005).
- [44] V. Patil, R. Kumar: “An Effective Motion Re-estimation in Frame- Skipping

Video Transcoding” , Proc. of International Conference on Computing: Theory and Applications (ICCTA), pp. 655–659 (2007).

[45] Lonetti, F., Martelli, F.: Motion Vector Composition Algorithm in H.264 Transcoding. In: IWSSIP, pp. 401 – 404, 27 – 30 (June 2007)

[46] Joint Video Team of ITU-T and ISO/IEC JTC 1, “Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec.H.264 ISO/IEC 14496-10 AVC)”, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, March 2003.

[47] Wenxin Yu, Xin Jin, Satoshi GOTO, “Temporal scalable decoding process with frame rate conversion method for surveillance video”, PCM, Part II, LNCS 6298, Page(s): 297 - 308, Shanghai, China, September 2010

[48] C. Fehn, “Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV,”, Proceedings of SPIE, vol. 5291, pp. 93-104, 2004.

[49] C.-M. Cheng, S.-J. Lin, S.-H. Lai, and J.-C. Yang, “Improved novel view synthesis from depth image with large baseline,”, Proc. of the 19th International Conference on Pattern Recognition (ICPR), pp. 1 – 4, 2008.

[50] K.-J. Oh, S. Yea, and Y.-S. Ho, “Hole filling method using depth based in-painting for view synthesis in free viewpoint television and 3-D video,” Proc. of the Picture Coding Symposium (PCS), pp. 1-4, 2009.

[51] K. Müller, A. Smolic, K. Dix, P. Merkle, P. Kauf, and T. Wiegand, “View synthesis for advanced 3-D video systems,” EURASIP J. Image Video Process., vol. 2008, 2008, Art. ID 438148.

[52] M. Bertalmio, A. Bertozzi, and G. Sapiro, “Navier-stokes, fluid dynamics, and image and video inpainting,” Proc. of the IEEE Computer Society Conference on

Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. I-355-I-362 vol.1, 2001.

[53] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," IEEE Transactions on Image Processing, vol. 13, no. 9, pp. 1200-1212, 2004.

[54] D. Lowe, "Distinctive image features from scale-invariant key points," IJCV, vol.60, pp.91-110, 2004.

[55] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves," ITU SG16 Doc. VCEG-M33, 2001.

[56] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[57] W. Yu, W. Wang, Z. Guo, and S. Goto, "An Integrated Hole-Filling Algorithm for View Synthesis," Pacific-Rim Conference on Multimedia (PCM), LNCS 7674, pp. 80–92, 2012.

[58] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nistér, "High-quality realtime stereo using adaptive cost aggregation and dynamic programming", 3DPVT, 2006.

[59] O. Stankiewicz and K. Wegner, "Depth map estimation software version 2", ISO/IEC MPEG meeting M15338, 2008.

[60] O. Veksler, "Stereo correspondence by dynamic programming on a tree", CVPR, 2005.

[61] F. Hawi and M. Sawan, "Phase based passive stereovision systems dedicated to cortical visual stimulators", ICCD, 2012.

[62] Weichen Wang, Satoshi Goto, "Stereo Matching with Pixel Classification and Reliable Disparity Propagation", ISCAS, Seoul, Korea, pp.1891-1894, 2012.

- [63] “Use cases and requirements for the enhancement of MPEG frame-compatible”, ISO/IEC JTC1/SC29/WG11 MPEG2010/N11681, Oct, Guangzhou, China, 2010.
- [64] “Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification”, ITU-T Rec.H.264/ISO/IEC, 2003.
- [65] G.J.Sullivan, “Draft ACX amendent text to specify constranined baseline profile, Stereo igh profile, and frame packing SEI message”, ITU-T SG16/Q6, 31th meeting, Doc. JVT-AE204, London, UK, 2009.
- [66] HDMI liscensing, LLC. “HDMI Specification 1.4”, 2009.
- [67] Zhuoying Zeng, Xin Jin, Satoshi Goto, “A Fast Intra Encoder of Frame-compatible Format Based on Content Similarity for 3D Distribution”, EUSIPCO, Barcelona, Spain, 2011.
- [68] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. On building an accurate stereo matching system on graphics hardware. GPUCV 2011.
- [69] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In Proc. ECCV, pages 151–158, 1994.
- [70] Andreas Geiger, Martin Roser and Raquel Urtasun, “Efficient Large-Scale Stereo Matching”, Computer Vision – ACCV 2010 Lecture Notes in Computer Science, Volume 6492/2011, 25-38, 2011.
- [71] Lai-Man Po, Shihang Zhang, Xuyuan Xu, Yuesheng Zhu, “A New Multidirectional Extrapolation Hole-Filling Method for Depth-Image-Based Rendering”, 2011 18th IEEE International Conference on Image Processing (ICIP), Page(s): 2589 – 2592, 2011.
- [72] Ismael Daribo and Beatrice Pesquet-Popescu, “Depth-aided image inpainting for Novel View Synthesis”, 2010 IEEE International Workshop on Multimedia Signal Processing (MMSP), Page(s): 167 – 170, 2010

- [73] A. Efros and T. Leung, “Texture synthesis by non-parametric sampling,” in Proc. of the 7th IEEE International Conference on Computer Vision (ICCV), vol. 2, 1999, pp. 1033–1038 vol.2.
- [74] Gary J. Sullivan, “Draft AVC amendment text to specify Constrained Baseline profile, Stereo High profile, and frame packing SEI message”, JVT of MPEG and VCEG, Doc. JVT-AE204, London, UK, June, 2009.
- [75] HDMIlicensing, LLC. “HDMI Specification 1.4”, 2009.
- [76] Vetro A., “Frame compatible formats for 3D video distribution”, 17th IEEE International Conference on Image Processing (ICIP), Page(s): 2405 – 2408, 2010
- [77] Ying Chen, Rong Zhang, Karczewicz M., “MVC based scalable codec enhancing frame-compatible stereoscopic video”, IEEE International Conference on Multimedia and Expo (ICME), Page(s): 1 – 4, 2011
- [78] Taoran Lu, Ganapathy H., Lakshminarayanan G., “Orthogonal Muxing Frame Compatible Full Resolution technology for multi-resolution frame-compatible stereo coding”, IEEE International Conference on Multimedia and Expo (ICME), Page(s): 1 – 6, 2013
- [79] Speranza F., Renaud R., Vincent A., Tam W.J., “Perceived Picture Quality of Frame-Compatible 3DTV Video Formats”, 2012 IEEE International Conference on Multimedia and Expo (ICME), Page(s): 640 – 645, 2012
- [80] Ballocca G., D'Amato P., Grangetto M., Lucenteforte M., “Tile format: A novel frame compatible approach for 3D video broadcasting”, IEEE International Conference on Multimedia and Expo (ICME), Page(s): 1 – 4, 2011
- [81] Siao-Wei Chen, Ming-Feng Tsai, Jui-Chiu Chiang, “Video retargeting based frame-compatible stereovideo coding”, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Page(s): 1854 – 1858, 2013

- [82] T. Y. Kuo, C. K. Yeh, H. Y. Tsai, "A novel method for global disparity vector estimation in multiview video coding," in Proc. Int. Symposium Circuits Systs., pp. 864-867, 2009.
- [83] K. H. Liu, T. J. Liu, H. H. Liu, "A SIFT descriptor based method for global disparity vector estimation in multiview video coding," in Proc. Int. Conf. Multimedia and Expo, pp. 1214-1218, 2010.
- [84] Tanimoto M, Fujii T, Fukushima N. 1D parallel test sequences for MPEG-FTV[S]. ISO/IEC JTC1/SC29/WG11 MPEG2008/M15378, Ar-champs, France, April 2008
- [85] ISO/IEC JTC1/SC29/WG11, "Multiview Video Test Sequence and Camera Parameters," M15419, April 2008
- [86] "Yuv qcif and cif video file." [Online]. Available: <http://trace.eas.asu.edu/yuv/index.html>
- [87] [Online]. Available: https://www.3dtv-research.org/3dav_CfP_FhG_HHI/
- [88] [Online]. Available: <http://sp.cs.tut.fi/mobile3dtv/stereo-video/>
- [89] [Online]. Available: <http://sp.cs.tut.fi/mobile3dtv/stereo-video/>

Publication List

Journal:

- [1] Wenxin Yu, Weichen Wang, Zhengyan Guo and Satoshi Goto, “An Integrated Hole-filling Algorithm for View Synthesis”, The Institute of Electronics, Information and Communication Engineers (IEICE), Vol.E96-A, No.6, pp.1306-1314, Jun. 2013
- [2] Wenxin Yu, Xin Jin and Satoshi Goto, “Low Power Video Decoding with Adaptive Granularity in Temporal Scalability”, The Journal of the Institute of Image Electronics Engineers of Japan, Vol.42, No.2, pp.249-261,2013
- [3] Ning Jiang, Jiu Xu, Wenxin Yu and Satoshi Goto, “An intra-combined feature for pedestrian detection”, IEEE Transactions on Image Electronics and Visual Computing Vol.1, No.1, pp.88-96, December 2013

International Conference:

- [1] Wenxin Yu, Weichen Wang, Gang He, Satoshi Goto, “Combined Hole-Filling with Spatial and Temporal Prediction”, The International Conference on Image Processing (ICIP), pp.3196 – 3200, Sep. 2013
- [2] Jiu Xu, Ning Jiang, Xinwei Xue, Heming Sun, Wenxin Yu and Satoshi Goto, “Multi-scale Bidirectional Local Template Patterns for Real-time Human Detection”, MMSP, pp.379-383, Sep. 2013
- [3] Ning Jiang, Jiu Xu, Wenxin Yu and Satoshi Goto, “Gradient Local Binary Patterns for Human Detection”, The IEEE International Symposium on Circuits and Systems (ISCAS), pp.978 – 981, May 2013
- [4] Wenxin Yu, Weichen Wang, Zhengyan Guo, and Satoshi Goto, “An Integrated Hole-Filling Algorithm for View Synthesis”, Pacific-Rim Conference on Multimedia (PCM), LNCS 7674, pp.80–92, Dec. 2012
- [5] Wenxin Yu, Xin Jin, Satoshi Goto, "Adaptive Temporal Scalable Decoding Scheme with Temporal and Spatial Prediction Method", The International Symposium on Intelligent

Signal Processing and Communication Systems (ISPACS), Chiangmai, Thailand, Dec. 2011

- [6] Wenxin Yu, Ning Jiang, Xin Jin, Satoshi Goto, “Adaptive low power decoding process with temporal prediction method for common video”, IEEE International Colloquium on Signal Processing & its Applications (CSPA), pp.163 – 166, Mar. 2011
- [7] Ning Jiang, Wenxin Yu, Shaopeng Tang, Satoshi Goto, “A cascade detector for rapid face detection”, IEEE International Colloquium on Signal Processing & its Applications (CSPA), pp 155–158, Mar. 5, 2011
- [8] Wenxin Yu, Xin Jin and Satoshi Goto, “Adaptive solution of temporal scalable decoding process with frame rate conversion method for surveillance video,” The International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Cheng Du, China, pp. 333-336. Dec. 6-8, 2010
- [9] Wenxin Yu, Xin Jin, Satoshi GOTO, “Temporal scalable decoding process with frame rate conversion method for surveillance video”, Pacific-Rim Conference on Multimedia (PCM), Part II, LNCS 6298, Page(s): 297 - 308, Shanghai, China, Sep. 2010

Domestic Conference:

- [1] Wenxin Yu, Xin Jin, Satoshi GOTO, “Low complexity decoding with frame-skipping for surveillance video”, 2010 General Conference of IEICE, D-11-21, Seida, Japan, January 2010