Vision, Modeling, and Visualization (2013)
Michael Bronstein, Jean Favre, and Kai Hormann (Eds.)

Image Based Rendering from Perspective and Orthographic Images for Autostereoscopic Multi-View Displays

Daniel Jung and Reinhard Koch

Computer Science Department, Christian-Albrechts-University Kiel
Hermann-Rodewald-Str. 3, 24118 Kiel, Germany

Abstract

Current autostereoscopic (AS) multi-view displays for video that are targeted at the market allow typically up to 60 frames per second and offer between 20 and 60 different views per pixel. Future full parallax AS displays may well require thousands of views simultaneously. With the large number of different views video displays consume a huge amount of data, either transferred to the display or to be computed on demand from a 3D scene representation. In the following a novel depth-image based rendering interpolation algorithm targeted at multi-view video displays is introduced that combines the results of an interpolation on orthographic and perspective images. The same idea is further utilised to implement an efficient computer graphic rendering algorithm for full parallax AS displays.

Categories and Subject Descriptors (according to ACM CCS):
I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms
I.4.0 [Image Processing and Computer Vision]: General—Image displays and Image processing software

1. Introduction

In the past few years several full parallax displays for static content became available to the consumer market that encode between 50,000 and up to 200,000 different views per pixel, based on holography and arrays of microlenses (corporations: Zebra Imaging [HC02], REALEYES GmbH [JK11]). Typical AS displays for video range from 20 to 60 different views per pixel, found in horizontal-parallax lenticular displays [HDFP11] and up to about 300 different views per pixel in full 2D parallax microlens arrays, deduced from [AOK*10]. A comprehensive survey of current multi-view displays was performed by Holliman et al. [HDFP11] and Yaraş et al. [YKO10].

The static content displays show that it is possible to manufacture and assemble displays capable of thousands of different views. With the progress in miniaturisation in LED displays and laser video projectors an increasing number of views for multi-view video displays is to be expected.

There are several different approaches to realise a full 2D parallax AS display. On the left hand side of Fig. 1 a draft of a display based on a microlens array is shown. The number of lenses in width (o_w) and height (o_h) define the resolution of one view. One microlens of the display is called a *display element* in the remainder and the distance between two display elements (o_d) is the pixel pitch, which defines the *spatial resolution* of the display. The right hand side of Fig. 1 shows a display element that is composed of a microlens and a full 2D colour image behind the lens, called an *elemental image* (e) of dimension e_w x e_h. Depending on the position of an observer the microlens limits the visible part of the elemental image to the corresponding viewing direction, yielding the observed view.

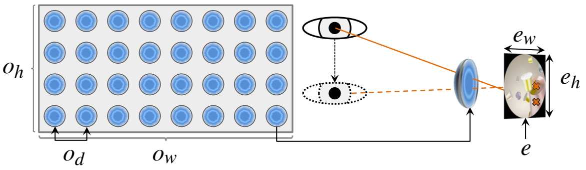

Figure 1: Draft of an AS multi-view display based on a microlens array.

With an increasing number of views the requirements for storage and transmission of the high data rate of AS displays becomes a challenge. Arai et al. [AOK*10] and Mishina [Mis11] introduced a 3D TV system that allows the

DOI: 10.2312/PE.VMV.VMV13.187-194

capturing of a scene and playback on an AS display based on a microlens array. For capturing they used a high resolution camera and for playback a high resolution projector of the same resolution. According to their specification the capturing and display components had an uncompressed data rate of 100 MB per frame resulting in 6.0 GB per second at the full frequency of 60 frames per second.

Balogh et al. [BKB07] came to the conclusion that the data rate of future multi-view display technology will increase by 10^2 to 10^4 compared to current HDTV. Xu et al. [XPLL11] analysed the bandwidth requirements of a 3D holographic display. The bandwidth requirement of their display was about 1.3 GB per second and they solved the transmission challenge via a local network consisting of ten 1 Gbps channels. They came to the conclusion that the bandwidth requirement of AS 3D displays will increase to the range of 12.5 GB to 125 GB per second due to the increasing resolution of AS displays. The authors proposed lossless compression or the transmission of 3D object data to deal with the increasing bandwidth requirements.

This work has the goal to reduce the input data rate of multi-view video displays. The idea is to transfer only a sparse sub-set of computer generated colour and depth maps to the device and let specialised hardware on the device interpolate the full data set with a depth-image based rendering (DIBR) algorithm. The goal is to reduce the data rate by a factor that will allow to use off-the-shelf hardware to operate AS displays with a very high number of views and avoid some of the drawbacks of DIBR by combining perspective and orthographic input data.

1.1. Previous Work

Halle and Kropp [HK97] introduced an algorithm for the efficient rendering of perspective computer graphics images for full parallax displays. Based on their work Holzbach and Chen [HC02] developed an algorithm that avoids rendering artefacts introduced by the clipping of polygons at the near clipping plane and handles degenerate cases, allowing for a commercial application. Balogh et al. [BKB07] uses an *OpenGL* wrapper between the application and their display to render computer graphics content for their display, allowing to display content rendered with *OpenGL*. Annen et al. [AMZ*06] used a distributed rendering system to render images for AS displays at interactive rates. They implemented several distributed rendering algorithms for Chromium [HHN*02] and rendered images for front and rear projection AS displays. Jung and Koch [JK11] introduced a DIBR algorithm for the elemental images of a full parallax AS display in order to reduce the rendering time for ray traced content. Their initial approach of using a sparse set of regular sampled depth and colour elemental images to build a point based scene representation lead to missing parts of the scene in the interpolated images when scene content was near the display plane. They solved the problem by using a scene analysis to select the most relevant elemental images for their viewpoint interpolation. Farre et al. [FWL*11] introduced an algorithm to render novel views based on image domain warping that implicitly handles artefacts introduced by occlusions. Most recently, Heide et al. [HWRH13] proposed to render light fields by an iterative approach of alternating taking plenoptic samples, followed by an optimisation in order to find the best samples for the next iteration, minimising the residual. Depending on the scene they reduced the rendered light rays to 1.62% - 11.1%. For a light field of $1.1 \cdot 10^9$ rays they reported a rendering time of about 99 hours with an additional hour for the optimisation. In horizontal parallax displays, which are available commercially, usually layered depth video (LDV) is utilised to render the different views out of a central view, a depth map and an occlusion layer. The LDV format emerged from the layered depth images that were introduced by Shade et al. [SGHS98].

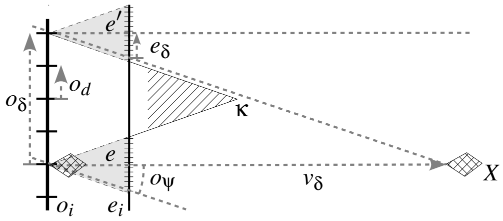

Figure 2: Relationship between object distance and disparity for perspective and orthographic input images.

Buehler et al. [BBM*01] evaluated image based rendering algorithms on the basis of a set of desirable goals and introduced an algorithm to meet these goals. By evaluating a sparse blending field for the input images they utilised the graphics hardware to interpolate a dense blending field and blend the input images, rendering free viewpoints at interactive rates. Magnor et al. [MRG03] used a model-based coding to compress the large amount of image data needed for IBR and achieved high compression ratios of over 2000:1. Matusik and Pfister [MP04] implemented the full production chain for 3D TV from light field acquisition to display on an AS display. For transmission they used a temporal encoding of the individual views with MPEG-2. Merkle et al. [MMSW06] utilised an Hx264/MPEG4-AVC codec to encode multi-view video streams exploiting similarities in the temporal and viewpoint domain.

1.2. Depth-Image Based Rendering

Figure 2 shows a draft of the display plane (o_i) and a virtual object (X). The two elemental perspective images (e and e') are considered as input images for a DIBR algorithm with the goal to interpolate the elemental images between both input positions. The projection centre for all elemental images is located in the display plane. At (κ) the field-of-view of the input images overlap, hence objects within the shaded region can't be reconstructed from the elemental images (e) and (e'), leading to holes in the interpolated images. One way to avoid such holes is a scene analysis (see [JK11]) that computes an optimised set of elemental images, depending on the objects position in the scene. With increasing depth (v_δ) of the object (X) its disparity (e_δ) in image (e') becomes smaller, converging to zero for (v_δ) towards infinity.

Orthographic images consist of parallel viewing rays, one pixel per display element with a pixel pitch of (o_d). Two orthographic input images are sketched in Fig. 2, the direction orthogonal to (o_i) and the direction under an angle of (o_ψ). The projection centre of the orthographic images are at infinity and the viewing rays between (e) and (e') are omitted for clarity. The disparity (o_δ) between the orthogonal orthographic view and the view direction (o_ψ) is zero for objects located in the display plane and becomes infinity for (v_δ) towards infinity. Depending on the display size there is a distance from where on two different orthographic view directions have no overlapping field-of-view, especially for objects at infinity an interpolation solely on orthographic input images will lead to holes in the interpolated images.

The proposed idea is to use interpolation on orthographic images for content that is close to the display plane and switch to the interpolation on perspective images when the content has a given minimum distance to the display plane.

2. Our Contribution

This paper introduces an DIBR algorithm suited for the interpolation of elemental images for full parallax AS displays. The motivation to use a DIBR algorithm to interpolate the full frame of the AS display from a sparse sub-set of depth and colour images is to reduce the data rate between the computer that stores the AS video data and the display device. Unlike prior work the proposed algorithm avoids a computational costly full scene analysis [JK11], an optimised scene sampling [HWRH13] or encoding in favour of an algorithm that could be ported with little effort to specialised hardware and offers a structured and constant input data rate. For transmission the input data of the DIBR algorithm could be further reduced by a suitable compression algorithm.

Our contribution further covers the rendering of input data for the interpolation algorithm. The input data consist of perspective and orthographic images and corresponding depth maps. Most modelling tools that use a ray tracer for rendering grant native support for the creation of such images or offer a ray camera that can be utilised to render the elemental input images. Based on the work of Halle and Kropp [HK97] a method for rendering of the elemental images with *OpenGL* is introduced that offers a new solution for positioning the near clipping plane and solves the problem of viewpoint shifting without a scene analysis.

The evaluation demonstrates that a DIBR algorithm can benefit from the combination of orthographic and perspective input images in contrast to the same DIBR algorithm that uses the same number of input pixels solely from orthographic images and perspective images respectively.

2.1. Overview

The first step of the proposed algorithm is the interpolation (I_p) of all perspective elemental images of the display out of a sparse set of regular sampled colour and depth elemental images of the display. Afterwards a second interpolation (I_o) is executed, based on a sparse set of orthographic colour and depth images. The orthographic input images are evenly distributed in the domain of all viewing directions supported by the display. After all orthographic images of the display are interpolated the interpolation results are combined (I_c) to yield the final interpolation result. The final output are all elemental images of the display. Hence, the full set of interpolated orthographic images has to be translated into a perspective representation (${}_oTo_p$), which is done while combining the interpolation results.

2.2. Algorithm

Interpolation of the images is achieved by forward mapping of the colour information from the input images into the interpolated image using the corresponding depth maps. All orthographic images share the same image plane (o_i) and the elemental perspective images share the image plane (e_i, see Fig. 2). The image warping uses as input a sparse set of colour images and their corresponding depth maps to interpolate all elemental images of the display. For orthographic input images of dimension $o_w \cdot o_h$ every n^{th} image is taken out of the domain of all viewing directions and for perspective elemental images of dimension $e_w \cdot e_h$ every m^{th} image in width and height is taken out of the spatial domain of all display elements $o_w \cdot o_h$.

First, the closest set of up to four input images is selected for the image that should be interpolated. For every input image of the set a weighting (w_{dist}) is computed as the inverse L_1 norm of the distance between the position of the input image (i) and the image that should be interpolated (t) as

$$w_{dist}(i,t) := \frac{1}{\| p(i) - p(t) \|_1}; \; p(i) \neq p(t). \qquad (1)$$

This weight is calculated only once per interpolated image and normalised after all weights are calculated for the current interpolated view. For orthographic images the function ($p()$) returns the two angles of the viewing direction, whereas for perspective images the function returns the position in the spatial domain of display elements. For perspective images this weighting favours small spatial distances, i.e., small disparities to the input image. For orthographic images small angular distances are favoured, as described by [BBM*01] as the minimal angular deviation criterion.

Let $t_{(x,y)}$ be a pixel at position (x,y) in the input image (t). For every pixel of (t) a cost (c_f) is computed as its distance to the centre of the image $(c_x,c_y) = [width/2, height/2]^T$, with the constraint of $(c_x = c_y)$, by

$$c_f\left(x,y,c_x\right) := max\left(\frac{\| (x,y)^T - (c_x,c_x)^T \|_2}{c_x}, a\right), \qquad (2)$$

normalised by the maximum distance to the centre of the image. A threshold $(a = 0.8)$ is used to ensure constant costs around the centre of the image. Towards the border of the image the costs increase, allowing for a smooth blending. The result is similar to the field-of-view penalty, as described by [BBM*01]. Normalisation is assured by the maximum field-of-view of the elemental image. This weight is only computed for perspective images as the interpolation on orthographic images is on the nearest neighbours in the angular domain.

Then a pixel wise forward mapping of the closest set of input images to the target view is applied. The visibility problem is solved by the use of a depth buffer, blending the forward mapped colour values within a given interval to account for a limited precision of the depth buffer. The colour information is distributed in the 3 x 3 neighbourhood of the warped sub-pixel position (u,v) of the target view (t) to avoid holes due to quantisation. Under the assumption of squared pixels the 1D neighbourhood $b = \| (x,y) - (x+1,y) \|_2$ is used to limit the blending to distances between the half 1D neighbourhood and the half diagonal 2D neighbourhood in order to limit foreground fattening at object borders. Given the distance of the warped sub-pixel position to the current candidate in the 3 x 3 neighbourhood (x,y) as $d_p = \left(\| (x,y)^T - (u,v)^T \|_2\right)$ the blending costs are computed as

$$c_s(d_p) := max\left(d_p, \frac{b}{2}\right) \qquad (3)$$

if $d_p < \frac{b}{\sqrt{2}}$, otherwise the warping of the pixel is aborted.

2.3. Interpolation on Perspective Images

The interpolation on perspective images iterates over all display elements. The final blending weight for warping of an input pixel is computed as

$$w_p\left(c_f, c_s, w_{dist}\right) := \frac{w_{dist}}{c_f} + \frac{w_{dist}}{c_s}. \qquad (4)$$

The weights are accumulated during the interpolation of a pixel for normalisation of the interpolated result.

2.4. Interpolation on Orthographic Images

The interpolation iterates over the domain of all viewing directions, supported by the display. The blending weight of an orthographic input pixel is computed according to Eq. 1, which is already normalised.

In the orthographic images each pixel is represented by a display element. Due to the relatively large distance of two and four millimetre respectively between the display elements (see evaluation 3.3) resulting from the microlens array of the display a discrete forward warping into the interpolated view is used instead of the sub-pixel forward mapping (Eq. 3). Otherwise the distributed forward warping would lead to a large foreground fattening at object borders.

2.5. Combining the Interpolation Results

The interpolation results of the perspective and orthographic input images are combined to yield the final interpolation result. The orthographic input images are primarily used to cover the area around the display plane, where the perspective images don't have an overlapping field-of-view. The disparity in an orthographic image (o_δ) is directly correlated with the number of display elements the image content is transferred across, see Fig. 2. After the perspective interpolation (I_p) all orthographic images are interpolated (I_o) and translated into the perspective representation $({}_oTo_p)$. The final interpolation result (I_c) is decided for each pixel at position (x,y) by

$$I_{c(x,y)} = \begin{cases} {}_oTo_p(I_o)_{(x,y)} & \text{if } o_\delta < m \\ I_{p(x,y)} & \text{else,} \end{cases} \qquad (5)$$

therefore replacing (I_p) with the result of (I_o) when the disparity in the orthographic image (o_δ) is smaller as the distance between neighbouring perspective input views (m). This limits the disparity for both interpolations, because for perspective images the disparity decreases with an increasing distance to the display plane, whereas for orthographic images the disparity decreases with a decreasing distance to the distance plane, reducing the size of potential holes in the combined interpolated views. The threshold also offers the opportunity for an early abort of the warping algorithm that runs second. Due to the low disparity in the orthographic images near the image plane the discrete forward mapping doesn't introduce severe artefacts in the combined result.

2.6. Suitability for a Highly Parallel Execution

The interpolation algorithm was designed to be ported to a FPGA in the future and should allow for a modular composition of independent display devices. A fixed input data access pattern is achieved by limiting the interpolation to the four nearest neighbours. The algorithm allows for a modular composition of display devices because the input data depends only on the actively used part of the display plane. The warping algorithm allows for an efficient computation on specialised hardware, especially when disparity is used instead of depth, and instead of the L_2 norm, e.g., the L_1 norm is used for calculation of the costs. A further adaption will be the use of a fixed-point number representation. Depending of the computational capability of the FPGA the costly distribution of the colour information could be omitted, which would result in a simplified blending weight and would allow to skip the computation of (c_s).

One requirement for the efficient computation is a full frame and depth buffer. Hence, either the fraction of the display that one FPGA interpolates or the resolution of the elemental images has to be adjusted to the available memory. The frame buffer is used for the accumulation of the colour values and transferring the orthographic representation into the perspective elemental image, because every elemental image depends on all orthographic images. The depth buffer is used to solve the visibility problem.

2.7. Rendering for Full Parallax Displays

One reason perspective and orthographic images were selected as input for the interpolation algorithm is that the input data can be rendered with virtually every modelling tool. For *OpenGL* based rendering systems the proposed algorithm illustrates a rendering method that solves the problem of placing the camera centre and the near clipping plane, as described by Halle and Kropp [HK97] and Holzbach and Chen [HC02] without the need of shifting the camera centre or a scene analysis to find a suitable position for the near clipping plane. The orthographic images are rendered with a parallel projection through the virtual display that is placed in the scene and the desired view direction. The near and far clipping planes can be extended beyond the zone used for viewpoint interpolation to avoid clipping of polygons.

The orthoscopic and pseudoscopic parts of the elemental images are rendered as described by [HK97]. The content near the display plane is rendered by the orthographic images, therefore, the near clipping planes can be placed relatively far away from the display plane, allowing for an unmodified camera centre in the display plane and less z-fighting due to a larger distance between the camera centre and the near clipping plane. Again, the near clipping planes can be extended beyond the zone used for viewpoint interpolation to avoid clipping of polygons.

3. Evaluation

The proposed algorithm was evaluated on two artificial scenes. First, the evaluation method consisting of simulated views of a multi-view display is introduced. Afterwards, the path of simulated viewpoints is described, followed by the evaluation of the interpolation on both scenes.

3.1. Simulation of a Multi-View Display

In order to simulate a view from a given viewpoint the rays for every display element to the viewpoint are computed. According to the viewing direction the colour information is looked up in the elemental images of every display element using bi-linear interpolation. Finally the simulated view is constructed from the coloured viewing rays.

The simulated ground truth views of the evaluated scenes show aliasing for content near the display plane (see Fig. 3 (centre), capsules borders and Fig. 3 (right), the colour ornamentation). The aliasing results from the sampling of a high frequency in the scene with the low spatial resolution of the display elements. Near the display plane the relatively large distance between the display elements did not allow for a smooth transition between neighbouring display elements. This kind of aliasing would also be visible on real displays and occur on all spatially sparse AS displays. In general all lens based AS displays reduce spatial resolution [LWH*12] and are therefore prone to aliasing for content near the display plane. It can be remedied by avoiding high frequency in the modelled scene or by a depth dependent low-pass filtering of the elemental images near the display plane.

3.2. Simulation of an Observer

For the evaluation a series of viewpoints in front of the virtual display are generated to simulate a moving observer. The viewer is centred 6 meters in front of the display and the size of each pixel of the display is set to overlap with its neighbours to avoid the background colour in the active display area. This approach doesn't compare the whole data set, only the light rays used in the simulated views. This is justified by the advantage that the evaluation is done on views that are relevant to a potential observer. The path of the observer is shown in Fig. 3 (left) and consists of 315 positions that are used in the remainder to generate all results. The axis of abscissae shows the horizontal deviation from the centre of the display and the axis of ordinate shows the vertical deviation. The first position P_0 is placed centred before the display. The observer then moves to the left hand side, afterwards following a rhombus like path in counter clockwise direction, closing the rhombus at position P_{314}. The observers view is always directed at the centre of the display.

3.3. Evaluation

The *Coffee Capsules* data set shows a couple of coffee capsules, floating in mid-air in front of a uni-coloured background. A simulated view of the display is shown in Fig. 3 (centre). The scene is challenging because of the many small objects that occlude each other and the highlights on the metallic surfaces of the coffee capsules. The display is placed approximately in the centre of the coffee capsules, such that the objects extend about one meter in front and behind the display plane. The data set is rendered for a display with a spatial resolution of 303 x 207 pixels, a distance of two millimetre between the display elements and a resolution of the elemental images of 512 x 512 pixels, allowing for about 200,000 views, distributed over a field-of-view of 40 degree. The total number of display pixels

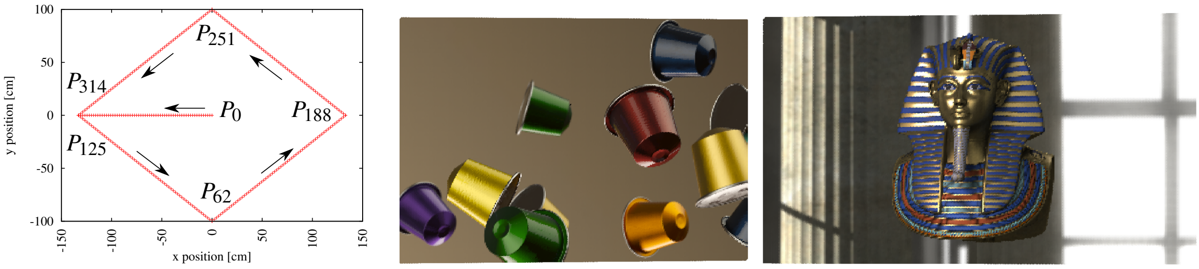

Figure 3: Trajectory of the viewpoint positions (left), simulated ground truth view from position P_{65} of the scene *Coffee Capsules* (centre) and position P_{196} of the scene *Tutankhamun* (right).

is $303 \cdot 207 \cdot 202,963 = 1.27 \cdot 10^{10}$, yielding the number of pixels of the ground truth data set (see Table 1). The proposed mixture of orthographic and elemental images (I_c) is compared to an interpolation on elemental images only (I_p) and an interpolation on orthographic images (I_o) with 0.22% of all pixels used as input (see Table 1). The background has a colour gradient, dependent on the viewing angle. For the orthographic interpolation a ground truth background image was inserted when the elemental images were assembled in order to avoid large interpolation errors on the background.

The second data set is the mask of Tutankhamun, placed in a large hall. The data set is rendered for a display with a spatial resolution of 320 x 180 pixels, a distance of four millimetre between the display elements and a resolution of 512 x 512 pixels for the elemental images. The display plane runs through the centre of the mask displayed in Fig. 3 (right). At the end of the hall is a window, placed about 25 meters behind the display plane. The scene is challenging due to the reflecting surfaces of the ground, the high level of geometric detail on the mask and a high resolution texture on the mask with highlights on the golden ornamentation. In order to avoid large interpolation errors behind the background window frame a white ground truth background image was inserted when the elemental images were assembled for the interpolation on orthographic images. The full interpolated data sets are then used to render all viewpoints of the evaluated path (see Fig. 3, left) and compared against the rendered viewpoints of the ground truth data set. The peak signal to noise ration (*PSNR*) for the interpolation on the different kind of input data is shown in Fig. 4. Table 1 summarises the mean, minimum and maximum PSNR of the interpolated views. With an equivalent number of input pixels the proposed algorithm (I_c) achieved the highest mean PSNR for both scenes.

For the scene *Coffee Capsules* around viewpoint 275 (Fig. 4, left) the interpolation on perspective images (I_p) outperforms the proposed algorithm (I_c). From that viewpoint the objects close to the display plane are occluded by foreground objects. The consequence is that the combined interpolation (I_c) can't benefit from the orthographic input images and interpolates almost exclusively with the interpolation on perspective images (I_p), but with less input images, leading to a lower PSNR. Fig. 5 shows the simulated views of position P_{65} and the negated difference view to the ground truth view (see Fig. 3, centre). One observes that the interpolation errors on orthographic images mainly occur at object borders, due to the low spatial resolution of the orthographic images.

The evaluation of the *Tutankhamun* Scene (Fig. 4, right) shows that for almost the whole evaluated path the proposed approach (I_c) has a considerably higher PSNR then the interpolation on elemental or orthographic images. Fig. 6 shows the simulated views of position P_{196} and the negated difference view to the ground truth view (see Fig. 3). The interpolation on perspective images (I_p) show artefacts on the mask, due to a non-overlapping field-of-view of the input images. The interpolation on orthographic images (I_o) shows severe artefacts on the background of the scene where parts of the window frame and the background wall are missing. Due to the low spatial resolution of the orthographic images (four millimetre per pixel) and the discrete forward mapping spatial warping errors of one pixel are introduced. Besides the spatial warping errors the proposed approach (I_c) does not reveal distinctive artefacts, although, the negated difference image reveals that there are more errors on the background than in the interpolation on perspective images (I_p) and more errors on the foreground object than in the interpolation on orthographic images (I_o), due to a smaller number of input pixels.

The average time for the interpolation of one pixel on an Intel(R) Core(TM) i7 950 with 3.07 GHz are shown in the right column of Table 1. The runtime measurement excludes the load and write operations on the image data. The proposed interpolation (I_c) of the full *Coffee Capsules* data set would therefore require about 58 minutes and about 54 minutes on the *Tutankhamun* data set respectively, when time measurement is restricted to the interpolation. Current AS displays are capable to show up to three orders of magnitude fewer views (about 200), which suggest that porting to a FPGA could achieve real-time. The ratio of the complete set of pixels of the simulated displays against the number of pixels that were used as input for the different interpolation

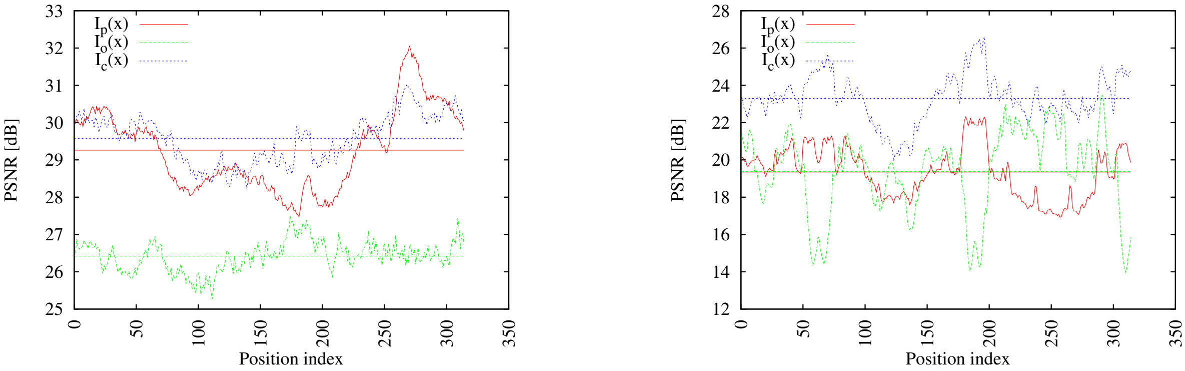

Figure 4: PSNR for I_o (green), I_p (red) and the proposed I_c (blue) for the *Coffee Capsules* data set (left) and the *Tutankhamun* data set (right). The axis of abscissae shows the viewers position according to Fig. 3 (left).

Method	Coffee Capsules					Tutankhamun					Avg. time per pixel [s]·10^{-7}
	# input pixels [%]	Fac.	Mean [dB]	Min. [dB]	Max. [dB]	# input pixels [%]	Fac.	Mean [dB]	Min. [dB]	Max. [dB]	
I_p	0.22	447.4	29.26	27.47	32.06	0.28	352.5	19.35	16.92	22.31	1.61
I_o	0.22	447.4	26.42	25.27	27.50	0.28	352.5	19.37	13.97	23.47	1.14
I_c	0.22	447.4	29.58	28.23	31.03	0.28	352.5	23.30	19.94	26.60	2.75

Table 1: Results of *Coffee Capsules* and *Tutankhamun* scene showing the number of input pixels, the mean, the minimum and the maximum PSNR of the sequence of viewpoints. The average interpolation time for one pixel is shown on the right side.

approaches are shown in Table 1. For the *Coffee Capsules* scene the full colour data of $1.27 \cdot 10^{10}$ pixels of the display was reduced by a factor of about 440. The actual input data of the display was reduced by a factor of 330 when accounting for the input depth maps required by the interpolation algorithm. For the *Tutankhamun* scene the full colour input data of $1.17 \cdot 10^{10}$ pixels was reduced by a factor of 350 and a factor of 260 respectively. For the proposed interpolation I_c in each case 0.05 percent of the input pixels were from orthographic images and the remaining from perspective ones.

4. Conclusion and Future Work

With the growing number of views available on AS displays, transferring the image data to multi-view displays will be a challenge in the near future, due to the massive increase in the required data rate. This work proposed to utilise a depth-image based rendering algorithm to reduce the required data rate by interpolating the full set of images on the device. It has been shown that a DIBR algorithm achieved better interpolation results with a fixed amount of input pixels by combining orthographic and perspective input images in contrast to an interpolation solely on orthographic or perspective images. Another benefit of the combination of perspective and orthographic images is that degenerative cases of DIBR, e.g., objects in the display plane and at infinity, are implicit handled and an upper bound is applied for the disparity, reducing holes introduced by occlusions. For the evaluated data sets the data rate could be reduced by a factor of 260 and 330 respectively. This can be seen as a lossy compression that allows the transfer of large amounts of data to a large scale AS display. A larger compression factor for transmission of the data over a network could be achieved by a lossy or lossless compression of the input data of the proposed algorithm.

The problem of being able to render the input images from a variety of modelling tools was solved by restricting the proposed algorithm to orthographic and perspective images. By combining the interpolation results the proposed algorithm performs well on objects near the display plane as well as objects far away from the display plane, towards infinity.

For the rendering of elemental images with an *OpenGL* based rendering systems an algorithm was proposed that solves the problem of placing the camera centre and the near clipping plane. The benefit of the proposed algorithm is that no scene analysis is required and a reduced probability of z-fighting due to a larger distance between the camera centre and the near clipping plane. The drawback is that for an efficient rendering of elemental images a full frame buffer has to be stored in memory to transfer the orthographic images back into a perspective elemental image.

Future work will be porting the proposed algorithm to a FPGA and investigate the feasible input data rate that can be processed in real-time, in dependence on the final specification of the AS display.

References

[AMZ*06] Annen T., Matusik W., Zwicker M., Pfister H., Seidel H.-P.: Distributed Rendering for Multiview Parallax Displays. In *Proceedings of Stereoscopic Displays and Virtual Reality Systems XIII* (San Jose, USA, 2006), SPIE Press, pp. 231–240. 2

[AOK*10] Arai J., Okano F., Kawakita M., Okui M., Haino Y., Yoshimura M., Furuya M., Sato M.: Integral Three-Dimensional Television Using a 33-Megapixel Imaging System. *Display Technology, Journal of 6*, 10 (oct. 2010), 422–430. 1

[BBM*01] Buehler C., Bosse M., Mcmillan L., Gortler S., Cohen M.: Unstructured Lumigraph Rendering. In *Proceedings of SIGGRAPH 2001* (2001), ACM, pp. 425–432. 2, 3, 4

[BKB07] Balogh T., Kovacs P., Barsi A.: Holovizio 3D Display System. In *3DTV Conference, 2007* (may 2007), pp. 1–4. 2

[FWL*11] Farre M., Wang O., Lang M., Stefanoski N., Hornung A., Smolic A.: Automatic content creation for multiview autostereoscopic displays using image domain warping. In *Proceedings of the 2011 IEEE International Conference on Multimedia and Expo* (Washington, DC, USA, 2011), ICME '11, IEEE Computer Society, pp. 1–6. 2

[HC02] Holzbach M. E., Chen D. T.: Rendering methods for full parallax autostereoscopic displays, United States patent US 6,366,370, April 2002. 1, 2, 5

[HDFP11] Holliman N., Dodgson N., Favalora G., Pockett L.: Three-Dimensional Displays: A Review and Applications Analysis. *Broadcasting, IEEE Transactions on 57*, 2 (june 2011), 362 –371. 1

[HHN*02] Humphreys G., Houston M., Ng R., Frank R., Ahern S., Kirchner P. D., Klosowski J. T.: Chromium: a stream-processing framework for interactive rendering on clusters. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2002), SIGGRAPH '02, ACM, pp. 693–702. 2

[HK97] Halle, Kropp: Fast Computer Graphics Rendering for Full Parallax Spatial Displays. *Proc. SPIE Vol. 3011* (1997), p. 105–112. 2, 3, 5

[HWRH13] Heide F., Wetzstein G., Raskar R., Heidrich W.: Adaptive Image Synthesis for Compressive Displays. *ACM Trans. Graph. (Proc. SIGGRAPH) 32*, 4 (2013), 1–11. 2, 3

[JK11] Jung D., Koch R.: Efficient Rendering of Light Field Images. In *Video Processing and Computational Video*, Cremers D., Magnor M., Oswald M., Zelnik-Manor L., (Eds.), vol. 7082 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2011, pp. 184–211. 1, 2, 3

[LWH*12] Lanman D., Wetzstein G., Hirsch M., Heidrich W., Raskar R.: Beyond parallax barriers: applying formal optimization methods to multilayer automultiscopic displays. 82880A–82880A–13. 5

[Mis11] Mishina T.: Three-dimensional television system based on integral photography. In *Visual Communications and Image Processing (VCIP), 2011 IEEE* (nov. 2011), pp. 1–4. 1

[MMSW06] Merkle P., Muller K., Smolic A., Wiegand T.: Efficient Compression of Multi-View Video Exploiting Inter-View Dependencies Based on H.264/MPEG4-AVC. In *Multimedia and Expo, 2006 IEEE International Conference on* (july 2006), pp. 1717–1720. 2

[MP04] Matusik W., Pfister H.: 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. In *ACM SIGGRAPH 2004 Papers* (New York, NY, USA, 2004), SIGGRAPH '04, ACM, pp. 814–824. 2

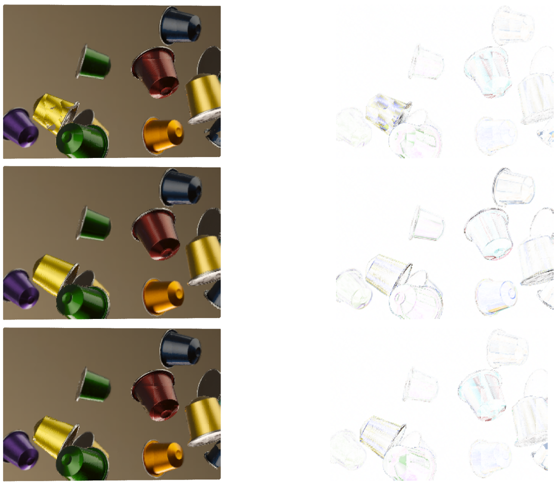

Figure 5: Results *Coffee Capsules* from P_{65}.

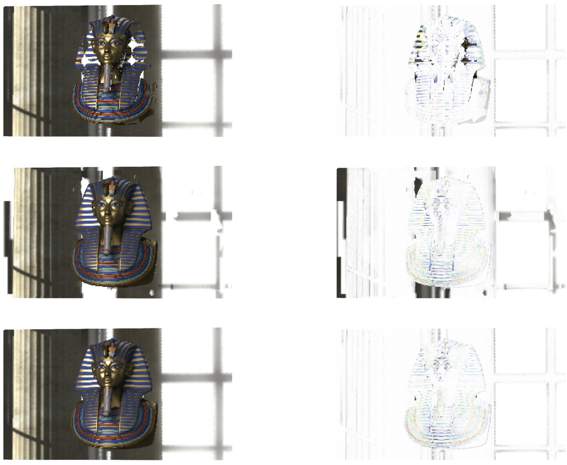

Figure 6: Results *Tutankhamun* from P_{196}. In Fig. 5 and 6 I_p is on top, I_o is centred and the proposed I_c is placed at the bottom. Left are the interpolated views, right are the negated difference to the ground truth (see Fig. 3, centre, right).

[MRG03] Magnor M., Ramanathan P., Girod B.: Multi-view coding for image-based rendering using 3-D scene geometry. *Circuits and Systems for Video Technology, IEEE Transactions on 13*, 11 (nov. 2003), 1092–1106. 2

[SGHS98] Shade J., Gortler S., He L.-w., Szeliski R.: Layered depth images. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1998), SIGGRAPH '98, ACM, pp. 231–242. 2

[XPLL11] Xu X., Pan Y., Lwin P. P. M. Y., Liang X.: 3D holographic display and its data transmission requirement. In *Information Photonics and Optical Communications (IPOC), 2011 International Conference on* (oct. 2011), pp. 1–4. 2

[YKO10] Yaraş F., Kang H., Onural L.: State of the Art in Holographic Displays: A Survey. *Display Technology, Journal of 6*, 10 (oct. 2010), 443–454. 1