Light Field Segmentation Using a Ray-Based Graph Structure

Matthieu Hog[1,2(✉)], Neus Sabater[1], and Christine Guillemot[2]

[1] Technicolor R&I, Rennes, France
matthieu.hog@technicolor.com
[2] Inria, Rennes, France

Abstract. In this paper, we introduce a novel graph representation for interactive light field segmentation using Markov Random Field (MRF). The greatest barrier to the adoption of MRF for light field processing is the large volume of input data. The proposed graph structure exploits the redundancy in the ray space in order to reduce the graph size, decreasing the running time of MRF-based optimisation tasks. Concepts of *free rays* and *ray bundles* with corresponding neighbourhood relationships are defined to construct the simplified graph-based light field representation. We then propose a light field interactive segmentation algorithm using graph-cuts based on such ray space graph structure, that guarantees the segmentation consistency across all views. Our experiments with several datasets show results that are very close to the ground truth, competing with state of the art light field segmentation methods in terms of accuracy and with a significantly lower complexity. They also show that our method performs well on both densely and sparsely sampled light fields.

Keywords: Light field · Segmentation · Markov Random Field

1 Introduction

Image segmentation is a key step in many image processing and computer vision problems. Many powerful solutions for image segmentation have been proposed in the image editing domain to this ill-posed problem. However, user interaction is still necessary to compensate for the lack of high level reasoning of segmentation algorithms. In parallel, the past decade has seen an increasing interest in multiview content to offer immersive user experiences or personalised applications with higher interactivity, stressing the need to develop new tools to interact with such multiview content.

One example of such emerging media for highly interactive applications is the light field technology. Different types of devices have been proposed to capture light fields, such as plenoptic cameras [1,2] or camera arrays [3,4] which capture the scene from slightly different positions. The recorded flow of rays (the so-called light field) is in the form of large volumes of highly redundant data yielding a very rich description of the scene enabling advanced creation of novel images from a

B. Leibe et al. (Eds.): ECCV 2016, Part VII, LNCS 9911, pp. 35–50, 2016.
DOI: 10.1007/978-3-319-46478-7_3

single capture. The data redundancy enables a variety of post-capture processing functionalities such as refocusing [5], depth estimation [6,7], or super-resolution [8,9]. However, the volume of captured data is the bottleneck of the light field technology for applications such as interactive editing, in terms of running time and memory consumption but also ease of use. This limitation becomes even more critical for platforms with limited hardware (e.g. mobile devices).

Meanwhile, MRF has proved to be a very powerful tool for multiview segmentation and co-segmentation [10,11]. In that framework, MRF are coupled with optimisation techniques such as graph-cuts [12]. Multiview segmentation and co-segmentation are in some aspects similar to light field segmentation. However, the principal challenge with light fields is the very large volume of input data which makes the MRF unsuitable for this task. The definition of the underlying graph structure and the corresponding energy terms are indeed crucial in the performance in terms of accuracy and complexity of the segmentation algorithm. For instance, our preliminary tests showed that a straightforward implementation of [13], using one node per ray of the light field and a simple 8-neighbourhood on the four light field dimensions, has a high computational complexity (about one hour of computation for a Lytro 1 light field image).[1]

In this paper, we propose a novel graph structure aiming to overcome the above problem. The philosophy of the approach is to consider that views of a light field, densely sampled or not, mostly describe the same scene (with the exception of occlusion and non-Lambertian surfaces). Therefore, it is unnecessary to segment separately each captured ray. Rays corresponding to the same scene point are detected thanks to a depth estimation of the scene on each view. Placed in a graph context, this means that all rays coming from the same scene point, according to their local depth measure, are represented as a *ray bundle* in a graph node. And rays having an incoherent depth measure, because of occlusion, non-Lambertian surfaces or faults in depth estimation, are represented as a *free ray* in a graph node. Pairwise connectivity is defined from the spatial neighbourhood on the views. Finally, in order to apply the graph-cut algorithm, energy terms are defined on the simplified graph structure based on free rays, ray bundles and the relationships between these entities.

To summarise, our contributions are twofold. First, we give a new representation of the light field based on a graph structure, where the number of nodes does not strictly depend on the number of considered views, decreasing greatly the running time of further processing. Second, we introduce an energy function for object segmentation using graph-cut on the new graph structure. This strategy provides a coherent segmentation across all views, which is a major benefit for further light field editing tasks.

Our experiments on various datasets [15–17] show first that the proposed segmentation method yields the same order of accuracy as the state of the art [18], with a notably lower complexity and second that the approach is very efficient for both densely and sparsely sampled light fields.

[1] This is the approach of [14], published in parallel to this work. They use a *full* graph structure and as a consequence, the authors report memory and computational time issues and experiment with only 25 of the 81 available views.

2 Related Work

In the current literature, few papers focus on interactive light field editing. One solution consists in propagating the user edits. In [19] the light field and input edits are first downscaled using a clustering based on colour and spatial similarity. While the complexity problem is solved, the propagated edit greatly depends on the quality of the clustering. On the other hand, the solution of [20] relies on a space voxelisation to establish correspondences between rays of different views. The approach has been demonstrated on circular light field but needs dense user input.

Concerning light field segmentation, two approaches have been proposed. In [21–23], level sets are used to extract objects with coherent depth in a scene. The method is fully automatic but is unfortunately limited to layer extraction. In [18], the most related work to ours, the authors use a random forest to learn a joint colour and depth ray classifier from a set of input scribbles on the central view. The output of the random forest classification is then regularised to obtain a segmentation close to the ground truth on synthetic images. Nevertheless, the authors report an important running time for the regularisation, over 5 min on a modern GPU, to compute the segmentation on 9×9 views of size 768×768.

The problem of extracting one or more visible objects in a set of images has been addressed in the co-segmentation and multiview segmentation literature using MRF and graph representations. The authors in [24] present a co-segmentation approach which extracts a common object from a set of images. Other approaches build an appearance model based on colour [24] or more advanced cues [25] and then use a MRF for each view to iteratively extract the objects with the graph-cut technique [12]. The model is updated until convergence is reached. In [10], the authors propose to model explicitly the correspondences between pixels that are similar in appearance by linking them to an introduced *similarity node*. Image geometry has also been used in a similar way to establish correspondences between pixels of the different views. Indeed, to avoid handling a space voxelisation [26], pixels or superpixels are linked directly using epipolar geometry [27] or as in [11] where extra nodes, corresponding to 3D scene samples, are used to propagate the labelling across a set of calibrated views. Equally, in [28], a graph structure is used to propagate a pre-segmented silhouette, assumed constant, to another view.

Those works show how powerful MRF modelling is to represent arbitrarily defined relationships between arbitrarily defined nodes. However, the problem of light field segmentation differs from those approaches in two points. First, the light field views are much more correlated than in co-segmentation and multiview segmentation, therefore labelling consistency can be furtherly enforced. Second, where multiview and co-segmentation consider a relatively limited number of views, light fields typically consist in a dozen to a hundred of views, causing a serious increase in running time during the energy minimisation. In the next sections, we describe how, from the same idea of MRF modelling with arbitrarily defined nodes, we design an MRF model that copes with the above mentioned problems.

3 Ray-Based Graph Structure

In this section, we define the proposed graph structure to perform the light field segmentation. We first give the formal definitions and then explain the motivations of the design.

We consider an input light field, $C(s,t,x,y)$ represented with the two plane parametrisation (as in [29]), where (s,t) are the angular (view) coordinates and (x,y) the spatial (pixel) coordinates.

3.1 Free Rays and Ray Bundles

Let r_i be a light ray represented by its 4-D coordinates (s_i,t_i,x_i,y_i) in the light field. We denote $D(r_i)$ its local disparity measurement. $D(r_i)$ is estimated along s and/or t in the adjacent views, either by traditional disparity estimation for sparsely sampled light fields, or by studying intensity variations on epipolar images [7] for densely sampled light fields. We define a *ray bundle* b_i as the set of all rays describing the same 3D scene point, according to its depth measurement $D(r_i)$. Formally, two rays r_i and r_j belong to the same bundle if and only if they satisfy the left-right coherence check

$$\begin{cases} [x_i + (s_i - s_j)D(s_i,t_i,x_i,y_i)] & = x_j\,, \\ [x_j + (s_j - s_i)D(s_j,t_j,x_j,y_j)] & = x_i\,. \end{cases} \tag{1}$$

where $[a]$ denotes the rounded value of a. The same test is performed for the $t-y$ direction. Note that Eq. (1) holds for uniformly sampled and calibrated light fields but can straightforwardly be adapted to a light field with different geometry.

A ray bundle gathers all rays emitted by the same 3D scene point according to their local depth measurement. On the contrary, a ray is called *free* if it has not been assigned to any ray bundle. Generally *free rays* correspond to occlusions or light rays having wrong depth estimates.

Now let R be the set of all free rays and B the superset that contains all ray bundles. In this setup, if LF denotes the set of all rays (i.e. the light field), regardless if they are free or not, then $LF = R \dot{\cup} B$. Figure 1 summarises this light field representation.

3.2 Graph Construction

For constructing the graph, we need to define the neighbouring relationships between free rays and ray bundles. Let $\mathcal{N}(r_i)$ be the 4-connect neighbourhood of r_i on each view, that is to say the set of rays $\{r_j, r_k, r_l, r_m\}$ with r_j of coordinates (s_i,t_i,x_i-1,y_i), r_k of coordinates (s_i,t_i,x_i+1,y_i), r_l of coordinates (s_i,t_i,x_i,y_i-1) and r_m of coordinates (s_i,t_i,x_i,y_i+1). One ray r_i is neighbour of a ray bundle b_i if and only if one ray element of b_i is neighbour of r_i:

$$r_i \in \mathcal{N}(b_i) \iff b_i \cap \mathcal{N}(r_i) \neq \emptyset. \tag{2}$$

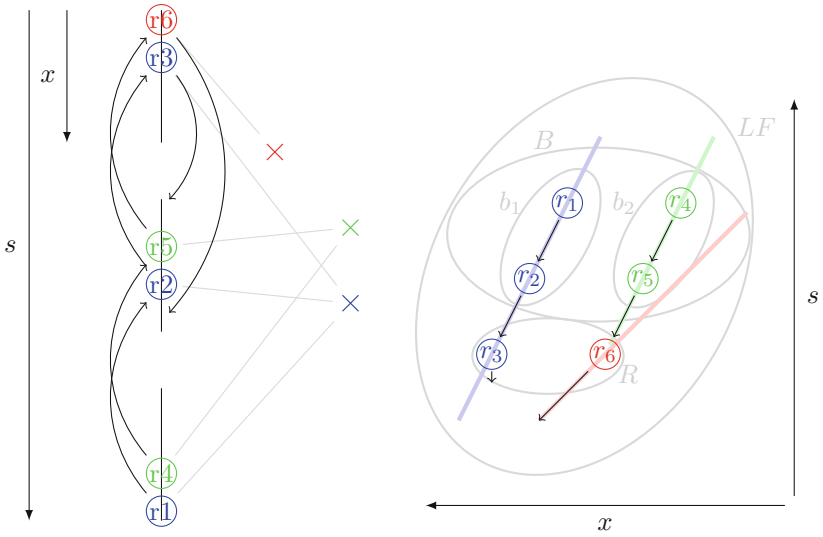

Fig. 1. Proposed light field representation of a 2D flatland illustrated as scene/view (left) and EPI (right). We show three scene points as red, green and blue crosses (and their resp. lines in the EPI). 6 rays r_i (in gray) come from those points and hit three different views. The black arrows represent the local depth measurement. The rays r_1 and r_2 are assigned to the same ray bundle b_1 because their depth measurement satisfies the left-right coherence check (Eq. (1)). Similarly r_4 and r_5 are assigned to b_2. On the contrary, r_3 has an incoherent (noisy) depth estimate and is classified as a free ray and not as a ray of b_1. Finally, the red scene point occludes the green scene point in the first view, so r_6 is also classified as a free ray and not as a ray of b_2. (Color figure online)

Similarly, two ray bundles b_i and b_j are neighbours if they have at least one element in the neighbourhood of the elements of each other, i.e.,

$$b_i \cap \mathcal{N}(b_j) \neq \emptyset. \tag{3}$$

Finally, we build the graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where each node $\mathcal{V}$ corresponds to either one element of R or one element of B, and the edges $\mathcal{E}$ are defined by the neighbouring relationships between two rays, two bundles, and between rays and bundles:

$$\begin{cases} \mathcal{V} = B \dot{\cup} R\,, \\ \mathcal{E} = \big\{(r_i, r_j),\ r_j \in \mathcal{N}(r_i)\big\} \cup \big\{(b_i, r_i),\ r_i \in \mathcal{N}(b_i)\big\} \cup \\ \qquad \cup \big\{(b_i, b_j),\ b_i \cap \mathcal{N}(b_j) \neq \emptyset\big\}\,, \quad \forall r_i, r_j, b_i, b_j \in \mathcal{V}. \end{cases} \tag{4}$$

The main motivation behind our graph construction is to reduce the amount of data to process compared to a naive graph (one node per light ray). With our approach, in the best case scenario, when the depth is perfect and almost all light rays are grouped in bundles, the number of nodes of our graph is roughly

divided by the number of views with respect of the number of nodes of the naive graph (minus the occlusions). This is of a particular interest for problems that need global or semi global optimisations, such as image segmentation, which are usually not solvable in polynomial time (they are NP-complete problems).

The strategy of keeping free rays which are not grouped in bundles allows the use of a relatively coarse - and fast - depth estimation methods. With our approach, a low quality depth estimation only affects the number of free rays compared to the number of ray bundles, increasing the running time, but it has limited impact on the segmentation quality.

However, one problem arises when two rays r_i and r_j have wrong depth estimates, while still satisfying the left-right coherence check Eq. (1). In practice, we will see that these errors do not have many consequences on the overall result, since the mismatch usually happens on rays having very similar appearances, thus likely to belong to the same object.

4 Energy Function

The goal is now to express the energy function for the segmentation in a way that takes into account the proposed hybrid graph structure. Let us denote L the labelling function that assigns a label α to each free ray and ray bundle. The energy we seek to minimise is of the form:

$$\varphi_L = \sum_{r_i \in R} U(r_i) + \sum_{b_i \in B} U(b_i) + m\left(\sum_{\substack{r_i, r_j \\ r_i \in R,\, r_j \in \mathcal{N}(r_i)}} P(r_i, r_j) + \sum_{\substack{b_i, r_i \\ b_i \in B,\, r_i \in \mathcal{N}(b_i)}} P(b_i, r_i) + \sum_{\substack{b_i, b_j \\ b_i \in B,\, b_j \cap \mathcal{N}(b_i) \neq \emptyset}} P(b_i, b_j) \right), \quad (5)$$

where U denotes the data terms and P the smoothness terms. As, in conventional non-iterative graph-cut, m is the parameter that balances the data term with the smoothness term. In practise, we find the labelling L that yields the minimum energy using the alpha-expansion algorithm [30,31].

We now give the details of the energy function terms.

4.1 Unary Energy Terms

An annotated image is obtained by asking the user to draw scribbles of different colours over the objects he wants to segment on the reference view of the light field. We call S the scribble image of the same size as the reference view. Each pixel value under a scribble represents a label code (from 1 to the number of scribbles) and 0 otherwise. These scribbles are used to build a colour and depth model for each free ray and ray bundle using the following approach.

Defining and learning a joint colour and depth model is still an active research problem. Colour and depth are by nature hard to fuse because they represent different physical attributes. One solution is to learn a separate colour and depth

model and use a weighted fusion for classification, but that introduces extra data-dependent parameters to be either fine-tuned [32] or approximated [33]. Deep learning algorithms have proven to be efficient to overcome this limitation but are usually heavy and beyond the scope of the paper. On the other end, multivariate Gaussian Mixture Models (GMM) have proven to be efficient to model colour. The learning step of GMM however can be very time consuming depending on the number of mixture components. Fortunately, 5 to 8 components have been shown to be enough for most cases [34].

In our approach, a joint colour and depth GMM is learnt for each label. A fixed number of $K = 8$ components is used to infer the GMM with the Expectation Maximisation algorithm [35]. While mixtures of Gaussian are sub-optimal to infer depth, previous work [36] has shown convincing results and we will see that it suffices to demonstrate the interest of the proposed graph structure. One way of further improving the method could be to use a more specific type of joint distribution to characterise the depth [37] but the study of colour and depth statistical models is not the point of this work.

Now, since our segmentation method is a human-guided task, we first convert the input light field from RGB to *CIELab* colour-space to have a perceptually uniform colour distance in the segmentation process. Let the colour value of a ray be denoted $C(r_i)$. Then, the colour of a ray bundle is defined as the average colour of its element rays $C(b_i) = \frac{1}{|b_i|} \sum_{r_i \in b_i} C(r_i)$. Similarly, the depth of a bundle is the mean depth of its components $D(b_i) = \frac{1}{|b_i|} \sum_{r_i \in b_i} D(r_i)$.

The data term of a ray bundle b_i for a label α is then defined as the negative log likelihood of the bundle joint colour and depth probability $\mathcal{P}$ to belong to an object of label α, i.e. the data term is computed as

$$U(b_i) = \begin{cases} -log\Big(\mathcal{P}\big(C(b_i), D(b_i) | L(b_i) = \alpha\big)\Big) & \text{if } \exists r_i \in b_i, S(r_i) = 0, \\ \infty & \text{if } \exists r_i \in b_i, S(r_i) = \alpha, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

The joint colour and depth probability $\mathcal{P}$ is computed from the GMM. In Eq. (6) above, we use the input scribbles as hard constraints by setting $U(b_i)$ to 0 and ∞ if at least one of the rays of b_i is under a scribble.

Unfortunately, the depth information for free rays is unreliable. To compute $\mathcal{P}$ we assume the colour and depth values for a given ray to be independent. Hence, we can compute the probability $\mathcal{P}$ of the 3-dimensional sample r_i from the learnt 4 dimensional multivariate mixture Gaussian by removing the depth component from the learnt covariance matrix and mixture component means. Similarly to ray bundles, the scribbles are used as a hard constraint to compute the unary term for free rays as

$$U(r_i) = \begin{cases} -log\Big(\mathcal{P}\big(C(r_i) | L(r_i) = \alpha\big)\Big) & \text{if } S(r_i) = 0, \\ \infty & \text{if } S(r_i) = \alpha, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

4.2 Pairwise Energy Terms

Because of the new graph structure, we need to define 3 types of pairwise energy terms (i.e. edge weights): between two rays, between one ray and one bundle and between two bundles. One of the specificity of the proposed graph structure is that the connectivity between ray bundles depends on the captured geometry of the scene. One solution could be to define ray bundles connectivity from the 3D scene points they represent and keep the free ray pairwise energy as in conventional monocular segmentation. However, the combination of the two terms in a single energy function would require tuning an extra coefficient to balance their relative importance. Moreover, it involves surface reconstruction which is still a challenging and computationally expensive problem.

Instead, we propose to *derive* the energy function from a classical monocular framework. We start from the classical 4-connect neighbourhood to define the pairwise energy for free rays and ray bundles in order to obtain consistent energy terms.

The pairwise term between two rays is not different from the one used in classical image segmentation and is defined from the colour distance of the rays:

$$P(r_i, r_j) = \delta_{L(r_j) \neq L(r_i)} \, exp\left(\frac{-\Delta E(C(r_i), C(r_j))}{\sigma_{Lab}}\right), \tag{8}$$

where σ_{Lab} is the local image colour variance, ΔE the euclidean distance in the *CIELab* color space and δ the Kronecker delta so that our term is on the form of a contrast sensitive Potts model [31]. Similarly, since one ray bundle can only have one of its component as a neighbour of a free ray r, the pairwise between a free ray and a ray bundle is defined as:

$$P(b_i, r_i) = \delta_{L(b_i) \neq L(r_i)} \, exp\left(-\frac{\Delta E(C(b_i), C(r_i))}{\sigma_{Lab}}\right). \tag{9}$$

One specificity of the proposed graph structure is that the connectivity is dependent of the scene geometry. In fact, as illustrated in Fig. 2, an occlusion yields a *duplicated* neighbourhood for points at the border of foreground objects. If the weights on the corresponding edges were defined between two bundles having at least one neighbouring ray (minimal connectivity), red nodes corresponding to points at the border of foreground objects would be more connected to background points than to their foreground neighbours. To overcome this issue, we define the strength of the connections between two scene points as the sum of the colour differences of its corresponding rays (summed connection), which is a major twist to conventional pairwise energy design. Doing so, the sum of edge weights at the border of objects compensates for the over-connectivity.

In addition, we use the depth information of each bundle to favour the assignment of the same label to two neighbouring bundles which are on the same depth layer. The bundle pairwise probability term is then expressed as

$$P(b_i, b_j) = \\ \delta_{L(b_i) \neq L(b_j)} \, |b_j \cap \mathcal{N}(b_i)| \, exp\left(-\frac{\Delta E(C(b_i), C(b_j))}{\sigma_{Lab}} - \frac{(D(b_i) - D(b_j))^2}{\sigma_D}\right), \tag{10}$$

where σ_{Lab} and σ_D are the local colour and depth variances.

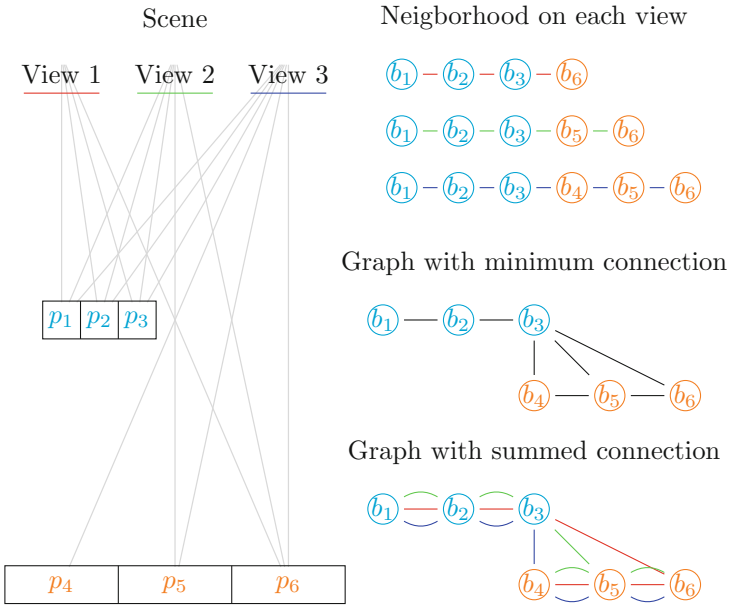

Fig. 2. Illustration of the over-connectivity problem. We show what happens to the neighbourhood of a ray bundle b_3 in our approach. Given a simple scene with 2 planes composed of 6 scene points p_i and their corresponding rays bundles b_i, we see that b_3 has 4 different neighbours across the 3 views (represented in red, green, and blue). (Color figure online)

5 Experiments

We first perform a quantitative evaluation of our light field segmentation approach using the dataset proposed in [15]. It is composed of 4 densely sampled synthetic light fields with known depth and ground truth labelling, along with a set of pre-defined input scribbles. The input data contains 9×9 views of 768×768 pixels. We compare the obtained segmentation with the results in [18]. Similarly, we use the ground truth labelling to find the optimum parameter m and we use the same input scribbles.

Figure 3 shows that our method yields a segmentation which is visually closer to the ground truth segmentation than the one obtained with the method of [15]. Table 1 gives the percentage of successfully segmented rays with respect to the ground truth. We can observe that this percentage is very close in terms of accuracy to the ground truth segmentation. It is also close to the one obtained in [18], even if in some cases it can be slightly lower.

We have seen that our wrongly labeled pixels (less than 1 % of total) are on the 1-pixel wide outskirt of the segmented objects.

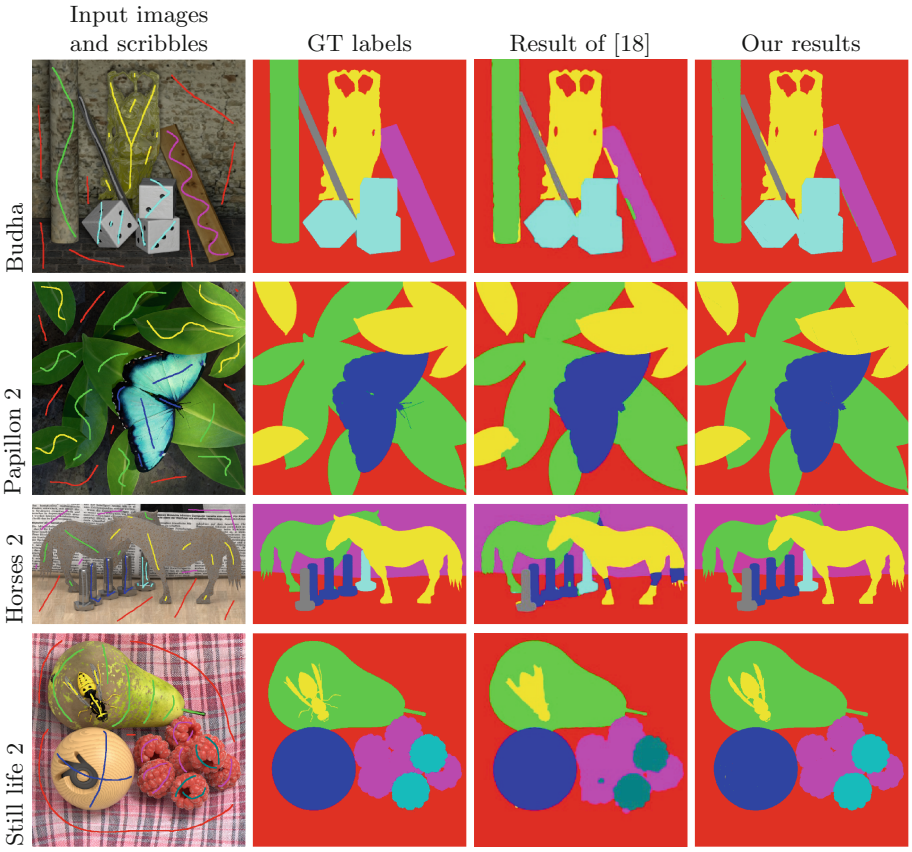

Fig. 3. Light field segmentation results obtained with the synthetic light field dataset proposed in [15]. From left to right, we show, the input central view with scribbles, the ground truth labelling, the results in [18] and our results. While both algorithms have a similar performance, in general, our results are more accurate in some challenging cases (see 'Horses 2').

Table 1. Segmentation accuracy comparison as the percentage of successfully segmented pixels. The results are for the entire light field views.

Dataset:	Still life 2	Papillon 2	Horses 2	Budha
Result of [18]:	99.3	99.4	99.3	98.6
Our results:	99.2	99.5	99.1	99.1
Our results w/o depth:	98.91	99.4	95.5	98.8

However, the big advantage of the method is the very significant gain in terms of running time. With a mono-thread CPU implementation of alpha-expansion[2], we perform the optimisation in 4 to 6 s depending on m, on an Intel Xeon E5640. Using the ground truth depth, we typically reduce the number of nodes by a factor of 50 (from $4.77 \cdot 10^7$ to $8.19 \cdot 10^5$ on 'Budha').

Another interesting point is that, with our framework, using depth in the unary term is only required to segment complex scenes. Indeed, we can see in Table 1 that running the same experiment without the depth in the unary term (Eq. 6), we can obtain very similar results. The only challenging case was the dataset 'Horses 2', for which the depth is required to differentiate adjacent objects having the same colour. The first row in Fig. 5 shows the segmentation result on a 4×4 synthetic sparsely sampled light field we produced. The segmentation result is very close to the ground truth showing that our approach is not limited to densely sampled light fields.

The approach has also been validated on the real, sparsely sampled light field of the Middlebury dataset [16] 'Tsukuba'. The input light field is composed of 5×5 rectified views of 288×384 pixels. We estimate, for each view, a disparity map using the algorithm presented in [38], which is real-time and accurate. More precisely, we only compute 25 right-to left conventional disparity maps for each view, without any fusion of the obtained depth maps. The first row of Fig. 6 shows the input image, the scribbles, the depth map and the segmentation result using $m = 20$. The segmentation step takes 3 s. Figure 4 visualises as a point cloud the obtained graph nodes for the light field 'Tsukuba'. We represent free rays as a 2D array on the background and the ray bundle as 3D points. We can see that, because the connectivity is defined from the views neighbourhood, the bundles do not need to be accurately estimated to have a coherent segmentation. As shown on the second row of Fig. 6, we further tested the approach on the densely sampled 'Legos' dataset from the new Stanford light field archive [17]. The images have been down-sampled by a factor of two to decrease the effect of mis-rectification. We see that our approach can handle challenging setups, where very few elements differentiate the scene objects.

We also tested the method on several 3D sparse light fields from the Middlebury [16] dataset. Initially proposed for multiview depth estimation, the light fields are composed of 7 high resolution views with important baselines. As visible on the 3 last rows of Fig. 6, we see that the free ray strategy copes efficiently with errors in the depth maps, while being able to segment arbitrarily defined objects.

Finally, a major advantage of the proposed method is that a coherent segmentation across all views is available. This is of particular interest for light field editing tasks. As an example, we show (see second row of Fig. 5) how the obtained segmentation can be used to remove an occluding object from a scene during synthetic aperture refocusing [39].

[2] http://vision.csd.uwo.ca/code/.

Fig. 4. Visualisation of the graph nodes for the dataset 'Tsukuba'. Points on the background planes are free rays, points projected in 3D represent ray bundles. We invite the reader to see the video on our website (see footnote 3) for more details.

Fig. 5. Experiments with our synthetic, sparsely sampled light field. The first row shows, from right to left, the input image and scribbles, the ground truth and our result. The second row shows an example of application for the light field segmentation: object removal via synthetic aperture [39]. From right to left, the obtained light field segmentation with only two labels (the object to remove in red), the image refocused using the full light field and the image refocused using the segmented light field. (Color figure online)

We make the dataset, along with the results of our experiments and supplementary video available on our website.[3]

Discussion: Our experiments allow us to draw conclusions at several levels. First, we show that the proposed framework is an efficient solution to reduce the computational load of MRF-based light field processing problems. In terms of accuracy, objective comparison on ground truth data shows results competing with the state of the art. We also validate our approach on real data, showing the flexibility of the proposed framework and its robustness to faults in depth estimation. The running time for the graph cut on CPU being of the order of the second, a GPU implementation as in [40] will most likely give real-time performances. As a limitation of our approach, we can see that it requires a relatively accurate depth estimation on all the views. Indeed, a too incoherent

[3] https://www.irisa.fr/temics/demos/RayBasedGraphStructure/index.html.

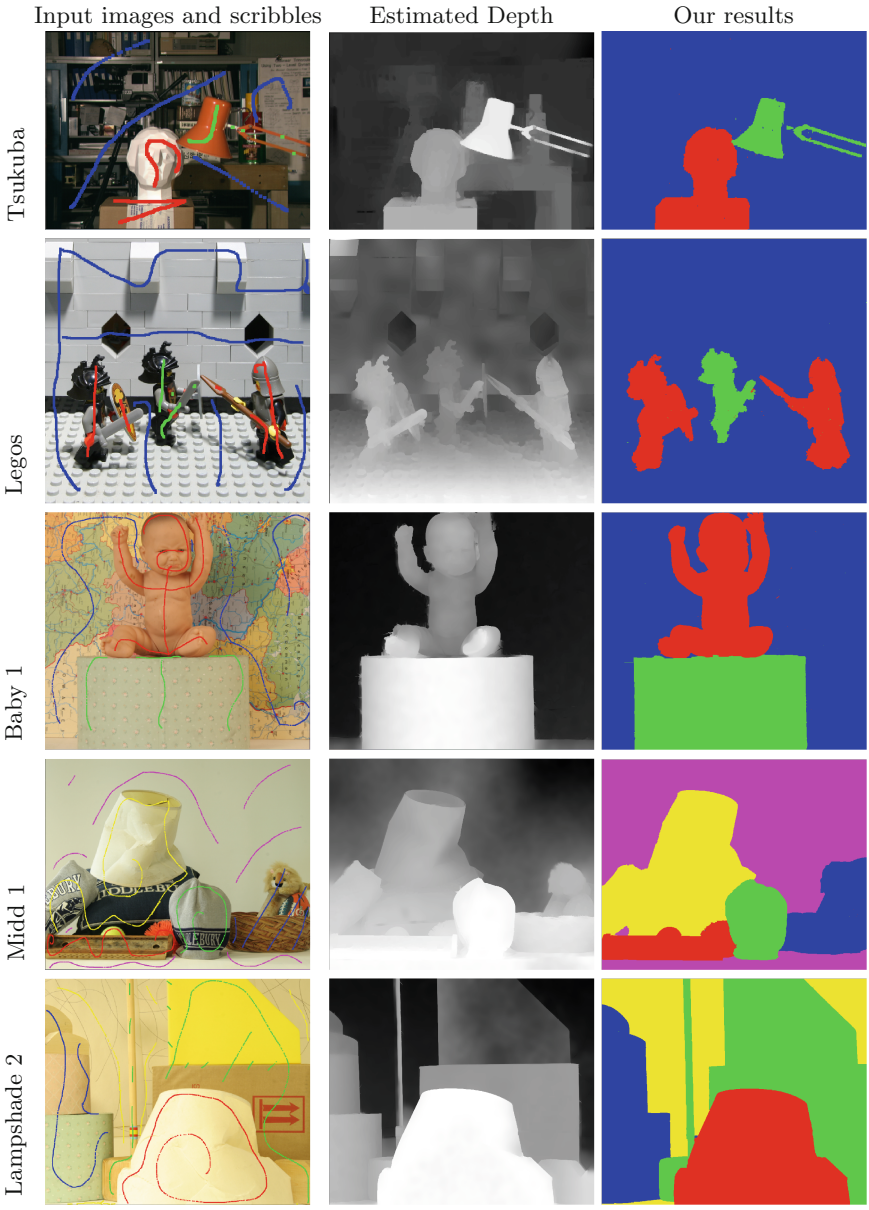

Fig. 6. Ligh-field segmentation results on real datasets from [16,17].

depth estimation will result in too many free rays, greatly increasing the running time but also losing segmentation coherence.

In that case, the angular neighbourhood concept newly introduced in [14] (for densely sampled light fields) or interactive scribbling of several views (for sparsely sampled light fields) could be good workarounds.

Hopefully, this is mitigated by the fact that, for sparsely sampled light fields, research on disparity estimation is mature, proposing a lot of reliable and fast disparity estimation. For densely sampled light fields, depth estimation is one of the main research interests and several efficient approaches have been proposed [7,41]. Equally, in some rare cases, two rays with faulty depth estimate will still satisfy the re-projection constraint, leading to the creation of a bundle that does not exist. The bundle has generally a depth value different from its neighbourhood, making it isolated according to the smoothness term. As a consequence it can be assigned a label different from its neighbourhood. One solution could be to increase the smoothness parameter to force consistency, but this also triggers loss in small details. Another solution could be to forbid the creation of bundles containing very few rays.

6 Conclusion

We present a novel approach to deal with light field processing needing a MRF formulation. Instead of using the full ray space in a MRF, the solution exploits the redundancy of the captured data estimated from a fast, local depth estimation to reduce the amount of nodes, in order to cope with the fact that the optimisation of MRF problems scales badly with the input size. We demonstrate the efficiency of the framework by proposing a user guided multi-label light field segmentation, where scribbles on a light field view are used to learn a colour and depth model for each object to segment. Unary and pairwise terms are defined according to the new graph representation. Graph-cut is then used to find the optimal segmentation. Comparison on synthetic light fields, with known ground truth show that our approach is close to state of the art in accuracy, while keeping a lower running time. Experiments on real light fields show that the proposed approach is not too sensitive to the errors in the required depth estimation, and is rather flexible regarding the arbitrary definition of objects to segment. Moreover, the solution is shown to be as effective for sparse light fields as for dense light fields. Future work will focus on adapting the proposed method for light field video segmentation.

References

1. Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P.: Light field photography with a hand-held plenoptic camera. Comput. Sci. Tech. Rep. **2**(11), 1–11 (2005)
2. Lumsdaine, A., Georgiev, T.: The focused plenoptic camera. In: ICCP, pp. 1–8. IEEE (2009)

3. Zhang, C., Chen, T.: A self-reconfigurable camera array. In: SIGGRAPH Sketches, p. 151. ACM (2004)
4. Wilburn, B., Joshi, N., Vaish, V., Levoy, M., Horowitz, M.: High-speed videography using a dense camera array. In: CVPR, vol. 2, p. II-294. IEEE (2004)
5. Ng, R.: Fourier slice photography. TOG **24**, 735–744 (2005). ACM
6. Tao, M.W., Hadap, S., Malik, J., Ramamoorthi, R.: Depth from combining defocus and correspondence using light-field cameras. In: ICCV, December 2013
7. Wanner, S., Goldluecke, B.: Globally consistent depth labeling of 4D light fields. In: CVPR, pp. 41–48. IEEE (2012)
8. Bishop, T.E., Zanetti, S., Favaro, P.: Light field superresolution. In: ICCP, pp. 1–9. IEEE (2009)
9. Wanner, S., Goldluecke, B.: Variational light field analysis for disparity estimation and super-resolution. PAMI **36**(3), 606–619 (2014)
10. Hochbaum, D.S., Singh, V.: An efficient algorithm for co-segmentation. In: ICCV, pp. 269–276. IEEE (2009)
11. Djelouah, A., Franco, J.S., Boyer, E., Clerc, F., Pérez, P.: Multi-view object segmentation in space and time. In: ICCV, pp. 2640–2647 (2013)
12. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. PAMI **23**(11), 1222–1239 (2001)
13. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient ND image segmentation. IJCV **70**(2), 109–131 (2006)
14. Mihara, H., Funatomi, T., Tanaka, K., Kubo, H., Nagahara, H., Mukaigawa, Y.: 4D light-field segmentation with spatial and angular consistencies. In: ICCP (2016)
15. Wanner, S., Meister, S., Goldluecke, B.: Datasets and benchmarks for densely sampled 4D light fields. In: VMV Workshop, pp. 225–226 (2013)
16. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47**(1–3), 7–42 (2002)
17. Andrew, A.: The (new) stanford light field archive. http://lightfield.stanford.edu/lfs.html. Accessed 3 Aug 2016
18. Wanner, S., Straehle, C., Goldluecke, B.: Globally consistent multi-label assignment on the ray space of 4D light fields. In: CVPR, pp. 1011–1018. IEEE (2013)
19. Jarabo, A., Masia, B., Gutierrez, D.: Efficient propagation of light field edits. In: SIACG (2011)
20. Seitz, S.M., Kutulakos, K.N.: Plenoptic image editing. IJCV **48**(2), 115–129 (2002)
21. Berent, J., Dragotti, P.L.: Unsupervised extraction of coherent regions for image based rendering. In: BMVC, pp. 1–10 (2007)
22. Dragotti, P.L., Brookes, M.: Efficient segmentation and representation of multiview images. In: SEAS-DTC Workshop, Edinburgh (2007)
23. Berent, J., Dragotti, P.L.: Plenoptic manifolds-exploiting structure and coherence in multiview images. Sig. Process. Mag. **24**, 34–44 (2007)
24. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching-incorporating a global constraint into MRFS. In: CVPR, vol. 1, pp. 993–1000. IEEE (2006)
25. Mukherjee, L., Singh, V., Peng, J.: Scale invariant cosegmentation for image groups. In: CVPR, pp. 1881–1888. IEEE (2011)
26. Reinbacher, C., Rüther, M., Bischof, H.: Fast variational multi-view segmentation through backprojection of spatial constraints. Image Vis. Comput. **30**(11), 797–807 (2012)
27. Campbell, N.D., Vogiatzis, G., Hernández, C., Cipolla, R.: Automatic object segmentation from calibrated images. In: CVMP, pp. 126–137. IEEE (2011)

28. Sormann, M., Zach, C., Karner, K.: Graph cut based multiple view segmentation for 3D reconstruction. In: 3DPVT, pp. 1085–1092. IEEE (2006)
29. Gortler, S.J., Grzeszczuk, R., Szeliski, R., Cohen, M.F.: The lumigraph. In: SIGGRAPH, pp. 43–54. ACM (1996)
30. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? PAMI **26**(2), 147–159 (2004)
31. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. PAMI **26**(9), 1124–1137 (2004)
32. Dal Mutto, C., Zanuttigh, P., Cortelazzo, G.M.: Scene segmentation by color and depth information and its applications. University of Padova (2010)
33. Mutto, C.D., Zanuttigh, P., Cortelazzo, G.M.: Fusion of geometry and color information for scene segmentation. J-STSP **6**(5), 505–521 (2012)
34. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. TOG **23**, 309–314 (2004). ACM
35. Bilmes, J.A., et al.: A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. ICSI **4**(510), 126 (1998)
36. Harville, M., Gordon, G., Woodfill, J.: Foreground segmentation using adaptive mixture models in color and depth. In: Workshop on Detection and Recognition of Events in Video, pp. 3–11. IEEE (2001)
37. Hasnat, M.A., Alata, O., Trémeau, A.: Unsupervised RGB-D image segmentation using joint clustering and region merging. J-STSP **6**(5), 505–521 (2012)
38. Drazic, V., Sabater, N.: A precise real-time stereo algorithm. In: IVCNZ, pp. 138–143. ACM (2012)
39. Yang, T., Zhang, Y., Yu, J., Li, J., Ma, W., Tong, X., Yu, R., Ran, L.: All-in-focus synthetic aperture imaging. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8694, pp. 1–15. Springer, Heidelberg (2014)
40. Vineet, V., Narayanan, P.: CUDA cuts: Fast graph cuts on the gpu. In: CVPR, pp. 1–8. IEEE (2008)
41. Bishop, T.E., Favaro, P.: Plenoptic depth estimation from multiple aliased views. In: ICCV Workshops, pp. 1622–1629. IEEE (2009)