

# Light-Field Surface Color Segmentation with an Application to Intrinsic Decomposition

Elena Garces

Erik Reinhard

Technicolor

{elena.garces|erik.reinhard}@technicolor.com

## Abstract

*To enable light fields of large environments to be captured, they would have to be sparse, i.e. with a relatively large distance between views. Such sparseness, however, causes subsequent processing to be much more difficult than would be the case with dense light fields. This includes segmentation. In this paper, we address the problem of meaningful segmentation of a sparse planar light field, leading to segments that are coherent between views. In addition, uniquely our method does not make the assumption that all surfaces in the environment are perfect Lambertian reflectors, which further broadens its applicability. Our fully automatic segmentation pipeline leverages scene structure, and does not require the user to navigate through the views to fix inconsistencies. The key idea is to combine coarse estimations given by an over-segmentation of the scene into super-rays, with detailed ray-based processing. We show the merit of our algorithm by means of a novel way to perform intrinsic light field decomposition, outperforming state-of-the-art methods.*

## 1. Introduction

Plenoptic cameras produce light fields that are typically dense: after view extraction, the differences in viewpoint of each view is small. This favors the capture of small scenes, and mostly covers macro-photography. Camera arrays produce views by placing a set of cameras in a grid, where the spacing between the cameras determines how dense or sparse the capture is. To capture large scenes, the spacing between the cameras will have to be large, or the number of cameras would have to be intractably large.

A wide spacing between views, however, results in sparse light fields, and these tend to be difficult to process, analyze or edit. In particular, any processing of dense light fields rely on the derivatives in the angular domain to impose constraints or fit parametric models to extract surface properties. However, in sparse light fields the estimation of

such derivatives may become unreliable, so that many light field processing techniques do not perform well on sparse data.

While sparse light fields could be processed with techniques designed for multi-view data or even for video, light fields offer some opportunities that make direct application of multi-view or video processing techniques unattractive. Specifically, multi-view data, as well as video, tends to be unstructured and therefore usually requires 3D reconstruction and navigation through the views. Finally, if we were to process the light field as a video sequence, we cannot guarantee angular consistency, and even if we could filter out the resulting artifacts [9], we would miss an opportunity to gain fidelity by ignoring the geometric information of the capture device: even if a light field is sparse, the data would have been captured from equally spaced locations in space. This feature alone should be sufficient to avoid the full reconstruction of a 3D point cloud, and thereby offers advantages in terms of algorithmic complexity.

One type of low-level image processing that often lies at the heart of more advanced image processing, analysis and editing, is that of segmentation. Consequently, segmentation is an important task, and it would therefore be an advantage to have a good segmentation algorithm that can be applied to light fields. Of course, it would be possible to segment each of the views of a light field independently. Such an approach, however, would ignore the coherence that exists between views. To create a meaningful segmentation on which to base subsequent processing, a segment in one view should have a one-to-one mapping with a corresponding segment in each of the other views.

Here, we present an algorithm to create a surface color segmentation of sparse planar light fields that exhibits this desirable feature, and we show its benefits in the context of a specific application, namely intrinsic light field decomposition.

In segmentation, often the simplifying assumption is made that all surfaces in a scene are reflecting light in a perfectly diffuse (Lambertian) manner. However, the real world is full of non-Lambertian surfaces such as specular,



Figure 1. Color variations between views due to specular reflections. From left to right: the light field, the top left view and the bottom right view. Original scene by *guismo*, *BlendSwap*.

glossy, or mirror-like surfaces. The problem with such materials is that their appearance may change depending on the point of view of the scene. This problem is particularly critical for sparse light fields with a large baseline. The same glossy surface may have different colors in different views (Figure 1). Thus, if we want to segment the surface of the object into its different colors, we need to take into account these differences between views.

The problem to circumvent is that the amount of information available in sparse light fields does not allow for rich surface predictions. In addition, we want the task of light field segmentation to be automatic, obviating the need for a user to navigate through the views to fix possible inconsistencies. This requires the design of novel techniques that take into account the structure of sparse light field data in an efficient and intelligent manner.

To find a good trade-off between the quality of the results and computational complexity, we shun obvious approaches which may be based on optical flow [10] or patch-match algorithms [30]. These would be more appropriate for unstructured data such as videos or multi-view captures. Instead, we rely on an over-segmentation of the light field into super-rays [22], which can be seen as the light field equivalent of super-pixels [32] in that they represent a coherent grouping of pixels across views. Further efficiency gains can be obtained by first calculating a color palette from the central view, and by then sampling just the central view as well as the corner views. These samples are then mapped to the central view, where a dense fully connected Conditional Random Field (CRF) algorithm [24] performs the color segmentation on the sampled points. The super-ray representation is then used to propagate the segmentation results to all the views. The key contributions of our method are therefore:

- Use perceptual color palettes as an initial estimation of the colors of a scene. This provides more robust estimation than k-means clustering, which is widely used but requires the number of segments to be set by the user.
- Segment a light field into distinct surface colors and specular highlights without user interaction robust to view-dependent effects.
- Use implicit geometry given by the light field super-rays and take advantage of the structure of the planar

light fields.

- Use the proposed segmentation to improve the state-of-the-art of intrinsic decomposition techniques for sparse light fields.

## 2. Related Work

Segmentating light fields is a surprisingly underexplored problem, leaving us with only a few papers to review. To our knowledge, methods for the segmentation of sparse light fields currently do not exist at all.

A special case of segmentation is layer segmentation, whereby a light field is segmented such that objects at different depths are assigned different labels. This is a useful problem to solve, as it allows viewpoint interpolation. It is, however, a somewhat different problem from the one we wish to solve, in that different objects/textures in the same layer may not be distinguished by such methods. An early method for layer segmentation uses epipolar plane volumes [12], which can be constructed by stacking a row or column of views into a volumetric pixel representation. A cut through such a volume is known as an epipolar plane image (EPI). Objects will form line structures in EPIs at an angle that relates to the spacing between the cameras as well as the depth of the object, making layer segmentation possible. Such techniques can be extended to consider not only a row or column of views, but to consider all views simultaneously [4, 5].

In the context of standard object segmentation, Wanner and colleagues observe that while segmentation is a difficult problem, the estimation of disparity in light fields provides an additional cue that can be leveraged to increase segmentation accuracy [28]. They propose a variational framework that operates in ray-space, effectively utilizing the structure of EPIs. For this to work effectively, the disparity between neighboring views has to be limited, so that the angles of the lines seen in EPIs do not become too shallow, and can therefore be robustly estimated. Maintaining accuracy, therefore, requires the light field to be dense.

Alternatively, segmentation and co-segmentation may be based on Markov Random Fields (MRFs), as for instance practiced in multi-view segmentation [20, 15]. Light fields tend to be represented by enormous amounts of data, and this makes direct application of MRFs on light fields intractable. However, a graph structure may be imposed upon a light field, and MRF-based segmentation may be applied to the graph representation to produce an interactive and coherent light field segmentation algorithm [21].

Note that the most related methods [28, 21, 25] require user-provided input scribbles to guide the segmentation, whereas our aim is to provide a fully automatic object segmentation.

Light field	$\mathbf{L}$
Image plane	$\Omega_{xy}$
Sensor plane	$\Pi_{uv}$
Number of views	$m \times n$
View	$\mathbf{L}_{ij}$
Reference view	$\mathbf{L}_{u_c, v_c}$
Key views	$\mathbf{L}' = \{\mathbf{L}_{00}, \mathbf{L}_{m0}, \mathbf{L}_{u_c v_c}, \mathbf{L}_{0n}, \mathbf{L}_{mn}\}$
Ray	$\mathbf{r} = (x, y, u, v)$
Super-ray	$\mathbf{p}$
Set of all super-rays	$\mathcal{P}(\mathbf{L}) = \{\mathbf{p}\}$
Segment of a ray	$\phi(\mathbf{r})$
Color palette: set of CIELab colors	$\mathcal{P}(\mathbf{L}_{u_c v_c}) = \{\alpha_0, \alpha_1, \dots, \alpha_k\}$
Label	$\mathbf{x}$
CIELab color of a label	$\alpha(\mathbf{x})$
Color channel of a label	$\alpha^c(\mathbf{x}), c \in \{L, a, b\}$
Energy for the CRF	$E$
Feature vector of a ray	$\mathbf{f}(\mathbf{r}) = [x', y', L, a, b]$
Label compatibility	$\tau(i, j)$
Unique segments per super-ray	$\phi_u(\mathbf{p})$
Color coherence	$\beta^{\mathbf{p}}$
Fuzziness per super-ray and segment	$\lambda^{\mathbf{p}}(c_k)$
Fuzziness	$\rho^{\mathbf{p}}$
Edge weights	$w_{i,k}$

Table 1. Notation

### 3. Method

We take as input a light field  $\mathbf{L}(x, y, u, v)$ , represented with the two-plane parametrization on ray space, where a light ray  $\mathbf{r} = (x, y, u, v) \in \mathbf{L}$  passes through two parallel planes: the sensor plane  $\Pi_{uv}$ , and the virtual camera plane or image plane  $\Omega_{xy}$ . We denote as  $\mathbf{L}_{ij}(x, y)$  or  $\mathbf{L}_{ij}$  the slice or view of the light field that cuts the sensor plane at the coordinates  $(u = i, v = j)$ . We define the reference view of the light field as  $\mathbf{L}_{u_c v_c}$  with  $u_c = \lfloor \frac{m}{2} \rfloor$  and  $v_c = \lfloor \frac{n}{2} \rfloor$  for a sensor plane of size  $m \times n$ . Table 1 summarizes the notation of the paper.

An overview of the method is shown in Figure 2. Our goal is to find a set of segments  $\Phi = \{c_k\}$  with coherent surface color across the multiple views of the light field. The first step is to select the key views  $\mathbf{L}'$ . In this case, for the key views we choose the extrema and the reference view of the light field  $\mathbf{L}' = \{\mathbf{L}_{00}, \mathbf{L}_{m0}, \mathbf{L}_{u_c v_c}, \mathbf{L}_{0n}, \mathbf{L}_{mn}\}$ , since they cover the highest variability in terms of occlusions and color variations with view-point changes.

Given the light field, in step 2 we compute the set of super-rays  $\mathcal{P}(\mathbf{L}) = \{\mathbf{p}\}$  [22]. Each super-ray contains a group of rays of the same area of the scene with uniform color. The chosen method produces super-rays with a regular shape and size. This kind of over-segmentation provides a soft geometric decomposition of the scene, and has been used before in the context of image-based rendering [11]. Thus, we use this information as implicit structural cue. We additionally rely on disparity values as given by the super-rays and with  $d(x, y, u, v)$  we denote the disparity at each ray of the light field.

Our pipeline also accepts an object mask, which the user may define in the central view. The mask will be automatically propagated to the remaining views in step 3. In step 4, we then compute a perceptual and automatic color palette  $\mathcal{P}(\mathbf{L}_{u_c, v_c})$  from the reference view [14]. Given the color palette  $\mathcal{P}$  and the key views  $\mathbf{L}'$ , we use a fully connected dense Conditional Random Field (CRF) [24] to obtain a fine-grained segmentation in step 5. This initial segmentation is further refined using the soft geometry given by the super-rays (step 6), and propagated to the remaining views (step 7). We use the disparity to reparametrize all the rays to the central view. Consequently, each ray of the light field will have new  $(x, y)$  coordinates:  $x' = x + (u - u_c) d(x, y, u, v)$  and  $y' = y + (v - v_c) d(x, y, u, v)$ .

The benefits of this approach are as follows. First, we implicitly leverage scene geometry thanks to the use of super-rays. Second, we are able to consider occlusions and view-dependent effects by analysing the extremal views as well as the reference view, while keeping computational complexity low. Finally, we obtain a fine-grained segmentation which would be difficult to obtain with super-rays alone. In the following, the non-trivial steps 3 to 7 are explained in more detail.

**Step 3: Mask Propagation** There are many potential ways to select an object in a light field and propagate its selection to the other views [28, 21]. The most straightforward way given our architecture would be to simply select the super-rays by means of a user interface. However, given the variation in size of super rays that may occur at object boundaries due to occlusions, we perform a few additional operations for the propagation.

Given a mask for the reference view, we first project the mask into the super-rays directly using the reference view as a guide. In this process, as the mask is defined at the pixel level, some super-rays will be completely selected, and some will be partially selected. For each super-ray partially selected, we make a decision about the selection as follows. We build a bounding box with the selected rays in the reference view. Then, we reparametrize all the other rays of the super-ray to the reference view and count the number of rays which lie inside the bounding box. If the number of points inside the bounding box is greater than 75% of the size of the super-ray, then the super-ray is marked as selected. This simple process allows us to discard super-rays which might be erroneously selected due to aliasing effects at occlusions.

**Step 4: Color Palette** We start by computing a color palette  $\mathcal{P}(\mathbf{L}_{u_c, v_c}) = \{\alpha_0, \alpha_1, \dots, \alpha_k\}$  by using the non-parametric method of Delon et al. [14]. This method automatically finds modes in a histogram and has been successfully used for color transfer [16]. Computing a palette is

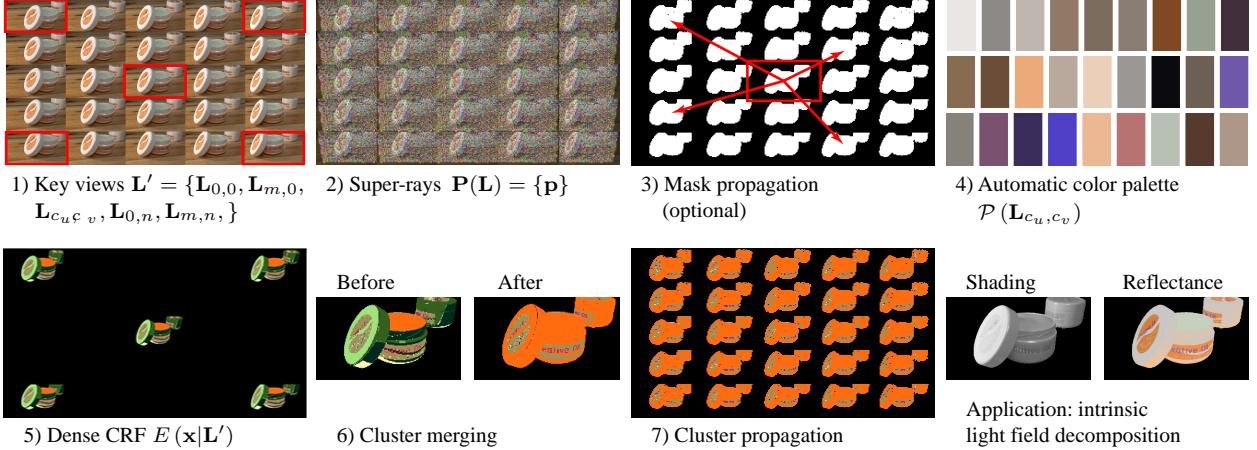


Figure 2. Overview of the method. Step 1: Select the extrema and the reference view as *key views*. Step 2: Compute the super-rays. Step 3: Optionally propagate the mask given by the user in the reference view of the light field. Step 4: Compute the color palette for the reference view. Step 5: Color segment the key views using a Conditional Random Field algorithm for the given color palette. Step 6: Process the segments and merge them according to the soft structure given by the super-rays. Step 7: Propagate the segmentation of the key views to the remaining views. Application: we show an application of the segmentation to intrinsic light field decomposition.

an alternative to using  $k$ -means clustering as most previous work does [18, 2, 19]. The main advantage is that we do not need to input the number of segments, and the method thus enables a palette to be generated while adapting fully automatically to the number of reflectances of a scene. We show in Section 4 how the choice of palette affects the color segmentation. The color palette forms the set of candidate labels used for segmentation in the following step.

**Step 5: Dense CRF over a Sparse Set of Views** We pose segmentation as a discrete labeling problem where for each ray  $\mathbf{r}_i$  of  $\mathbf{L}'$  we aim to find the associated color, or label  $\mathbf{x}_i$ , among the candidates given by the reflectance palette  $\mathcal{P}$ , such that  $\mathbf{x}_i \in \mathcal{P}$ . We follow previous work and formulate the following energy function [2, 3, 6]:

$$E(\mathbf{x}|\mathbf{L}') = \sum_{i \in \mathbf{L}'} \psi_u(i) + \sum_{i < j} \psi_p(i, j)$$

The first term constitutes a unary potential:

$$\psi_u(i) = \sum_{c \in a, b} |\alpha^c(\mathbf{x}_i) - \mathbf{f}^c(\mathbf{r}_i)|$$

and the second term is the pairwise potential:

$$\psi_p(i, j) = \tau(i, j) \exp \left( -\frac{\|\mathbf{w} \odot (\mathbf{f}(\mathbf{r}_i) - \mathbf{f}(\mathbf{r}_j))\|_2^2}{2} \right)$$

where the penalty for label compatibility  $\tau(i, j)$  equals  $\|\log \alpha(\mathbf{x}_i) - \log \alpha(\mathbf{x}_j)\|_1$  and  $\mathbf{f}(\mathbf{r}_i) = [x', y', L, a, b]$  is the feature vector per ray. The color of label  $\mathbf{x}_i$  of a ray is

$\alpha(\mathbf{x}_i)$ , whereby  $\alpha^c(\mathbf{x}_i)$  is the value for each color channel. We use the CIELab color space, so that  $c \in \{L, a, b\}$ . Further,  $\mathbf{w} = [\omega_1, \omega_2, \omega_3, \omega_4, \omega_5]$  where  $\omega_i$  are the weights for each feature, which in our experiments are  $\omega_{1-3} = 0.1$  and  $\omega_{4-5} = 0.025$ . Finally,  $\odot$  indicates the Hadamard product.

The unary potential penalizes each pixel that has a chroma value very different from the chroma of the label, and the pairwise potential forces points with similar feature vectors to have similar labels. The penalty is chosen as the Euclidean difference in CIELab which corresponds to perceptual differences in color. This problem may be solved by optimization with the method of Krähenbühl and Koltun [24].

**Step 6: Merging of Segments** The previous step produces a color segmentation of the light field which tends to be over-segmented, typically due to strong shadows or highlights (see Figure 3). Also, color variations due to speculars might be assigned to different segments in different views (Figure 4) resulting in incoherent segmentation.

For these reasons we perform further analysis of the light field to merge segments which correspond to the same surface. The key idea of the merging is that if the boundary between two segments is fuzzy and their color is similar, then it is likely that they might have been separated because of a shadow or a highlight. In that case they should be merged (see insets of Figure 3). To decide the merging of two segments, we analyze the boundaries between them taking the super-rays as reference. We compute two measures per super-ray: color coherence  $\beta^P$  and fuzziness  $\rho^P$ .

Color coherence measures the color variation of the pix-

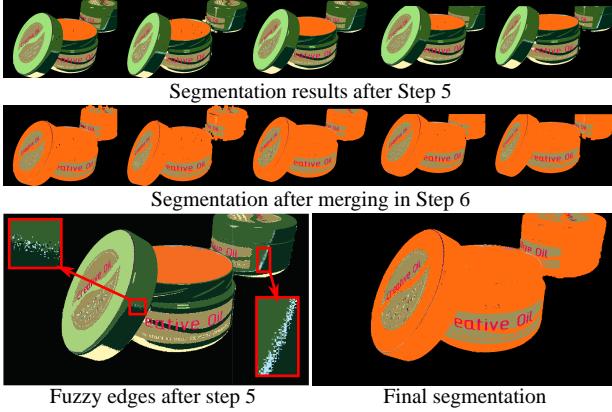


Figure 3. Comparison of segmentation results after Steps 5 and 6. Note that in Step 6 fuzzy edges are merged, leading to a cleaner result.

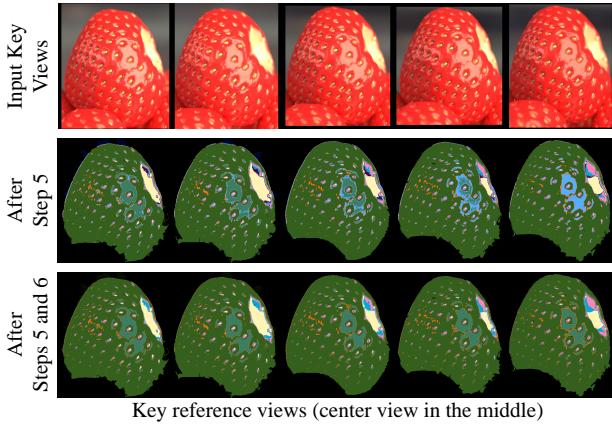


Figure 4. Segmentation results for Steps 5 and 6. Note that in the strawberry the segment representing the highlight is highly inconsistent between views.

els within the super-ray:

$$\beta^P = \exp \left( \frac{\|\sigma(\alpha^L(P)), \sigma(\alpha^a(P)), \sigma(\alpha^b(P))\|_2}{-0.01} \right)$$

where  $\sigma(\alpha^c(P))$  is the standard deviation in color channel  $c$  for the set of rays  $\{r_i\}$  inside super-ray  $P$ .

Fuzziness measures the spread of the values of each segment inside the super-ray with respect to its centroid. We start by measuring fuzziness per segment  $c_k$  and super-ray

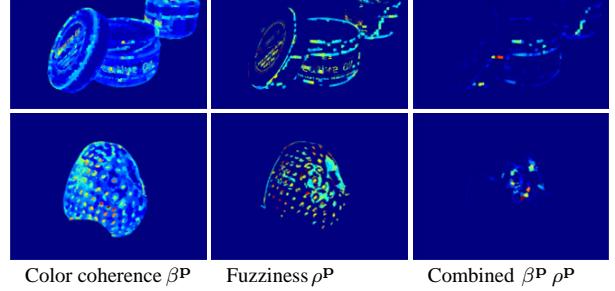


Figure 5. Color coherence and fuzziness measures computed per super-ray.

$P$  as follows:

$$\lambda^P(c_k) = \frac{\sum_{i \in q_k} \|r_i - \text{centroid}(\{r_m | m \in q_k\})\|_2^2}{\max(w, h)}$$

$$w = \max_{i \in q_k}(x'_i) - \min_{i \in q_k}(x'_i)$$

$$h = \max_{i \in q_k}(y'_i) - \min_{i \in q_k}(y'_i)$$

$$q_k = \{ i | r_i \in P \wedge \phi(r_i) = c_k \wedge c_k \in \phi_u(P) \wedge |c_k| > \mu |P| \}$$

where  $\phi_u(P)$  is the distinct set of segments which are present in the super-ray,  $|c_k|$  is the number of rays of segment  $c_k$  which are inside the super-ray,  $|P|$  is the total number of rays of the super-ray, and  $\mu = 0.1$  for all the results shown in the paper. Then, we compute fuzziness per super-ray as:

$$\rho^P = \max(\{\lambda^P(c_k) | c_k \in \phi_u(P)\})$$

Once we have these measures per super-ray, we build a graph  $G = (V, E)$  where  $V$  is the set of nodes  $v_i$  which corresponds to each segment, i.e.  $V = \{v_i = c_i\}$ ; and  $E$  is the set of weighted edges, where two nodes are neighbors if they are connected in a 4-neighborhood in the image plane of the light field. The weight  $w_{i,k}$  of each edge is given by:

$$w_{i,k} = \frac{1}{2|B(c_i, c_k)|} \sum_{P \in B(c_i, c_k)} \rho^P \beta^P$$

$$B(c_i, c_k) = \{P | c_i, c_k \in \phi_u(P) \wedge c_i \neq c_k \wedge P \in P(L)\}$$

where  $B$  is the set of super-rays which intersect with the boundaries between the segments  $c_i$  and  $c_k$ . We merge nodes if  $w_{i,k} < 0.02$ , an empirically selected value which has been used for all the results shown in the paper. The resulting values for the measures can be seen in Figure 5 for two examples. In Figures 3 and 4 we show some final results before and after the segment merging step. Note that the result of this process is better if done for a single object

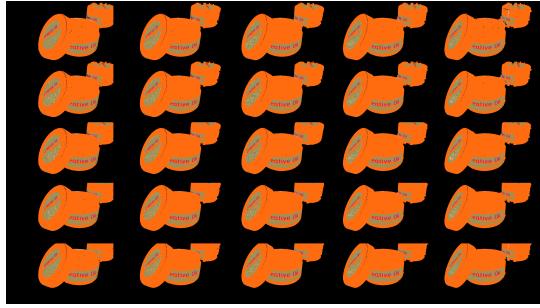


Figure 6. Propagation example. Results for all the scenes shown in the paper are displayed in the Supplementary Material.

rather than a whole scene. As this is a global process, if the initial segmentation in Step 5 clusters together two different surfaces with similar reflectance colors, the measures computed for the merging might be inaccurate, as we integrate the values over non local areas of the image. Further results are shown in Section 4.

**Step 7: Propagation of Segments** The previous step yields a coherent surface segmentation for the extrema and the central views of the light field  $\mathbf{L}'$ . However, we still need to propagate this information to the intermediate views. Inspired by Ye et al. [29], we use a look-up-table (*lut*) to propagate segments values per super-ray. We build a *lut* per super-ray which stores segment values indexed by RGB values. If there are two or more RGB pixels inside a super-ray with a different segment value, we keep the segment with the highest occurrence. Figure 6 shows an example obtained following this approach.

## 4. Results

As far as we know there is no existing method which can automatically segment light fields, nor is there any dataset for evaluation. Thus, in this section we evaluate each part of the pipeline separately. First, we provide qualitative comparisons to measure the impact of using automatic color palettes against conventional  $k$ -means clustering. Then, we evaluate the robustness of our algorithm in light fields for different palette initializations and compare with the state of the art of light field segmentation from user scribbles.

**Palette Initialization** We start by measuring how the choice of the initial palette impact the resulting color segmentation in single images. In Figure 7, right, we apply Step 5 of the method on the reference view of the light field varying the way to compute the initial reflectance palette. We test the automatic histogram segmentation approach of Delon et al. [14] and  $k$ -means clustering with a different number of clusters ( $k = 10$  and  $k = 20$ ). We observe that the results are scene-dependent. For example, *still-life*

has a larger number of different reflectances than *buddha* or *horses*. Consequently, a low value of  $k$  might be insufficient. On the other hand, the automatic palette provides a richer representation of the scene (a larger number of reflectances) and is adaptable to the complexity of the scene.

**Single Image vs. Light Field Segmentation** We measure how using five key views of the light field and the posterior merging operation as explained in Steps 5 and 6 improves the coherence of the segmentation for any choice of the initial palette. This is shown in Figure 7, left. In the same figure, we also compare against a light field segmentation method [21] which takes as input a set of user scribbles. Although this comparison is not fair, as our method does not require user input, it provides some intuition about the performance of the algorithm. Note, however, that our merging operation that integrates estimations globally works better for single objects. The reason is that, occasionally, the dense clustering of Step 5 group together different surfaces with the same color. As the merging is a global operation, this kind of inconsistent grouping prevents merging some segments. This effect is particularly visible in Figure 8. As can be seen, the segments of the jar are not merged together if the scene is globally processed (bottom row), but they are merged if the jar is processed locally (middle row).

## 5. Application: Intrinsic Light Field Decomposition

Intrinsic decomposition is generally understood to be the process of reverse engineering the process of image formation [23]. In particular, any pixel recorded by a camera system corresponds to light having reflected off a surface in a scene. In its simplest form, intrinsic decomposition splits each image  $\mathbf{I}$  into a component representing the reflectance  $\mathbf{R}$  of the imaged surface, and a second component  $\mathbf{S}$  representing the illumination (or shading) incident upon that surface, so that  $\mathbf{I} = \mathbf{R} \odot \mathbf{S}$ . This is an under-constrained problem in single images, as well as in light fields (for which  $\mathbf{I} = \mathbf{L}$ ).

In light fields, however, the same scene point is imaged in different views from somewhat different vantage points. The broader the baseline of the light field, the more potential for analysis a light field offers in this respect, even if this increases sparsity and normally makes analysis difficult. A robust and coherent sparse light field segmentation method as described above, however, enables a meaningful intrinsic decomposition that would be difficult to achieve with existing methods.

We rely on our robust color segmentation to guide the decomposition. Then, following previous work [19, 2, 6], we find a gray-scale shading component  $s = \log(\mathbf{S})$  by minimizing a linear system of equations. As a consequence

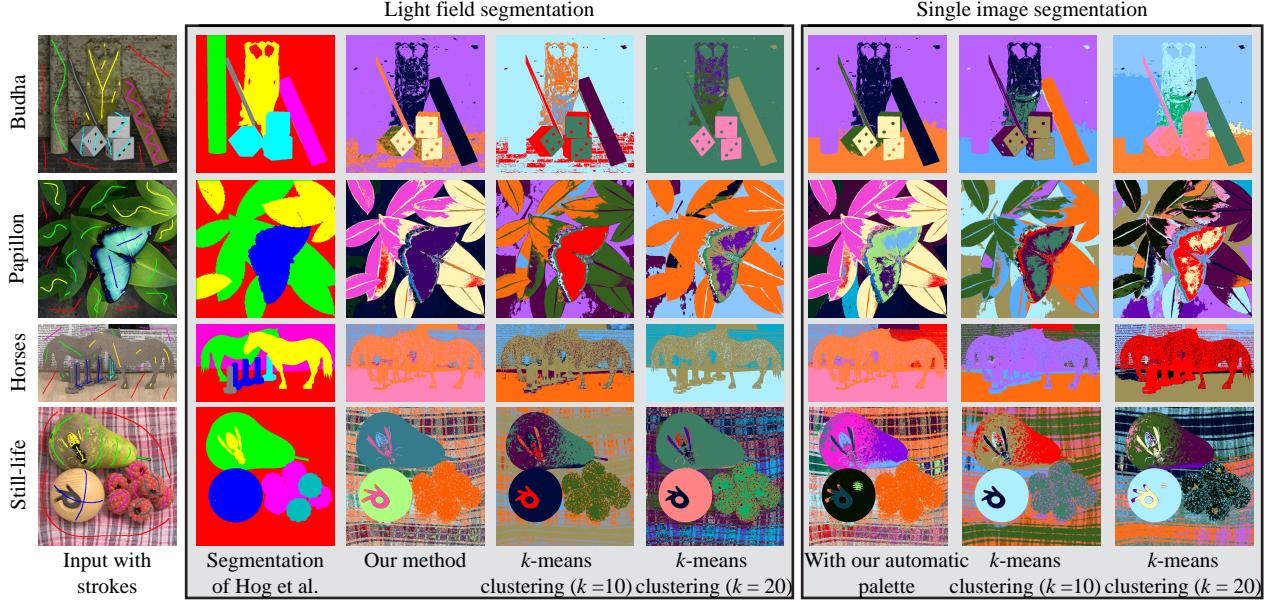


Figure 7. Color segmentation comparison. Left: Input scene and user strokes as used by Hog et al. [21]. Middle: light field segmentation comparison. Right: single image segmentation comparison.

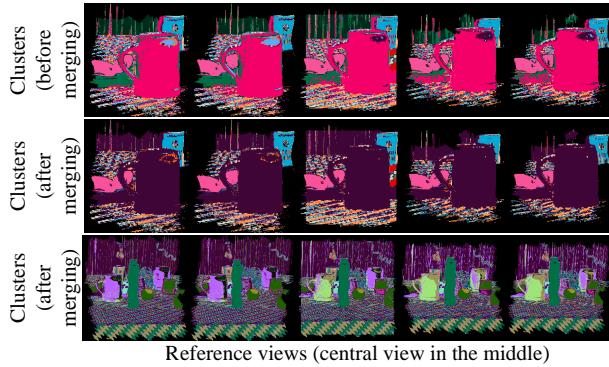


Figure 8. For each of the key views, we show the segments before and after merging.

of having segmented the lightfield, we now have just one unknown per segment  $c_k$ , so that we can impose shading smoothness constraints between pairs of neighboring segments:

$$\min_s \sum_{c_k \sim c_l} (s_k - s_l)^2$$

where  $c_k$  and  $c_l$  are neighboring segments in the image plane, and the shading components  $s_k$  (and  $s_l$ ) are computed as:

$$s_k = \bar{i}_k - r_k$$

$$\bar{i}_k = \frac{1}{|c_k|} \sum_{j \in c_k} \log \left( \frac{\mathbf{r}_j^R + \mathbf{r}_j^G + \mathbf{r}_j^B}{3} \right)$$

Here,  $r_k$  refers to a reflectance color. Further,  $\bar{i}_k$  is the average value for all the rays within the segment  $k$ . This optimization can be solved globally in closed-form for the entire light field, as the number of unknowns is significantly reduced by working with segments instead of rays. Then the light field's reflectance image  $\mathbf{R}$  is computed as a per pixel and per channel (Hadamard) division:  $\mathbf{R} = \mathbf{L} \oslash \mathbf{S}$ . Note, however, that a better decomposition would be possible in two ways: 1) by including non-local reflectance constraints given by the segmentation [19, 6]; 2) by separating the segments by checking connected components in the image plane in an 8-neighborhood.

**Results** Current intrinsic light field algorithms [17, 1] cannot cope with sparse light fields, so that comparison with such methods would not produce meaningful results. Our method, on the other hand, is suitable for both dense and sparse light fields as we show here and in the Supplementary Material. As argued in Section 1 and by Garcés et al. [17], intrinsic video decomposition methods [8, 29] are also not suitable for sparse light field data.

We are, however, able to compare our intrinsic decomposition against the state-of-the-art intrinsic image decomposition method of Bell et al. [2], as shown in Figures 9 and 10. Additionally, Figure 9 shows the ground truth decomposition.

We show in Figure 10 our decomposition only for the reference view taking the light field segmentation and, similar to previous work [19, 2], splitting the segments in image space as explained before. As we can observe, our decom-

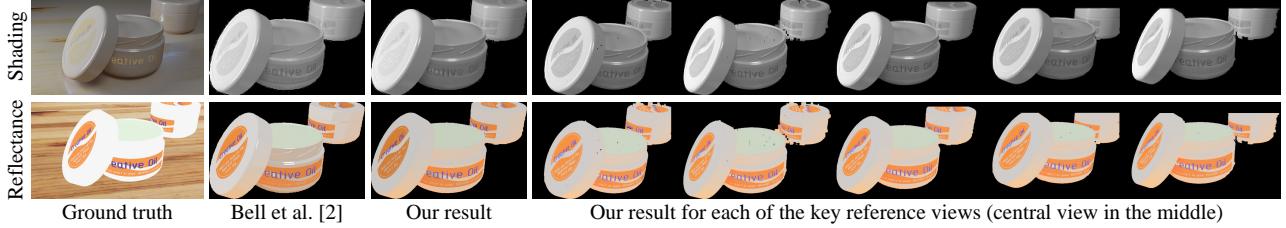


Figure 9. Intrinsic decomposition result. Note that highlights are wrongly assigned to the reflectance layer in Bell’s solution, and shading gradients are not smooth.

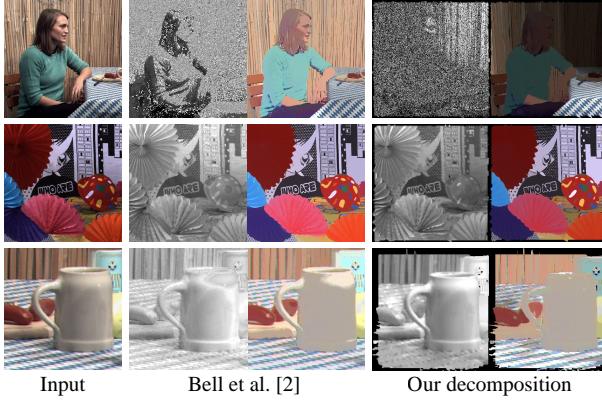


Figure 10. Comparison of our intrinsic decomposition with Bell et al. [2]. Our decomposition is done for the reference view only by splitting the segments in image space. Note that our result has highlights better associated with the shading component. Scenes taken from Sabater et al. [27] and Dabala et al. [13]

position for the reference view tends to be more accurate than the result of Bell et al. for most cases, even though there is an exception for the *girl* scene of Figure 10 (top row). Here, our reflectance result is unbalanced, some areas are too dark and others too bright. As explained in Section 5 this is due to the lack of global reflectance constraints that could eventually be incorporated. On the other hand, the checkered tablecloth exhibits a high level of accuracy.

The decomposition for the entire light field is a bit coarser since the segments are handled without splitting, thus, potentially causing inaccuracies when imposing the shading smoothness constraints. Further segmentation and intrinsic decomposition results on real scenes are shown in Figure 11 and in the Supplementary Material.

## 6. Conclusions

We have presented a method to coherently segment a light field into different colors, showing results for both real and synthetic scenes. Our color segmentation leverages the light field structure robustly and consistently, even in sparse light fields. In addition, our method deals with view-point

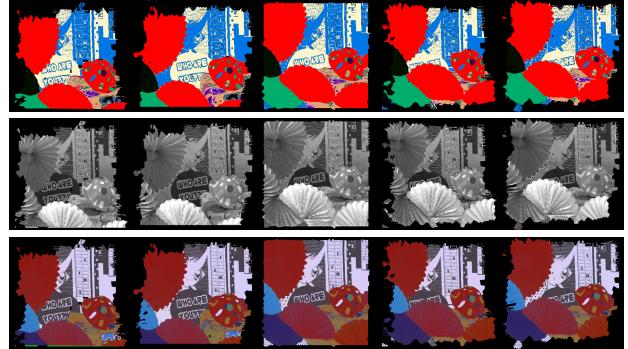


Figure 11. Segmentation and intrinsic decomposition of a real scene for each of the four corner views as well as the central view (shown in the middle). From top to bottom: our segmentation result, intrinsic shading and reflectance.

dependent material changes by means of a structural merging. The utility of the method was demonstrated in the context of intrinsic light field decomposition, giving superior decompositions than could be obtained in single images.

Light fields will have to become sparse to enable large scenes to be imaged and to be efficiently processed. Our work is an early example of how such processing may be accomplished. We hope that this work will open new applications in light field analysis and processing.

As an example, per-object intrinsic image decomposition could be explored, enabling as the purpose of intrinsic image decomposition often is image editing [7]. Finally, recent advances in intrinsic image decomposition, such as pre-filtering [2, 6, 26, 17] and the inclusion of non-local texture cues [31, 19] may be combined with our technique, and could thus further improve intrinsic light field decomposition results.

**Acknowledgments** We thank the reviewers for their insightful comments, Matthieu Hog and Neus Sabater for useful discussions and for helping with the superpixels code, and Nicolas Bonneel for providing the rendering files for the synthetic scenes.

## References

- [1] A. Alperovich and B. Goldluecke. A Variational Model for Intrinsic Light Field Decomposition. In *Asian Conference on Computer Vision*. Springer, 2016.
- [2] S. Bell, K. Bala, and N. Snavely. Intrinsic Images in the Wild. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 33(4), 2014.
- [3] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material Recognition in the Wild with the Materials in Context Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.
- [4] J. Berent and P. L. Dragotti. Segmentation of Eepipolar-plane Image Volumes with Occlusion and Disocclusion Competition. In *IEEE Workshop on Multimedia Signal Processing*, 2006.
- [5] J. Berent and P. L. Dragotti. Unsupervised Extraction of Coherent Regions for Image Based Rendering. In *British Machine Vision Conference*, 2007.
- [6] S. Bi, X. Han, and Y. Yu. An L1 Image Transform for Edge-Preserving Smoothing and Scene-Level Intrinsic Decomposition. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 34(4), 2015.
- [7] N. Bonneel, B. Kovacs, S. Paris, and K. Bala. Intrinsic Decompositions for Image Editing. *Computer Graphics Forum (Proc. Eurographics STAR)*, 36(2), 2017.
- [8] N. Bonneel, K. Sunkavalli, J. Tompkin, D. Sun, S. Paris, and H. Pfister. Interactive Intrinsic Video Editing. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 33(6):197, 2014.
- [9] N. Bonneel, J. Tompkin, D. Sun, O. Wang, K. Sunkavalli, S. Paris, and H. Pfister. Consistent Video Filtering for Camera Arrays. *Computer Graphics Forum (Proc. Eurographics)*, 36(2), 2017.
- [10] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High Accuracy Optical Flow Estimation based on a Theory for Warping. *European Conference on Computer Vision*, pages 25–36, 2004.
- [11] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis. Depth Synthesis and Local Warps for Plausible Image-based Navigation. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 32(3):30, 2013.
- [12] A. Criminisi, S. B. Kang, R. Swaminathan, R. Szeliski, and P. Anandan. Extracting layers and analyzing their specular properties using epipolar-plane-image analysis. *Computer Vision and Image Understanding*, 97(1), 2005.
- [13] L. Dabala, M. Ziegler, P. Didyk, F. Zilly, J. Keinert, K. Myszkowski, H.-P. Seidel, P. Rokita, and T. Ritschel. Efficient Multi-image Correspondences for On-line Light Field Video Processing. *Computer Graphics Forum (Proc. Eurographics)*, 2016.
- [14] J. Delon, A. Desolneux, J.-L. Lisani, and A. B. Petro. A Non-parametric Approach for Histogram Segmentation. *IEEE Transactions on Image Processing*, 16(1), 2007.
- [15] A. Djelouah, J.-S. Franco, E. Boyer, F. Le Clerc, and P. Pérez. Multi-view Object Segmentation in Space and Time. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2640–2647, 2013.
- [16] O. Frigo, N. Sabater, V. Demoulin, and P. Hellier. Optimal Transportation for Example-guided Color Transfer. In *Asian Conference on Computer Vision*, pages 655–670. Springer, 2014.
- [17] E. Garces, J. I. Echevarria, W. Zhang, H. Wu, K. Zhou, and D. Gutierrez. Intrinsic Light Field Images. *Computer Graphics Forum*, 36(8), 2017.
- [18] E. Garces, D. Gutierrez, and J. Lopez-Moreno. Graph-Based Reflectance Segmentation. In *Ibero-American Symposium on Computer Graphics*. Eurographics, 2011.
- [19] E. Garces, A. Munoz, J. Lopez-Moreno, and D. Gutierrez. Intrinsic Images by Clustering. *Computer Graphics Forum (Proc. EGSR)*, 31(4), 2012.
- [20] D. S. Hochbaum and V. Singh. An Efficient Algorithm for Co-segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 269–276. IEEE, 2009.
- [21] M. Hog, N. Sabater, and C. Guillemot. Light Field Segmentation using a Ray-based Graph Structure. In *European Conference on Computer Vision*, pages 35–50. Springer, 2016.
- [22] M. Hog, N. Sabater, and C. Guillemot. Super-rays for Efficient Light Field Processing. *IEEE Journal of Selected Topics in Signal Processing*, 2017.
- [23] B. K. P. Horn. Determining Lightness from an Image. *Computer Graphics and Image Processing*, 3(4):277–299, 1974.
- [24] P. Krähenbühl and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In *Advances in Neural Information Processing Systems*, pages 109–117, 2011.
- [25] H. Mihara, T. Funatomi, K. Tanaka, H. Kubo, Y. Mukaigawa, and H. Nagahara. 4d light field segmentation with spatial and angular consistencies. In *IEEE International Conference on Computational Photography*, pages 1–8, 2016.
- [26] T. Nestmeyer and P. V. Gehler. Reflectance Adaptive Filtering Improves Intrinsic Image Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.
- [27] N. Sabater, G. Boisson, B. Vandame, P. Kerbiriou, F. Babon, M. Hog, R. Gendrot, T. Langlois, O. Bureller, A. Schubert, et al. Dataset and Pipeline for Multi-View Light-Field Video. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 30–40, 2017.
- [28] S. Wanner, C. Straehle, and B. Goldluecke. Globally Consistent Multi-label Assignment on the Ray Space of 4d Light Fields. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1011–1018, 2013.
- [29] G. Ye, E. Garces, Y. Liu, Q. Dai, and D. Gutierrez. Intrinsic Video and Applications. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 33(4), 2014.
- [30] F.-L. Zhang, J. Wang, E. Shechtman, Z.-Y. Zhou, J.-X. Shi, and S.-M. Hu. Plenopatch: Patch-based Plenoptic Image Manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 23(5):1561–1573, 2017.
- [31] Q. Zhao, P. Tan, Q. Dai, L. Shen, E. Wu, and S. Lin. A Closed-Form Solution to Retinex with Nonlocal Texture Constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1437–1444, 2012.
- [32] H. Zhu, Q. Zhang, and Q. Wang. 4d Light Field Superpixel and Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.