JPEG BACKWARD COMPATIBLE FORMAT FOR 3D CONTENT REPRESENTATION

Philippe Hanhart, Pavel Korshunov, Martin Řeřábek, and Touradj Ebrahimi

Multimedia Signal Processing Group, EPFL, Lausanne, Switzerland

ABSTRACT

Different formats and compression algorithms have been proposed for 3D video content, but 3D images are still mostly represented as a stereo pair only. However, for enhanced 3D rendering capabilities, such as depth perception adjustment or display size adaptation, additional depth data is necessary. To facilitate the standardization process of a common 3D format, backward compatibility with legacy technologies is necessary. In this paper, we propose to extend the JPEG file format, as the most popular image format, in a backward compatible manner to represent a stereo pair and additional depth data. We propose an architecture to achieve such backward compatibility with JPEG. The coding efficiency of a simple implementation of the proposed architecture is compared to the state of the art stereoscopic 3D image compression and storage formats.

1. INTRODUCTION

The popularity of immersive contents, such as 3D, high dynamic range, and ultra-high definition images and video sequences, is increasing. However, there is a lack of unified and standardized approach to compression and storage of such contents in a backward compatible fashion with current legacy technologies. Most available solutions are ad hoc patches to existing approaches adapting them to cope with new types of content. For instance, the most popular formats for stereoscopic 3D images are JPEG Stereoscopic (JPS) and Multi-Picture Format (MPO). These formats encode a stereoscopic image as a JPEG image of double width (JPS) or a series of JPEG images (MPO), without taking into account any similarities between the views, making the compression inefficient. Moreover, whereas JPS can be considered as a primitive solution of JPEG backward compatible format for a single stereo pair (this format does not support multiview images), MPO is not compatible with legacy decoders. Therefore, there is a need in standardized backward compatible solutions for compression and storage of stereoscopic 3D images.

Efficient compression of stereo pairs has been of research interest since 1986, when Lukacs [1] introduced disparity compensated prediction. Since then, many other algorithms have been proposed [2, 3, 4, 5], most of which use block-based disparity compensated prediction to compress the second view of the stereo pair. While the idea of exploiting similarities between the different views has a lot of merit, such block-based solutions are not optimal for 3D contents represented as stereo plus depth or multiview plus depth, and often suffer from a lack of backward compatibility.

To resolve the drawbacks of existing approaches, in this paper, we propose a general architecture for a JPEG backward compatible format for stereoscopic 3D images. According to the format, one view of the stereoscopic 3D image is encoded as JPEG, with additional information about other view and depth data stored in a lossy or lossless way inside its APPn marker. Such approach ensures that at least one view can be decoded by any legacy JPEG decoder, whereas a specialized decoder would be able to recover (or synthesize) additional views of the 3D content. Therefore, the proposed format allows improved applications such as depth perception adjustment and display size adaptation.

To demonstrate the feasibility of the proposed JPEG backward compatible format, we implemented a simple prototype, which assumes as input the left and right views together with one depth map. The left view is encoded with JPEG. A downsampled version of the depth map is encoded with JPEG 2000. The difference between the original right view and its synthesized approximation (from left view and depth map), called *residual*, is also encoded with JPEG 2000. To ensure seamless decoding, the depth map is stored together with the residual. For evaluation of the coding performance, we used a set of key frames from the 3D video sequences used by the Joint Collaborative Team on 3D Video Coding Extension Development [6]. The 3D data, i.e., stereo pair and depth map, was encoded using our prototype and as separated images, as it is done in the JPS and MPO formats. The compression efficiency was then measured using the Bjøntegaard model.

The remainder of the paper is organized as follows. The proposed stereoscopic 3D image compression architecture is described in Section 2. In Section 3, the coding efficiency of a simple implementation of the proposed architecture is reported and analyzed. Finally, concluding remarks and discussion on future work are given in Section 4.

This work has been conducted in the framework of the Swiss National Foundation for Scientific Research (FN 200021-143696-1), EC funded Network of Excellence VideoSense, Swiss SER project Quality of Experience in 3DTV, and COST IC1003 European Network on Quality of Experience in Multimedia Systems and Services QUALINET.

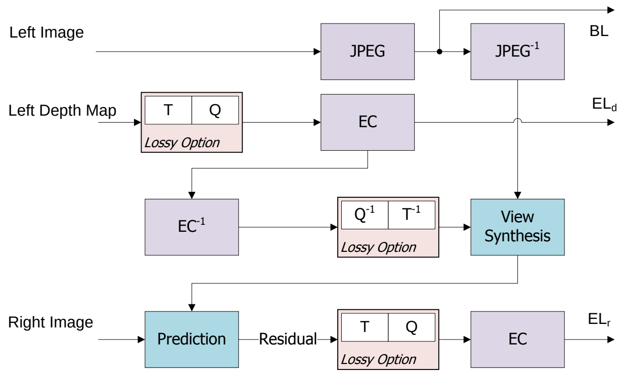

Fig. 1: Scheme of JPEG backward compatible encoding process.

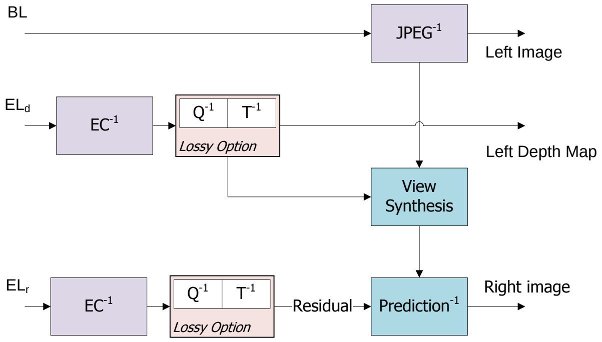

Fig. 2: Scheme of JPEG backward compatible decoding process.

2. PROPOSED STEREOSCOPIC 3D IMAGE COMPRESSION

In this section, we first present a general overview of the proposed JPEG backward compatible compression architecture for stereoscopic 3D images, followed by the description of an example prototype implementation of this architecture.

2.1. Architecture of encoder and decoder

Figure 1 depicts the generic block diagram of the encoder architecture proposed in this paper for stereoscopic 3D image compression, with the feature of being JPEG backward compatible, while offering a more optimal solution when compared to the state of the art.

By JPEG backward compatibility, we mean that when the resulting bitstream is fed into a conventional JPEG decoder, the latter can decode it into an image and display one view of the original stereoscopic 3D image.

In the diagram of Figure 1, *JPEG* and $JPEG^{-1}$ indicate a conventional JPEG compression and decompression, respectively. In particular, the most widely used JPEG compression format relies on a Y'CbCr color image representation, with all three components represented as 8-bits unsigned integer, and where the Cb and Cr components are sub-sampled by a factor of 2 in both horizontal and vertical directions.

In the diagram, three inputs are assumed to be available: left view, right view, and left depth map. In practical scenarios, the depth map can be estimated directly from left and right views, or captured from the scene using specific sensors. Furthermore, the position of the depth map could be in between the views, in which case the proposed architecture can be adapted in a straightforward manner.

No further assumption is made and the compression ratio, or any other JPEG compression parameters such as the choice of quality factor, quantization and entropy coding tables, or any pre- and post- processing for the purpose of JPEG compression and decompression are left to the encoder, with the largest degree of flexibility.

BL indicates the baseline portion of the resulting bitstream and consists of a fully compatible JPEG format, readable by any JPEG compliant decoder. As in many extensions of JPEG, the additional bitstream necessary for decoding both views of the stereoscopic 3D image will be included in the baseline JPEG format thanks to an appropriate APPn application marker, as proposed in the JPEG standard.

The left depth map is transformed and quantized (if lossy compression is used) and then entropy coded resulting in EL_d. The depth map and view synthesis process, which are used to estimate the right view from the decoded left view, are essential components of the proposed algorithm. View synthesis is a technique used to synthesize a virtual (or real) view at a selected position from available view(s) and associated depth map(s). One example of view synthesis technique is depth-image-based rendering (DIBR).

The output of *View Synthesis* is then fed into a prediction component, which will compute the residual using the original right view of the input stereoscopic 3D image. The prediction can be either in the form of a differential, a ratio, or a more complex mechanism. The residual image is transformed and quantized (if lossy compression is used) and then entropy coded. The enhancement layer containing the residual of the input image (EL_r), along with the depth map (EL_d), is then embedded inside the JPEG compressed file format, using the APPn marker as indicated earlier.

Figure 2 depicts the proposed decoder and essentially performs the dual operation of the encoder.

2.2. Illustrative implementation

To illustrate the compression scheme presented in Figure 1 and Figure 2, we implemented a variant of a simple codec based on the proposed solution. The main goal of this implementation is to demonstrate the feasibility of the proposed general compression scheme by successfully exploiting the similarity between the left and right views of a stereoscopic 3D image, consecutively, showing that we can achieve efficient compression even via simple means.

Although, the proposed solution can cope with both lossless and lossy compression, in our illustration, we focused only on lossy scenario. For implementation of various components, we relied on MATLAB implementation of JPEG and JPEG 2000.

The detailed process of stereoscopic 3D image compression in a JPEG backward compatible manner consists of the following steps:

1. Compress the left view of the stereoscopic 3D image with JPEG and save it as baseline (*BL*).
2. Downsample the left depth map to quarter resolution and compress it with JPEG 2000 (EL_d).
3. Decode the left view and depth map.
4. Upsample the decoded depth map.
5. Synthesize the right view of the stereoscopic 3D image using the decoded left view and upsampled depth map:
   (a) Warp the left depth map to the right view.
   (b) Fill holes using background propagation.
   (c) Apply 3 × 3 median filtering.
   (d) Apply reverse warping to synthesize the right view.
6. Construct the residual as the difference between the original right view and its synthesized estimation.
7. Compress the residual (16-bits) with JPEG 2000 (EL_r).
8. Store the compressed residual (EL_r) and depth map (EL_d) inside the final JPEG file using an APPn marker.

The decoding process performs a similar but dual process when compared to encoding.

3. RESULTS AND DISCUSSION

Seven multiview video plus depth (MVD) contents were used in the experiments (see Table 1). These contents are used by the Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) of VCEQ and MPEG [6], as well as by other researchers, to evaluate the performance of 3D video compression algorithms. *Undo Dancer* and *GT Fly* are computer-generated scenes with ground truth depth maps, whereas the five remaining contents consist of real scenes where the depth maps are estimated using the Depth Estimation Reference Software (DERS) [7]. One key frame, which maximizes the amount of depth, was selected for each content (see Table 1) to evaluate the performance of the proposed format to handle large disoccluded areas. The encoded views used in the experiments were the same as the ones specified in the Common Test Conditions [6] of the 3DV Core Experiments conducted by JCT-3V.

Table 1: Stereoscopic contents used in the experiments.

Content	Resolution	Encoded views	Frame no.
Poznan Hall2	1920 × 1088	$7-6$	200
Poznan Street	1920 × 1088	$5-4$	250
Undo Dancer	1920 × 1088	$1-5$	148
GT Fly	1920 × 1088	$9-5$	158
Kendo	1024 × 768	$1-3$	241
Balloons	1024 × 768	$1-3$	10
Newspaper	1024 × 768	$2-4$	196

To compare the performance of the proposed coding architecture, the PSNR based rate-distortion analysis was conducted. The PSNR was computed as the average PSNR of the left and right views of the stereo pair. The following bit rates were targeted for each view of the stereo pair: 0.25, 0.50, 0.75, 1.00, 1.25, and 1.50 bpp. The 3D data, i.e., stereo pair and depth map, was encoded using our prototype and as separated images, as it is done in the JPS (without depth) and MPO formats. Whereas the proposed coding architecture processes the left view in the same manner as JPS and MPO, the bit rate of the right view is divided between depth map and residual. As depth map images can be compressed at bit rates significantly lower than 20% of the bit rates of color images [8], the bit rate of the right view was asymmetrically divided between depth map and residual. More particularly, 10% of the bit rate of the right view was devoted to the depth map, at most, and the rest to the residual image. For the MPO format, the left and right views of the stereo pair were encoded at the targeted bit rate and the left depth map was encoded at 10% of the bit rate of one view, at most.

Figure 3 shows the rate-distortion graphs of two real contents (*Poznan Hall2* and *Kendo*) and one synthetic content (*Undo Dancer*), to illustrate the performance of the proposed and JPS coding schemes. For real contents, the performance of the proposed compression architecture is better than JPS for bit rates below 0.50 bpp, whereas JPS outperforms our architecture for higher bit rates. However, the difference in PSNR values above 0.50 bpp is less than 0.8 dB. Note also that for bit rates above 0.50 bpp, the PSNR values are over 40.8 dB on all real contents. For synthetic contents, the performance of the proposed coding architecture is better for all targeted bit rates. The difference in PSNR values is between 0.8 and 2.8 dB.

The results for synthetic and real contents are different because of the different ways the depth maps are produced in both cases. For synthetic content, the depth map is accurately generated as it is computer-generated, whereas for real content, it is estimated using the DERS algorithm [7]. Thus, inaccuracies in depth maps of real contents degrade the accuracy of the synthesized right view, consequently, increasing the size of the residual, which negatively impacts the rate-distortion performance.

The Bjøntegaard model [9] was used to measure the coding efficiency of the proposed architecture using PSNR measurements. The average bit rate difference, called Bjøntegaard delta rate (BD-Rate), was estimated using a third order logarithmic polynomial fitting [9]. Table 2 reports the average

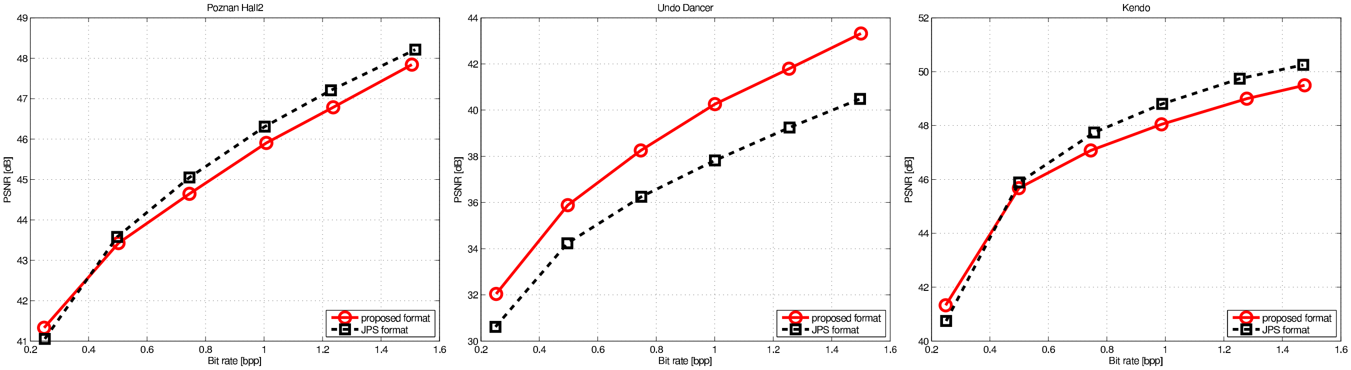

Fig. 3: Rate-distortion performance.

Table 2: Average bit rate difference (BD-Rate) for the proposed coding algorithm in comparison to JPS and MPO formats.

Content **Coding Scheme**	*Poznan Hall2*	*Poznan Street*	*Undo Dancer*	*GT Fly*	*Kendo*	*Balloons*	*Newspaper*	**Average**
JPEG3D vs JPS	$+6.25\%$	-1.57%	-27.86%	-17.00%	$+4.54\%$	$+2.92\%$	$+0.64\%$	-4.58%
JPEG3D vs MPO (JPS+Z)	$+2.31\%$	-6.15%	-30.90%	-20.80%	-0.30%	-1.83%	-4.08%	-8.82%

bit rate difference, for each content separately, for the proposed coding architecture when compared to JPS and MPO formats. These values show that up to 27.9% and 30.9% bit rate savings can be achieved for synthetic contents and, on the other hand, up to 6.3% and 2.3% bit rate increasing would be needed for real contents, when compared to JPS and MPO formats, respectively. Averaging the BD-Rate values across contents gives promising results of about 4.6% and 8.8% bit rate savings for the proposed coding architecture when compared to JPS and MPO formats, respectively. The coding efficiency of the proposed coding scheme is better when compared to MPO than when compared to JPS due to the transmission of additional information for the depth map in the MPO format.

4. CONCLUSION AND FUTURE WORK

In this paper, we proposed a general architecture for a JPEG backward compatible format for stereoscopic 3D images. One view of the stereoscopic 3D image is encoded as JPEG, with additional information about the other view and depth data stored inside its APPn marker. Consequently, at least one view can be decoded by any legacy JPEG decoder. Results showed that a simple implementation of the proposed architecture can efficiently compress stereoscopic 3D images with an average bit rate reduction of 4.6% and 8.8% when compared to JPS and MPO formats, respectively.

The results presented in this paper can be extended in several directions. Different view synthesis techniques, with improved inpainting, can be exploited to construct a better estimation of the right view. An alternative way to compute the residual can be investigated. Different compression of the residual image, both in lossy and lossless fashions, can be explored as well. Finally, a rigorous subjective evaluation could be performed to compare the proposed compression scheme to state of the art formats, from subjective quality point of view rather than PSNR, as reported in this paper.

5. REFERENCES

[1] M. Lukacs, "Predictive coding of multi-viewpoint image sets," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Apr. 1986, vol. 11, pp. 521–524.

[2] M.G. Perkins, "Data compression of stereopairs," *IEEE Trans. on Communications*, vol. 40, no. 4, pp. 684–696, Apr. 1992.

[3] W. Woo and A. Ortega, "Stereo image compression with disparity compensation using the MRF model," in *Proc. SPIE 2727, Visual Communications and Image Processing*, Feb. 1996, pp. 28–41.

[4] M.S. Moellenhoff and M.W. Maier, "Transform coding of stereo image residuals," *IEEE Trans. on Image Processing*, vol. 7, no. 6, pp. 804–812, June 1998.

[5] T. Frajka and K. Zeger, "Residual image coding for stereo image compression," *Optical Engineering*, vol. 42, no. 1, pp. 182–189, Jan. 2003.

[6] ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, "Common Test Conditions of 3DV Core Experiments," Doc. JCT3V-D1100, Incheon, Korea, Apr. 2013.

[7] ISO/IEC JTC1/SC29/WG11, "Draft Report on Experimental Framework for 3D Video Coding," Doc. N11478, Geneva, Switzerland, July 2010.

[8] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proc. SPIE 5291, Stereoscopic Displays and Virtual Reality Systems*, pp. 93–104, May 2004.

[9] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," Tech. Rep. VCEG-M33, ITU-T SG16/Q6, Austin, Texas, USA, Apr. 2001.