

# MPEG-C PART 3: ENABLING THE INTRODUCTION OF VIDEO PLUS DEPTH CONTENTS

*Arnaud Bourge\**, *Jean Gobert\** and *Fons Bruls\*\**

\*Philips Applied Technologies  
Suresnes, France

\*\*Philips Research  
Eindhoven, The Netherlands

## ABSTRACT

We present the ISO/IEC 23002-3 (a.k.a. MPEG-C part 3) specification, which we are currently editing inside the Motion Picture Experts Group. This specification gives a representation format for depth maps which allows encoding them as conventional 2D sequences. Some useful parameters are also defined, which can be conveyed at system level to correctly interpret the decoded depth values at the receiver side. This standardized video plus depth solution provides interoperability of the content, flexibility regarding transport and compression techniques, display independency and ease of integration.

We also demonstrate a practical implementation of a mobile video plus depth system: we modified a version of the Renoir visual IP for 3D rendering. The system is connected to a 5-view lenticular display with a resolution of 800x480 RGB pixels and a screen width of 9cm.

## I. INTRODUCTION

The introduction of three-dimensional visualization is expected to be the next technological differentiator in consumer-oriented video devices. Although the complete chain, from the content generation to the viewing experience, is not totally mature yet, the sales of “3D screens” has already started and their growth indicates that it could become a mass market very fast.

Most of these screens are auto-stereoscopic displays, generally based on lenticular technology, sometimes on parallax barrier. Advantageously, these displays provide a depth impression without the need to wear glasses, so they are well-suited for TV at home, mobile devices and public spaces. Other techniques are glasses-based and include polarized glasses, anaglyph and shutter glasses synchronized with alternate frame sequencing. This type of solution is envisioned for 3D digital cinema in theatres for instance.

At the other side of the chain, 3D content generation is also growing. Sequences can now be directly shot with stereo cameras or with an RGB+infrared one. Another important source of stereoscopic material is the “2D to 3D

conversion” of existing movies, for which fully automated and semi-manual algorithms are proposed.

So, as stereoscopic displays are entering the market and 3D video content generation is progressing, the need for a standardized way of exchanging data was urging. Responding to a strong industry demand in that direction, the Motion Picture Experts Group recently launched the ISO/IEC 23002-3 specification (a.k.a. MPEG-C part 3) [1]. It will be published as a final standard in January 2007.

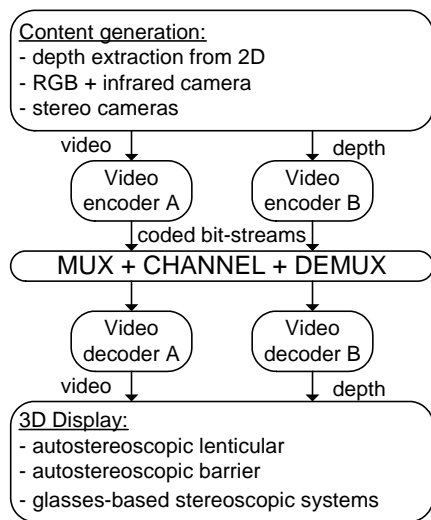
Indeed the success of 3D stereoscopic video will not only depend on the quality of the displays and on the availability of suitable content. The acceptance of a new technology by the market is also linked to some fundamental rules: interoperability, display technology independency, backwards compatibility and compression efficiency. MPEG-C part 3 fulfils these four requirements. Moreover, it also allows encoding depth-map sequences with existing (and even future) video compression standards. This efficient re-use of well-known techniques obviously reduces industrialization costs and time-to-market delays.

MPEG-C part 3 is based on the encoding of 3D content as video plus depth, a concept that is already well known and has been studied inside the European project ATTEST [2]. In section II we briefly recall the benefits of this solution and then describe the main features and advantages of the new specification. As an illustration, we present in section III an implementation of video plus depth decoding and rendering on a combined software/hardware platform. By reusing “as is” the 2D functions that were already present in our hardware renderer (called Renoir), we managed to upgrade it from 2D to 3D capabilities while increasing its number of gates by only 8%. It decodes and renders compressed 2D+Z QVGA sequences at roughly 10 frames per second.

## II. DESCRIPTION OF MPEG-C PART 3: CODING DEPTH AS AUXILIARY VIDEO DATA

### 2.1 Benefits of the video plus depth approach

Figure 1 illustrates a complete stereoscopic chain, from the capture to the display.



**Figure 1: The complete stereoscopic chain.**

In this chain, using video plus depth as the exchange format allows [3]:

- Backwards compatibility with 2D,
- Independency regarding the display technology,
- Independency regarding the capture technology,
- Direct compatibility with most “2D to 3D” algorithms,
- Good compression efficiency (low overhead),
- User-controlled global depth range.

Of course this approach has some limitations: a very strong depth is difficult to handle at the display because occlusion areas are too big. However, it is known that a good viewing comfort requires a small depth range (otherwise headaches appear) [6][7]. Furthermore, a limited depth impression is suitable for an introduction scenario of 3DTV and can pave the way for free-viewpoint video [8].

## 2.2 Mapping depth to a generic representation format

More generically than just depth, the ISO/IEC 23002-3 standard specifies an Auxiliary Video Data format. It simply consists of an array of N-bit values that are associated with the individual pixels of a regular video stream. These data can then be compressed like conventional luminance signals using already existing (and even future) MPEG video codecs.

The type of auxiliary video data is signaled by the 1-byte syntax element *aux\_video\_data\_type*. One of the strengths of the standard is that it only specifies the semantics of *aux\_video\_data\_type* depending on its value and leaves open the way it is actually conveyed. This brings a great flexibility because any transport specification can include this syntax element in a way that suits its own structure and just refer to ISO/IEC 23002-3 (see 2.3). Initially, two types are defined: depth and parallax (which can be seen as a

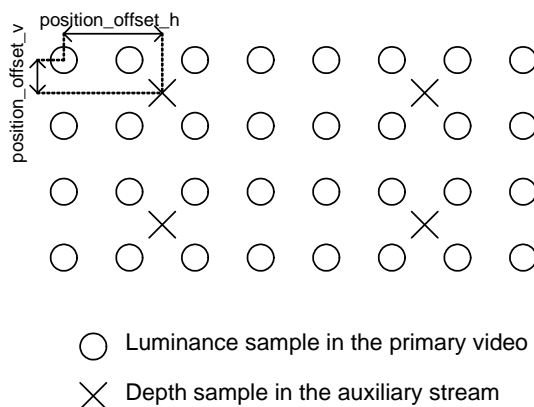
hyperbolic representation of depth), respectively corresponding to 0x10 and 0x11. Other values are reserved for future use. In case a new data type is included (for instance infra-red temperature maps) this is transparent for the transport layers, which do not need to be updated.

Following this generic format, a depth value  $z_p$  is represented by an unsigned N-bit value  $m$ :

$$z_p = \frac{m}{2^N} (k_{\text{near}} W + k_{\text{far}} W) - k_{\text{far}} W$$

$W$  represents the screen width at the receiver side, so it does not need to (and cannot) be transmitted.  $W$  can also be viewed as a control on the depth range.  $k_{\text{near}}$  and  $k_{\text{far}}$  specify the relative range of depth respectively behind and in front of the display. Their values are transmitted via a specific metadata structure: *aux\_video\_params*, which can be viewed as a list of parameters. The content, syntax and semantics of this structure are described in the specification and depend on *aux\_video\_data\_type*. For each parameter, default (typical) values are defined in case *aux\_video\_params* is not explicitly sent. Finally, the specification allows new parameters to be easily added for future extensions, in a backwards compatible way.

Furthermore, ISO/IEC 23002-3 allows for optional sub-sampling of the depth map in both the spatial and temporal domain. This can be beneficial for some particular applications where very low bitrates are needed [4]. The up-sampling to the original resolution at the receiver side is voluntarily left open. However, in order to avoid unkeying between the 2D video and its depth map, a position offset indicating the phase of the down-sampling filters is also present in *aux\_video\_params* (Figure 2).



**Figure 2: position offset of 2D and depth.**

## 2.3 Signaling depth at systems level

Thanks to ISO/IEC 23002-3, depth maps can be encoded as 2D video sequences. However, a receiver must be able to distinguish these two types of data to correctly reconstruct three-dimensional views (and more simply to prevent 2D

receivers from displaying depth maps). This is done through a signaling at systems level. So as to use depth map coding in broadcast and optical media, we have thus developed an amendment to the MPEG-2 Systems standard in order to transport auxiliary video data [5]. A new descriptor is specified, which contains the value of *aux\_video\_data\_type* (depth, parallax, etc.) and the metadata structure *aux\_video\_params*. Carrying these data there allows the use of strictly unmodified video codecs, for which the process is fully transparent.

### III. APPLICATION TO A MOBILE DISPLAY PLATFORM

Rendering texture and depth requires functionality that has a lot in common with the features already supported by the Renoir visual renderer. Based on this observation, we designed a prototype of a modified version of Renoir enabling it to support 3D as video plus depth.

#### 3.1 Adaptation of a visual 2D Renderer

The rendering algorithm was adapted to the tiled-based processing of Renoir. We reuse “as such” functions common to 3D and 2D rendering, which are: reading texture bitmap, interpolating the texture when samples are read from non integer positions and write result pixels into memory. We implemented the following modifications:

- Introduce a 3D mode. In this mode, instead of reading an alpha plane (defining the transparency of 2D objects), Renoir reads a depth plane.
- In this 3D mode, the Geometric Transform Unit, normally used for computing affine geometric transformations, is bypassed and replaced by a new block called “Projected Disparity Unit” (PDU).

The PDU takes in charge functions that are 3D specific: compute projected disparity for the different views, manage occlusion and de-occlusions, and interweave views according to the display lens layout. The PDU is organized in three pipelined steps at the level of a line of a tile. The first step is input driven while the last two are output driven. The first step scans the input depth map to find the corresponding projected disparity, the second step fills the gaps which may appear in the projected disparity map. The scan direction of these steps takes into account the view position so that occlusions are properly handled. Then, the third step scans the pixels in the output space, computes texture coordinates from the projected disparity, interpolates input samples and multiplexes views in the appropriate format.

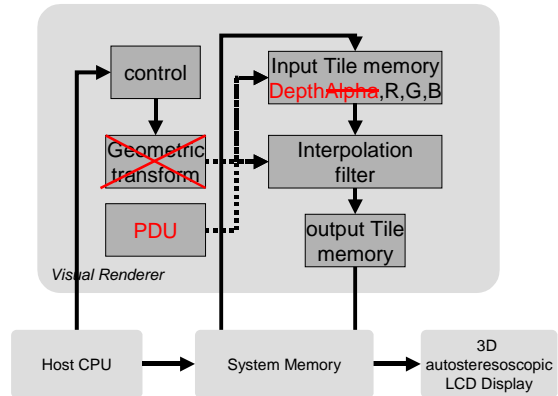


Figure 3: Simplified block diagram of the renderer and modification in 3D mode

#### 3.2 Optimizations

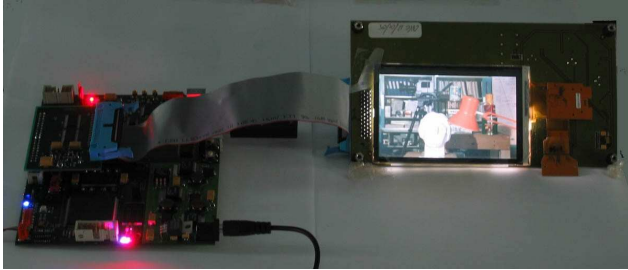
On a lenticular display, a given subpixel of the screen (a red, green or blue dot) will be allocated to one view only. As a consequence, there is no need to compute all views for all subpixels. We can take advantage that the generation of views for the display and their interweaving are performed on the same engine, to optimize the processing and only generate pixels components for those dots that are actually selected for the views. This reduces also the bus bandwidth, as Renoir does no transfer to memory intermediary views at full resolution, but only the multiplexed bitmap.

We introduced a further bandwidth optimization. The texture and depth map are read at half of the LCD resolution in both directions and expanded in the rendering process. This saves 75 % of the input bandwidth and has little impact on the resulting quality, as the effective resolution of views is anyway only a fraction of the native LCD one (e.g. 1/5<sup>th</sup> on a five view module). In the process of geometric transformation, the expansion needed by the fact that we read half resolution bitmaps is combined with the horizontal offset of the disparity.

#### 3.3 Implementation

The modification was implemented in VHDL for a prototype version. Simulations show that satisfactory image quality can be achieved. The increase of complexity is about 15 Kgates, which represents 8 % of the whole renderer. In order to test the concept, we implemented a video plus depth decoder and the multiview renderer into a mobile display platform developed in collaboration with “GET/Télécom Paris”. The board includes a CPU and the FPGA in which the multiview renderer prototype was integrated. We connected the system to a 5-view lenticular display module with a resolution of 800x480 RGB pixels

and a screen width of 9cm (“Moscow” display module). The complete system will hopefully be shown during the poster session.



**Figure 4: Picture of the 3D platform and display.**

#### IV. CONCLUSIONS

ISO/IEC 23002-3 provides an efficient way for attaching depth (or parallax) values to the individual pixels of a regular video stream. It allows re-using existing video compression standards, while its flexible structure makes it easily upgradeable. Furthermore, the advantages of the video plus depth approach are fully exploited (simplicity, low 3D overhead, display independency, backwards compatibility with 2D, adjustable depth-effect at display). Finally a prototype for mobile applications has been developed, which demonstrates the effectiveness of such a method.

#### VI. REFERENCES

- [1] ISO/IEC JTC 1/SC 29/WG 11. Committee Draft of ISO/IEC 23002-3 Auxiliary Video Data Representations. WG 11 Doc. N8038. Montreux, Switzerland, April 2006.
- [2] C. Fehn, P. Kauff, M. Op de Beeck, F. Ernst, W. Ijsselsteijn, M. Pollefeys, L. Van Gool, E. Ofek and I. Sexton, “An Evolutionary and Optimised Approach on 3D-TV”, *Proceedings of International Broadcast Conference*, pp. 357-365, Amsterdam, The Netherlands, September 2002.
- [3] C. Fehn, “Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV”, *Proceedings of SPIE Stereoscopic Displays and Virtual Reality Systems XI*, pp. 93-104, San Jose, CA, USA, January 2004.
- [4] ISO/IEC JTC 1/SC 29/WG 11. Applications and Requirements for StereoScopic Video (SSV), WG 11 Doc. W7777, Bangkok, Thailand, January 2006.
- [5] ISO/IEC JTC 1/SC 29/WG 11. Proposed Draft Amendment of ISO/IEC 13818-1:200X/AMD 2. WG 11 Doc. N8094. Montreux, Switzerland, April 2006.
- [6] W.A Ijsselsteijn, H. de Ridder and J. Vliegen, “Subjective evaluation of stereoscopic images: effects of camera parameters and display duration”, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 2, March 2000.
- [7] W.A Ijsselsteijn, H. de Ridder and R. Hamberg, “Perceptual factors in stereoscopic displays. The effect of stereoscopic filming parameters on perceived quality and reported eye-strain”, *Proc. SPIE*, vol. 3299, pp. 282-291, 1998.

- [8] P. Kauff, A. Smolic, P. Eisert, C. Fehn, K. Müller, R. Schäfer, “Data format and coding for free-viewpoint video”, *Proceedings of International Broadcast Conference*, Amsterdam, The Netherlands, September 2005.