Nonuniform Spatial Deformation of Light Fields by Locally Linear Transformations

CLEMENS BIRKLBAUER, DAVID C. SCHEDL, and OLIVER BIMBER
Johannes Kepler University Linz

Light-field cameras offer new imaging possibilities compared to conventional digital cameras. However, the additional angular domain of light fields prohibits direct application of frequently used image processing algorithms, such as warping, retargeting, or stitching. We present a general and efficient framework for nonuniform light-field warping, which forms the basis for extending many of these image processing techniques to light fields. It propagates arbitrary spatial deformations defined in one light-field perspective consistently to all other perspectives by means of 4D patch matching instead of relying on explicit depth reconstruction. This allows processing light-field recordings of complex scenes with non-Lambertian properties such as transparency and refraction. We show application examples of our framework in panorama light-field imaging, light-field retargeting, and artistic manipulation of light fields.

Categories and Subject Descriptors: I.4.9 [**Image Processing and Computer Vision**]: Applications

General Terms: Algorithms

Additional Key Words and Phrases: Light field, deformation, retargeting, panorama

ACM Reference Format:

Clemens Birklbauer, David C. Schedl, and Oliver Bimber. 2016. Nonuniform spatial deformation of light fields by locally linear transformations. ACM Trans. Graph. 35, 5, Article 156 (June 2016), 12 pages.
DOI: http://dx.doi.org/10.1145/2928267

1. INTRODUCTION

With the introduction of first commercial and compact plenoptic cameras, capturing light fields has become widely available. Plenoptic imaging enables, for instance, synthetic refocusing, multiperspective recording, and depth-based filtering by providing not only spatial but also angular information. However, the additional angular domain makes it difficult to apply conventional image processing algorithms. They cannot be used directly, because in addition to spatial integrity, angular integrity must be preserved as well. Thus, finding a general framework that consistently deforms light fields is critical to the success of many light-field processing applications.

Existing approaches (e.g., Seitz and Kutulakos [2002] and Wang et al. [2005]) require knowledge of the captured scene's underlying 3D geometry gathered either by automatic depth reconstruction or manual editing. Although this works well for Lambertian scenes, transparent objects or specular effects can cause problems, especially during automatic processing.

We present a nonuniform deformation framework that allows the application of an arbitrary spatial warping function consistently to a light field. In contrast to existing methods, this is done fully automatically. We demonstrate that uniform 2D linear transformations can be applied without violating the angular consistency when they are used for both the spatial and the angular domain. Expressing nonuniform warping functions with locally linear transformations and utilizing 4D patch matching allows transferring arbitrary deformations consistently from one perspective to all others. Our approach does not require explicit depth reconstruction and thus works well for complex non-Lambertian scenes containing reflections and transparent objects, where many dense depth estimation algorithms fail, which results in visual artifacts.

Our algorithm is designed for deforming two-plane light fields captured with regular plenoptic cameras. We do not aim for a physically correct modification of objects in the captured scene, as this would require geometry, material, and lighting to be separated. Our goal is to quickly create plausible and consistent results by directly altering the light-field data.

2. APPLICATIONS

The ability to deform a light field consistently with an arbitrary warping function is the basis for many processing algorithms.

Artistic deformations are the most obvious examples. Figure 1 shows a light field that was modified with a twirl and a fish-eye effect defined in the central perspective and propagated to all other perspectives with our framework.

Another application is light-field retargeting. Image-based methods allow changing the aspect ratio of a picture while preserving the shape of important objects. Most approaches either carve out seams of nonsalient pixels or deform the image with a nonuniform warping grid, as summarized in Shamir and Sorkine [2009]. The latter can be extended to light fields with our framework. The results shown in Figure 2 were gained by applying the image-retargeting approach presented in Zhang et al. [2009] to the central perspective and propagating the resulting warping to all other perspectives with our framework.

This project was funded by the Austrian Science Fund (FWF) under contract P 24907-N23.

Authors' address: C. Birklbauer (corresponding author), D. C. Schedl, and O. Bimber, Johannes Kepler University Linz, Institute of Computer Graphics, Altenberger Strasse 69, 4040 Linz, Austria; emails: firstname.lastname@jku.at.

DOI: http://dx.doi.org/10.1145/2928267

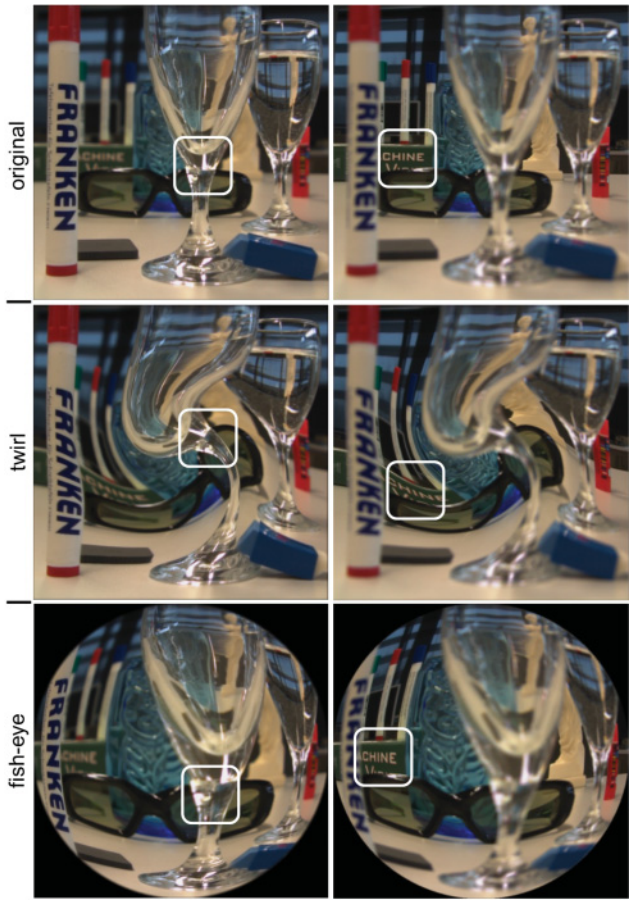

Fig. 1. Artistically deformed light fields processed with our approach. Images were rendered using a wide aperture from light fields with a resolution of 662x662x9x9. Processing time was approximately 4 minutes. The focus is indicated by the white squares.

Fig. 2. Light-field retargeting. The guitars in the original light field are distorted when linearly stretched. Light-field retargeting better preserves salient objects. Processing the light field (output resolution: 1324x662x9x9) took approximately 8 minutes.

The objective of light-field panorama imaging, as a last application example (Figure 3), is to stitch multiple sub-light fields to a single wide field-of-view light field. Previous approaches, such as that in Birklbauer and Bimber [2014], require exact positioning of the camera with a tripod to guarantee overlapping ray spaces. Otherwise, parallax is introduced between the sub-light fields, making registration and stitching difficult. Our framework enables the application of handheld plenoptic cameras and allows stitching of unstructuredly recorded light fields. For this, we correlate all sub-light fields with the image-based registration method described in Zaragoza et al. [2013] being applied to the central perspectives. Propagating these registrations correctly to other corresponding perspectives is then supported by our framework.

Whereas our deformation framework is discussed in Section 4, further details on the implementation of the preceding application examples are presented in Section 5. More results can be found in Section 6 and Figure 16.

3. RELATED WORK

Light-field deformation has been used in previous work mainly in the context of editing and morphing. In Seitz and Kutulakos [2002], a method for plenoptic image editing was presented. As a first step, the light field is decomposed into geometry and radiance information. Deformations in one perspective are applied to the reconstructed geometry and can thereby be propagated to all other perspectives. The editing approach described in Chen et al. [2005] allows deforming a light field by transforming the corners of a bounding box that parameterizes the 4D data, achieving mainly uniform free-form deformations. To warp different parts of the light field independently, it is separated into distinct ray bundles with individual bounding boxes by manual segmentation or special recording setups. The morphing algorithm presented in Wang et al. [2005] also deforms light fields during processing. A user defines 2D polygons in multiple perspectives that approximate the underlying geometry required for propagating the modifications. In contrast to our approach, the nonuniform deformation methods in these examples require either explicit depth reconstruction or time-consuming manual editing. High-quality depth estimation cannot realistically be achieved for complex non-Lambertian scenes. In contrast, a main advantage of plenoptic imaging is the ability to capture effects such as transparency and specularity by a dense angular sampling. Note that the main goal of our method is spatial deformation. Other operations, such as recoloring or depth modifications, which are partially realized in related approaches, are not supported by our algorithm.

Other plenoptic processing approaches describe optimal user interfaces for consistent editing [Jarabo et al. 2014] or the automatic propagation of appearance changes [Jarabo et al. 2011]. However, they cannot be used directly for deformations that require additional mechanisms for handling occlusions and disocclusions. The completion algorithm described in Yatziv et al. [2004] is also related to our method, as it employs 4D patch matching to fill holes in a light field. In contrast, we use this idea to find the correct deformation for each patch in all perspectives.

Surface light fields (e.g., Wood et al. [2000] and Jeong et al. [2003]) and surface reflectance fields (e.g., Weyrich et al. [2005]) are hybrid scene representations between conventional 3D geometry and light fields. They map view-dependent illumination effects directly on a 3D object that can easily be deformed. However, complex acquisition steps are required to gather such scene representations in adequate quality. We process regularly sampled two-plane light fields captured in a matter of milliseconds to minutes with handheld plenoptic cameras.

Fig. 3. Wide aperture renderings of a light-field panorama (2208x637x9x9). Seven individual sub-light fields captured with a handheld Lytro camera (first generation) were stitched in approximately 23 minutes.

One application of our approach is spatial light-field retargeting. A good overview of image retargeting algorithms can be found in Shamir and Sorkine [2009]. As discussed earlier, we use the warping-based technique described in Zhang et al. [2009] as a basis for our method. An alternative approach to light-field retargeting was presented in Birklbauer and Bimber [2012]. To achieve consistent results, seam carving is applied to volumetric scene representations reconstructed for each perspective. In contrast, our method is applied directly to the 4D plenoptic data, avoiding time-consuming and lossy conversions.

Another application is the computation of plenoptic panoramas by stitching multiple two-plane sub-light fields. Again, we build upon image-based methods such as those presented in Zaragoza et al. [2013], Brown and Lowe [2007], and Szeliski [2006]. In contrast to the light-field stitching method described in Birklbauer and Bimber [2014], we do not require precise positioning of the plenoptic camera during recording to achieve overlapping ray spaces, but support unstructured handheld capturing. The technique presented in Birklbauer et al. [2013] could also allow this in principle; however, it converts 4D plenoptic data to 3D focal stacks before stitching and is therefore limited to mainly Lambertian scenes. The method described in Guo et al. [2015] applies 4D graph cut for blending to hide slight misregistrations in case of imprecise positioning but still assumes overlapping ray spaces. In Johannsen et al. [2015], a light-field registration based on structure from motion is described. The approach allows registering light fields with nonoverlapping ray spaces but does not consider suitable stitching and blending methods to handle ghosting artifacts. Finally, in Xue et al. [2014], a theoretical motion model for light fields was discussed that can be used to compute plenoptic panoramas, but no actual registration method was presented, and the motion is assumed to be known in the simulations shown.

4. SPATIAL LIGHT-FIELD DEFORMATION

Our method works with light fields that can be described by a regular two-plane parameterization. This representation is widely used and allows easy processing and rendering due to its uniform structure. One important requirement is to preserve the regularity and simplicity after processing. The deformed light field must therefore have the same parameterization as the original one. As well, no additional information about the applied transformations should be required for further processing or rendering of the resulting data.

We use the analogy of a light field that was captured by a camera array in the following description for better readability. Plenoptic data captured with microlenses in front of the image sensor can easily be converted to this representation. Rays are parameterized by their intersections with two parallel planes: the camera plane ST where the cameras are located, representing the angular domain, and the common image plane UV placed at a fixed distance from ST toward the objects to capture, representing the spatial domain (Figure 4). The indices s, t describe different camera positions on

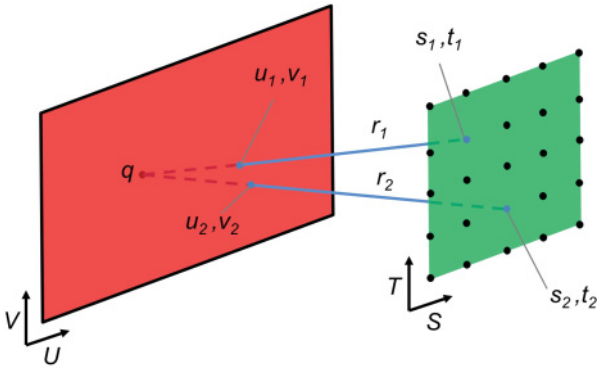

Fig. 4. Two-plane parameterization. Rays are defined by their intersections with two parallel planes: camera plane ST and image plane UV. The disparity of a 3D point q can be computed from two rays r_1 and r_2 intersecting the planes at (s_1, t_1), (u_1, v_1) and (s_2, t_2), (u_2, v_2).

ST, and u, v address pixels in the captured perspective images. We assume the light field to be regularly parameterized, and thus the sampling steps along both the S and T and the U and V domains are equal and axis aligned. This ensures that the disparity of an object relative to one sampling step is the same along the SU and TV domains. Note that the s, t coordinates of the central perspective (i.e., the image captured by the camera in the center of the array) are zero in all of our equations to simplify computations.

4.1 Overview

Our goal is to extend a 2D deformation given in the spatial domain of one reference perspective consistently to other perspectives (i.e., to the angular domain of the light field). Our method does not depend on reconstructed depth information, as this is unreliable for complicated scenes that justify light-field recordings. Instead, we use uniform linear transformations to deform the light field since they can be applied consistently without requiring any information about the underlying 3D geometry, as we show in Section 4.2.

To support more complex nonlinear deformations, we assume that a given warping function can be approximated by locally uniform linear transformations as explained in Section 4.3. Thus, the resulting light field can be computed by piecewise linear transformations (i.e., stitching the required data from different linearly transformed light fields). For clarity, we show a simplified example in Figure 5: a light field of a sinusoid-like surface with varying disparities (different slopes in Figure 5(a)) is piecewise deformed with different linear transformations (indicated by colors in Figure 5(a)). Multiple instances of the light field are deformed with each linear transformation (Figure 5(b)), and the required pieces are combined to compute the result (Figure 5(c)).

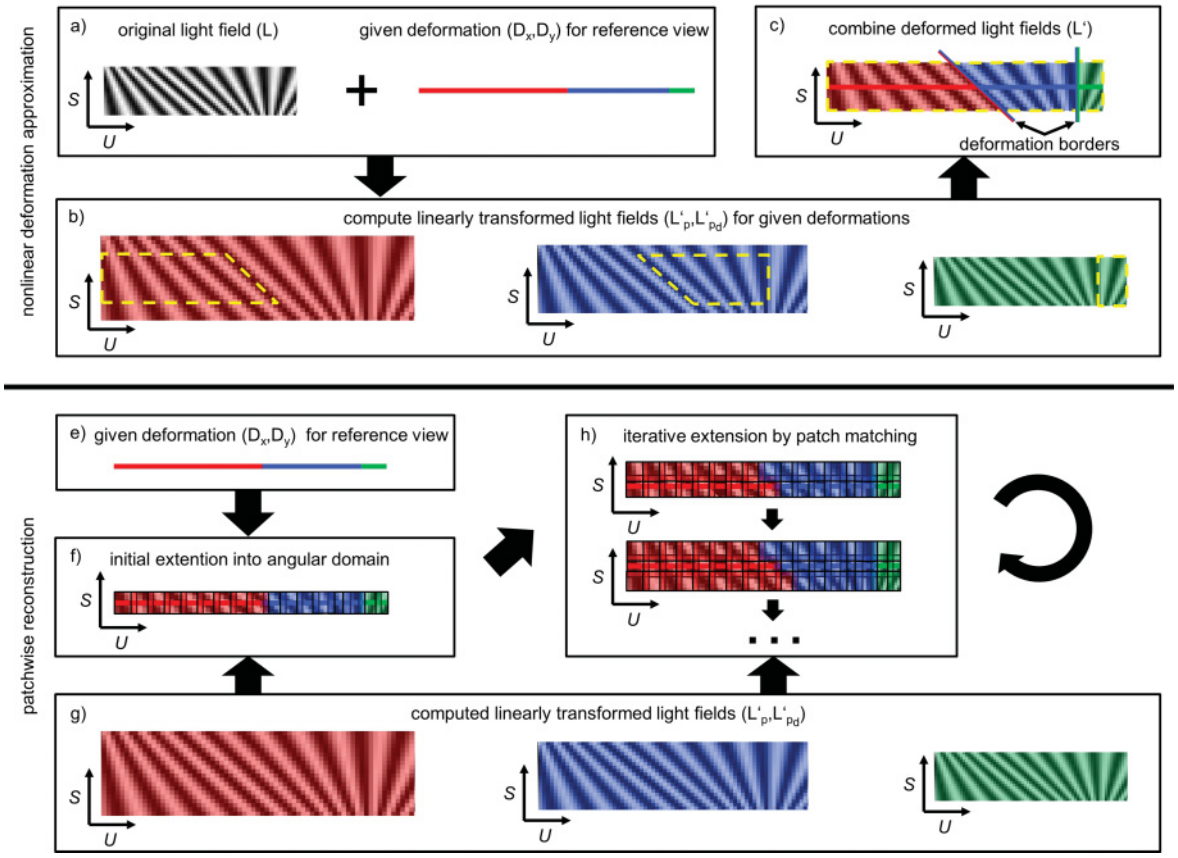

Fig. 5. Overview of our deformation approach. A given nonlinear deformation in the reference perspective is extended to the entire light field by piecewise linear transformations indicated with red, blue, and green (a–c). Since the depth-related spatial-angular slopes at the deformation borders are unknown, reconstruction is achieved by iteratively using patch matching (e–h). Note that rescaling is the only linear transformation that we can show in this illustrative 2D example. Red, blue, and green indicate different degrees of scaling.

A primary reason this approximation is sufficient is that the disparities of objects are preserved when applying uniform linear transformations to their local neighborhoods. This allows merging of differently transformed light fields without noticeable artifacts.

In the idealized example in Figure 5(c), the depth-related spatial-angular slopes at the deformation borders are assumed to be known. Since this is not the case for actual light fields, we approximate them iteratively—starting at perspectives nearest to the reference perspective where deformations are assumed to be linear within a local neighborhood. We compute the deformed nearest neighbor perspectives by stitching together small 4D patches from linearly deformed light fields that are approximating the given local deformations in the reference perspective (Figure 5(f)).

Next, we continue with the computation of light-field perspectives that are farther away from the reference perspective (Figure 5(h)). Similar to before, they are approximated patchwise. However, deformations might differ compared to the reference perspective, as the slopes of their borders are unknown. To approximate them, we use patch matching and test—for each new patch—which linear transformation results in the most consistent light field before stitching. We demand new patches to partially overlap with the already computed light field and ensure consistency by minimizing the difference in those overlapping regions. This process is repeated until the whole light field is computed.

Note that although approximating the deformation border slopes is still depth dependent, our approach has several advantages over direct disparity reconstruction. A primary benefit is that depth is irrelevant for all scene parts that do not map to deformation borders. Furthermore, errors in the slope approximation usually do not cause salient artifacts if the corresponding deformations change slowly. A more detailed evaluation and discussion is presented in Sections 6 and 7.

4.2 Spatial and Angular 2D Linear Transformations

In this section, we show that a 2D linear transformation applied to the spatial and angular domains of a light field does preserve consistency. More specifically, it preserves the disparity of all objects in the deformed light field to ensure correct focus effects when rendering.

Synthetic aperture reconstruction algorithms shift the perspective images in the light field $L(s,t,u,v)$ based on a desired focal disparity d_f and then average them to render the 2D image I focused at d_f. This is explained, for example, in Isaksen et al. [2000]. Our discussion is based on the simplified equation presented in Kubota et al. [2007]:

$$I(x,y) = \sum_{s,t} w(s,t) L(s,t,x - d_f s, y - d_f t). \quad (1)$$

The weight function $w(s,t)$ defines the desired aperture shape and size. A point q at a disparity $d = d_f$ appears sharp in the rendered image I due to the inverse shift by d_f in each perspective. By ensuring that the disparity d of q stays the same in the deformed light field, correct focus effects are achieved when rendering the processed light field.

4.2.1 *Disparity Consistency.* The disparity d of a 3D point q captured by a regular two-plane light field L can be computed from any two rays (r_1 at (s_1, t_1, u_1, v_1), r_2 at (s_2, t_2, u_2, v_2)) intersecting q as follows (Figure 4):

$$d = \frac{u_2 - u_1}{s_2 - s_1} = \frac{v_2 - v_1}{t_2 - t_1} \mid s_1 \neq s_2, t_1 \neq t_2. \tag{2}$$

A 2D linear transformation matrix A is then applied to the spatial and angular domain of the light field for computing the deformed coordinates s', t', u', v':

$$\begin{aligned} A &= \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix}, \\ \begin{bmatrix} u' \\ v' \end{bmatrix} &= A \begin{bmatrix} u \\ v \end{bmatrix}, \\ \begin{bmatrix} s' \\ t' \end{bmatrix} &= A \begin{bmatrix} s \\ t \end{bmatrix}. \end{aligned} \tag{3}$$

Equation (2) is also used to compute the disparity d' from s', t', u', v' in the deformed light field L'. By combining Equations (2) and (3), we can show that $d' = d$ (see Appendix A). Consequently, any 2D linear transformation can be applied without changing the disparity of captured objects.

Note that this does not result in a physically correct deformation of the underlying 3D structure. It rather represents a resampling of the light-field planes that does not affect disparity (and consequently refocusing), as we explained earlier. Note also that other sampling parameters, such as the distance between the UV and ST planes, must not change.

4.2.2 *Linear Light-Field Transformation.* To compute a deformed light field L', we use the inverse matrix A^{-1} of A. It is used to obtain (for any ray r in L' at s'_r, t'_r, u'_r, v'_r) the matching coordinates s_r, t_r, u_r, v_r in the original light field L required for ray lookup:

$$\begin{aligned} \begin{bmatrix} u_r \\ v_r \end{bmatrix} &= A^{-1} \begin{bmatrix} u'_r \\ v'_r \end{bmatrix}, \\ \begin{bmatrix} s_r \\ t_r \end{bmatrix} &= A^{-1} \begin{bmatrix} s'_r \\ t'_r \end{bmatrix}, \\ L'(s'_r, t'_r, u'_r, v'_r) &= L(s_r, t_r, u_r, v_r). \end{aligned} \tag{4}$$

The first two lines of the preceding equation explain how to determine the required look up coordinates s_r, t_r, u_r, v_r. The third line indicates how to obtain the intensity of a ray r in the deformed light field L' from the original light field L.

4.3 Nonuniform Deformation by Locally Linear Transformations

Next, we extend our approach from homogeneous and linear transformations to nonuniform deformations. As input, we need the 2D warping functions $u = D_x(u', v'), v = D_y(u', v')$ that define the desired changes to the central light-field perspective. They relate any coordinate u', v' in the modified image to the corresponding u, v in the original perspective.

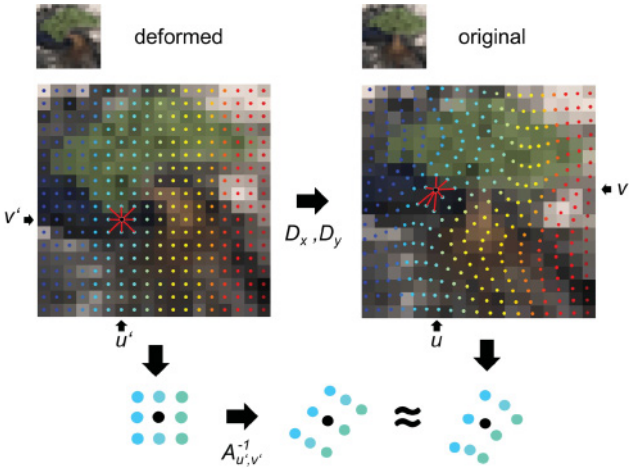

Fig. 6. Locally linear transformations. For any pixel position u', v' in the deformed image (left), the warping functions D_x, D_y define the corresponding coordinates u, v in the original data (right). The matrices $A^{-1}_{u',v'}$ are computed for each u', v' from its neighborhood (red lines). Although they only approximate the local modifications, we need them to consistently extend the changes to the angular domain. 3D scene © Blender Foundation.

4.3.1 *Local Transformation Estimation.* To be able to extend the spatial deformations consistently to the angular domain, we need to approximate the provided warping functions locally by 2D linear transformations. Thus, as a preparation step for the following algorithm, we estimate a linear transformation matrix $A^{-1}_{u',v'}$ for each pixel in the modified central perspective from its local neighborhood $N_{u',v'}$ in the spatial domain. It is computed by minimizing the difference between the approximated and known local deformation for each neighbor $[u'_n, v'_n] \in N_{u',v'}$ (Figure 6). We obtain a solution for $A^{-1}_{u',v'}$ by solving Equation (6) via singular value decomposition:

$$A^{-1}_{u',v'} = \begin{bmatrix} a'_{1,1} & a'_{1,2} \\ a'_{2,1} & a'_{2,2} \end{bmatrix}, \tag{5}$$

$$\begin{aligned} &\begin{bmatrix} u'_{n1} - u' & v'_{n1} - v' & 0 & 0 \\ 0 & 0 & u'_{n1} - u' & v'_{n1} - v' \\ u'_{n2} - u' & v'_{n2} - v' & 0 & 0 \\ 0 & 0 & u'_{n2} - u' & v'_{n2} - v' \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} a'_{1,1} \\ a'_{1,2} \\ a'_{2,1} \\ a'_{2,2} \end{bmatrix} \\ &= \begin{bmatrix} D_x(u'_{n1}, v'_{n1}) - D_x(u', v') \\ D_y(u'_{n1}, v'_{n1}) - D_y(u', v') \\ D_x(u'_{n2}, v'_{n2}) - D_x(u', v') \\ D_y(u'_{n2}, v'_{n2}) - D_y(u', v') \\ \cdots \end{bmatrix}. \end{aligned} \tag{6}$$

4.3.2 *Piecewise Deformation.* The deformed light field L' is then computed by transforming it patchwise, as shown in Figure 7. We proceed iteratively and compute all 4D patches centered at a particular perspective of the light field during each pass before continuing with the next. Parts computed in later iterations must overlap with already existing ones to find their correct deformations, as explained later. Note that also in the angular domain, the estimated linear transformations only approximate the modifications locally. Thus, the patch size generally does not span across all desired perspectives.

First, the patches positioned on the central perspective are computed, where the 2D warping functions and approximated matrices

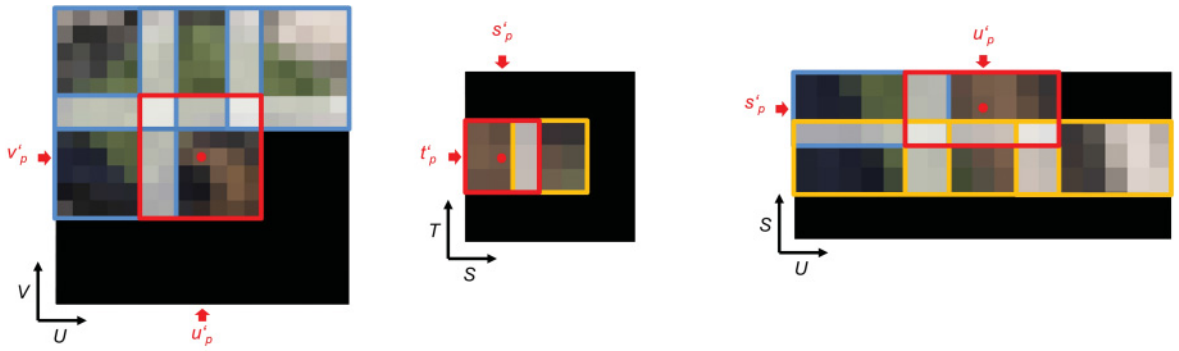

Fig. 7. Example patch distribution on a partially deformed light field shown in the spatial domain (left), the angular domain (center), and a spatial-angular slice along SU (right). The red rectangle indicates the currently processed patch positioned at s'_p, t'_p, u'_p, v'_p. The yellow patches located at the central perspective were computed first. In the second iteration, we estimated the blue/red ones, which should match the patches of previous iterations at overlapping regions (white areas). Black areas were not yet reconstructed. 3D scene © Blender Foundation.

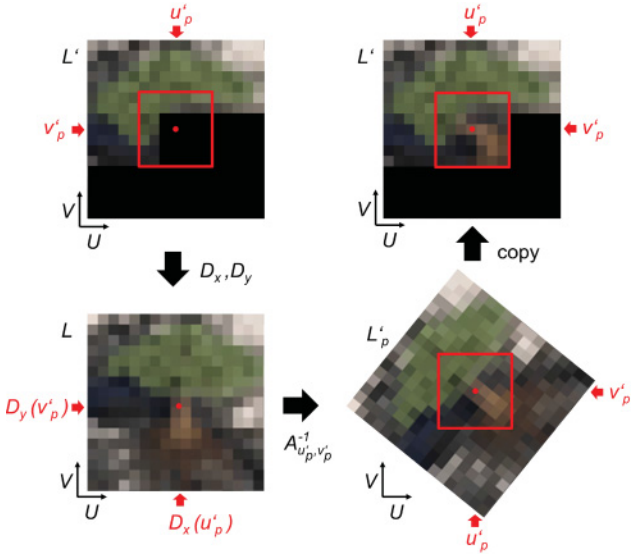

Fig. 8. The desired patch at u'_p, v'_p is copied from the linearly transformed light field L'_p, which is computed by modifying the original data in L with the matrix $A^{-1}_{u'_p,v'_p}$. The deformation origin in L is found with the warping functions D_x, D_y. 3D scene © Blender Foundation.

are known. We extract each patch p from a different linearly transformed light field L'_p. Figure 8 shows a 2D example in the spatial domain. The matrix $A^{-1}_{u'_p,v'_p}$ required to compute L'_p is looked up at the coordinates u'_p, v'_p of the patch center. To allow copying the patch directly from L'_p to L', we make sure that the patch is also centered at u'_p, v'_p in L'_p by adapting Equation (4) as follows:

$$\begin{aligned}\begin{bmatrix} u_r \\ v_r \end{bmatrix} &= \begin{bmatrix} D_x(u'_p) \\ D_y(v'_p) \end{bmatrix} + A^{-1}_{u'_p,v'_p} \begin{bmatrix} u'_r - u'_p \\ v'_r - v'_p \end{bmatrix}, \\ \begin{bmatrix} s_r \\ t_r \end{bmatrix} &= A^{-1}_{u'_p,v'_p} \begin{bmatrix} s'_r \\ t'_r \end{bmatrix}, \\ L'_p(s'_r, t'_r, u'_r, v'_r) &= L(s_r, t_r, u_r, v_r).\end{aligned} \tag{7}$$

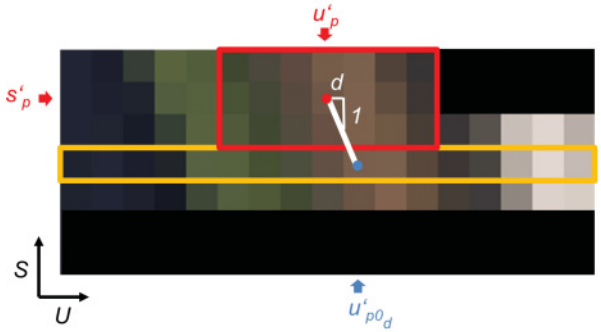

Fig. 9. Computation of the corresponding patch position u'_{p0_d}, v'_{p0_d} in the central perspective (yellow rectangle). The original coordinates u'_p, v'_p are shifted based on the estimated disparity d. This example shows a 2D slice along the SU domain of the light field. 3D scene © Blender Foundation.

Note that the warping functions D_x, D_y in this equation only calculate the spatial offset of the patch to preserve its position in L'_p. They cannot be used to deform the light field, as we cannot apply them to the angular domain.

The required patch can then be copied directly from L'_p to L'. Its size is given by the radii w_s, w_t, w_u, w_v in each light-field dimension, which define the area around the patch center coordinate s'_p, t'_p, u'_p, v'_p to be extracted.

In the following iterations, we compute patches positioned in off-center perspectives. In this case, the warping functions and transformation matrices are unknown, and we must search for the deformation leading to a patch that best matches the light field already processed. To preserve consistency, the same deformation must be applied to the scene part visible in the patch at an off-center perspective as to the corresponding part in the central perspective. We therefore reduce the search space to deformations defined in the central perspective. According to epipolar geometry theory, the matching scene part for the patch positioned at s'_p, t'_p, u'_p, v'_p can only lie along a corresponding epipolar line in the central perspective, which further limits our search range. Coordinates u'_{p0_d}, v'_{p0_d} along this line can be computed with varying disparity d (Figure 9)

as follows:

$$u'_{p0_d} = u'_p - ds'_p,$$
$$v'_{p0_d} = v'_p - dt'_p. \tag{8}$$

The linearly deformed light field L'_{p_d} for the patch is then computed in a way similar to that in Equation (7) by using u'_{p0_d}, v'_{p0_d} instead of u'_p, v'_p.

4.3.3 *Optimal Patch Transformation.* So far, we have explained how to compute a patch with a given disparity. Next, we show how to find the optimal d that leads to the best transformation. As mentioned earlier, we require the patches to overlap in the spatial and angular domains (Figure 7). This makes it possible to compare different patch candidates (computed with varying disparities d) to the existing patches and to search for the best match for preserving consistency.

For comparison, we compute the sum of intensity differences over all overlapping rays between the already computed light field L' and the patch candidate L'_{p_d}. We use the L_1 norm ($|\cdot|_1$) to determine the absolute difference between a ray in L' and L'_{p_d} as follows:

$$D_{exist}(i,j,k,l) = \begin{cases} |L'(i,j,k,l) - L'_{p_d}(i,j,k,l)|_1, & \text{if } L'(i,j,k,l) \text{ exists} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Note that we default the function to 0 for any ray in L' that was not computed in a previous iteration. Summing up the differences for all rays belonging to the patch results in the error value E_{exist}:

$$E_{exist} = \sum_{o_s=-w_s}^{w_s} \sum_{o_t=-w_t}^{w_t} \sum_{o_u=-w_u}^{w_u} \sum_{o_v=-w_v}^{w_v} D_{exist}(s'_p + o_s, t'_p + o_t, u'_p + o_u, v'_p + o_v). \tag{10}$$

It is not always guaranteed to find a perfectly matching patch in this process. Small errors can quickly accumulate toward outer perspectives and result in noticeable artifacts and wrong deformations. To prevent this, we not only compare the patch candidate to its local neighborhood but also compare the angular center slice of a candidate with the ground truth central perspective. The corresponding rays are computed based on the disparity d of the candidate, resulting in the following difference function:

$$D_{center}(k,l) = |L'(0,0,k - ds'_p, l - dt'_p) - L'_{p_d}(s'_p, t'_p, k, l)|_1. \tag{11}$$

The difference of all rays belonging to the central slice of the patch is then accumulated to compute the error value E_{center}:

$$E_{center} = \sum_{o_u=-w_u}^{w_u} \sum_{o_v=-w_v}^{w_v} \left(D_{center}(u'_p + o_u, v'_p + o_v)\right). \tag{12}$$

Both error values are finally combined to compute E_d, which our algorithm minimizes for a patch over different disparities d (between a given d_{min} and d_{max} using a step width d_{step}):

$$E_d = \frac{\alpha E_{exist}}{(2\,w_s + 1)(2\,w_t + 1)} + \beta E_{center}. \tag{13}$$

We weight the error values by the scene-dependent factors α and β. Note that the comparison to the central perspective is only valid for mainly Lambertian surfaces. Therefore, we usually weight β down to avoid problems with specular and transparent scene parts and occlusion boundaries. To compensate for the influence of the angular patch size, which does not change E_{center}, the error value E_{exist} is furthermore normalized by the angular extent.

4.3.4 *Algorithmic Details.* Number and resolution of angular samples in the deformed light field should match the original ones. Thus, the number of required iterations depends on the size and overlap of the patches in the angular domain.

The order according to which the patches are determined during each iteration in the spatial domain also influences the result slightly. Since patch comparisons in completely uniform light-field parts can easily lead to ambiguous disparity estimations, we first compute patches with a high image variance in their angular center slice. In the first iteration, the order is irrelevant, as the deformations are given. In all other iterations, we guarantee that the angular center slice of a patch was already computed in a previous iteration by choosing a large overlap of the patches in the angular domain.

To smoothly blend overlapping patches in the spatial domain, we use a linear function that weights pixels based on their distances to the patch center. In the angular domain, they are not blended. When a new patch is inserted, we simply do not overwrite existing ones that have been computed in a previous iteration. Thereby, we prefer patches generated in iterations closer to the reference perspective because they suffer less from error accumulation.

An additional advantage of estimating the disparity d is that it can be used to improve the interpolation of color values queried from L between discrete light-field perspectives during patch extraction. Depending on the deformation, it may also happen that rays from perspectives that lie outside the angular range of L are required. In this case, the disparity allows us to compute a corresponding ray within L and thus to extrapolate the required data. Depth-assisted light-field lookups are explained, for example, in Gortler et al. [1996].

5. IMPLEMENTATION

We implemented our framework in Matlab but used C++ and CUDA to speed up performance-relevant parts. Our tests were conducted on a PC with an i5-650 CPU (3.2Ghz) and an Nvidia GTX 580 graphics card. The computation time mainly depends on the size of the 4D data and the chosen patch extent and step width. Processing a typical light field captured with a first-generation Lytro camera (output resolution: 662x662x9x9; patch size: 5x5x3x3; step width: 2x2x1x1) takes approximately 4 minutes. Note that all durations stated in this article do not include the time required to convert the raw plenoptic data to a Matlab array.

Unless stated otherwise, we used the following parameters in our experiments: $d_{min} = -3, d_{max} = 3, d_{step} = 0.05, w_s = 1$ or $2, w_t = 1$ or $2, w_u = 2, w_v = 2, \alpha = 1, \beta = 0.1$ to 0.5. Furthermore, we chose the patch step width to be the same as the radius. More details regarding the influence of the different parameters to the performance and quality of the results can be found in Section 6.

The application examples described in Section 2 are all based on existing image processing techniques that define the warping functions D_x, D_y required for our method. Thus, the implementation of artistic effects and retargeting is straightforward.

For panorama stitching, we apply our deformation framework to warp one sub-light field to a second one according to the registration of the corresponding central perspectives computed with the method described in Zaragoza et al. [2013]. Similar to the approach presented in Agarwala et al. [2004], we then blend the sub-light fields along seams that are computed for the central perspectives by applying graph cut. Thereby, we utilize the same penalty

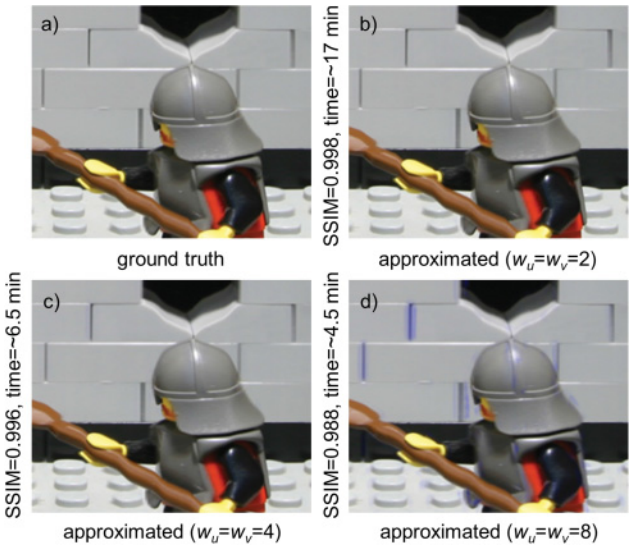

Fig. 10. Comparison of the given ground truth deformation in the central light-field perspective (a) to the patchwise linear approximation (b–d) computed with our approach using different spatial patch radii w_u and w_v. A too large patch size leads to noticeable differences, indicated by a blue overlay in the presented close-ups. It is computed using the SSIM. Light field courtesy of Stanford Computer Graphics Laboratory.

function as described in Agarwala et al. [2004] but consider the average difference of all corresponding light-field perspectives (resulting in a single 2D difference image) as interaction penalty. The same seams are applied to all other perspectives, and multiband blending [Brown and Lowe 2007] is finally used for combining them. The whole process is applied iteratively to stitch more than two sub-light fields.

6. EVALUATION

In this section, we show the impact of the different parameters on performance and reconstruction results. Furthermore, we compare our approach to two alternatives for light-field deformation.

6.1 Impact of Parameter Settings

To show the influence of different parameters on the reconstruction result, we deform an exemplary light field with one of the most challenging deformations for our approach: a rapidly changing transformation that quickly alternates between weak and strong horizontal stretches. Note that the given timings indicate the processing duration for deforming the whole light field with an output resolution of 768x512x17x17. If not stated otherwise, the default parameters discussed in Section 5 are used (choosing $w_s = w_t = 1$ and $\beta = 0.5$).

First, we evaluate the influence of the spatial patch radii w_u and w_v when approximating the given ground truth deformation in the central perspective. A larger spatial patch size leads to a faster reconstruction but to noticeable differences. We present this in Figure 10 by computing the structural similarity index (SSIM) [Wang et al. 2004] between the fully reconstructed and the ground truth perspective. The difference map is overlaid in blue.

Reconstruction in the angular domain is strongly influenced by the patch radii w_s and w_t. Again, a larger patch radius leads to a faster reconstruction but does not allow precise reconstruction of the slope of the deformation borders. This leads to inconsistencies in the light field as shown in Figure 11(b) through (d). The result computed with optimal parameters is shown in Figure 11(a) for comparison. Reconstruction in a single iteration with a patch size spanning the whole angular extent of the light field introduces severe inconsistencies and showcases the importance of using iterative patch matching (Figure 11(d)).

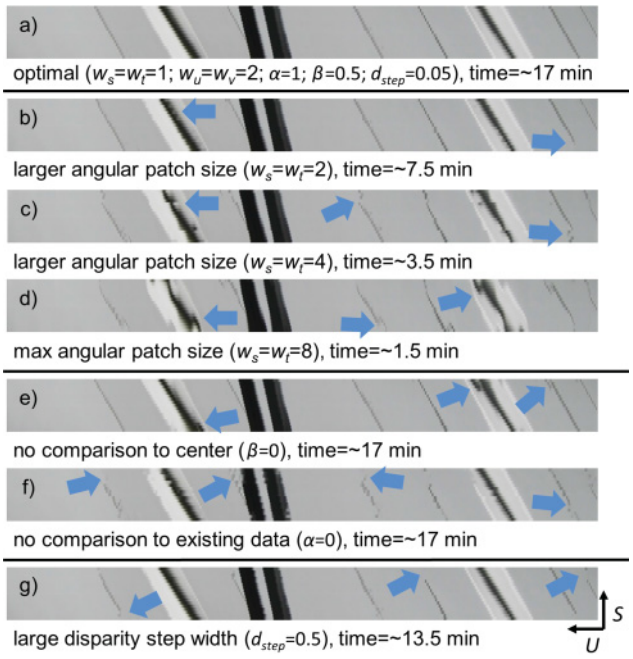

Fig. 11. Comparison of the impact of different parameters on the angular consistency. The presented spatial-angular slices along the SU domain are computed using the optimal set of parameters (a), varying angular patch sizes (b–d), varying energy weights (e, f), and a large disparity step width (g). Examples for angular inconsistencies are indicated with the blue arrows. Light field courtesy of Stanford Computer Graphics Laboratory.

Furthermore, we like to show the impact of the error terms E_{center} (Equation (12)) and E_{exist} (Equation (10)). Figure 11(e) presents the error accumulation in outer perspectives that can arise when disabling E_{center} by setting $\beta = 0$. Similarly, E_{center} alone ($\alpha = 0$) does not preserve the consistency in the light field, as shown in Figure 11(f).

Finally, we discuss the sampling of the disparity range when minimizing E_d (Equation (13)). A high step width $d_{step} = 0.5$ leads to problems in outer perspectives due to too coarse disparity estimation (Figure 11(g)). The required min and max disparities (d_{min},d_{max}) are light field dependent. Our experiments indicate that a significantly larger disparity range or angular extent (exceeding the ones shown in our examples) might degrade quality. We cannot specify strict limits since the optimal parameterization and the success of our method in general also depend on the scene content and the desired deformation. Limitations in this regard are discussed in Section 7.

6.2 Alternative Deformation Approaches

Next, we compare our approach to two alternatives for light-field deformation.

A naive approach would be to simply apply the deformation defined for the central perspective to all other perspectives, ignoring

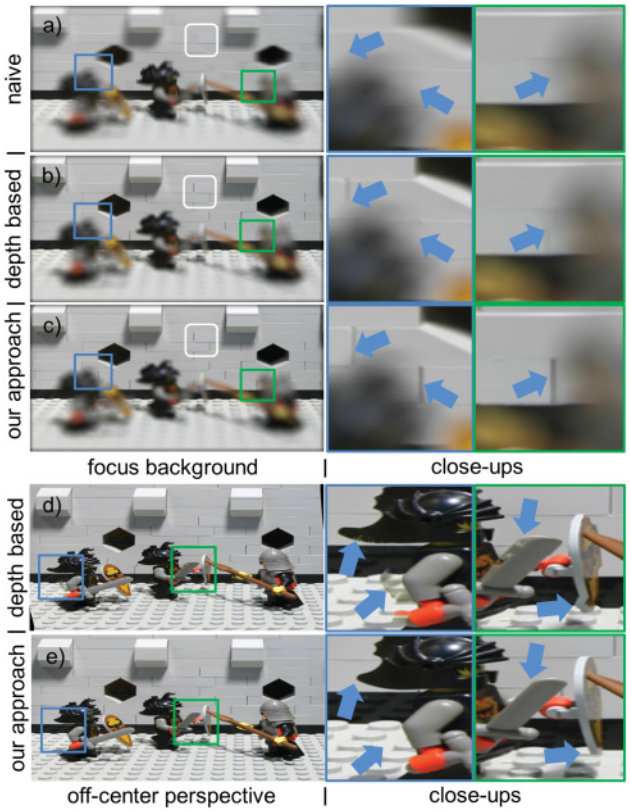

Fig. 12. Comparison of our method with a naive and a depth-based deformation approach. Wide aperture renderings (a, b) show focus problems due to angular inconsistencies, although focusing is still possible when the light field has been deformed by our approach (c). The depth-based method additionally suffers from artifacts caused by unknown deformations or disocclusions in off-center perspectives (d). Light field courtesy of Stanford Computer Graphics Laboratory.

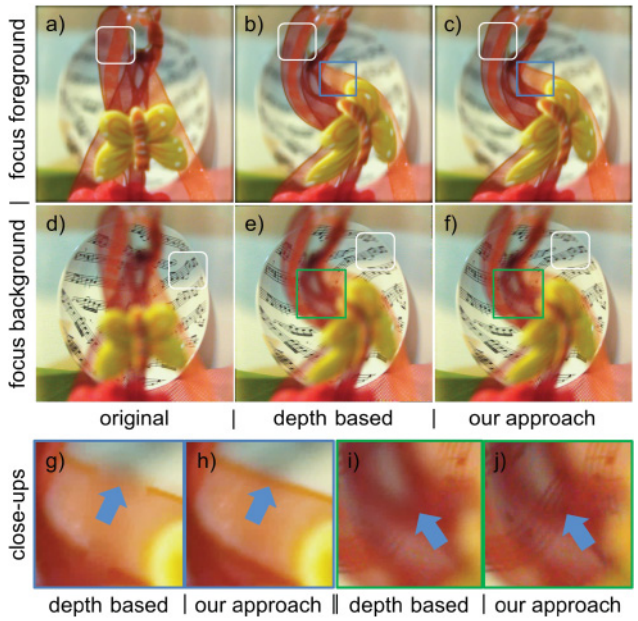

Fig. 13. Comparison of our method with a depth-based deformation approach in the case of transparency. The original light field is deformed with a twirl effect. The wide aperture renderings show focus problems due to angular inconsistencies, while there are no problems when the light field has been deformed with our approach. The blue and green rectangles highlight the areas shown in the close-ups.

the angular domain. However, as shown in Figure 12(a), even a primitive linear stretching transformation will corrupt the angular consistency and thereby prevent correct refocusing when the light field is rendered with a wide aperture.

An alternative would be to reconstruct depth and use this to transfer the deformations defined in the central perspective to all other perspectives. The modifications are given as offsets between the undeformed and deformed coordinates of a pixel and are propagated to different perspectives by shifting the displacements with the estimated disparities. Shifts are computed in a way similar to that in Equation (8), and we use the method described in Jeon et al. [2015] to reconstruct a depth map for the central perspective. Note that the offsets are not constant but vary even if a uniform transformation is applied. Therefore, correct disparity information is also required in such a simple case. This approach leads to better results compared to the naive one, but it has two main drawbacks.

First, deformations are defined only in one reference image, and thus they are unknown for objects that are occluded in this perspective but visible in others. Further, although the offsets change smoothly in the central perspective, this is not the case in the other perspectives after shifting. Rapid changes in the displacements lead to holes in the processed perspectives. We show the resulting visual artifacts in Figure 12(d). More advanced propagation and inpainting algorithms, as used in Seitz and Kutulakos [2002], can improve such results but require correct depth values for all rays in the light field.

The second problem is that incorrect depth information leads to angular inconsistencies and consequently to focusing issues during rendering, as shown in Figure 12(b). Depth reconstruction algorithms often fail for non-Lambertian scenes, especially in the case of transparency. Such scene parts will result in two or more conflicting depth values. This causes problems for either the foreground or the background, as shown in Figure 13(g) and (i).

Our approach also implicitly computes a disparity value for each patch, which is used to find the best deformation along a limited epipolar line in the central perspective, as explained in Section 4.3.2. We assume that the local transformations only change marginally along this line. If this is the case, different disparities lead to very similar deformations, and consequently, conflicting depth values do not cause salient artifacts. Other challenging situations for depth estimation, such as occlusion boundaries, are handled in the same way.

Our algorithm relies only on correct disparities for interpolation and extrapolation. Incorrect values can lead to local inconsistencies. However, they are generally not noticeable during rendering for moderate deformations, and the overall consistency is much better preserved than with the depth-based approach, as shown in Figure 13(h) and (j) and the supplementary video.

Interpolation is less error prone than extrapolation. The latter is required, for example, in the case of compressive deformations. As explained in Section 4.2, such transformations also condense the angular domain and thus cause missing perspectives when reconstruction of the complete angular extent of the original light field is attempted (Figure 14). By applying an additional uniform stretching

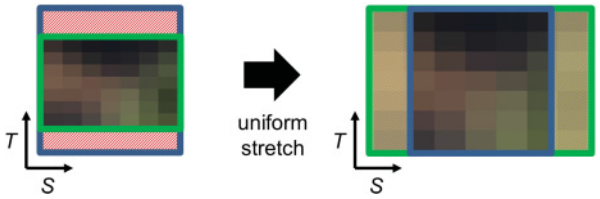

Fig. 14. Avoiding extrapolation. The green rectangle on the left represents the valid angular domain resulting from a compressive transformation. When reconstructing the full angular extent of the original light field, indicated by the blue square, all perspectives in the red area must be extrapolated. By applying an additional uniform stretching operation, as on the right, this can be avoided. The orange area indicates unused angular samples in this case. 3D scene © Blender Foundation.

operation to all domains of the 4D data, we can avoid extrapolation. Although this leads to unused perspectives and requires more interpolation, it causes fewer artifacts.

7. LIMITATIONS

One limitation of our approach results from the previously mentioned assumption that the local transformations change only moderately in the spatial domain. Rapid changes can lead to conflicting deformations for a patch, especially at occlusion boundaries and transparent objects. Although our technique is in many cases more tolerant than a depth-based algorithm, this can cause visual artifacts in the light field, as shown in Figure 15(a) and (b).

Furthermore, extreme transformations in the angular domain will require intensive interpolations and extrapolations to reconstruct the same angular extent and resolution as in the original light field. As explained earlier, these operations depend on correct disparities and may thus lead to artifacts.

The potential mix of patches from different perspectives in the case of nonuniform deformations means that we do not preserve the original appearance of view-dependent effects, such as specular highlights, in each perspective. However, we believe that this will rarely result in disturbing artifacts, as such effects are often small and change smoothly across neighboring perspectives. It can be difficult to judge their correctness for a human observer. To emphasize the problem, the example shown in Figure 15(c) and (d) was generated by deforming a highly specular light field with a rapidly changing deformation.

Limitations also exist for our panorama stitching application. Due to the handheld capturing nature, the ray spaces of the recorded sub-light fields do not overlap. This makes the computation of physically correct panoramas impossible (as shown in Birklbauer and Bimber [2014]) and requires several assumptions to be made for supporting stitching. Since our deformation method does not allow changing the disparity of captured scene points, we require them to remain constant over all sub-light fields. However, this assumption is violated in cases where the camera is moved in the direction of the optical axis. Although this did not cause salient artifacts with the small baseline light fields used in our experiments, it might be a problem if the disparity resolution is increased for a wider baseline. In case of camera rotations, another cause for mismatching disparities is that the planes of different sub-light fields do not remain parallel. This might lead to problems when stitching recordings captured with a wider field of view. Finally, our method is only suitable for increasing the field of view (spatial domain) of the light field but does not allow extension of the aperture (angular domain), as shown in Guo et al. [2015].

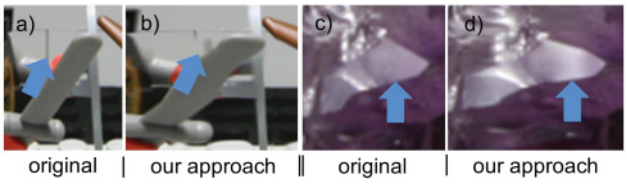

Fig. 15. Artifacts in the case of rapidly changing deformations. The close-ups on the left show inconsistencies at occlusion boundaries when comparing the original and the processed light-field perspective. The close-ups on the right show additional intensity variations at specular highlights in the processed perspective when comparing it to the original. In both cases, a rapidly changing transformation that quickly alternates between weak and strong horizontal stretches was applied. Light fields courtesy of Stanford Computer Graphics Laboratory.

8. CONCLUSION AND FUTURE WORK

We have presented an efficient and general deformation framework that applies arbitrary transformations defined in one 2D perspective to the whole 4D light field. Our approach does not rely on explicit depth reconstruction but is based on linear transformations that do not influence the angular consistency when applied to the spatial and angular domains. By deforming the light field patchwise, we achieve nonuniform deformations. Our approach can be used in a wide range of processing algorithms. Possible applications of our framework include artistic deformations, light-field retargeting, and the generation of 4D panoramas captured with a handheld plenoptic camera.

If perfect depth reconstruction is possible, methods as explained in Seitz and Kutulakos [2002] can produce comparable or even better results. However, we believe that 4D light-field sampling is not necessary in this case, as all scene information can sufficiently be represented in 3D. For common anisotropic or specular reflections, scattering, transparency, and other complex light transport a perfect depth reconstruction is difficult to achieve.

Our algorithm is mainly targeted at densely and regularly sampled two-plane light fields with limited disparity range and angular extent. A low sampling rate in the angular domain would make interpolation artifacts caused by incorrect depth values more visible. Furthermore, excessive disparity and angular extent might increase error accumulation toward outer perspectives. For performance reasons, we used an iterative extension strategy in our patch matching algorithm. In the future, we will investigate global optimization approaches to further improve consistency and reduce error accumulation toward outer perspectives.

We additionally plan to revise our retargeting and panorama examples. A 4D graph cut approach, as used in Guo et al. [2015], would improve light-field stitching. Furthermore, creating 360-degree panoramas and using bundle adjustment for registering multiple light fields, as done for images [Zaragoza et al. 2014], will be part of future work.

APPENDIX

A. PROOF OF DISPARITY CONSISTENCY

This appendix proofs the disparity consistency explained in Section 4.2.1. By combining Equations (2) and (3), we can prove that $d = d'$. This is demonstrated in the following equation by a series of equivalent transformations. Note that we show this exemplarily for the SU domain, assuming $s_1 \neq s_2, t_1 \neq t_2, u_1 \neq u_2$ and $v_1 \neq v_2$.

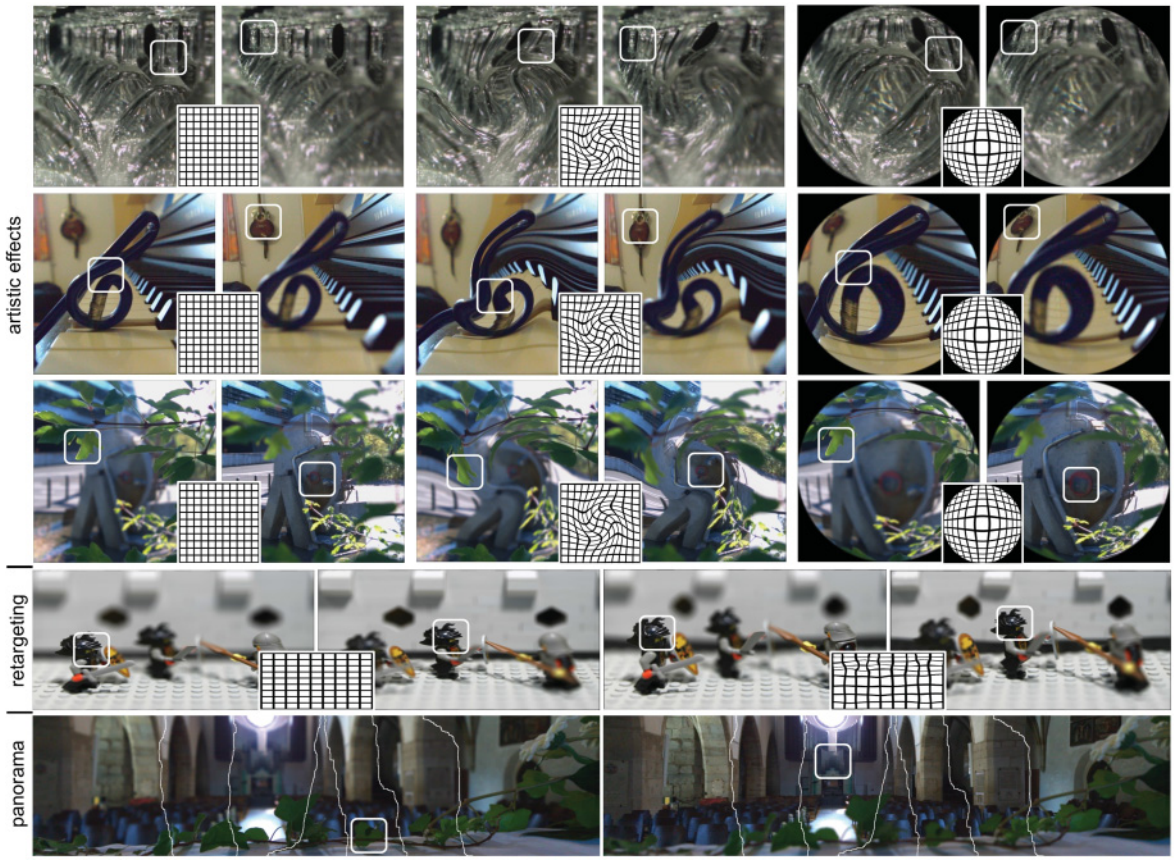

Fig. 16. Light fields deformed with the presented approach showing three different applications (from top to bottom): artistic effects, retargeting, and panorama stitching. They were rendered using a wide aperture. Lego knights light field courtesy of Stanford Computer Graphics Laboratory.

All other cases can be proven similarly.

$$
\begin{aligned}
d' &= \frac{u_2' - u_1'}{s_2' - s_1'}, \text{(Eq. (2))} \\
&= \frac{a_{1,1}u_2 + a_{1,2}v_2 - a_{1,1}u_1 - a_{1,2}v_1}{a_{1,1}s_2 + a_{1,2}t_2 - a_{1,1}s_1 - a_{1,2}t_1}, \text{(Eq. (3))} \\
&= \frac{a_{1,1}(u_2 - u_1) + a_{1,2}(v_2 - v_1)}{a_{1,1}(s_2 - s_1) + a_{1,2}(t_2 - t_1)}, \\
&= \frac{\left(a_{1,1} + a_{1,2}\frac{v_2 - v_1}{u_2 - u_1}\right)(u_2 - u_1)}{\left(a_{1,1} + a_{1,2}\frac{t_2 - t_1}{s_2 - s_1}\right)(s_2 - s_1)}, \\
&= \frac{\left(a_{1,1} + a_{1,2}\frac{v_2 - v_1}{u_2 - u_1}\right)(u_2 - u_1)}{\left(a_{1,1} + a_{1,2}\frac{v_2 - v_1}{u_2 - u_1}\right)(s_2 - s_1)}, \text{(Eq. (2))} \\
&= \frac{u_2 - u_1}{s_2 - s_1} = d. \text{(Eq. (2))}
\end{aligned}
\tag{14}
$$

ACKNOWLEDGMENTS

We thank the Stanford Computer Graphics Laboratory for providing some of the light fields used in this article and the supplementary video. The 3D scene used in Figures 6, 7, 8, 9, and 14, and the supplementary video, is licensed under the creative commons attribution 3.0 license (http://creativecommons.org/licenses/by/3.0/) with © copyright Blender Foundation | www.sintel.org.

REFERENCES

Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen. 2004. Interactive digital photomontage. *ACM Transactions on Graphics* 23, 3, 294–302. DOI:http://dx.doi.org/10.1145/1015706.1015718

Clemens Birklbauer and Oliver Bimber. 2012. Light-field retargeting. *Computer Graphics Forum* 31, 2, 295–303. DOI:http://dx.doi.org/10.1111/j.1467-8659.2012.03008.x

Clemens Birklbauer and Oliver Bimber. 2014. Panorama light-field imaging. *Computer Graphics Forum* 33, 2, 43–52. DOI:http://dx.doi.org/10.1111/cgf.12289

Clemens Birklbauer, Simon Opelt, and Oliver Bimber. 2013. Rendering gigaray light fields. *Computer Graphics Forum* 32, 2, 469–478. DOI:http://dx.doi.org/10.1111/cgf.12067

Matthew Brown and David G. Lowe. 2007. Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision* 74, 1, 59–73. DOI:http://dx.doi.org/10.1007/s11263-006-0002-3

Billy Chen, Eyal Ofek, Heung-Yeung Shum, and Marc Levoy. 2005. Interactive deformation of light fields. In *Proceedings on the Symposium on Interactive 3D Graphics and Games*. 139–146. DOI:http://dx.doi.org/10.1145/1053427.1053450

Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. 1996. The lumigraph. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'96)*. 43–54. DOI:http://dx.doi.org/10.1145/237170.237200

X. Guo, Z. Yu, S. Kang, H. Lin, and J. Yu. 2015. Enhancing light fields through ray-space stitching. *IEEE Transactions on Visualization and Computer Graphics* PP, 99, 1. DOI:http://dx.doi.org/10.1109/TVCG.2015.2476805

Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. 2000. Dynamically reparameterized light fields. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'00)*. 297–306. DOI:http://dx.doi.org/10.1145/344779.344929

Adrian Jarabo, Belen Masia, Adrien Bousseau, Fabio Pellacini, and Diego Gutierrez. 2014. How do people edit light fields? *ACM Transactions on Graphics* 33, 4, 146:1–146:10. DOI:http://dx.doi.org/10.1145/2601097.2601125

Adrian Jarabo, Belen Masia, and Diego Gutierrez. 2011. Efficient propagation of light field edits. In *Proceedings of the Ibero-American Symposium on Computer Graphics (SIACG'11)*. 75–80.

Hae-Gon Jeon, Jaesik Park, Gyeongmin Choe, Jinsun Park, Yunsu Bok, Yu-Wing Tai, and In So Kweon. 2015. Accurate depth map estimation from a lenslet light field camera. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*. 1547–1555. DOI:http://dx.doi.org/10.1109/CVPR.2015.7298762

Eunhee Jeong, Mincheol Yoon, Seungyong Lee, Minsu Ahn, Seungyong Lee, and Baining Guo. 2003. Feature-based surface light field morphing. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*. 215–223. DOI:http://dx.doi.org/10.1109/PCCGA.2003.1238263

O. Johannsen, A. Sulc, and B. Goldluecke. 2015. On linear structure from motion for light field cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'15)*. 720–728. DOI:http://dx.doi.org/10.1109/ICCV.2015.89

Akira Kubota, Kazuya Kodama, and Yoshinori Hatori. 2007. Image-based refocusing by 3D filtering. In *Proceedings of the Pacific Rim Symposium on Image and Video Technology*. 385–398. DOI:http://dx.doi.org/10.1007/978-3-540-77129-6_35

Steven M. Seitz and Kiriakos N. Kutulakos. 2002. Plenoptic image editing. *International Journal of Computer Vision* 48, 2, 115–129. DOI:http://dx.doi.org/10.1023/A:1016046923611.

Ariel Shamir and Olga Sorkine. 2009. Visual media retargeting. In *Proceedings of the ACM SIGGRAPH ASIA 2009 Courses*. Article No. 11. DOI:http://dx.doi.org/10.1145/1665817.1665828

Richard Szeliski. 2006. Image alignment and stitching: A tutorial. *Foundations and Trends in Computer Graphics and Vision* 2, 1, 1–104.

Lifeng Wang, Stephen Lin, Seungyong Lee, Baining Guo, and Heung-Yeung Shum. 2005. Light field morphing using 2D features. *IEEE Transactions on Visualization and Computer Graphics* 11, 1, 25–34. DOI:http://dx.doi.org/10.1109/TVCG.2005.11

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4, 600–612. DOI:http://dx.doi.org/10.1109/TIP.2003.819861

Tim Weyrich, Hanspeter Pfister, and Markus H. Gross. 2005. Rendering deformable surface reflectance fields. *IEEE Transactions on Visualization and Computer Graphics* 11, 1, 48–58. DOI:http://dx.doi.org/10.1109/TVCG.2005.14

Daniel N. Wood, Daniel I. Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H. Salesin, and Werner Stuetzle. 2000. Surface light fields for 3D photography. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'00)*. 287–296. DOI:http://dx.doi.org/10.1145/344779.344925

Zhou Xue, Loic Baboulaz, Paolo Prandoni, and Martin Vetterli. 2014. Light field panorama by a plenoptic camera. In *Proceedings of SPIE 9020: Computational Imaging XII*. 90200S–90200S–11. DOI:http://dx.doi.org/10.1117/12.2038254

L. Yatziv, G. Sapiro, and M. Levoy. 2004. Lightfield completion. In *Proceedings of the 2004 International Conference on Image Processing (ICIP'04)*, Vol. 3. 1787–1790. DOI:http://dx.doi.org/10.1109/ICIP.2004.1421421

J. Zaragoza, T. J. Chin, Q. Tran, M. S. Brown, and D. Suter. 2014. As-projective-as-possible image stitching with moving DLT. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7, 1285–1298. DOI:http://dx.doi.org/10.1109/TPAMI.2013.247

Julio Zaragoza, Tat-Jun Chin, Michael S. Brown, and David Suter. 2013. As-projective-as-possible image stitching with moving DLT. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'13)*. 2339–2346. DOI:http://dx.doi.org/10.1109/CVPR.2013.303

Guo-Xin Zhang, Ming-Ming Cheng, Shi-Min Hu, and Ralph R. Martin. 2009. A shape-preserving approach to image resizing. *Computer Graphics Forum* 28, 7, 1897–1906. DOI:http://dx.doi.org/10.1111/j.1467-8659.2009.01568.x

Received October 2015; revised April 2016; accepted April 2016