3DTV at Home: Eulerian-Lagrangian Stereo-to-Multiview Conversion

PETR KELLNHOFER, MIT CSAIL and MPI Informatik
PIOTR DIDYK, Saarland University, MMCI and MPI Informatik
SZU-PO WANG, MIT CSAIL
PITCHAYA SITTHI-AMORN, Chulalongkorn University
WILLIAM FREEMAN, MIT CSAIL
FREDO DURAND, MIT CSAIL
WOJCIECH MATUSIK, MIT CSAIL

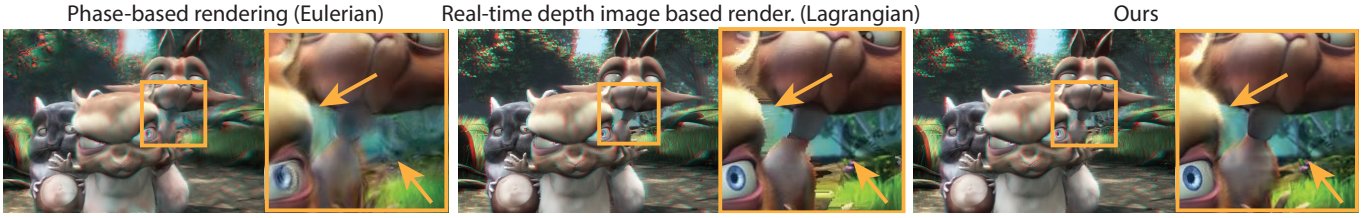

Fig. 1. Stereoscopic views of multiview content generated from stereoscopic content using our method and two previous approaches. Monocular insets highlight limitations of the Lagrangian approach when dealing with fuzzy depth edges and ringing artifacts caused by exhaustive input disparities in the case of the Eulerian approach. Our method successfully avoids such problems and produces outputs visually closest to the original input. Scene copyright: Blender Foundation (https://orange.blender.org/)

Stereoscopic 3D (S3D) movies have become widely popular in the movie theaters, but the adoption of S3D at home is low even though most TV sets support S3D. It is widely believed that S3D with glasses is not the right approach for the home. A much more appealing approach is to use automultiscopic displays that provide a glasses-free 3D experience to multiple viewers. A technical challenge is the lack of native multiview content that is required to deliver a proper view of the scene for every viewpoint. Our approach takes advantage of the abundance of stereoscopic 3D movies. We propose a real-time system that can convert stereoscopic video to a high-quality, multiview video that can be directly fed to automultiscopic displays. Our algorithm uses a wavelet-based decomposition of stereoscopic images with per-wavelet disparity estimation. A key to our solution lies in combining Lagrangian and Eulerian approaches for both the disparity estimation and novel view synthesis, which leverages the complementary advantages of both techniques. The solution preserves all the features of Eulerian methods, e.g., subpixel accuracy, high performance, robustness to ambiguous depth cases, and easy integration of inter-view aliasing while maintaining the advantages of Lagrangian approaches, e.g., robustness to large disparities and possibility of performing non-trivial disparity manipulations through both view extrapolation and interpolation. The method achieves real-time performance on current GPUs. Its design also enables an easy hardware implementation that is demonstrated using a field-programmable gate array. We analyze the visual quality and robustness of our technique on a number of synthetic and real-world examples. We also perform a user experiment which demonstrates benefits of the technique when compared to existing solutions.

CCS Concepts: • **Computing methodologies** → **Image-based rendering**; *Perception*; • **Hardware** → *Displays and imagers*;

Additional Key Words and Phrases: automultiscopic displays, view synthesis, interperspective antialiasing, view transitions

ACM Reference format:
Petr Kellnhofer, Piotr Didyk, Szu-Po Wang, Pitchaya Sitthi-Amorn, William Freeman, Fredo Durand, and Wojciech Matusik. 2017. 3DTV at Home: Eulerian-Lagrangian Stereo-to-Multiview Conversion. *ACM Trans. Graph.* 36, 4, Article 146 (July 2017), 13 pages.
DOI: http://dx.doi.org/10.1145/3072959.3073617

0730-0301/2017/7-ART146 $15.00
DOI: http://dx.doi.org/10.1145/3072959.3073617

1 INTRODUCTION

Stereoscopic 3D (S3D) has become much more popular during the last decade. Today, many movie blockbusters are released in a stereo format. However, the popularity of S3D in the movie theaters has not translated to equivalent popularity at homes. Despite the fact that most current TV sets support S3D and the content providers offer streaming stereoscopic content, the adoption of S3D at home remains very low. It is widely believed that the use of stereoscopic glasses is not practical in a home setting [Chinnock 2012], and we believe that the right approach to S3D at home is to use automultiscopic displays that provide a glasses-free, 3D stereoscopic experience to multiple viewers. These displays are rapidly improving due to the industry drive for a higher and higher display resolution (e.g., even current 4K UHD displays can be easily converted to a 3D automultiscopic display with 8 views and an HD spatial resolution). However, using these displays presents one fundamental challenge – while there is plenty of stereoscopic content available,

there is practically no multiview content for automultiscopic displays. Therefore, there is a clear need for methods and systems that can convert streaming, high-resolution, stereoscopic video available from the standard delivery channels to high-quality, multiview content in real time. Furthermore, the methods should be amenable to hardware implementations such that they can be incorporated in future streaming TV devices and smart TV sets. Finally, the systems should support some customization of the 3D video – viewers desire different levels of the 3D experience.

We propose a method that addresses all of the requirements we have outlined. It works with existing stereoscopic content expanding it to a high-quality, multiview format in real time. The method can be implemented efficiently in hardware and naturally supports disparity manipulations. It is inspired by the recent advances in phase-based approaches [Didyk et al. 2013; Fleet et al. 1991; Wadhwa et al. 2013] which provide robustness, enable inter-view anti-aliasing at almost no cost, and allow for simple disparity manipulation. In contrast to standard depth image-based rendering methods [Riechert et al. 2012; Zitnick et al. 2004], such techniques are limited to small disparities. Inspired by the work in physics-based simulation [Fan et al. 2013], we overcome this problem by combining a phase-based approach (an Eulerian method) with standard depth image-based rendering (a Lagrangian approach). Our technique starts by decomposing the input signal using a set of filters inspired by a steerable pyramid decomposition [Simoncelli and Freeman 1995; Simoncelli et al. 1992]. The basis functions of this transform resemble Gabor-like wavelets; therefore, we will refer to them as wavelets. Next, disparity information is estimated for each of them separately using a combination of standard disparity estimation and phase-based measures. To synthesize new views, our method applies a wavelet re-projection which moves wavelets according to their disparities. Such an approach allows us to both handle large disparities and preserve all the advantages of the Eulerian approach [Didyk et al. 2013]. We demonstrate that our method can provide real-time performance both on a GPU and a field-programmable gate array (FPGA). We evaluate our method on a variety of stereoscopic test scenes and Hollywood movies.

2 PREVIOUS WORK

Multiview content can be captured with camera arrays [Matusik and Pfister 2004; Wilburn et al. 2005, 2001]. Such setups are expensive and hard to manage due to their size. Smaller lightfield cameras [Lytro Inc. 2015; Raytrix GmbH 2015] have become available, but the amount of parallax they offer is insufficient to create good stereoscopic effects. A great alternative to capturing multiview content is to use image-based techniques which can convert existing, widely available stereoscopic footage to a multiview version. In this section, we provide an overview of these techniques.

Image-based rendering techniques can be categorized into Lagrangian and Eulerian methods. We borrow this terminology from recent work on image manipulations [Didyk et al. 2013; Wadhwa et al. 2013; Wu et al. 2012]. Originally, both terms refer to handling fluid dynamics. Lagrangian techniques analyze the trajectory of the individual particles, whereas Eulerian methods analyze the local change of different characteristics, such as pressure and velocity, over time. Similarly, in the context of novel-view synthesis, Lagrangian techniques analyze correspondence in the input images, i.e., depth/disparities, whereas Eulerian approaches process local changes of pixel values.

Lagrangian Techniques. Lagrangian techniques recover depth information first [Brown et al. 2003], and then use re-projection [Mark et al. 1997] to create novel views [Smolic et al. 2008]. Using such an approach, Riechert et al. [2012] and Liao et al. [2013] built systems for real-time stereo-to-multiview conversion, and similar techniques are used in the context of view reprojection for virtual reality [Anderson et al. 2016]. Although many sophisticated techniques for depth estimation have been proposed, this is still a challenging problem, especially in the case of real-time applications. The methods still suffer from low-quality depth maps, if the performance of the system is of high importance. To overcome this limitation, it is possible to improve depth information using an additional filtering [Matsuo et al. 2013; Richardt et al. 2012], or by applying more sophisticated matting techniques [Hasinoff et al. 2006; Zitnick et al. 2004]. Although such refinements can lead to significant quality improvement, this often comes at the price of reduced performance. For example, the solution by Zitnick et al. requires an additional off-line preprocessing step. Another approach to overcome the problem of poor depth estimation is to use sparse depth information together with an image warping technique [Farre et al. 2011; Stefanoski et al. 2013]. Such methods have an additional advantage as they do not need to deal with missing information in disocclusion regions. This, however, comes at the price of poor depth quality at sharp depth discontinuities and in regions with fine depth details. The resolution of the mesh is usually too coarse to handle such cases. Very recently, a hardware implementation of such a technique has also been presented [Schaffner et al. 2015]. A different kind of approach was proposed by Flynn et al. [2015]. They trained a deep convolution network for view interpolation. A similar approach was presented by Kalantari et al. [2016]. They trained a deep convolutional network for both disparity and in-between view computation for narrow baseline data. Both techniques achieve superior performance in challenging regions with occlusions, but they cannot perform significant extrapolation. It is also unclear how they can handle arbitrary baselines. Both of these aspects are crucial for creating content for automultiscopic display. Furthermore, real-time performance was not demonstrated for these techniques, especially in the context of high-resolution content as in our case.

Most of the Lagrangian approaches rely explicitly on per-pixel depth information. This is often insufficient when depth cannot be uniquely defined. Examples include motion blur, depth-of-field effects, and transparencies, which commonly appear in the case of movie content. This problem has been recently acknowledged by view synthesis techniques that handle the case of highly reflective surfaces. The most common techniques decompose the input image into layers, e.g., diffuse and specular, [Sinha et al. 2009; Szeliski et al. 2000] and perform the view synthesis separately for each of them. Recently, a more robust technique that does not require the explicit layer separation has been proposed by Kopf et al. [2013]. Although these techniques offer a significant step towards handling difficult cases, they deal only with reflections. Also, none of these

works provide a complete real-time end-to-end system for novel view synthesis.

Eulerian Techniques. Eulerian techniques estimate local changes using local phase information, as opposed to recovering depth or optical flow information explicitly. The advantages of phase-based processing have been presented by Fleet et al. [1990; 1991], and more recently in [Didyk et al. 2013; Wadhwa et al. 2013]. They are often attributed to the overcomplete representation, i.e., instead, of one per-pixel depth value, phase-based approaches consider localized, per-band information. This leads to better results in difficult cases where per-pixel information cannot be reliably estimated, e.g., depth-of-field, motion blur, specularities, etc. [Didyk et al. 2013], and more accurate estimates due to the sub-pixel precision of these techniques [Fleet et al. 1991; Wadhwa et al. 2013]. Another argument is that phase-based manipulations are semi-local and cannot have catastrophic failures like pixel warping does. As a result, such methods provide graceful quality degradation. Unfortunately, phase-based techniques have one significant limitation: the disparity/depth range that they can deal with is relatively small [Didyk et al. 2013].

Although there exist multiscale phase-based disparity estimation techniques which extend the supported disparity range [Pauwels and Van Hulle 2008; Zhou et al. 2007], their goal is to estimate per-pixel disparity. Instead, we address the problem of limited disparity support by combining a phase-based technique with a Lagrangian approach which pre-aligns views to reduce disparity so that the Eulerian approach can be applied. In this regard, the most similar work to ours is the technique proposed by Zhang et al. [2015], which addresses the problem of reconstructing a light field from a micro-baseline image pair. Similarly to our work, they also rely both on disparity and phase information. However, in contrast to their view-synthesis method which relies on per-pixel disparity information, we use a concept of per-wavelet disparity, which provides much richer representation. Another difference is that we propose a real-time solution which is capable of performing the stereo-to-multiview conversion on the fly. To our knowledge, there have been no attempts at designing hardware implementations of Eulerian techniques for view expansion.

Discussion. In comparison to previous techniques, the main contribution of this work is the end-to-end solution for multiview content creation that exploits complementary advantages of Lagrangian and Eulerian techniques and overcomes their limitations. We draw inspiration from the steerable pyramid decomposition [Simoncelli and Freeman 1995; Simoncelli et al. 1992] which was recently used [Didyk et al. 2013; Wadhwa et al. 2013], but we augment it with depth information. This enables handling large disparities, which was the main limitation of previous phase-based methods. Although initialization of depth estimation techniques with a good guess provided by another or the same technique is not new [Fleet et al. 1991; Nishihara 1984], the idea was mostly exploited in the context of multi-scale approaches, also called coarse-to-fine propagation techniques, where the goal is to estimate per-pixel disparity. In our work, we explicitly avoid such strategies and do not share disparity information between different frequency levels of our decomposition. This leads to a more flexible representation for cases where a single per-pixel disparity is not defined, as for multiple depth-separated image layers. We also reduce our conversion problem to a set of 1D problems, which significantly improves performance. To synthesize novel views, we introduce a new view synthesis approach which reprojects wavelets. To this end, we employ a non-uniform Fourier transform. Despite some similarities to the idea of pixel reprojection (e.g., [Mark et al. 1997]), the domain and technique are significantly different. All the above steps make our technique suitable for hardware implementation (Section 5). We believe that this is the first attempt to implement a phase-based view expansion in hardware.

3 STEREO TO MULTIVIEW CONVERSION

For expanding stereoscopic content to its multiview version, our method takes as an input a rectified stereoscopic image pair together with corresponding disparity maps. In the first step, the images are decomposed into wavelet representations (Section 3.1), and disparity maps are used to compute per-wavelet disparity. For efficiency reasons, we allow the disparity maps to be low quality. In our method, we are concerned with reproduction of horizontal parallax, and use low-resolution disparity maps computed using the work of Hosni et al. [2013]. We also rectify the input views using [Fusiello et al. 2000]. Next, we refine per-wavelet disparity by incorporating phase information. To reconstruct novel views, we propose a new image-based rendering approach tailored to our decomposition (Section 3.2). In contrast to standard image-based rendering techniques which use pixel reprojection to compute novel views [Mark et al. 1997], our technique reprojects whole wavelets. It supports both view interpolation and extrapolation in a unified way. The two operations differ only in the direction in which wavelet locations are altered.

3.1 Per-wavelet Depth Estimation

Disparity is an important cue to synthesize novel views. For stereoscopic content, disparity maps (D_l and D_r) encode the correspondence between left and right views (L and R). More formally, if for a given position in the world space, its projections into the left and the right views are $\mathbf{x}_l$ and $\mathbf{x}_r$, the disparity is defined as the distance between those locations in the screen space. A signed distance is considered to distinguish between locations in front of and behind the zero-disparity plane. For rectified views disparity maps represent a horizontal translation and can be defined as follows: $D_l(\mathbf{x}_l) = \mathbf{x}_{lx} - \mathbf{x}_{rx}$ and $D_r(\mathbf{x}_r) = \mathbf{x}_{rx} - \mathbf{x}_{lx}$, where $\mathbf{x}_{lx}$ and $\mathbf{x}_{rx}$ denote the horizontal components of $\mathbf{x}_l$ and $\mathbf{x}_r$.

In contrast to previous approaches, we consider per-wavelet, instead of per-pixel, disparity. This allows us to use phase information to improve the quality of the estimates and overcome limitations of previous Lagrangian and Eulerian approaches. To compute per-wavelet disparities, we first decompose the input images into wavelet representations. Then, for each wavelet, the initial disparity is computed from the input disparity maps. In the next step, this information is refined by additionally considering local phase information. The whole process is depicted in Figure 2. Our disparity information is not a single disparity map. Instead, we obtain one disparity map for each pyramid level.

Wavelet Decomposition. Because the input views are rectified, we can limit our analysis to scanlines. We consider each pair of

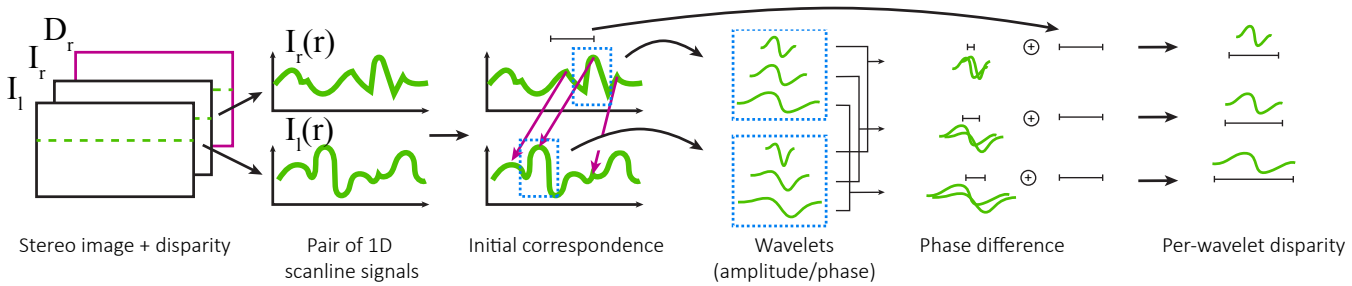

Fig. 2. In contrast to most approaches where a per-pixel disparity is estimated, in our method we consider wavelets as basic elements of a picture and estimate disparity for each of them. To this end, from left to right, we start with a stereoscopic image pair and consider each image scanline independently. We decompose the scanlines into wavelets and find the initial correspondence between wavelets from the left and the right views based on the input disparity maps. The position difference of the corresponding wavelets defines the initial disparity information. To further refine it, the phase difference of the wavelets is computed and combined with the initial disparity estimation.

corresponding scanlines (1D signals) (I_r and I_l) of the right and left views separately and represent them as a sum of basis functions b_f with a frequency response defined as:

$$\hat{b}_f(\xi) = \cos(\frac{\pi}{2}\log_w(\xi/f)) \cdot \Pi(\frac{1}{2}\log_w(\xi/f)), \quad (1)$$

where $f \in \mathcal{F}$ specifies the central frequency of the filter, Π is a rectangular function centered around zero that extends from -0.5 to 0.5, and w defines the width of filters – the ratio of central frequencies of neighboring levels. In this work, we perform an octave decomposition and use $w = 2$. We found no visible difference when reconstructing new views using $\mathcal{F}$ with frequency below 16. Consequently, in all our results, we let

$$\mathcal{F} = \{2^n | n = \{4 \ldots \log_2(\text{length}(I))\}\}.$$

The filters in Equation 1 are 1D versions of filters used by Simoncelli et al. [1995; 1992]. Similarly to the original filters, ours allow for computing local phase and amplitude but lack information on orientation. An additional low-pass filter,

$$\hat{b}_0(\xi) = \prod_{f \in \mathcal{F}} \sqrt{1 - \hat{b}_f^2(\xi)}, \quad (2)$$

collects the residual low-frequency components. The filters are visualized in Figure 3.

Using such a filter bank, we compute a single wavelet coefficient for a given location x and frequency f as:

$$A_{fx} = (b_f * I)(x),$$

where $*$ denotes a convolution. As we use complex filters b_f, A_{fx} is also a complex number which contains local phase and amplitude. The decomposition can be easily inverted by summing up wavelets for all frequencies in ($\mathcal{F}$) and the additional residual component from Equation 2:

$$I = 2\,\text{Re}\left(\sum_{f \in \mathcal{F} \cup \{0\}} \frac{\text{length}(I)}{|X_f|}\left(\sum_{x \in X_f} A_{fx} b_f(t-x)\right)\right). \quad (3)$$

The additional factor of 2 compensates for the fact that the complex wavelets are obtained only from positive frequency components, and

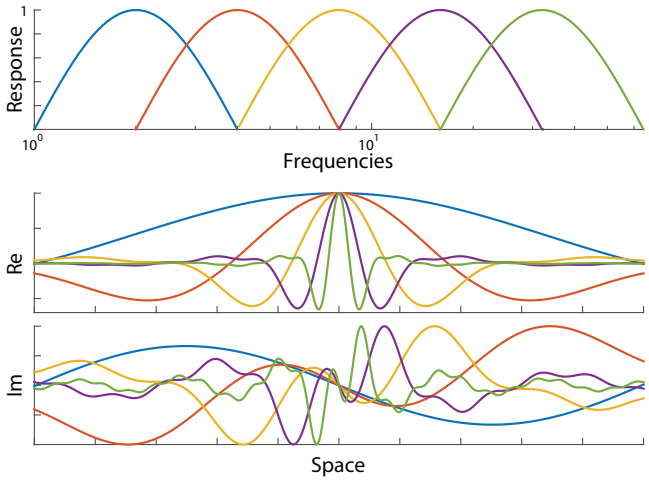

Fig. 3. Filters used to perform wavelet decomposition. From the top: Frequency response for several filters; real part in the spatial domain; and imaginary part. The plots in the spatial domain are scaled for visualization purposes.

factor of $length(I)/|X_f|$ is necessary to compensate for the energy loss due to only $|X_f|$ wavelets representing the signal. We choose

$$X_f = \{x \in \mathbb{Z} | \max(f - f/2, 1) \le x \le \min(f + f/2, \text{length}(I))\}, \quad (4)$$

for $f \in \mathcal{F}$ and $X_0 = X_{fmin}$ where $fmin$ is the lowest frequency in $\mathcal{F}$. These sets have overlapping regions such that each wavelet is sampling at least twice, so that it prevents aliasing.

In practice, both decomposition and reconstruction are performed in the frequency domain. To decompose the signal, we simply transform each 1D scanline into the frequency domain, multiply it with the filters, and transform the result to the pixel domain. The reconstruction is done similarly in the frequency domain, but in our case, this step requires a non-uniform Fourier transform (Section 3.2).

Initial Wavelet Disparity. After decomposing I_r and I_l into wavelets, we establish a correspondence between them using input disparity maps (D_r and D_l). More precisely, for each wavelet ψ_{rfx} from I_r we seek a corresponding wavelet $\psi_{lfx'}$ from I_l. To this end, for each ψ_{rfx} we compute a disparity value from D_r. Because each wavelet

spans a certain spatial extent, there is no direct correspondence between wavelets and disparity values. Therefore, we compute the disparity of a wavelet as an average of disparities in its local neighborhood whose size is equal to the wavelet spacing. Formally, the disparity for wavelet ψ_{rfx} is defined as

$$d_{rfx} = \sum_{y=x-s/2}^{x+s/2} D_r(y) \,/\, (s+1), \quad \text{where} \quad s = |\, I_r \,|\,/\,|\, X_f \,|.$$

$\psi_{lfx'}$ is then found as the closest wavelet to the location $x - d_{rfx}$. We perform the same step for all wavelets from I_l. An alternative to finding the closest wavelet would be to re-evaluate it at the exact same location. However, this would significantly increase the computational cost.

Wavelet Disparity Refinement. The disparity between wavelet pairs computed in the previous step is often inaccurate, due to insufficient quality of the input disparity maps, or additional effects such as transparency or depth of field that cannot be captured using a per-pixel disparity value. However, our observation is that the initial correspondence serves as a good pre-alignment, and the residual disparity that is not captured by the disparity is usually small. Such small, often sub-pixel differences can usually be effectively captured by the phase [Didyk et al. 2013; Fleet et al. 1991]. Therefore, we further improve the per-wavelet disparity estimation using phase difference between corresponding wavelets:

$$\Delta\varphi = \arg(A_{rfx}) - \arg(A_{lfx'}).$$

The phase difference can be easily transformed into the disparity residual by multiplying it by $f/2\pi$, and added to the initial disparity of wavelet as a correction. Consequently, we update disparity information d_{rfx} of wavelet ψ_{rfx} by adding $\Delta\varphi \cdot f/2\pi$. In this way, we obtain a continuous depth resolution without expensively numerous depth labels. For color images, we compute phase differences for each channel separately and combine them using a weighted sum to get the disparity refinement. The weights are proportional to the wavelet amplitudes to penalize the phase for weak signals that can be only poorly estimated.

Our per-wavelet disparity estimation is performed on individual 1D scanlines. To prevent inconsistencies between them, we apply an additional filtering to the disparity estimation. More precisely, we filter the per-wavelet disparity using a 2D mean filter with a kernel size equal to double wavelet spacing. To avoid filtering across significant discontinuities, we penalize contributions from wavelets with a large phase difference. To this end, we weight the contribution of each wavelet using a Gaussian function defined on the phase differences with $\sigma = \pi/4$.

As a result of our wavelet disparity refinement step, we obtain an accurate disparity estimation for each wavelet. Compared to standard depth-based methods which compute per-pixel disparity information, this is a much richer representation, as it stores disparity information separately for different frequencies. As we show later (Section 6), such additional information enables handling difficult cases when used for rendering novel views.

3.2 Novel Views Reconstruction

To compute novel views, we first modify the position of each wavelet. The new position for each wavelet ψ at location x and disparity d is computed as $x + a \cdot d$, where parameter a directly controls the new viewing position. After the position of each wavelet is updated, we convert the displaced wavelets back into uniform-spaced samples using a non-uniform Fourier transform as described in [Liu and Nguyen 1998]. The non-uniform Fourier transform process utilizes an oversampled grid with an oversampling factor $m = 2$. Each displaced wavelet is approximated as a weighted sum of $q = 4$ nearby samples on the oversampled grid, where the weights depend on the fractional residual in the displaced location. After the contributions from all wavelets are summed, a low-pass filter is used to downsample back into the original grid. We refer the reader to the original paper for more details. After the wavelets are converted back to the original uniform grid, we can reconstruct the 1D signal using a pyramid reconstruction. For lowest frequency wavelets corresponding to filter b_0, a linear interpolation of the wavelet values on the uniform grid is used. This is to prevent low-passed wavelets from accumulating and creating color bands.

View Arrangement for a Screen. In the simplest case, we treat the two input views as the central views of an autostereoscopic screen, and reconstruct every other view from the closest original one. More formally, if the targeted display requires a set of $2N$ views $\{V_i\}$, then V_N view corresponds to the left input view, and V_{N+1} view corresponds to the right input view. The set of views $\{V_i : i < N\}$ is then reconstructed from the left view by setting $a = |N - i|$. Views $\{V_i : i > N + 1\}$ are reconstructed from the right view by setting $a = |i - N - 1|$. This strategy leads to a simple view expansion. Note that before reconstructing novel views, the pair of corresponding wavelets from the left and the right views can be moved closer to each other by scaling disparity between them by factor $s < 1$ and moving their positions accordingly. Effectively, such an operation reduces disparities in the original views, and when the strategy for new views reconstruction is applied, the disparities between neighboring views will be compressed by factor s. Similarly, one can increase disparity in the multiview content by scaling the disparities between the initial wavelets by $s > 1$. Please note that if $s < 1$, some of the views will be a result of interpolation between the input views and others will be extrapolated. Our technique is, however, transparent to these cases and can treat both of them simultaneously.

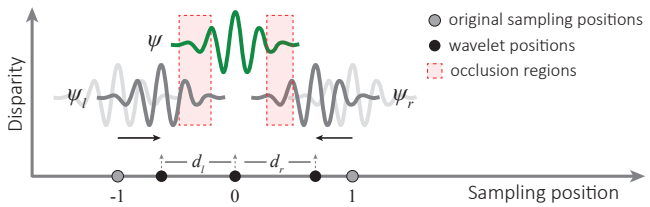

Fig. 4. We resolve occlusion of wavelet ψ by attenuating its amplitude. The attenuation is proportional to the part of the wavelet ψ that is occluded by the nearest left foreground wavelet ψ_l and the nearest right foreground wavelet ψ_l within the same view and frequency band.

Occlusions and Disocclusions. Moving individual wavelets of the same frequency independently has similar shortcomings as moving image patches in the Lagrangian approach. There might be two potential problems resulting from the non-uniform sampling. First, there can be missing information in the undersampled regions. This does not cause significant problems, as there is remaining information in lower frequency levels. Second, some of the wavelets may overlap. This leads to mixing background and foreground signals. To avoid this, we detect occluded wavelets and attenuate their frequency. This approach is conceptually similar to resolving pixel occlusions using depth information in DIBR.

To this end, for a given wavelet ψ we first find the closest wavelets to the left ψ_l and to the right ψ_r that have smaller disparities (i.e., they are in front of ψ). It is sufficient to consider wavelets corresponding to the same frequency. We then compute the portion of the wavelet ψ that is occluded by ψ_l and ψ_r. We assume that one wavelet completely occludes the other wavelet if the distance between them is at most half of the original sampling distance. As a result, we defined the occlusion using distances between sampling locations of ψ and the other two wavelets, i.e., the occlusion caused by ψ_l is defined as $\mathrm{O}_l = \max(2 - 2d_l, 0)$ and for ψ_r by $\mathrm{O}_r = \max(2 - 2d_r, 0)$. Here, d_l and d_r are the distances, as marked in Figure 4, and the original spacing between wavelets is assumed to be 1. The occlusions have constant value 1 if the neighboring wavelet moves halfway to ψ, and 0, if the distance between them is at least the original sampling distance. To combine occlusions for both wavelets, we define the effective occlusion of wavelet ψ as $\mathrm{O}_\psi = \mathrm{O}_l + \mathrm{O}_r$. $\mathrm{O}_\psi = 0$ indicates that neither ψ_l nor ψ_r occlude ψ. $\mathrm{O}_\psi = 1$ indicates that wavelet ψ is completely occluded. Next, we attenuate wavelet ψ according to a smooth function s that interpolates between 0 and 1.

$$s(x) = \begin{cases} 1 & \text{if } x \geq 1 \\ 3x^2 - 2x^3 & \text{if } x \in (0,1) \\ 0 & \text{if } x \leq 0 \end{cases}$$

The amplitude of the attenuated wavelet is then defined as $\overline{A_\psi} = s(\mathrm{O}_\psi) \cdot A_\psi$. For real-time performance, we find ψ_l and ψ_l by first placing all wavelets in buckets according to their location, and then considering wavelets only from neighboring buckets within a distance of the wavelet spacing at the current level.

4 ADDITIONAL PROCESSING

Computing high-quality views is not sufficient to assure perfect viewing quality. Due to the limited angular resolution of automultiscopic screens, displaying synthesized multiview content directly on a screen may lead to significant inter-view aliasing in regions with large disparities and fine texture. One of the results of such aliasing is ghosting (Figure 5, insets). To enhance the quality and provide a better experience, an inter-view antialiasing needs to be applied [Zwicker et al. 2006]. Moreover, due to the accommodation-vergence conflict, large disparities may introduce visual discomfort [Shibata et al. 2011]. To overcome this limitation, depth presented on such an automultiscopic display needs to be carefully adjusted to match its capabilities [Chapiro et al. 2014; Didyk et al. 2013; Masia et al. 2013]. Both angular antialiasing and depth manipulations can be easily incorporated into our method.

Antialiasing. Didyk et al. [2013] proposed to perform the inter-view antialiasing by attenuating local amplitude according to phase information. As we rely on a very similar decomposition, the filtering can be performed using a similar technique. To filter a view that was synthesized using our method, we attenuate every wavelet before the view is reconstructed. The amount of attenuation depends directly on the disparities between neighboring views, which can be easily obtained from our representation. We choose a Gaussian filter for our antialiasing filtering. For a given wavelet at frequency level f with disparity d, we filter the signal with $\frac{1}{\sqrt{2\pi}\sigma} \exp(-d^2/(2\sigma^2))$, where σ is the antialiasing width as defined in Didyk et al. [2013]. This corresponds to multiplying amplitude of each wavelet with $\exp(-\sigma^2(\frac{2\pi d}{f})^2/2)$. The example of a synthesized view and its filtered version is shown in Figure 5 (top). Note that this can lead to blur in areas with large disparities, e.g., the background, but in return, it avoids significant ghosting, which can impair stereo perception and is in general not desired [Zwicker et al. 2006].

Disparity Adjustment. Using our wavelet representation together with per-wavelet disparity information, we can easily apply non-linear disparity mapping operators, which was not possible for Eulerian methods [Didyk et al. 2013]. Such operators are usually defined as a disparity mapping function which maps disparity according to certain goals [Didyk et al. 2012; Lang et al. 2010]. In contrast to simple disparity scaling described in Section 3, a disparity mapping function usually scales disparities in a non-linear way. To apply such a mapping during our synthesis, it is sufficient to replace the scaling factor s with the desired non-linear function. The rest of our view synthesis technique in Section 3 remains unchanged. In Figure 5 (bottom), we demonstrate one example of such manipulations.

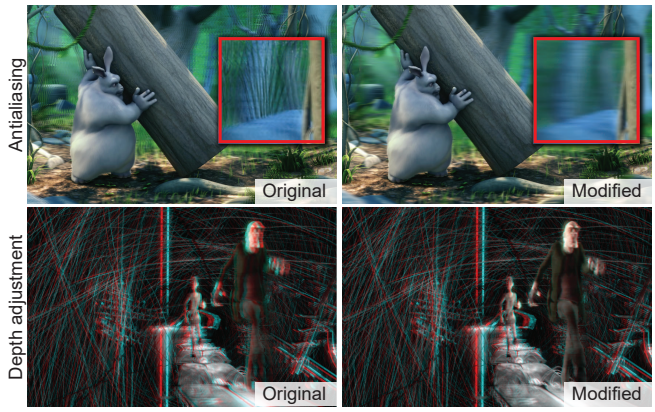

Fig. 5. This figure presents the additional processing. Top: A synthesized view using our technique (left) without inter-view antialiasing simulated as it would appear on an automultiscopic screen; the same view with the antialiasing. The inset shows a zoomed-in region. Note how aliasing in the form of ghosting is removed by the additional step. Bottom: An example of nonlinear disparity remapping. The depth for the foreground objects is compressed, resulting in this part of the scene being pushed close to the zero disparity plane (screen depth). Both original and modified images can be viewed using red-cyan anaglyph glasses.

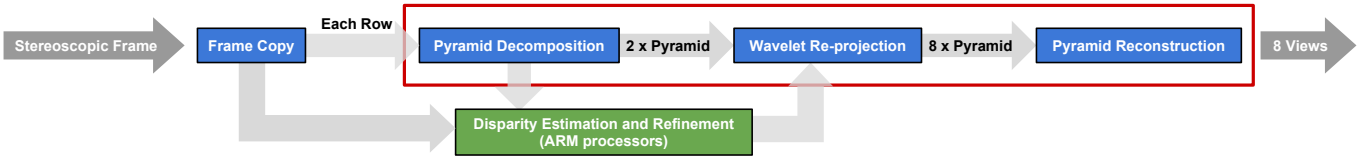

Fig. 6. The figure presents how our method can be mapped to hardware architecture composed of an FPGA board with ARM processors.

5 IMPLEMENTATIONS

Our technique provides performance that is necessary to convert a 4K stereoscopic content in real time. In this section, we describe two implementations: a CUDA-based GPU implementation and a hardware implementation using an FPGA with ARM processors.

5.1 GPU Implementation

We produce content for an 8-view 4K (3840 × 2160) automultiscopic display, where each of the output views has a resolution of 960 × 1080. Our implementation accepts a FullHD stereo video input and computes the initial disparity maps at a quarter the size of the input. The rest of the pipeline is computed at 960 × 1080. We implement our method on a GPU using CUDA. To test its performance, we run it on the Nvidia GeForce GTX Titan Z graphics card. For such a setup, our technique can perform the conversion with the additional steps in 25–26 FPS for all sequences presented in the paper and the supplemental material. The breakdown of the timing and the memory usage for the individual steps is presented in Table 1.

Stage	Timing (%)	Memory (GB)
Pyramid decomposition	9.9	1.05
Initial disparity estimation	4.9	0.31
Per-wavelet disparity refinement	18.5	0.23
Wavelet re-projection	30.5	0.50
Pyramid reconstruction	36.2	1.55

Table 1. Performance breakdown for the individual steps of the GPU implementation.

5.2 FPGA Implementation

One advantage of our technique is that most stages in our algorithm can be done in a scanline fashion. This eliminates the need for any external memory during the computation of these stages, and thus, it is suitable for a hardware implementation such as an FPGA or an ASIC. Our technique requires only low-resolution disparity maps. Therefore, we leverage the ARM processors inside the System-On-Chip (SoC) for this task. The ARM processor computes these disparity maps at the 240 × 180 resolution at 24 FPS.

Figure 6 describes each stage in our hardware implementation. The first stage decomposes the frame into two pyramids: one for the left view and the second for the right view. Both pyramids are sent to the second stage. In the second stage, each wavelet in the pyramid is re-projected according to the disparity from the ARM processor. The re-projected wavelets are filtered similarly to [Liu and Nguyen 1998] and sent to the final stage. The final stage reconstructs views from the synthesized pyramids and sends the result to the output.

We test each stage of our implementation on the FPGA SoC Xilinx ZC706 development board using Xilinx Vivado HLS 2015.4 software. The FPGA SoC has two ARM processors running at up to 1 GHz and programmable logic with 350K logic cells and a total of 19Mbit of internal RAM. Table 2 shows the resource utilization of our implementation. Each stage is customized to the target, generating 8 views of 512× 540 resolution at 24 FPS while running at 150 MHz. The total memory utilization of our implementation is only 13 Mbit of the internal memory. This is a much smaller memory footprint than our current GPU implementation. Moreover, the current FPGA implementation uses only about 50% of the hardware resource on the FPGA we are using. Therefore, it is possible to double the resolution to get a FullHD resolution in the future implementations.

Stage	RAMs (Kbits)	DSPs	LUTs	FFs
Pyramid decomposition	976	26	14K	12K
Wavelet re-projection	12,960	427	74K	85K
Pyramid reconstruction	1,476	75	13K	20K

Table 2. Resource utilization on our FPGA implementation.

6 RESULTS AND COMPARISONS

In this section, we provide an evaluation of our technique on both synthetic and real footage. We also compare our results to other techniques and ground truth data. In the paper, we show only stereoscopic and single-view images. Please refer to the supplemental videos, where we show several results of our real-time expansion to 32-view content for video content. The results also include a stereoscopic version of the content, as well as a capture of an autostereoscopic screen showing our results. We consider both interpolation and extrapolation for all results in our work.

6.1 Comparison to State of the Art

We compare our method to both Lagrangian and Eulerian techniques. The first group consists of a depth image-based rendering (DIBR) [Riechert et al. 2012] and an image-domain warping (IDW) [Schaffner et al. 2015]. Both of them target a real-time conversion of stereoscopic content to its multiview version. As the source code of the first method is not publicly available, we used our implementation. We compute the initial disparity map using [Hosni et al. 2013], and apply further depth refinement [Matsuo et al. 2013] to improve its quality. For the second technique [Schaffner et al. 2015], we provide a direct comparison to the results provided in the original paper. For reference, we also compare this to the technique by Riechert et al. [2012] which is supplied with a full-resolution depth map. However, this solution cannot be considered as being

Fig. 7. *Columns*: Comparison of the same red-cyan anaglyph stereoscopic views produced by PBR, offline full-resolution DIBR, real-time DIBR and our algorithm as presented to participants in our user study. *Rows*: "Big Buck Bunny" © by Blender Foundation (Scene 1 and 2) and "Ball" © by Eric Deren / Dzignlight Studios (Scene 3).

real-time due to the offline depth computation. We also compare to the Eulerian method proposed by Didyk et al. [2013] which applies a phase-based rendering approach (PBR).

Lagrangian Approaches. The most common artifacts in image-based rendering occur on object boundaries. Due to the lack of information in disoccluded regions, DIBR requires an additional hole filling step which provides only an approximate solution by filling in the information from neighboring regions. This step is very sensitive to any depth inaccuracies and a wrong assignment of pixels to foreground/background regions. Post-processing techniques applied to improve depth usually cannot provide a sufficient improvement due to boundary pixels sharing both foreground and background information. The above problems lead to an effect of stretching content over the disoccluded regions. Our technique does not need to explicitly perform hole filling. Instead, the missing information is filled during the non-uniform FFT. As a result, the local frequency spectrum in disoccluded regions is similar to the one in the neighborhood, which can be considered as hallucinating the unknown content. This is different from the interpolation performed by DIBR techniques, which leaves vertical frequencies and removes horizontal ones. The better performance of our technique on object boundaries can be observed in Figure 1 and Figure 7 (Scene 3, red).

Although the additional post-processing techniques such as [Matsuo et al. 2013] improve the quality of depth at depth discontinuities, in many cases they also lead to depth flattening. In regions without significant color boundaries or small depth variations, the additional filtering creates big flat depth regions, a so-called cardboard effect, when the content is viewed stereoscopically [Meesters et al. 2004]. While these artifacts can be observed in the real-time DIBR method as in Figure 7 (Scene 1, red and Scene 3, red), our step of phase-based depth correction can recover from these artifacts and provide a more correct depth percept. The offline DIBR technique supplied with a high-quality depth map usually does not produce such artifacts.

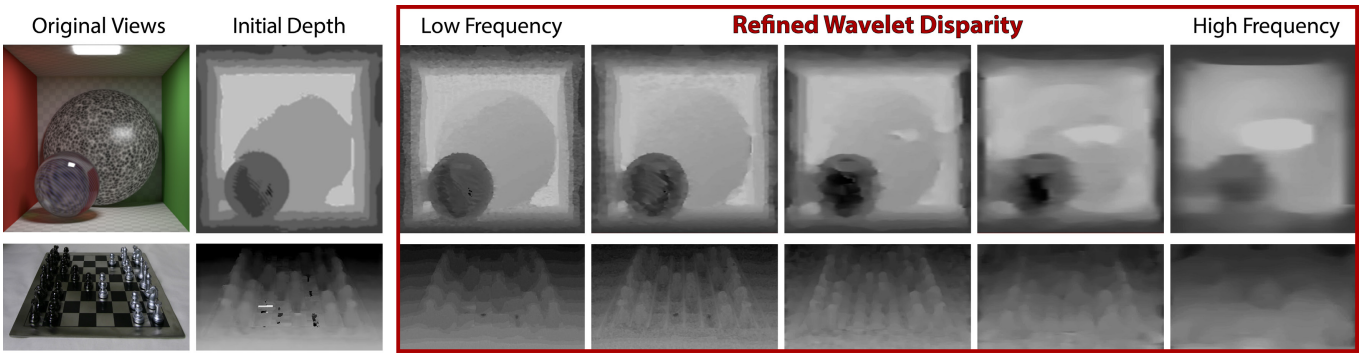

Fig. 8. Complex light effects. As highlights or refractions have a different disparity than the diffuse component, it is challenging to expand views correctly due to an incorrectly estimated initial depth map. Our approach can recover from such a situation by estimating disparity information separately for each wavelet frequency level. In this scene, the disparity information for diffuse components is captured by higher frequency levels, while lower frequency levels contain the disparity information for highlights and refractions.

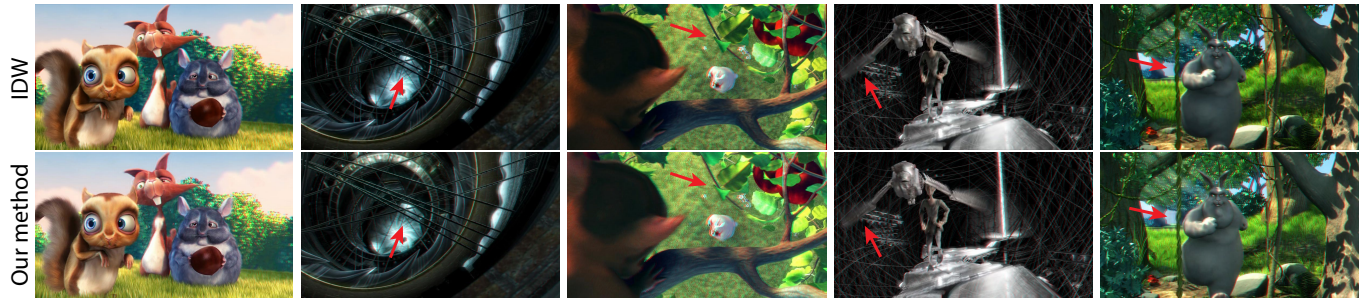

Fig. 9. The figure presents a comparison of our technique to a recent image-domain warping technique [Schaffner et al. 2015]. The top row images come directly from the original publication, while the bottom row contains corresponding results of our method. The stereoscopic images are presented in anaglyph colors. Besides the first image (the leftmost one), the images were reported as difficult cases for the IDW method. In all cases they suffer from inaccuracies coming from sparse depth representations and the artifacts include flattening the scene as well as loss of sharp depth discontinuities (arrows). Results provided by our technique offer a more correct depth reproduction. At the same time, for the images on which IDW technique performs well (the first image on the left), our technique provides equally good results.

Lagrangian techniques also have significant problems when per-pixel depth is insufficient to provide good depth representation in a scene. Such cases include, for example, depth-of-field, motion blur, transparencies, and complex reflectance effects such as highlights. Our technique, by leveraging the advantages of the Eulerian approach, can handle such situations more accurately. This is because instead of per-pixel disparity information, it uses a per-wavelet disparity which provides more information. In the case of motion blur and depth-of-field effects, Lagrangian techniques tend to introduce sharp edges that are not present in the original frames. The problem is also mentioned in [Riechert et al. 2012]. The effect can be observed in Figure 7 (Scene 2, red and yellow regions): motion blur and depth-of-field effects are not correctly reproduced by the real-time DIBR. The results are better for the offline DIBR. Our solution provides more accurate results. In Scene 3, the DIBR methods have problems with reproducing the correct depth of reflections, tiny particles, and the glassy ball. In contrast, our technique can reproduce the depth of these elements more precisely by leveraging the advantages of the Eulerian approach. Figure 8 provides a more in-depth analysis of how our wavelet representation helps us resolve ambiguous depth situations.

The problem of depth inaccuracies is addressed in image-domain warping (IDW) techniques. Such methods overcome the problem by warping the image according to sparse depth information. For example, Schaffner et al. [2015] use mesh resolution 180×100. This avoids many visual artifacts related to depth estimation, but introduces another type of artifact. As the depth information is represented using a sparse set of features, depth details cannot be reproduced. We compare our technique to [Schaffner et al. 2015] in Figure 9. In all cases, the IDW technique does not introduce visible 2D artifacts; however, when significant depth variations are present, it flattens the scene and smooths out depth discontinuities.

Eulerian approach. The major limitation of phase-based techniques is that they can handle only a limited range of disparities [Didyk et al. 2013]. As a result, the higher frequencies are incorrectly synthesized, which leads to significant ringing and blurring (Figure 1 and Figure 7, Scene 1, yellow). In our work, we overcome this problem by combining a phase-based approach with a standard Lagrangian approach. Our technique can handle much larger input disparities. This leads to better reproduction of high-frequency content. An important feature of our technique is the ability to perform sophisticated, nonlinear depth manipulations (Figure 5, bottom).

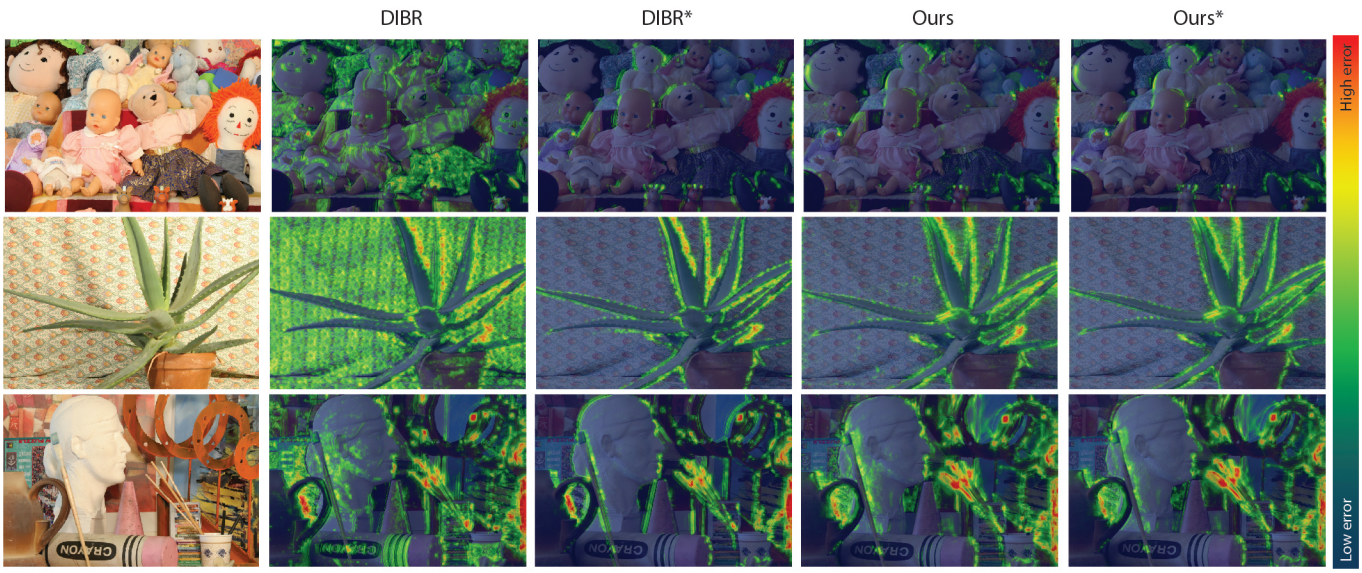

Fig. 10. The figure presents an objective performance of our technique and DIBR when compared to ground truth target view images from the Middlebury stereo datasets (http://vision.middlebury.edu/stereo/). Additionally, the results of both techniques with ground truth disparity information are shown (Ours* and DIBR*). The colormaps correspond to the differences between reconstructed images and original ground truth target views measured using the SSIM metric [Wang et al. 2004]. While our technique (fourth column) outperforms the DIBR technique (second column), it produces very similar results as the methods that use ground truth disparity information (third and fifth columns).

The standard phase-based technique can provide only linear scaling of disparities.

6.2 Comparison to Ground Truth

The input to our technique is a disparity map generated using [Hosni et al. 2013] at a quarter of the input resolution. We then resample it to the input resolution using bilinear interpolation. The real-time DIBR technique that we compare against uses the same disparity maps, but it performs a number of processing steps to improve it before it is finally used for view synthesis. To demonstrate the robustness of our technique to low-quality disparity information, we checked how our technique performs if ground truth disparity information is available. Figure 10 presents results of ours vs. DIBR techniques for three images from the Middlebury stereo datasets [Hirschmuller and Scharstein 2007] as differences with respect to known ground-truth target views. Additionally, we computed the results using the same techniques but supplied with ground truth disparity information. This is indicated by "*" next to the method names. The results are compared to original views using the SSIM metric [Wang et al. 2004], and the differences are reported using colormaps. It can be seen that our technique outperforms the DIBR technique, even though it uses improved disparity information. At the same time, our technique provides similar results to the DIBR technique, which relies on ground truth disparity information. Interestingly, our method does not significantly benefit from better disparity information. This demonstrates that our technique can use lower-quality disparity information without overall quality loss and would most likely not benefit from costly depth estimations such as [Zhang et al. 2015]. This is crucial for high-quality view synthesis, as ground truth disparity is usually unavailable. Although our technique performs similarly to DIBR with ground truth disparity, the scenes used in these tests consist mostly of diffuse surfaces without ambiguous depth situations like reflections, depth of field or motion blur. In more difficult cases with complex light effects, depth-of-field, and motion blur effects, our method can recover from a poorly estimated low-resolution initial depth map and provide plausible results (Figure 8).

We have also investigated the quality provided by our technique as a function of time and the magnitude of interpolation and extrapolation that has to be performed. To this end, we used a short, multiview animation which consists of eight views. This gave us four different stereo pairs with different baselines that served as an input to our algorithm for computing missing views. Each synthesized view was later compared to the ground truth using an SSIM metric. Sample results, as well as the plots of SSIM scores, are presented in Figure 11. As expected, the scores decrease with the increasing difference between the input and output baseline. This is expected, as a large difference in baselines results in a more extensive modification of the image content. In such cases, the quality loss is an expected behavior for any synthesis method. For example, when a synthesis from view 1 is considered, the lowest SSIM score is observed for view 4 (the top plot in Figure 11b). Conversely, if the synthesis from view 4 is considered, the quality score is the lowest for view 1 (the bottom plot in Figure 11b). Interestingly, when the difference in baselines is similar, e.g., synthesizing views 1 and 3 from view 2, the quality remains similar regardless of whether the technique performs intra- or extrapolation. Despite the error reported by SSIM metric, both interpolation and extrapolation lead to visually pleasing results (Figure 11c). As can be observed in Figure 11b, the quality

Fig. 11. A comparison to the ground-truth multiview video of "Big Buck Bunny" © by Blender Foundation. a) Various input view pairs tested to evaluate role of input camera baseline. b) The SSIM between our results and the rendered ground truth as a function of time. Each graph shows three plots that correspond to error (SSIM score) generated by synthesizing a particular view (indicated by the color of the plot) from a source view (indicated by the title of the graph). The color coding corresponds to the visualization in a). c) Sample views (indicated by columns) synthesized from different input views (indicated by rows). The orange triangle of the grid (bottom left) demonstrates interpolation from a wider to a narrower camera baseline, and the green triangle (upper right) represents extrapolation from a narrower to a wider camera baseline. The images on the diagonal are the original ground truth views.

of each view is stable across the whole animation, which indicates a good temporal stability of our technique.

7 SUBJECTIVE EVALUATION

To validate our method, we have applied it to several stereoscopic movie sequences together with other competing approaches. We then ran a user study comparing the visual quality of the results.

Fig. 12. Preview of stimuli used in our user study. All copyrights belong to their respective owners. Images and text owned by other copyright holders are used here under the guidelines of the Fair Use provisions of United States copyright law.

Stimuli. 10 short stereoscopic movies with duration ranging from 2 to 5 seconds and with both captured and computer-animated content were used as inputs for all methods (Figure 12). The disparities were linearly remapped to limit the maximum angular disparity with respect to the screen plane to 36 arcmin. Antialiasing was applied to prevent ghosting artifacts due to screen limitations. The same parameters were used to process the videos using our real-time method with a low-resolution disparity estimate (*Ours*), a standard real-time Eulerian method (*PBR*), an offline Lagrangian method utilizing a full resolution disparity estimate (*Offline DIBR*), and a real-time Lagrangian method utilizing bilateral disparity upscaling of a low-resolution disparity estimate (*RT DIBR*).

Task. Participants were presented with pairs of videos with the same sequence processed by two different methods. The sequences were displayed on a custom 8-view automultiscopic display utilizing a parallax barrier on top of a 4K 39″ LCD panel. A dimmed office light was used to avoid any reflections that could interfere with 3D perception. The participants were seated at an optimal distance of 2 meters. A single video was played in a loop at a time. Participants used a keyboard to switch between two versions at will. Participants were suggested to move their head in order to explore the multi-view content. No time limit was applied. Participants used a confirmation button to select the video which provided a better overall image quality. The study consisted of 60 video pairs and on average took 30 minutes. 12 participants naïve to the purpose of the experiment took part in the study. All of them had normal or corrected-to-normal vision and none suffered from stereo blindness.

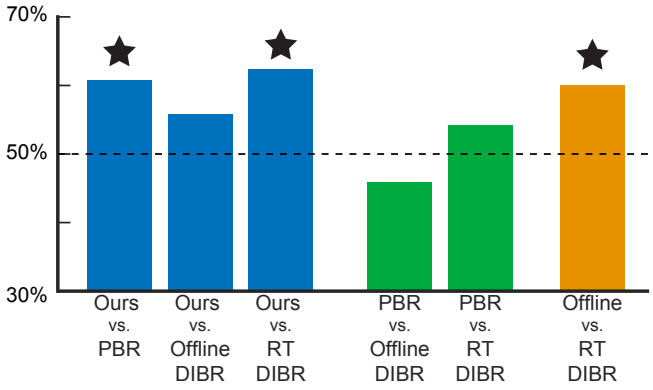

Fig. 13. Results of the user study pairwise comparing our method and other competitors. Values above 50% denote preference of the first named method. Stars mark statistical significance in the binomial test for $p < 0.05$.

Results. The results of the study are summarized in Figure 13. Our method achieved a statistically significant preference (binomial test, $p < 0.05$) over the Eulerian PBR method (60.8%) and the real-time version of the Lagrangian DIBR method (62.5%). There is no effect with respect to the offline DIBR (55.8% preference of our method). This shows that our method is superior to other real-time alternatives. With only a low-resolution disparity map, it achieves a visual quality comparable to that of a method which relies on an offline full-resolution disparity estimation.

The study further showed that the alternative methods are not significantly outperforming each other, as each suffers from a different type of problem. We observe an expected significant preference for the offline DIBR method over its real-time counterpart (60.0%). The relatively small effect size even for this trivial comparison confirms that a detection of visual differences between videos is a difficult task. This further strengthens results achieved by our method.

8 DISCUSSION, LIMITATIONS, AND FUTURE WORK

Similarly to a pure Eulerian approach, our disparity refinement is limited to small disparity errors. Nevertheless, the total wavelet disparity is not. This is because the wavelet disparity also includes the component from the initial disparity. This allows our method to outperform Lagrangian methods in many situations. For example, additional refinement steps performed by DIBR methods lead to cardboard effects. These can be easily handled by our correction (Figure 7, Scene 1, red). In ambiguous cases, such as reflections, motion blur, etc., the initial disparity is usually wrong. However, these phenomena usually correspond to lower luminance frequencies, for which the range of corrections we can perform is sufficiently large (Figure 7, Scene 2 and Scene 3, yellow). Our Eulerian-based correction cannot handle large errors in the initial disparity map. These, however, usually correspond to untextured regions and although we are not able to correct high frequencies in these areas, this does not create severe artifacts, as the corresponding amplitudes are usually low.

There are two aspects that distinguish our approach from other multi-resolution techniques. First, we avoid the notion of searching through a range and taking an optimal value. After the initial disparity is estimated, all refinements are expressed as closed-form expressions on wavelet phases. Second, we do not propagate disparity across frequencies. This independent estimation is the strength of our technique. This is also why our refinement is not a coarse-to-fine method.

The main limitation in our disparity refinement is that it can only refine the disparity using the phase information at the particular wavelet. A large error in the initial disparity maps may lead to incorrect view synthesis and ringing artifacts. Large occlusion regions in the input can also cause significant ringing artifacts in the synthesis view. However, these regions typically have large disparity, and our antialiasing may be able to reduce the artifacts. Another limitation is that our method performs expansion only in the horizontal direction. Although this is sufficient for standard automultiscopic displays, it would be interesting to consider extending the technique to the vertical direction. Currently, our method assumes that input views are already rectified. For future work, it would be interesting to combine a real-time rectification technique with our conversion. We also believe that our approach opens up new ways of improving methods where view synthesis is necessary, e.g., temporal interpolation.

Our technique does not apply any correction for transition artifacts [Du et al. 2014] which lead to a visible ghosting if the viewing location for parallax and lenticular-based automultiscopic screens is not optimal. This effect is purely a display limitation and not a limitation of our technique. The artifacts can be observed in the supplemental materials where a capture of a 4K autostereoscopic screen is shown. In the future, it would be interesting to combine our technique with a technique that compensates for such artifacts, e.g., [Du et al. 2014].

9 CONCLUSIONS

In this paper, we have presented a method that opens the door to practical 3D television systems at home. Our real-time method converts existing stereoscopic content to a high-resolution, high-quality, multi-view format that is suitable for automultiscopic displays. Our approach leverages advantages of both Lagrangian and Eulerian techniques by combining them into one method. This allows us to handle larger disparities than the Eulerian approach can deal with when applied alone, and to resolve difficult cases such as motion blur, depth of focus, and reflections which are challenging for Lagrangian approaches. To this end, we propose to decompose input images to wavelet-like representations where disparity information is estimated for each wavelet separately. This decomposition is later used in our new wavelet-based view synthesis method which computes necessary views for autostereoscopic displays. Additional steps such as inter-view antialiasing or nonlinear disparity manipulations can be easily integrated to provide content customization. Our method operates locally, mostly on 1D scanlines, which allows for an efficient implementation both using a GPU and an FPGA. Our hardware implementation demonstrates that Eulerian techniques and their combination with Lagrangian approaches are good alternatives to hardware solutions that are based on a Lagrangian approach. Our approach opens the door to having 3D television without glasses at home.

REFERENCES

Robert Anderson, David Gallup, Jonathan T. Barron, Janne Kontkanen, Noah Snavely, Carlos Hernández, Sameer Agarwal, and Steven M. Seitz. 2016. Jump: Virtual reality video. *ACM Trans. Graph.* 35, 6, Article 198 (Nov. 2016), 13 pages. https://doi.org/10.1145/2980179.2980257

Myron Z Brown, Darius Burschka, and Gregory D Hager. 2003. Advances in computational stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 25, 8 (2003), 993–1008.

Alexandre Chapiro, Simon Heinzle, Tunç Ozan Aydın, Steven Poulakos, Matthias Zwicker, Aljosa Smolic, and Markus Gross. 2014. Optimizing stereo-to-multiview conversion for autostereoscopic displays. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, 63–72.

Chris Chinnock. 2012. Trends in the 3D TV market. In *Handbook of Visual Display Technology*. Springer, 2599–2606.

Piotr Didyk, Tobias Ritschel, Elmar Eisemann, Karol Myszkowski, Hans-Peter Seidel, and Wojciech Matusik. 2012. A luminance-contrast-aware disparity model and applications. *ACM Trans. Graph.* 31, 6 (2012), 184.

Piotr Didyk, Pitchaya Sitthi-Amorn, William Freeman, Frédo Durand, and Wojciech Matusik. 2013. Joint view expansion and filtering for automultiscopic 3D displays. *ACM Trans. Graph.* 32, 6 (2013), 221.

Song-Pei Du, Piotr Didyk, Frédo Durand, Shi-Min Hu, and Wojciech Matusik. 2014. Improving visual quality of view transitions in automultiscopic displays. *ACM Trans. Graph.* 33, 6 (2014), 192:1–192:9.

Ye Fan, Joshua Litven, David IW Levin, and Dinesh K Pai. 2013. Eulerian-on-Lagrangian simulation. *ACM Trans. Graph.* 32, 3 (2013), 22:1–22:9.

Miquel Farre, Oliver Wang, Manuel Lang, Nikolce Stefanoski, Alexander Hornung, and Aljoscha Smolic. 2011. Automatic content creation for multiview autostereoscopic displays using image domain warping. In *IEEE International Conference on Multimedia and Expo*.

David J Fleet and Allan D Jepson. 1990. Computation of component image velocity from local phase information. *International Journal of Computer Vision* 5, 1 (1990), 77–104.

David J Fleet, Allan D Jepson, and Michael RM Jenkin. 1991. Phase-based disparity measurement. *CVGIP: Image Understanding* 53, 2 (1991), 198–210.

John Flynn, Ivan Neulander, James Philbin, and Noah Snavely. 2015. DeepStereo: Learning to predict new views from the world's imagery. *arXiv preprint arXiv:1506.06825* (2015).

Andrea Fusiello, Emanuele Trucco, and Alessandro Verri. 2000. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications* 12, 1 (2000), 16–22.

Samuel W Hasinoff, Sing Bing Kang, and Richard Szeliski. 2006. Boundary matting for view synthesis. *Computer Vision and Image Understanding* 103, 1 (2006), 22–32.

Heiko Hirschmuller and Daniel Scharstein. 2007. Evaluation of cost functions for stereo matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 1–8.

Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. 2013. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 35, 2 (2013), 504–511.

Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. 2016. Learning-based view synthesis for light field cameras. *ACM Trans. Graph. (Proc. of SIGGRAPH Asia 2016)* 35, 6 (2016).

Johannes Kopf, Fabian Langguth, Daniel Scharstein, Richard Szeliski, and Michael Goesele. 2013. Image-based rendering in the gradient domain. *ACM Trans. Graph.* 32, 6 (2013), 199.

Manuel Lang, Alexander Hornung, Oliver Wang, Steven Poulakos, Aljoscha Smolic, and Markus Gross. 2010. Nonlinear disparity mapping for stereoscopic 3D. *ACM Trans. Graph.* 29, 4 (2010), 75:1–75:10.

Chao-Kang Liao, Hsiu-Chi Yeh, Ke Zhang, Vanmeerbeeck Geert, Tian-Sheuan Chang, and Gauthier Lafruit. 2013. Stereo matching and viewpoint synthesis FPGA implementation. In *3D-TV System with Depth-Image-Based Rendering*. Springer, 69–106.

QH Liu and N Nguyen. 1998. An accurate algorithm for nonuniform fast Fourier transforms (NUFFT's). *IEEE Microwave and Guided Wave Letters* 8, 1 (1998), 18–20.

Lytro Inc. 2015. (January 2015). https://www.lytro.com/.

William R Mark, Leonard McMillan, and Gary Bishop. 1997. Post-rendering 3D warping. In *Proc. of the 1997 Symposium on Interactive 3D Graphics*. ACM, 7–ff.

Belen Masia, Gordon Wetzstein, Carlos Aliaga, Ramesh Raskar, and Diego Gutierrez. 2013. Display adaptive 3D content remapping. *Computers & Graphics, Special Issue on Advanced Displays* 37, 6 (2013), 983–996.

Takuya Matsuo, Norishige Fukushima, and Yutaka Ishibashi. 2013. Weighted joint bilateral filter with slope depth compensation filter for depth map refinement. In *VISAPP (2)*. 300–309.

Wojciech Matusik and Hanspeter Pfister. 2004. 3D TV: A scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *ACM Trans. Graph.* 23, 3 (2004), 814–824.

Lydia MJ Meesters, Wijnand A IJsselsteijn, and Piter JH Seuntiens. 2004. A survey of perceptual evaluations and requirements of three-dimensional TV. *IEEE Trans. on Circuits and Systems for Video Technology* 14, 3 (2004), 381–391.

H Keith Nishihara. 1984. Practical real-time imaging stereo matcher. *Optical Engineering* 23, 5 (1984), 235536–235536.

Karl Pauwels and Marc M Van Hulle. 2008. Realtime phase-based optical flow on the GPU. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*. IEEE, 1–8.

Raytrix GmbH. 2015. (January 2015). http://www.raytrix.de/.

Christian Richardt, Carsten Stoll, Neil A Dodgson, Hans-Peter Seidel, and Christian Theobalt. 2012. Coherent spatiotemporal filtering, upsampling and rendering of RGBZ videos. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 247–256.

Christian Riechert, Frederik Zilly, Peter Kauff, Jens Güther, and Ralf Schäfer. 2012. Fully automatic stereo-to-multiview conversion in autostereoscopic displays. *The Best of IET and IBC* 4, 8 (2012), 14.

Michael Schaffner, Frank Gurkaynak, Pierre Greisen, Hubert Kaeslin, Luca Benini, and Aljosa Smolic. 2015. Hybrid ASIC/FPGA system for fully automatic stereo-to-multiview conversion using IDW. *Circuits and Systems for Video Technology, IEEE Trans. on* (2015).

T. Shibata, J. Kim, D.M. Hoffman, and M.S. Banks. 2011. The zone of comfort: Predicting visual discomfort with stereo displays. *Journal of Vision* 11, 8 (2011), 11:1–11:29.

Eero P Simoncelli and William T Freeman. 1995. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Image Processing, International Conference on*, Vol. 3. IEEE Computer Society, 3444–3444.

Eero P Simoncelli, William T Freeman, Edward H Adelson, and David J Heeger. 1992. Shiftable multiscale transforms. *IEEE Trans. on Information Theory* 38, 2 (1992), 587–607.

Sudipta N Sinha, Drew Steedly, and Richard Szeliski. 2009. Piecewise planar stereo for image-based rendering.. In *ICCV*. 1881–1888.

Aljoscha Smolic, Karsten Muller, Kristina Dix, Philipp Merkle, Peter Kauff, and Thomas Wiegand. 2008. Intermediate view interpolation based on multiview video plus depth for advanced 3D video systems. In *IEEE International Conference on Image Processing*. 2448–2451.

Nikolce Stefanoski, Oliver Wang, Michael Lang, Pierre Greisen, Simon Heinzle, and Aljoscha Smolic. 2013. Automatic view synthesis by image-domain-warping. *Image Processing, IEEE Trans. on* 22, 9 (2013), 3329–3341.

Richard Szeliski, Shai Avidan, and P Anandan. 2000. Layer extraction from multiple images containing reflections and transparency. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, Vol. 1. IEEE, 246–253.

Neal Wadhwa, Michael Rubinstein, Frédo Durand, and William T Freeman. 2013. Phase-based video motion processing. *ACM Trans. Graph.* 32, 4 (2013), 80:1–80:10.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Trans. on* 13, 4 (2004), 600–612.

Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. 2005. High performance imaging using large camera arrays. In *ACM Trans. Graph.*, Vol. 24. ACM, 765–776.

Bennett S Wilburn, Michal Smulski, Hsiao-Heng K Lee, and Mark A Horowitz. 2001. Light field video camera. In *Electronic Imaging 2002*. International Society for Optics and Photonics, 29–36.

Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John V Guttag, Frédo Durand, and William T Freeman. 2012. Eulerian video magnification for revealing subtle changes in the world. (2012).

Zhoutong Zhang, Yebin Liu, and Qionghai Dai. 2015. Light field from micro-baseline image pair. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*. 3800–3809.

Jun Zhou, Yi Xu, and Xiaokang Yang. 2007. Quaternion wavelet phase based stereo matching for uncalibrated images. *Pattern Recognition Letters* 28, 12 (2007), 1509–1522.

C Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. 2004. High-quality video view interpolation using a layered representation. In *ACM Trans. Graph.*, Vol. 23. ACM, 600–608.

Matthias Zwicker, Wojciech Matusik, Frédo Durand, and Hanspeter Pfister. 2006. Antialiasing for automultiscopic 3D displays. In *Proc. of the 17th Eurographics Conference on Rendering Techniques*. Eurographics Association, 73–82.