Globally Consistent Multi-Label Assignment on the Ray Space of 4D Light Fields

Sven Wanner Christoph Straehle Bastian Goldluecke
Heidelberg Collaboratory for Image Processing

Abstract

We present the first variational framework for multi-label segmentation on the ray space of 4D light fields. For traditional segmentation of single images, features need to be extracted from the 2D projection of a three-dimensional scene. The associated loss of geometry information can cause severe problems, for example if different objects have a very similar visual appearance. In this work, we show that using a light field instead of an image not only enables to train classifiers which can overcome many of these problems, but also provides an optimal data structure for label optimization by implicitly providing scene geometry information. It is thus possible to consistently optimize label assignment over all views simultaneously. As a further contribution, we make all light fields available online with complete depth and segmentation ground truth data where available, and thus establish the first benchmark data set for light field analysis to facilitate competitive further development of algorithms.

1. Introduction

Recent developments in light field acquisition systems [2, 18, 19, 20] strengthen the prediction that we might soon enter an age of light field photography [17]. Since compared to a single image, light fields increase the content captured of a scene by directional information, they require an adaption of established algorithms in image processing and computer vision as well as the development of completely novel techniques. In this work, we develop methods for training classifiers on features of a light field, and for consistently optimizing label assignments to rays in a global variational framework. Here, the ray space of the light field is considered four-dimensional, parametrized by the two points of intersection of a ray with two parallel planes, so that the light field can be considered as a collection of planar views, see figures 2 and 3.

Due to the planar sampling, 3D points are projected onto lines in cross-sections of the light field called epi-polar-plane images. In recent works, it was shown that robust disparity reconstruction is possible by analyzing this line structure [1, 4, 9, 23]. In contrast to traditional stereo matching, no correspondence search is required, and floating-point precision disparity data can be reconstructed at a very small cost. From the point of view of segmentation, this means that in light fields, we not only have access to the color of a pixel and information about the neighboring image texture. Instead, we can assume that disparity is readily available as a possible additional feature.

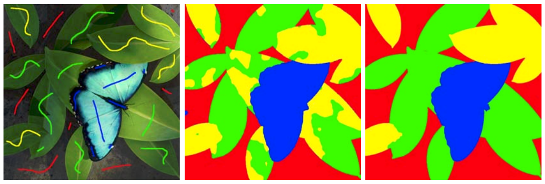

Figure 1. *Multi-label segmentation with light field features and disparity-consistent regularization across ray space leads to results which are superior to single-view labeling.*

Disparity turns out to be a highly effective feature for increasing the prediction quality of a classifier. As long as the inter-class variety of imaged objects is high and the intra-class variation is low, state of the art classifiers can easily discriminate different objects. However, separating for example background and foreground leafs poses a more difficult task, see figure 1. In general, there is no easy way to alleviate issues like this using only single images. However, for a classifier which also has geometry based features available, similar looking objects are getting readily distinguishable if their geometric features are separable.

In the following, we will show that light fields are ideally suited for image segmentation. One reason is that geometry is an inherent characteristic of a light field, and thus we can use disparity as a very helpful additional feature. While this has already been realized in related work on e.g. multi-view co-segmentation [15] or segmentation with depth or motion cues, which are in many aspects similar to disparity [21, 10], light fields also provide an ideal structure for a variational framework which readily allows consistent labeling across all views, and thus increases the accuracy of label assignments dramatically.

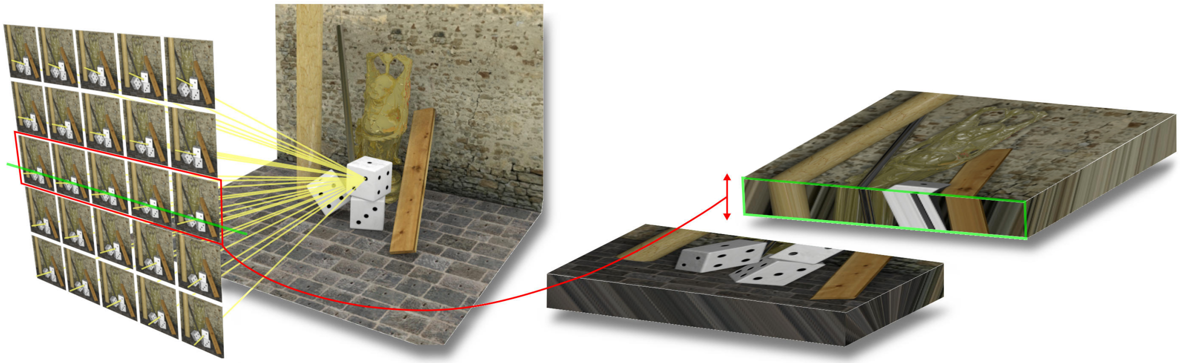

Figure 2. *One way to understand a 4D light field is as a collection of images of a scene, where the focal points of the cameras lie in a 2D plane. The rich structure becomes visible when one stacks all images along a line of view points on top of each other and considers a cut through this stack (denoted by the green border above). The 2D image in the plane of the cut is called an* epipolar plane image (EPI).

Contributions

In this work, we leverage the intrinsic geometry of 4D light fields to overcome problems of classical segmentation of single images. Typical examples are different objects with similar texture properties or identical objects on different spatial positions which cannot be discriminated by a classifier without geometry based features. The contribution of this work is threefold. First, we show that ray space based features enable a classifier to distinguish between objects similar in appearance. Second, we propose a variational multi-label optimization framework which makes use of the ray space regularizers in a related work [12] in order to obtain a consistent labeling over the complete ray space of the light field. To our knowledge, this is the first labeling framework which is designed to work on ray space. Third, we will provide all of our data sets online together with ground truth information for label assignments and depth where available, and thus establish the first benchmark data set for light field analysis.

2. Light field structure and parametrization

This paper builds upon our light field regularization framework, which is introduced in a related work [12]. We will first give a summary of the necessary ideas and notation here. Note that the first two subsections are almost verbatim copies, since they contain the basic definitions and are already formulated as compactly as possible.

Ray space

A 4D light field or Lumigraph is defined on a *ray space* $\mathcal{R}$, the set of rays passing through two planes Π and Ω in $\mathbb{R}^3$, where each ray can be uniquely identified by its two intersection points. For the sake of simplicity, we assume that both planes are parallel with distance $f > 0$, and equipped with 2D coordinate systems which are compatible in the sense that the base vectors are parallel and the origins lie on a line orthogonal to both planes.

The parametrization for ray space we choose is slightly different from the standard one for a Lumigraph [13], and inspired by [4]. A ray $R[s, t, x, y]$ is given by a point $(s, t) \in \Pi$ and $(x, y) \in \mathbb{R}^2$. The twist is that (x, y) is not a coordinate pair in Ω (as in the Lumigraph parametrization), but in the local coordinate system of the pinhole projection through (s, t) with image plane in Ω. This means that $R[s, t, 0, 0]$ is the ray which passes through the focal point (s, t) and the center of projection in the image plane, i.e. it is perpendicular to the two planes, see figure 3. In the following, coordinates (x, y) are always relative to the "base point" (s, t). We assume that the coordinate system of the pinhole view is chosen such that x is aligned with s and y is aligned with t, respectively.

Light fields and epipolar plane images

A light field L can now simply be defined as a function on ray space, either scalar or vector-valued for gray scale or color, respectively. Of particular interest are the images which emerge when ray space is restricted to a 2D plane. If we fix for example the two coordinates (y^*, t^*), the restriction L_{y^*,t^*} is the map

$$L_{y^*,t^*} : (x, s) \mapsto L(x, y^*, s, t^*), \qquad (1)$$

other restrictions are defined in a similar way. Note that L_{s^*,t^*} is the image of the pinhole view with center of projection (s^*, t^*). The images L_{y^*,t^*} and L_{x^*,s^*} are called *epipolar plane images*. They can be interpreted as horizontal or vertical cuts through a horizontal or vertical stack of the views in the light field, see figure 2, and have an interesting structure which seems to consist mainly of straight lines. The slope of the lines is linked to disparity, and determines correct regularization, as we will review now.

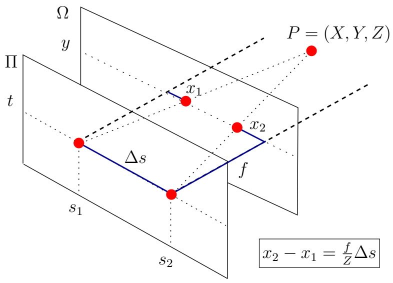

Figure 3. *Light field parametrization.* Each camera location (s, t) in the view point plane Π yields a different pinhole view of the scene. The two thick dashed black lines are orthogonal to both planes, and their intersection with the plane Ω marks the origins of the two different (x, y)-coordinate systems for the views (s_1, t) and (s_2, t), respectively.

Consistent functions on ray space

The planar camera movement leads to a linear dependency between the change of the view point and projection coordinates in the epipolar image plane. The rate of change depends on the depth of the scene point being projected, and is called the *disparity*. This dependency leads to the characteristic structure of epipolar plane images we have observed, since it implies that the projection of a 3D scene point in epipolar plane image space is a line. Previous works showed that this enables a very robust estimation of disparity on a light field, since line patterns can be detected without computing correspondences [9, 23].

In segmentation problems, when one wants to label rays according to e.g. the visible object class, the unknown function on ray space ultimately reflects a property of scene points. In consequence, all the rays which view the same scene point have to be assigned the same function value. Equivalent to this is to demand that the function must be consistent with the structure on the epipolar plane images. In particular, except at depth discontinuities, the value of such a function is not allowed to change in the direction of the epipolar lines, which are induced by the disparity field.

Regularization on ray space

The above considerations give rise to a regularizer $J_{\lambda\mu}(\boldsymbol{U})$ for vector-valued functions $\boldsymbol{U} : \mathcal{R} \to \mathbb{R}^n$ on ray space. It can be written as the sum of contributions for the regularizers on all epipolar plane images as well as all the views,

$$\begin{aligned} J_{\lambda\mu}(\boldsymbol{U}) &= \mu J_{xs}(\boldsymbol{U}) + \mu J_{yt}(\boldsymbol{U}) + \lambda J_{st}(\boldsymbol{U}) \\ \text{with } J_{xs}(\boldsymbol{U}) &= \int J_\rho(\boldsymbol{U}_{x^*,s^*})\, \mathrm{d}(x^*, s^*), \\ J_{yt}(\boldsymbol{U}) &= \int J_\rho(\boldsymbol{U}_{y^*,t^*})\, \mathrm{d}(y^*, t^*), \\ \text{and } J_{st}(\boldsymbol{U}) &= \int J_V(\boldsymbol{U}_{s^*,t^*})\, \mathrm{d}(s^*, t^*), \end{aligned} \tag{2}$$

where the anisotropic regularizers J_ρ act on 2D epipolar plane images, and are defined such that they encourage smoothing in the direction of the epipolar lines. This way, they enforce consistency of the function $\boldsymbol{U}$ with the epipolar plane image structure. For a detailed definition, we refer to our related work [12]. The spatial regularizer J_V encodes the label transition costs, as we will explore in more detail in the next section. Finally, the constants $\lambda > 0$ and $\mu > 0$ are user-defined and adjust the amount of regularization on the separate views and epipolar plane images, respectively.

3. Optimal label assignment on ray space

In this section, we introduce the new variational labeling framework on ray space. Its design is based on the representation of labels with indicator functions [6, 16, 25], which leads to a convex optimization problem. We can use the efficient optimization framework presented in [12] to obtain a globally optimal solution to the convex problem, however, as usual we need to project back to indicator functions and only end up within a (usually small) posterior bound of the optimum.

The variational multi-label problem

Let Γ be the (discrete) set of labels, then to each label $\gamma \in \Gamma$ we assign a binary function $u_\gamma : \mathcal{R} \to \{0, 1\}$ which takes the value one if and only if a ray is assigned the label γ. Since the assignment must be unique, the set of indicator functions must satisfy the simplex constraint

$$\sum_{\gamma \in \Gamma} u_\gamma = 1. \tag{3}$$

Arbitrary spatially varying label cost functions c_γ can be defined, which penalize the assignment of γ to a ray $R \in \mathcal{R}$ with the cost $c_\gamma(R) \geq 0$.

Let $\boldsymbol{U}$ be the vector of all indicator functions. To regularize $\boldsymbol{U}$, we choose $J_{\lambda\mu}$ defined in the previous section. This implies that the labeling is encouraged to be consistent with the epipolar plane structure of the light field to be labeled. The spatial regularizer J_V needs to enforce the label transition costs. For the remainder of this work, we choose a simple weighted Potts penalizer [24]

$$J_V(\boldsymbol{U}_{s^*,t^*}) := \frac{1}{2} \sum_{\gamma \in \Gamma} \int_\Omega g \left| (Du_\gamma)_{s^* t^*} \right| \, \mathrm{d}(x, y), \tag{4}$$

where g is a spatially varying transition cost. Since the total variation of a binary function equals the length of the interface between the zero and one level set due to the co-area formula [11], the factor $1/2$ leads to the desired penalization.

While we use the weighted Potts model in this work, the overall framework is by no means limited to it. Rather, we can use any of the more sophisticated regularizers proposed in the literature [6, 16], for example truncated linear penalization, Euclidean label distances, Huber TV or the Mumford-Shah regularizer. An overview as well as further specializations tailored to vector-valued label spaces can be found in [22].

The space of binary functions over which one needs to optimize is not convex, since convex combinations of binary functions are usually not binary. We resort to a convex relaxation, which with the above conventions can now be written as

$$\underset{\boldsymbol{U}\in\mathcal{C}}{\operatorname{argmin}} \left\{ J_{\lambda\mu}(\boldsymbol{U}) + \sum_{\gamma\in\Gamma} \int_{\mathcal{R}} c_\gamma u_\gamma \, \mathrm{d}(x,y,s,t) \right\}, \tag{5}$$

where $\mathcal{C}$ is the convex set of functions $\boldsymbol{U} = (u_\gamma : \mathcal{R} \to [0,1])_{\gamma\in\Gamma}$ which satisfy the simplex constraint (3). After optimization, the solution of (5) needs to be projected back onto the space of binary functions. This means that we usually do not achieve the global optimum of (5), but can only compute a posterior bound for how far we are from the optimal solution. An exception is the two-label case, where we indeed achieve global optimality via thresholding, since the anisotropic total variation also satisfies a co-area formula [25].

Optimization

Note that according to (2), the full regularizer $J_{\lambda\mu}$ which is defined on 4D ray space decomposes into a sum of 2D regularizers on the epipolar plane images and individual views, respectively. While solving a single saddle point problem for the full regularizer would require too much memory, it is feasible to iteratively compute independent descent steps for the data term and regularizer components.

The overall algorithm is detailed in [12]. Aside from the data term, the main difference here is the simplex constraint set for the primal variable $\boldsymbol{U}$. We enforce it with Lagrange multipliers in the proximity operators of the regularizer components, which can be easily integrated into the primal-dual algorithm [7]. An overview of the algorithm adapted to problem (5) can be found in figure 4.

On our system equipped with an nVidia GTX 580 GPU, optimization takes about 1.5 seconds per label in Γ and per million rays in $\mathcal{R}$, i.e. about 5 minutes for our rendered data sets if the result for all views is desired. If only the result for one single view (i.e. the center one) is required, computation can be restricted to view points located in a cross with that specific view at the center. The result will usually be very close to the optimization over the complete ray space. While this compromise forfeits some information in the data, it leads to significant speeds ups, for our rendered data sets to about 30 seconds.

To solve the **multi-label problem** (5) **on ray space**, we initialize the unknown vector-valued function $\boldsymbol{U}$ such that the indicator function for the optimal point-wise label is set to one, and zero otherwise. Then we iterate

- data term descent: $U_\lambda \leftarrow U_\lambda - \tau c_\lambda$ for all $\lambda \in \Lambda$,
- EPI regularizer descent:

$$\begin{aligned} \boldsymbol{U}_{x^*,s^*} &\leftarrow \operatorname{prox}_{\tau\mu J_\rho}(\boldsymbol{U}_{x^*,s^*}) \text{ for all } (x^*, s^*), \\ \boldsymbol{U}_{y^*,t^*} &\leftarrow \operatorname{prox}_{\tau\mu J_\rho}(\boldsymbol{U}_{y^*,t^*}) \text{ for all } (y^*, t^*), \end{aligned}$$

- spatial regularizer descent:

$$\boldsymbol{U}_{s^*,t^*} \leftarrow \operatorname{prox}_{\tau\lambda J_V}(\boldsymbol{U}_{s^*,t^*}) \text{ for all } (s^*, t^*).$$

The proximation operators prox_J compute subgradient descent steps for the respective 2D regularizer, and enforce the simplex constraint (3) for $\boldsymbol{U}$. The possible step size τ depends on the data term scale, in our experiments $\tau = 0.1$ led to reliable convergence within about 20 iterations.

Figure 4. *Algorithm for the general multi-label problem* (5).

4. Local class probabilities

We calculate the unary potentials c_γ in (5) from the negative log-likelihoods of the local class probabilities,

$$c_\gamma(R) = -\log p\big(\gamma|\mathbf{v}(R)\big), \tag{6}$$

so that by solving (5), we obtain the maximum a-posteriori (MAP) solution for the label assignment. The local class probabilities $p\big(\gamma|\mathbf{v}(R)\big) \in [0,1]$ for the label γ, conditioned on a local feature vector $\mathbf{v}(R) \in \mathbb{R}^{|F|}$ for each ray $R \in \mathcal{R}$, are obtained by training a classifier on a user-provided partial labeling of the center view. As features, we use a combination of color, Laplace operator of the view, intensity standard deviation in a neighborhood, Eigenvalues of the Hessian and the disparity computed on several scales.

While our framework allows the use of arbitrary classifiers, we specialize in this paper to a *Random Forest* [5]. These are becoming increasingly popular in image processing due to their wide applicability [8] and the robustness with regard to their hyper-parameters. Random Forests make use of *bagging* to reduce variance and avoid overfitting. A decision forest is built from a number n of trees, which are each trained from a random subset of the available training samples. In addition to bagging, extra randomness is injected into the trees by testing only a subset of $m < |F|$ different features for their optimal split in each split node. The above internal random forest parameters were fixed to $m = \sqrt{|F|}$ and $n = 71$ in our experiments.

Each individual tree is now built by partitioning the set of training samples recursively into smaller subsets, until the subsets become either class-pure or smaller then a given minimal split node size. The partitioning of the samples is achieved by performing a line search over all possible

	Classifier		
Features used	IMG	IMG-D	IMG-GT
RGB value	✓	✓	✓
Intensity standard deviation (in local neighbourhood)	✓	✓	✓
Eigenvalues of Hessian	✓	✓	✓
Laplace operator	✓		
Estimated disparity		✓	
Ground truth disparity			✓

Figure 5. *Combination of features used for the experiments in this paper. The individual scales of the features were determined via a grid search to find optimal parameters for each dataset individually.*

splits along a number of different feature axes for the optimal *Gini-impurity* of the resulting partitions, and repeating this process for the child partitions recursively. In each node, the chosen feature and the split value of that feature are stored.

After building a single tree, the class distribution of the samples in each leaf node is stored and used at prediction time to obtain the conditional class probability of samples that arrive at that particular leaf node. The leaf node with which a prediction-sample is associated is determined by comparing the nodes' split value for the split feature with the feature vector entry of a sample. Depending on whether the sample value is smaller (larger) than the node value, the sample is passed to the left (right) child of the split node, until a leaf node is reached.

Finally, the ensemble of decision tree classifiers is used to calculate the local class probability of unlabeled pixels by averaging their votes. In our experiments, we achieved total run-times for training and prediction between one and 5 minutes, depending on the size of the light field and the number of labels. However, we did not yet parallelize the local predictions, which is easily possible and would make computation a lot more efficient.

5. Experiments

In this section, we present the results of our multilabel segmentation framework on a variety of different data sets. To explore the full potential of our approach, we use computer graphics generated light fields rendered with the open source software Blender [3], which provides complete ground truth for depth values and labels. In addition, we show that the approach yields very good results on real world data obtained with a plenoptic camera and a gantry, respectively. A subset of the views in the real-world data sets were manually labeled in order to provide ground truth to quantify the results.

There are two main benefits of labeling in light fields. First, we demonstrate the usefulness of disparity as an additional feature for training a classifier, and second, we show the improvements from the consistent variational multilabel optimization on ray space.

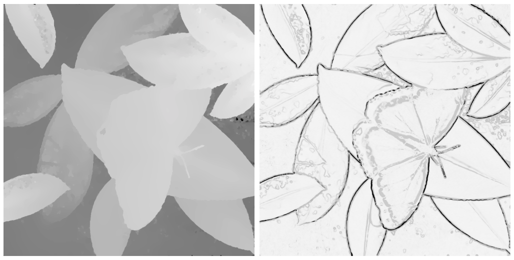

Figure 6. *Depth estimated using the method in [23] and spatial regularizer weight computed according to* (7) *for the light field view shown in figure 1.*

Disparity as a feature

The first step of our work flow does not differ from single image segmentation using a random forest. The user selects an arbitrary view from the light field, adds scribbles for the different labels, and chooses suitable features as well as the scales on which the features should be calculated. The classifier is then trained on this single view and, in a second step, used to compute local class probabilities for all views of the entire light field.

In advance, we have tested variations of common features for interactive image segmentation on our data sets to find a suitable combination of features which yields good results on single images. The optimal training parameters were determined using a grid search over the minimum split node size as well as the feature combinations and their scales for each data set individually. The number of different scales we used for each feature was fixed to four. This way, we can guarantee optimal results of the random forest classifier for all data sets and feature combinations, which ensures a meaningful assessment of the effects of our new ray space features.

Throughout the remainder of the paper, we use the three different sets of features detailed in figure 5. The classifier *IMG* uses only classical single-view features, while *IMG-D* and *IMG-GT* employ in addition estimated and ground truth disparity, respectively, the latter of course only if available. Estimated disparity maps were obtained using our method in [23] and are overall of very good quality, see figure 6. The achieved accuracy and the boundary recall for purely point-wise classification using the three classifiers above are listed in the table in figure 7. Sample segmentations for our data sets can be viewed in figure 9. It is clearly obvious that the features extracted from the light field improve the quality of a local classifier significantly for difficult problem instances.

	Classifier					
	IMG		IMG-D		IMG-GT	
Data set	acc	br	acc	br	acc	br
synthetic data sets						
Buddha	93.5	6.4	96.7	39.6	98.6	43.1
Garden	95.1	54.8	96.7	51.1	96.9	53.3
Papillon 1	98.6	59.3	98.3	57.4	99.0	78.9
Papillon 2	90.8	16.7	96.5	33.1	99.1	73.0
Horses 1	93.2	13.4	94.3	34.9	98.3	48.7
Horses 2	94.6	15.9	95.3	36.8	98.5	50.9
StillLife 1	98.6	36.3	98.7	41.2	98.9	45.3
StillLife 2	97.8	25.4	98.3	36.1	98.5	39.1
real-world data sets						
UCSD [26]	95.8	8.9	97.0	11.2	-	-
Plenoptic 1 [20]	93.7	3.5	94.5	4.4	-	-
Plenoptic 2 [20]	91.0	6.6	96.1	8.5	-	-

Figure 7. *Comparison of local labeling accuracy (acc) and boundary recall (br) for all datasets.* The table shows percentages of correctly labeled pixels and boundary pixels, respectively, for pointwise optimal results of the three classifiers trained on the features detailed in figure 5. Disparity for IMG-D is estimated using [23]. Ground truth disparity is used for IMG-GT to determine the maximum possible quality of the proposed method. It is obvious that in scenes like *Buddha*, *Papillon 2*, *Horses 2* or *StillLife 2*, where the user tries to separate objects with similiar or even identical appearance, the rayspace based feature leads to a large benefit in the segmentation results.

Global Optimization

In the second set of experiments, we employ our ray space optimization framework on the results from the local classifier. The unary potentials in (5) are initialized with the log-probabilities (6) from the local class probabilities, while the spatial regularization weight g is set to

$$g = \max\big\{0, 1 - \big(|\nabla I|_2 - \mathcal{H}(I)\big)\ |\nabla \rho|_2\big\}, \qquad (7)$$

where I denotes the respective single view image, $\mathcal{H}$ the Harris corner detector [14], and ρ the disparity field. This way, we combine the response from three different types of edge detectors. Experiments showed that the sum of the two different edge signals for the gray value image I leads to more robust boundary weights.

For all of the data sets, training classifiers with light field features and optimizing over ray space leads to significantly improved results compared to single view multi-labeling, see figures 8 and 9. The effectiveness of light field segmentation is revealed in particular on data sets which have highly ambiguous texture and color between classes.

In the light field *Buddha*, for example, it becomes possible to segment a column from a background wall having the same texture. In the scene *Papillon 2*, we demonstrate that it is possible to separate foreground from background leafs. Similarly, in *StillLife 2* we are able to correctly segment foreground from background raspberries. The data set *Horses 2* also represents a typical case for a problems only solvable using the proposed approach. Here, we perform a labeling of identical objects in the scene with different label classes.

6. Conclusion

If objects belonging to different classes do not vary much in appearance or if, even worse, identical objects appear in a scene, the corresponding segmentation problem usually cannot be solved with algorithms based on classical image features. In these cases, the fact that common images are only two-dimensional projections of the world leads to a loss of information which makes it impossible to distinguish between the classes.

In contrast, the light field of a scene densely samples rays from different view points, and thus not only implicitly encodes scene geometry, but also makes it possible to solve inverse problems consistently across all views by means of a relatively simple local prior. A light field thus allows to use the true geometric distance between pixels as an additional feature for a classifier, which increases the predictive power of a classifier enormously in many problematic cases. The consistent variational optimization for the label assignment is the first of its kind which is designed to work on ray space, and leads to significantly better results than a comparable single view multi-label framework.

As an additional contribution, we offer the data sets shown in this paper as the first light field benchmark with complete ground truth depth and label information available, to encourage competitive development of future algorithms.

References

[1] J. Berent and P. Dragotti. Segmentation of epipolar-plane image volumes with occlusion and disocclusion competition. In *IEEE 8th Workshop on Multimedia Signal Processing*, pages 182–185, 2006.

[2] T. Bishop and P. Favaro. The light field camera: Extended depth of field, aliasing, and superresolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):972–986, 2012.

[3] Blender Foundation. `www.blender.org`.

[4] R. Bolles, H. Baker, and D. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, 1987.

[5] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] A. Chambolle, D. Cremers, and T. Pock. A convex approach for computing minimal partitions. Technical Report TR-2008-05, Dept. of Computer Science, University of Bonn, 2008.

[7] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.*, 40(1):120–145, 2011.

[8] A. Criminisi. Decision forests: A unified framework for classification, regression, density estimation, manifold learning

Optimization	Single view (SV)						Ray space (RS)						Overall improvement	
Classifier	IMG		IMG+D		IMG+GT		IMG		IMG+D		IMG+GT		RS+IMG+D	RS+IMG+GT
	acc	imp	acc	imp	acc	imp	acc	imp	acc	imp	acc	imp	vs. SV+IMG	vs. SV+IMG
synthetic data sets														
Buddha	96.3	43.4	97.5	22.1	99.1	31.2	96.3	43.8	98.8	63.8	99.1	35.5	68.2	76.0
Garden	96.4	25.7	97.9	36.6	98.1	39.2	96.9	37.0	98.0	37.5	98.2	41.4	43.4	49.2
Papillon 1	99.1	34.6	99.1	45.5	99.3	31.7	99.3	46.3	99.2	50.9	99.7	65.3	7.9	60.1
Papillon 2	92.3	22.4	98.1	46.0	99.3	29.8	93.0	24.6	98.9	68.2	99.5	44.7	84.7	92.8
Horses 1	94.7	22.3	95.7	23.7	99.2	52.6	94.8	23.0	97.7	59.4	99.2	55.6	56.2	85.6
Horses 2	96.1	28.4	96.3	21.4	99.0	31.3	96.2	29.5	98.3	64.1	99.1	36.7	56.7	76.2
StillLife 1	99.1	38.2	99.3	50	99.4	45.6	99.2	43.8	99.4	54.5	99.6	64.0	31.5	53.4
StillLife 2	98.8	47.1	98.8	31.0	99.1	41.1	99.0	55.6	98.9	38.5	99.2	45.9	10.1	33.6
real-world data sets														
UCSD	97.6	44.3	99.1	70	–	–	97.8	48.6	99.3	76.3	–	–	69.9	–
Plenoptic 1	96.4	43.5	97.0	43.6	–	–	96.8	49.5	96.9	43.6	–	–	12.9	–
Plenoptic 2	94.1	34.4	96.1	33.2	–	–	94.5	39.4	96.1	33.9	–	–	34.6	–
Average	96.8	32.2	97.9	36.3	99.2	38.7	96.9	37.1	98.7	55.9	99.4	52.0	41.2	67.1

Figure 8. *Relative improvements by global optimization.* All numbers are in percent. The quantities in the columns *acc* indicate the percentage of correctly labeled pixels. The columns *imp* denote the relative improvement of the optimized compared to the respective raw result from the local classifier in figure 7. To be more specific, if acc_p is the previous and acc_n the new accuracy, then the column *imp* contains the number $(acc_n - acc_p)/(1 - acc_p)$, i.e. the percentage of previously erroneous pixels which were corrected by optimization. Optimal smoothing parameters λ, μ were determined using a grid search over the parameter space. We also compare our ray space optimization framework (RS) to single view optimization (SV), which can be achieved by setting the amount of EPI regularization μ to zero. Note that for every single classifier and data set, RS optimization achieves a better result than SV optimization. The last two columns indicate the relative accuracy of ray space optimization and the indicated ray space classifier versus single view optimization and single view features, computed the same way as the *imp* columns. In particular, they demonstrate the overall improvement which is realized with the proposed method.

and semi-supervised learning. *Foundations and Trends® in Computer Graphics and Vision*, 7(2-3):81–227, 2011.

[9] A. Criminisi, S. 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):51–85, 2005.

[10] S. Esedoglu and R. March. Segmentation with depth but without detecting junctions. *Journal of Mathematical Imaging and Vision*, 18(1):7–15, 2003.

[11] H. Federer. *Geometric measure theory*. Springer-Verlag New York Inc., New York, 1969.

[12] B. Goldluecke and S. Wanner. The variational structure of disparity and regularization of 4D light fields. In *Proc. International Conference on Computer Vision and Pattern Recognition*, 2013.

[13] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The Lumigraph. In *Proc. SIGGRAPH*, pages 43–54, 1996.

[14] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[15] A. Kowdle, S. Sinha, and R. Szeliski. Multiple view object cosegmentation using appearance and stereo cues. In *Proc. European Conference on Computer Vision*, 2012.

[16] J. Lellmann, F. Becker, and C. Schnörr. Convex optimization for multi-class image labeling with a novel family of total variation based regularizers. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[17] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, 2006.

[18] A. Lumsdaine and T. Georgiev. The focused plenoptic camera. In *In Proc. IEEE International Conference on Computational Photography*, pages 1–8, 2009.

[19] R. Ng. *Digital Light Field Photography*. PhD thesis, Stanford University, 2006. Note: thesis led to commercial light field camera, see also `www.lytro.com`.

[20] C. Perwass and L. Wietzke. The next generation of photography, 2010. `www.raytrix.de`.

[21] A. Stein, D. Hoiem, and M. Hebert. Learning to find object boundaries using motion cues. In *Proc. International Conference on Computer Vision*, 2007.

[22] E. Strekalovskiy, B. Goldluecke, and D. Cremers. Tight convex relaxations for vector-valued labeling problems. In *Proc. International Conference on Computer Vision*, 2011.

[23] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4D light fields. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 41–48, 2012.

[24] C. Zach, D. Gallup, J.-M. Frahm, and M. Niethammer. Fast global labeling for real-time stereo using multiple plane sweeps. In *Vision, Modeling and Visualization Workshop VMV 2008*, 2008.

[25] C. Zach, M. Niethammer, and J.-M. Frahm. Continuous maximal flows and Wulff shapes: Application to MRFs. In *Proc. International Conference on Computer Vision and Pattern Recognition*, 2009.

[26] M. Zwicker, W. Matusik, F. Durand, H. Pfister, and C. Forlines. Antialiasing for automultiscopic 3D displays. In *ACM Transactions on Graphics (Proc. SIGGRAPH)*, page 107. ACM, 2006.

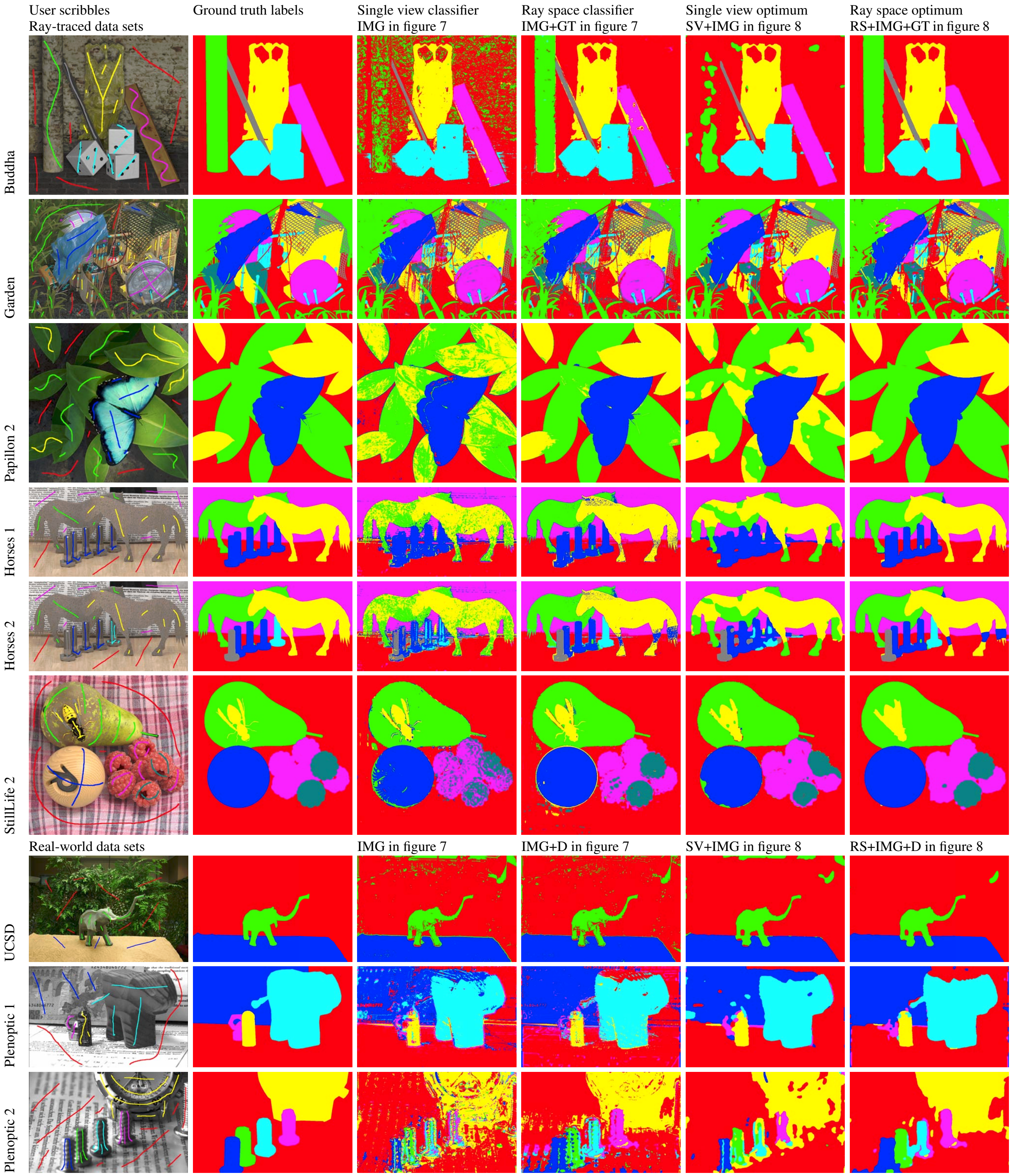

Figure 9. *Segmentation results for a number of ray-traced and real-world light fields.* The first two columns on the left show the center view with user scribbles and ground truth labels. The two middle columns compare classifier results for the local single view and light field features denoted on top. Since the focus of this paper is segmentation rather than depth reconstruction, here we show results for ground truth depth where available to compare to the optimal possible results from light field data. Finally, the two rightmost columns compare the final results after single view and ray space optimization, respectively. In particular for difficult cases, the proposed method is significantly superior.