Real-time adaptive content retargeting for live multi-view capture and light field display

Vamsi Kiran Adhikarla · Fabio Marton · Tibor Balogh · Enrico Gobbetti

Abstract The discrete nature of multiprojector light field displays results in aliasing when rendering scene points at depths outside the supported depth of field causing visual discomfort. We propose an efficient on-the-fly content-aware real-time depth retargeting algorithm for live 3D light field video to increase the quality of visual perception on a cluster-driven multiprojector light field display. The proposed algorithm is embedded in an end-to-end real-time system capable of capturing and reconstructing light field from multiple calibrated cameras on a full horizontal parallax light field display. By automatically detecting salient regions of a scene, we solve an optimization to derive a non-linear operator to fit the whole scene within the comfortable viewing range of the light field display. We *evaluate* the effectiveness of our approach on synthetic and real world scenes.

Keywords On-the-fly depth retargeting · GPU · Multiprojector light field display · Visually enhanced live 3D Video · Multi-view capture and display

1 Introduction

The main principle behind any glasses-free 3D display technology is direction-dependent light transmission. Ideally, to completely describe a scene in 3D, a display should reproduce light rays from all the points in all the directions within a given viewing zone. In practice, however, this is not possible due to the finite size of

Vamsi Kiran Adhikarla
Holografika & Pázmány Péter Catholic University, Hungary

Fabio Marton, Enrico Gobbetti
CRS4, Italy

Tibor Balogh
Holografika, Hungary

display hardware. Recent advances in computational displays showed several improvements in various dimensions such as color, luminance & contrast, spatial and angular resolution (see [17] for a detailed survey of these displays). Projection-based light-field displays, are among the most advanced solutions. Taking inspiration from the real-world, a light field display emits light rays from multiple perspectives using a set of optical modules. The various emitted light rays hit a holographic screen which performs the necessary optical modulation for reconstructing a 3D scene. Various scene points are described by intersecting light rays at corresponding depths. Even though these displays provide continuous views and improve over traditional automultiscopic solutions, the extent of practically displayable depth with reasonable 3D quality is still limited due to finite number of light generating optical modules. Scene points rendered outside this range are subjected to poor sampling and suffer from aliasing, which typically lead to excessive blurring in regions. This blurring makes it difficult to perceive details of very far objects from the screen, and leads to visual discomfort.

By matching the depth extent of scene to that of display by applying a process of *depth retargeting*, it is possible to greatly reduce the blurring artifacts, achieving all-in-focus rendering. An important consideration while retargeting the light field depth is that any depth compression results in flattening of objects and distorting the 3D structure of the scene. Thus, in order to provide compelling results, depth compression must be non-linear and content-adaptive. In the current work, we address this problem by proposing a low-complexity real-time solution to adaptively map the scene depth to display depth by taking into account the perspective effects of a light field display and the saliency of the scene contents. The proposed retargeting module, which

strives to reduce distortions in salient areas, is integrated into a real-time light field rendering pipeline that can be fed with a live multi-view video stream captured from multiple cameras.

Our approach. We propose an architecture coupling the geometry estimation and retargeting processes to achieve the real-time performance. While rendering the light field, our renderer estimates input scene geometry as seen from the positions of various display optical modules, using only multiview color input. The estimated depth is used for focusing the light field and is the basis for adaptive depth retargeting. In order to compute an optimal scene deformation, we formulate and solve a convex optimization problem by discretizing the depth range into regions, and using saliency information from the scene to preserve the 3D appearance of salient regions of the scene in retargeted space. We compute the scene saliency by analyzing the objects distribution in the scene depth space and weighting this distribution with appropriate texture gradient magnitudes. During retargeting, scene points are subjected to a perspective transformation using the computed non-linear mapping which changes depths and accordingly scales x-y positions. The quality and performance of our approach is demonstrated in an end-to-end system for real-time capture and all-in-focus display that achieves real-time performance using 18 cameras and 72 projection modules.

Contribution. Our improvements with respect to the state-of-the-art are the following:

- A perspective depth contraction method for live light field video stream that preserves the 3D appearance of salient regions of a scene. The deformation is globally monotonic in depth, and avoids depth inversion problems.
- A real-time plane sweeping algorithm which concurrently estimates and retargets scene depth. The method can be used for all-in-focus rendering of light field displays.
- An end-to-end system capable of real-time capturing and displaying with full horizontal parallax high-quality 3D video contents on a cluster-driven multiprojector light field display with full horizontal parallax.
- An evaluation of the objective quality of the proposed depth retargeting method.

Advantages and limitations. The proposed method for depth retargeting is content-adaptive and computationally light. It is general enough to be employed both for 3D graphics rendering on light field display and for real-time capture-and-display applications. The content-adaptive nature of the method makes it possible to employ a number of different measures to determine which depth intervals should be preserved most. We currently do not attempt to provide a model of the behavior of the human visual system to drive the optimization, and rather use a simple saliency estimator based on geometry and image gradients. The approach is general enough, however, to replace saliency estimation with more elaborate and domain-specific modules (e.g., face recognition in 3D video-conferencing applications).

2 Related work

Our end-to-end system enhances and integrates several state-of-the-art solutions for 3D video capture and rendering in wide technical areas. For comprehensive understanding, we refer the reader to established surveys (e.g., [12, 19]). In the subsequent paragraphs, we present some of the more relevant works.

Light field capture and display. A number of papers showing considerable advances in the areas of real-time 3D video or light field capture and display have been published in the recent years. Most of the approaches are based on pure light field conception and considers the sets of rays captured by the cameras as light field samples. During rendering, captured light field database is re-sampled to produce light rays from a required point of view [3, 18, 22]. These system do not take scene geometry in to account and thus, in accordance with the plenoptic sampling theory [5], for photo-realistic rendering, one may require very high number of cameras to substantially sample the light field. Estimating the scene geometry helps in producing higher quality views from arbitrary view positions using less cameras [13, 20, 21, 23]. We follow here the real-time approach of Marton et al. [15], which takes into account light field display characteristics in terms of both geometry and resolution of the reproduced light fields. In particular, they extend a multiple-center-of-projection technique [1, 8, 9] to map captured images to display space, and estimate depth to focus the light-field using a coarse-to-fine space-sweeping algorithm. We extend here their approach to embed a saliency aware depth retargeting step during depth evaluation to properly place the scene in the correct display range thus avoiding aliasing artifacts while maintaining correct depth for salient scene regions.

Adaptive depth retargeting. Content remapping is a well established approach for adapting image characteristics to limited displays, and is routinely used for adapting spatial and temporal resolution, contrast, colors, and aspect ratios of images. For the particular case of depth retargeting, Lang et al. [14] proposed a method for remapping stereoscopic 3D disparities us-

ing a non linear operator. The non-linear mapping is generated by sparse disparity estimation and combining the local edge and global texture saliency. The method is based on warping the stereo images independently to achieve depth retargeting. As the method relies on sparse disparities, warping can lead to noticeable artifacts especially near the depth discontinuities and may also distort any straight lines in the scene. Extending this method to remap full-parallax light field content would introduce artifacts because of the increased number of views. Kim et al. [10] extend the approach by proposing a framework for the generation of stereoscopic image pairs with per-pixel control over disparity, based on multi-perspective imaging from light fields. While their method might be extended to multiview images, the associated optimization problem is too costly to be solved in a run-time setting. Masia et al. [16] deal specifically with multiview displays by proposing a method for display-adaptive depth retargeting. They exploit the central view of a light field display to generate a mapping function and use warping to synthesize the rest of the light field. Their work strives to minimize perceived distortion using a model of human perception, but does not achieve real-time performance and does not include depth estimation. Piotr Didyk et al. [6] proposed a model for measuring perceived disparity and a way to automatically detecting the threshold for comfortable viewing. Their method can be used as a component to operate depth retargeting. We concentrate instead on overcoming device limitations. Birkbauer et al. [4] handle the more general problem of light field retargeting, using a seam-carving approach. The method supports visualizing on displays with aspect ratios that differ from those of the recording cameras, but does not achieve real-time performance. Content-aware remapping has also been proposed to achieve non-linear rescaling of complex 3D models, e.g. to place them in new scenes. The grid-based approach of Kraevoy et al. [11] has also been employed for image retargeting. Graf et al. [7] proposed an interesting approach for axis-aligned content aware 2D image retargeting, optimized for mobile devices. They rely on the image saliency information to derive an operator that non-linearly scales and crops insignificant regions of the image using a 2D mesh. Our method also takes the approach of using a discretized grid to quickly solve an optimization problem. In our case, we use a one-dimensional discretization of depth, which permits us to avoid depth inversion problems of solutions based on spatial grids.
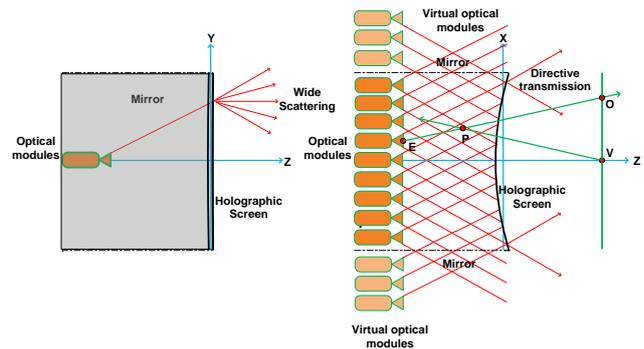

Fig. 1 Light field display model and optical characteristics. The display hardware setup consists of three parts: spatially arranged optical modules, a curved (cylindrical section) holographic screen and a pair of mirrors along display side walls. Left: vertically, the screen scatters widely and users perceive the same light field from any height. Right: horizontally, the screen is sharply transmissive with minimum aliasing between successive viewing angles.

3 Retargeting model

Given a 3D scene our aim is to produce a retargeted scene which meets the given depth constraints of a light field display.

Light field display. Projection based light field displays have three main components: spatially arranged optical modules, a curved (cylindrical section) holographic screen and a pair of mirrors along display side walls (see Fig. 1 left). The main purpose of the side mirrors is to bounce the light rays that are incident on the walls onto the holographic screen and thus they help in increasing the effective field of view of the display. Optical modules are arranged densely to yield sufficient angular resolution. The display screen is holographically recorded and has randomized surface relief structure that provides controlled angular light divergence. The optical properties of the screen enable sharp directive transmission along horizontal direction and allow us to achieve sub-degree angular resolution, creating continuous-like horizontal parallax. For rendering, we use a multiple-center-of-projection (MCOP) technique [1,9] that helps in preserving the motion parallax cue. We assume a right handed coordinate system for the display with screen center as origin of the coordinate space. Our perspective rendering approach assumes a viewing line in front of the display at fixed height and depth. Given an observer at $\mathbf{V}$, the ray origin passing through a point $\mathbf{P}$ is determined by $\mathbf{O} = (E_x + \frac{P_x - E_x}{P_z - E_z}(V_z - E_z), V_y, V_z)$, where $\mathbf{E}$ is the position of the optical module under consideration (see Fig. 1 right). The ray connecting $\mathbf{O}$ to $\mathbf{P}$ is used as projection direction in order to transform the model into normalized projection coordinates. 3D positions corresponding to the view port pixels of various optical

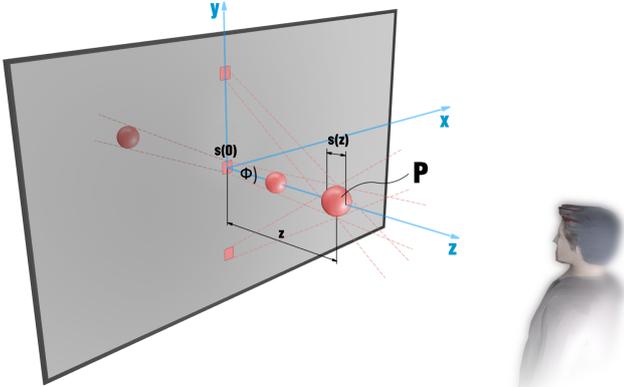modules are determined using the display geometry calibration information [1]. Using precise modeling and

Fig. 2 Varying spatial resolution of a light field display. The light field display's spatial resolution is depth dependent. The size of smallest displayable feature increases with distance from the screen. Thus, objects rendered on the screen surface appear sharper.

calibration procedures, light beams leaving from different positions and directions can be shaded to construct a 3D light field slice. One of the contributing factors to the degradation of visual quality in general 3D display setups is the varying spatial resolution with depth. In our case, the spatial resolution of the display changes with depth, according to:

$$s(z) = s_0 + 2\|z\| \tan(\frac{\Phi}{2}) \tag{1}$$

where z is the distance to the screen, and s_0 is the pixel size on the screen surface [1] (see Fig. 2). Thus, the spatial resolution is higher on the surface of the screen and the diminishes as we move away from screen. Therefore, to optimize the viewing experience on light field displays, the scene center must coincide with display's $Z = 0$ plane, total scene depth should comply with the limited depth of field of the display and the frequency details of the objects in the scene should be adjusted conforming to the displays spatial characteristics. While the latter can be addressed by adapting suitable rendering methods, our depth retargeting method focuses on addressing the former two goals.

Retargeting. If a 3D scene and display have the same depth extent no retargeting is required, but in a more general case, a depth remapping step is needed. Our aim is to generate an adaptive non-linear transform from scene to display that minimizes the compression of salient regions. To extract the scene saliency, we compute depth and color from perspectives of multiple display projection modules and combine this information. To make the process faster, we compute the saliency from

central and two lateral perspectives and use this information to retarget the light field from all the viewing angles. Depth saliency is estimated using a histogram of the pre-computed depth map (please refer to section 4 for details on how depths are computed). More specifically, we sweep the whole scene range, starting from camera near plane to far plane along originally defined steps from the three perspectives and collect the number of scene points located in each step. We then combine this information for extracting depth saliency. To estimate color saliency, we compute a gradient map of the color image associated to the depth map of the current view and dilate it to fill holes, as done in [7]. The gradient norm of a pixel represents color saliency. To avoid any abrupt depth changes we quantize the scene depth range into different depth clusters and accumulate the depth and color saliency inside each cluster. In real-world, objects far away from the observer appear flatter than the closer objects and thus impact of depth compression on closer objects is more than that of far objects. Taking into account this phenomena, we also apply weight to the saliency of each cluster based on the relative position from the observer. Using the length of a cluster and it's saliency, we solve a convex optimization to derive the retargeting function. We formulate the generation of
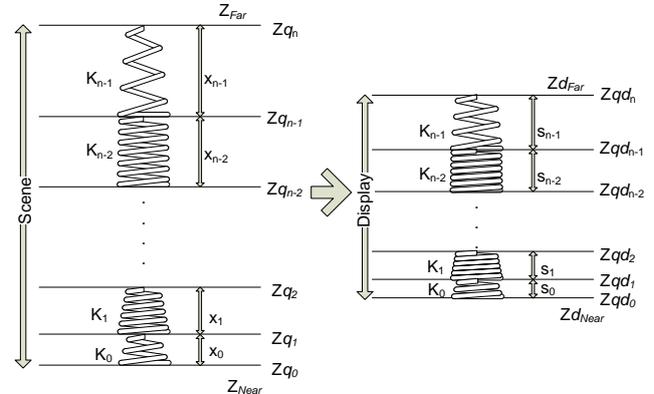

Fig. 3 Computation of content aware depth retargeting function. Scene depth space is quantized into n clusters and saliency is computed inside each. Cluster sizes in the retargeted display space is computed by solving a convex optimization. Z_{qi} and Z_{qdi} denote i^{th} quantized depth level in scene and display spaces respectively.

retargeted light field as a quadratic optimization program. Let's assume that the scene range is quantized into n clusters and a spring is assigned to each cluster as shown in Fig. 3 left. We denote the length and stiffness of a spring as the size and saliency of the representing cluster. Assuming that we compressed the n spring set within scene range to display range as shown in Fig. 3, the resulting constrained springs define the desired

new clusters in the display range which preserve the salient objects. To estimate the size of each constrained cluster, we define an energy function proportional to the difference between the potential energies of original and compressed spring. By minimizing this energy function summed over all the springs, we obtain the quantized spring lengths within the display space. Following the optimization of Graf et al. [7] for 2D image retargeting and adapting it to one-dimensional depth retargeting, our aim is to minimize:

$$\sum_{i=0}^{qn-1} \frac{1}{2} K_i \left(S_i - X_i \right)^2 \tag{2}$$

subject to:

$$\sum_{i=0}^{qn-1} S_i = D_d$$

$S_i > D_{cs}{}^{min}, i = 0, 1, ..., n-1$ Where, X_i and K_i are the length and stiffness of ith cluster spring, D_d is the total depth of field of the display and $D_{cs}{}^{min}$ are the minimum and allowable sizes of the resulting display space clusters. By expanding and eliminating the constant terms that do not contribute to the optimization, equation 2 can be re-written in the form of Quadratic Programming (QP) optimization problem as follows.

$$\frac{1}{2} \left(x^T G x + g_0 x \right) \tag{3}$$

subject to: $CE^T x + Ce_0 = 0$ & $CI^T x + Ci_0 \geq 0$
Where

$$G = \begin{pmatrix} K_0 & 0 & \cdots & 0 \\ 0 & K_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_{n-1} \end{pmatrix}$$

$$g_0 = [-2K_0 X_0 - 2K_1 X_1 \cdots - 2K_{n-1} x_{n-1}]$$

$$x = \begin{pmatrix} S_0 \\ S_1 \\ \vdots \\ S_{n-1} \end{pmatrix}; CE = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}; Ci_0 = \begin{pmatrix} -D_{cs}{}^{min} \\ -D_{cs}{}^{min} \\ \vdots \\ -D_{cs}{}^{min} \end{pmatrix}$$

$$CI = I_{(nxn)}; Ce_0 = -D_d$$

CE and Ci_0 are $(nX1)$ vectors.

For each point in the scene, we compute a new point in the display $z_{display} = f(z_{scene})$ using piecewise linear interpolation. It is important to note that while adapting the scene to display, displacement of depth planes

parallel to $XY = 0$ plane results in XY cropping of the scene background. Thus, in order to preserve the scene structure, we follow a perspective retargeting approach, i.e., along with z we also update XY position proportional to $\frac{1}{\delta Z}$, as it is done in a perspective projection (see Fig. 4). Thus in the retargeted space, the physical size of the background objects is less than the actual size. However, a user looking from the central viewing position perceives no change in the apparent size of the objects as the scene points are adjusted in the direction of viewing rays.
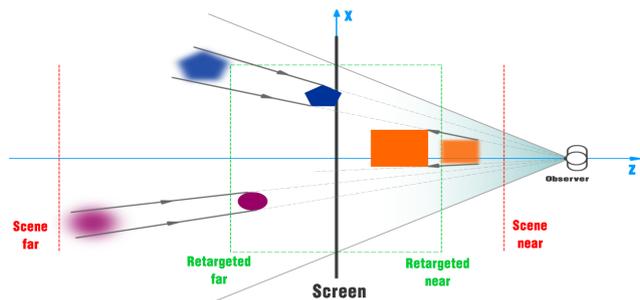

Fig. 4 Perspective Content adaptive retargeting. Objects are replaced within the display space with minimum distortion, trying to compress empty or not important depth ranges. Objects are not simply moved along z, but the xy coordinates are modified by a quantity proportional to $\frac{1}{\delta Z}$.

4 End-to-end capture and display system

Our retargeting method is simple and efficient enough to be incorporated in a demanding real-time application. While this use is straightforward in a 3D graphics setting, where retargeting can be implemented by direct geometric deformation of the rendered models, in this section, we introduce the first real-time multiview capture and light field display rendering system incorporating the adaptive depth retargeting method. The system seeks to obtain a video stream as a sequence of multiview images and render an all-in-focus retargeted light field in real-time on a full horizontal light field display. The input multiview video data is acquired from a calibrated camera rig made of several identical off the shelf USB cameras. The baseline length of the camera array is sufficiently chosen to meet the display FOV requirements. The captured multiview data is sent to a cluster of computers which drive the display optical modules. Each node in the cluster drives more than one optical module. Using the display geometry and input camera calibration data, each node estimates depth and color for corresponding light rays. As mentioned before, to maintain the real-time performance, we combine the
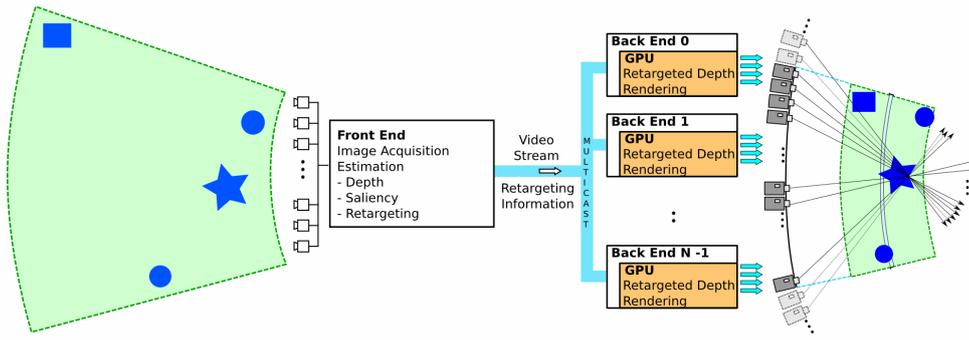
Fig. 5 End-to-end system overview. The front-end performs capture and adaptively computes retargeting parameters, while the back-end performs all-in-focus rendering

depth estimation and retargeting processes. The overall system architecture is shown in Fig. 5. The implementation details are elaborated in the following paragraphs.

Front end. The front end consists of a master PC and the capturing cameras. The master PC acquires video data from multiple software-synchronized cameras and streams it to several light field clients. The data is acquired in JPEG format with VGA resolution (640 X 480) at 15Hz over a USB 2.0 connection. At a given time stamp, the several captured multiview images are packed into a single multiview frame and sent to the backend over a Gigabit Ethernet connection. A reliable UDP multicast protocol is incorporated to distribute the data in parallel to all the clients. Apart from multiview capture and streaming, the master PC also runs in parallel, the rendering application instances for the display central and two lateral optical modules to compute the required light field retargeting function. This function describes the mapping between a set of quantized scene depth plane positions and their corresponding constrained and optimized positions in the retargeted display space. While scene depth plane positions are computed independently by all the clients, the quantized retargeted display plane positions are sent as metadata along with the current multiview frame to the backend.

Back end. The back end constitutes the rendering cluster and light field display. All the clients in the rendering cluster work independent of each other and produce a set of optical module images. Each client decodes the received multiview images and uploads the RGB channel data as a 3D array to the GPU. For a given display projection module, we compute the depth information for various viewport pixels extending the space sweeping approach of [15] to perform simultaneous estimation and retargeting. The method follows a coarse to fine approach. For each of the camera textures, we pre-compute a Gaussian RGBA pyramid, constructed with a 2D separable convolution of a filter of width 5 and factor of two sub-sampling. In parallel, we also generate and store a descriptor pyramid for pixels of

each level which will be used for depth computations. The descriptors are defined following the census representation [24]. Depth values are calculated iteratively by up-scaling the current optical module viewport from coarse to fine resolution with each iteration followed by median and min filtering to remove high frequency noise. To estimate depth for a given display light ray, space sweeping is performed in display space using a coarse-to-fine stereo-matching method described in [15]. Note that during stereo matching, the matching costs should be evaluated in the camera space where original scene points are located. Thus, while computing the matching cost at a particular depth level, we perspectively transform the candidate point position in display space to camera space using the inverse retargeting function. As mentioned earlier, a light ray may not be emitted by the same display optical module before and after retargeting. Thus, the scene ray corresponding to current display ray can have a different direction, which makes it necessary to re-compute the closest camera set (for matching cost evaluation) at every depth step of space sweeping. For a display-to-scene mapped voxel, ray direction is computed by finite differences, *i.e.*, at a given display depth step, we transform another display voxel position at a consecutive depth step to camera space using the same inverse retargeting function. The required ray direction in scene space at current display depth step is along the ray joining the two transformed points in camera space. Using this direction, we compute a set of four closest cameras over which the matching cost is evaluated and summed. The display depth step with best matching cost will be chosen as input candidate for the next iteration in the coarse-to-fine stereo matching method. As described in [15], matching cost is a function of two factors: the luminance difference that helps in tracking local depth variations and hamming distance between the census descriptors which helps in tracking texture areas and depth boundaries. After pre-defined number of iterations, we will have the depth map computed at finest resolution for all light rays of a display optical module. We then use this computed
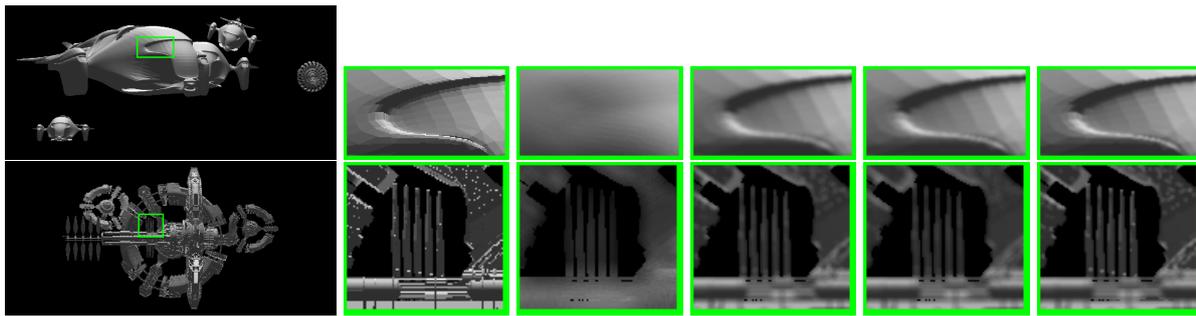
Fig. 6 Original and retargeted simulation results. Top row: Sungliders scene. Bottom row: Zenith scene. Left to right: ground truth central view and close-ups: ground truth, without retargeting, with linear, logarithmic and adaptive retargeting. Note that, as we present the content from display center viewing position, viewport content is not distorted in X-Y.
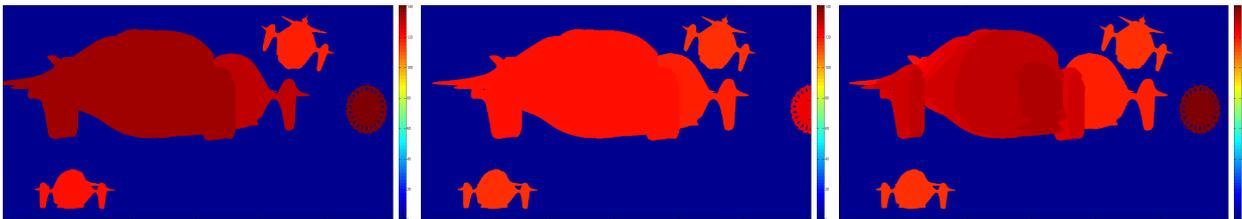

Fig. 7 Simulated retargeted display side view depth maps of the sequence - Sun gliders. Left: linear retargeting, middle: logarithmic retargeting and right: adaptive retargeting. The depth variations are better preserved for adaptive retargeting, thus producing increased parallax effect on light field display.

Fig. 8 Simulated retargeted display side views of the sequence - Zenith. Left to right - left view from linear retargeting, right view from linear retargeting, left view from logarithmic retargeting, right view from logarithmic retargeting, left view from adaptive retargeting and right view from adaptive retargeting. Observe better parallax effect of adaptive retargeting due to improved depth preservation of 3D objects.

depth information to calculate the color to be emitted from individual view port pixels of a given display optical module. Specifically, for a display light ray under consideration, we calculate a position in display space and along the display ray that falls at the computed depth level and transform this position to camera space using the inverse retargeting function. The final color for the display light ray is weighted average of the colors sampled at the transformed position from the four nearest cameras.

5 Results

We implemented the end-to-end capture and display pipeline in Linux. On-the-fly light field retargeting and rendering is implemented on GPU using CUDA. We tested the results of our content aware retargeting on a Holografika 72in light field display that supports 50°

horizontal Field Of View (FOV) with an angular resolution of 0.8°. The aspect ratio of the display is 16:9 with single view 2D-equivalent resolution of 1066×600 pixels. The display has 72 SVGA 800x600 LED projection modules which are pre-calibrated using an automatic multiprojector calibration procedure [2]. The front end is an Intel Corei7 PC with an Nvidia GTX680 4GB, which captures multiview images at 15 fps in VGA resolution using 18 calibrated Logitech Portable Web cameras. The camera rig covers a base-line of about 1.5m and is sufficient to cover the FOV of light field display. In the back end, we have 18 AMD Dual Core Athlon 64 X2 5000+ PCs running Linux and each equipped with two Nvidia GTX560 1 GB graphics boards. Each node renders images for four optical modules. Front-end and back-end communicate over a Gigabit Ethernet connection. In the following sub-sections, we present retargeting results using synthetic and real world light field content.

Retargeting synthetic light field content. Synthetic scenes are employed to evaluate our results and compare them with alternative approaches. As we aim to retarget the light field content in real-time, we limit objective quality evaluation of our method with ground truth and other real-time methods (in particualr, linear and logarithmic remapping [14]). The two synthetic scenes are Sungliders and Zenith. The ground truth central view and close-ups from the central views generated without retargeting, with linear, logarithmic and content adaptive retageting are shown in Fig. 6. The original depths of the scenes are 10.2m and 7.7m, that we remapped to a depth of 1m to match the depth range of our display. Similarly to Masia et al. [16], we generate the images by simulating the display behavior, as given by equation 1 and our display parameters. Figure 6 shows the simulation results: ground truth central view and close-ups from the central views generated without retargeting, with linear, logarithmic and content adaptive retageting. To generate the results for logarithmic retargeting, we use a function of the form $y = a + b * log(c + x)$, where y and x are the output and input depths. The parameters a, b & c are chosen to map the near and far clipping planes of the scene to the comfortable viewing limits of the display. When the original scene is presented on the display, voxels that are very close to the user appear more blurry. Note that in all the three retargeting methods, after retargeting, the rendered scene is less blurry. The adaptive approach better preserves the object depths, avoiding to flatten them. This is more evident for frontal objects between the screen and display near plane, which are almost flattened by the linear and logarithmic approaches and the blurry effect is still perceivable. We can see it from insets of Fig. 6, where near objects drawn with linear and logarithmic retargeting are less sharper than corresponding adaptive retargeted objects. Table 1 shows SSIM and RMSE values of various renderings from the two experimental sequences when compared to ground truth. The metrics show that our content adaptive retargeting performs better than linear, logarithmic and no retargeting. The flattening

Table 1 Central view SSIM and RMSE values obtained by comparison with ground truth image for Sungliders (S) and Zenith (Z) data sets. SSIM=1 means no difference to the original, RMSE=0 means no difference to the original

	Without	Linear	Logarithmic	Adaptive
SSIM-S	0.9362	0.9733	0.9739	0.9778
SSIM-Z	0.8920	0.9245	0.9186	0.9290
RMSE-S	3.6118	2.0814	2.0964	1.9723
RMSE-Z	3.6700	2.8132	2.8910	2.7882

of objects in case of linear and logarithmic retargeting

is clearly perceivable as we move away from central viewing position. Fig. 7 presents the color coded side view depth maps from the scene Sungliders for the three test cases. The global compression in linear retargeting results in the loss of depth resolution in the retargeted space. The non-linear logarithmic mapping leads large depth errors unless the objects are located very close to the display near plane. Adaptive retargeting approach produces continuous and better depth variations and thus preserves the 3D shape of objects. The flattening of objects manifests in the form of reduced motion parallax as shown in Fig. 8. The performance of our method can be better explained from the original and retargeted depth histograms (see Fig. 9). In Fig. 9, red lines represent the screen plane and the two green lines before and after a red line correspond to negative and positive comfortably displayable depth limits of the light field display. Linear retargeting compresses the depth space occupied by scene objects and the empty spaces in the same way, logarithmic retargeting is highly dependent on object positions and results in large depth errors after retargeting. In contrast, our approach best preserves the depth space occupied by objects and instead, compresses the less significant regions, thus maintains the 3D appearance of objects in the scene.
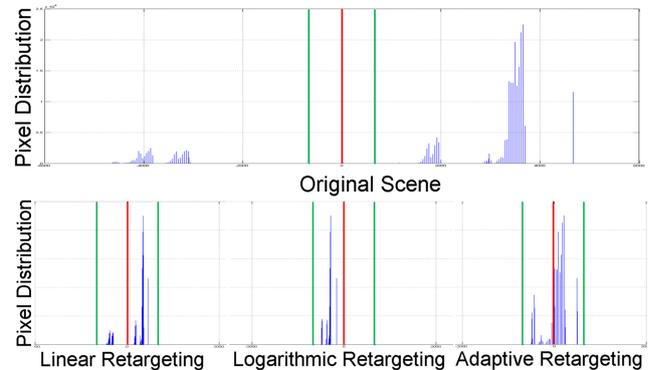

Fig. 9 Sunglider: linear, logarithmic and adaptive retargeting behavior explained using depth histograms. Top row: Original scene, bottom row : left to right - retargeted scene using linear, logarithmic and adaptive retargeting.

Retargeting live multiview feeds. To demonstrate the results of our method on real-world scenes, using a simple hand-held camera, we recorded the processes of live multiview capturing and real-time retargeted rendering(see the accompanying video). It should be noted that the 3D impression of our results on the light field display can not be fully captured by a physical camera. In Fig. 10, we present the screen shots of the light field display with various renderings at a single time instance of a multiview footage. For fair comparison, images

Fig. 10 Real-time light-field capture and retargeting. Our system achieves real-time capture and display (see accompanying video). This image illustrates the effect of retargeting methods on a frozen frame. From left to right: without retargeting, with linear retargeting, with adaptive retargeting. Without retargeting, everything far from the display is blurred. With linear retargeting, blur is reduced, but all objects are squeezed similarly, reducing motion parallax. With adaptive retargeting, empty space is deformed more than objects, which better preserve their 3D shape (e.g., the boxes in the foreground).

are captured from the same point of view to show the perceivable differences between plain rendering, linear retargeting and adaptive retargeting. Our experiments show that the results from the real-world scenes conform with our simulation results on the synthetic scenes. By following direct all-in-focus light field rendering, areas of the scene outside the displayable range are subjected to blurring. Linear retargeting achieves sharp light field rendering at the cost of flattened scene. Content aware depth retargeting is capable of achieving sharp light field rendering and also preserves the 3D appearance of the objects at the same time. The front end frame rate is limited at of 15fps by the camera acquisition speed. The back end hardware used in the current work supports an average frame rate of 11fps. However, our experiments showed that Nvidia GTX680 GPU is able to support 40fps. In the back end application the GPU workload is subdivided in this way: 30% to upsample depth values, 20% for census computation, 15% jpeg decoding, 13% to extract color from the depth map, other minor kernels occupy the remaining time. Retargeting is embedded in the upsampling and color extraction procedures.

6 Conclusions and future work

We presented a working method to perceptually enhance the quality of rendering on a projection-based light field display in a real-time capture and rendering framework. In particular, we addressed the problem of rendering scenes with greater depth than the tolerable depth of a light field display. To the best of our knowledge, we propose the first real-time setup that reconstructs an adaptively retargeted light field on a light field display from a live multiview feed. The method is very general and is applicable to 3D graphics rendering on light field display as well as to real-time capture-and-display applications. We showed that adaptive retargeting preserves the 3D aspects of salient objects in the scenes and achieves better results from all the viewing positions than linear and logarithmic approaches. One of the limitations of our end-to-end system is the inaccuracy of estimated depth values while retargeting and rendering. In future work, we plan to employ additional active sensors to get an initial depth structure of the scene, use human visual system aware saliency estimation, and conduct user studies to subjectively evaluate the performance of the method.

References

1. Agus, M., Gobbetti, E., Iglesias Guitián, J.A., Marton, F., Pintore, G.: GPU accelerated direct volume rendering on an interactive light field display. Computer Graphics Forum **27**(2), 231–240 (2008)
2. Agus, M., Gobbetti, E., Jaspe Villanueva, A., Pintore, G., Pintus, R.: Automatic geometric calibration of projector-based light-field displays. In: Proc. Eurovis Short Papers, pp. 1–5 (2013)
3. Balogh, T., Kovács, P.: Real-time 3D light field transmission. In: SPIE, vol. 7724, p. 5 (2010)
4. Birklbauer, C., Bimber, O.: Light-field retargeting. Computer Graphics Forum **31**(2pt1), 295–303 (2012)
5. Chai, J.X., Tong, X., Chan, S.C., Shum, H.Y.: Plenoptic sampling. In: Proc. SIGGRAPH, pp. 307–318 (2000)
6. Didyk, P., Ritschel, T., Eisemann, E., Myszkowski, K., Seidel, H.P.: A perceptual model for disparity. ACM Trans. Graph. **30**(4) (2011)
7. Graf, D., Panozzo, D., Sorkine-Hornung, O.: Mobile image retargeting. In: Proc. VMV (2013)
8. Iglesias Guitián, J., Gobbetti, E., Marton, F.: View-dependent exploration of massive volumetric models on large scale light field displays. The Visual Computer **26**(6–8), 1037–1047 (2010)
9. Jones, A., McDowall, I., Yamada, H., Bolas, M.T., Debevec, P.E.: Rendering for an interactive 360 degree light field display. ACM Trans. Graph **26**(3), 40 (2007)
10. Kim, C., Hornung, A., Heinzle, S., Matusik, W., Gross, M.: Multi-perspective stereoscopy from light fields. ACM Trans. Graph. **30**(6), 190:1–190:10 (2011)

11. Kraevoy, V., Sheffer, A., Shamir, A., Cohen-Or, D.: Non-homogeneous resizing of complex models **27**(5), 111 (2008)
12. Kubota, A., Smolic, A., Magnor, M., Tanimoto, M., Chen, T., Zhang, C.: Multiview imaging and 3DTV. IEEE Signal Processing Magazine **24**(6), 10 (2007)
13. Kunita, Y., Ueno, M., Tanaka, K.: Layered probability maps: basic framework and prototype system. In: Proc. VRST, pp. 181–188 (2006)
14. Lang, M., Hornung, A., Wang, O., Poulakos, S., Smolic, A., Gross, M.: Nonlinear disparity mapping for stereoscopic 3d. ACM Trans. Graph. **29**(4), 75:1–75:10 (2010)
15. Marton, F., Gobbetti, E., Bettio, F., Iglesias Guitián, J., Pintus, R.: A real-time coarse-to-fine multiview capture system for all-in-focus rendering on a light-field display. In: Proc. 3DTV-CON, pp. 1–4 (2011)
16. Masia, B., Wetzstein, G., Aliaga, C., Raskar, R., Gutierrez, D.: Display adaptive 3d content remapping. Comput. Graph. **37**(8), 983–996 (2013)
17. Masia, B., Wetzstein, G., Didyk, P., Gutierrez, D.: A survey on computational displays: Pushing the boundaries of optics, computation, and perception. Computers & Graphics **37**(8), 1012 – 1038 (2013)
18. Matusik, W., Pfister, H.: 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. ACM Trans. Graph. **23**(3), 814–824 (2004)
19. Smolic, A.: 3D video and free viewpoint video – from capture to display. Pattern Recognition (2010)
20. Taguchi, Y., Koike, T., Takahashi, K., Naemura, T.: TransCAIP: A live 3D TV system using a camera array and an integral photography display with interactive control of viewing parameters. IEEE Trans. Vis. Comput. Graph. **15**, 841–852 (2009)
21. Taguchi, Y., Takahashi, K., Naemura, T.: Real-time all-in-focus video-based rendering using a network camera array. In: Proc. 3DPVT, pp. 241–244 (2008)
22. Yang, R., Huang, X., Li, S., Jaynes, C.: Toward the light field display: Autostereoscopic rendering via a cluster of projectors. IEEE Trans. Vis. Comput. Graph. **14**, 84–96 (2008)
23. Yang, R., Welch, G., Bishop, G.: Real-time consensus-based scene reconstruction using commodity graphics hardware. In: Proc. Pac. Graph., pp. 225– (2002)
24. Zabih, R., Woodfill, J.: Non-parametric local transforms for computing visual correspondence. In: Proc. ECCV, pp. 151–158 (1994)

Fabio Marton is a senior researcher in Visual Computing at the CRS4 research center, Italy. He holds a Laurea (M.Sc.) degree (1999) in Computer Engineering from the University of Padua, Italy, as well as Associate Professor Habilitations in Computer Science and Information Processing from the Italian Ministry of University and Research. His research interests include real-time amd multiresolution graphics.

Tibor Balogh is the CEO and founder of Holografika Ltd, Hungary. He holds a master's Degree (M.Sc) (1980) from the Technical University of Budapest. He has extensive experience in the field of holography and optical engineering. Winner of Denis Gabor award, he actively follows the developments of 3D display technologies and is responsible for overall management, business strategy and the R&D team of Holografika.

Vamsi Kiran Adhikarla is a PhD student in the faculty of information technology, Pazmany Peter Catholic University, Budapest, Hungary. He holds a master's double degree (M.Tech, M.Sc.) (2011) from Blekinge Tekniska Hgskola,Sweden and Jawaharlal Nehru Technological University Hyderabad, India. His research interests include capture, processing, rendering and visual enhancement for 3DTV.

Enrico Gobbetti is the director of Visual Computing at the CRS4 research center, Italy. He holds an Engineering degree (1989) and a Ph.D. degree (1993) in Computer Science from the Swiss Federal Institute of Technology in Lausanne (EPFL), as well as Full Professor Habilitations in Computer Science and Information Processing from the Italian Ministry of University and Research.