

Occlusion-aware Depth Estimation Using Light-field Cameras

Ting-Chun Wang
UC Berkeley

tctwang0509@berkeley.edu

Alexei A. Efros
UC Berkeley

efros@eecs.berkeley.edu

Ravi Ramamoorthi
UC San Diego

ravir@cs.ucsd.edu

Abstract

Consumer-level and high-end light-field cameras are now widely available. Recent work has demonstrated practical methods for passive depth estimation from light-field images. However, most previous approaches do not explicitly model occlusions, and therefore cannot capture sharp transitions around object boundaries. A common assumption is that a pixel exhibits photo-consistency when focused to its correct depth, i.e., all viewpoints converge to a single (Lambertian) point in the scene. This assumption does not hold in the presence of occlusions, making most current approaches unreliable precisely where accurate depth information is most important – at depth discontinuities.

In this paper, we develop a depth estimation algorithm that treats occlusion explicitly; the method also enables identification of occlusion edges, which may be useful in other applications. We show that, although pixels at occlusions do not preserve photo-consistency in general, they are still consistent in approximately half the viewpoints. Moreover, the line separating the two view regions (correct depth vs. occluder) has the same orientation as the occlusion edge has in the spatial domain. By treating these two regions separately, depth estimation can be improved. Occlusion predictions can also be computed and used for regularization. Experimental results show that our method outperforms current state-of-the-art light-field depth estimation algorithms, especially near occlusion boundaries.

1. Introduction

Light-field cameras from Lytro [3] and Raytrix [18] are now available for consumer and industrial use respectively, bringing to fruition early work on light-field rendering [10, 15]. An important benefit of light-field cameras for computer vision is that multiple viewpoints or sub-apertures are available in a single light-field image, enabling passive depth estimation [4]. Indeed, Lytro Illum and Raytrix software produces depth maps used for tasks like refocusing after capture, and recent work [20] shows how multiple cues like defocus and correspondence can be combined.

However, very little work has explicitly considered oc-

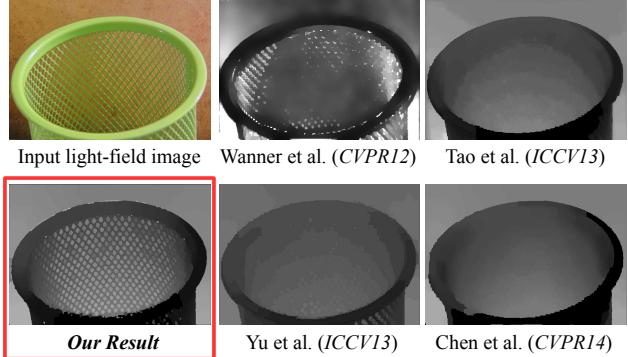


Figure 1: Comparison of depth estimation results of different algorithms from a light-field input image. Darker represents closer and lighter represents farther. It can be seen that only our occlusion-aware algorithm successfully captures most of the holes in the basket, while other methods either smooth over them, or have artifacts as a result.

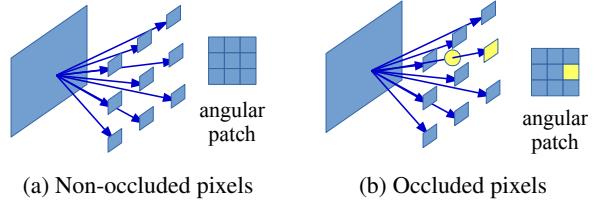


Figure 2: Non-occluded vs. occluded pixels. (a) At non-occluded pixels, all view rays converge to the same point in the scene if refocused to the correct depth. (b) However, photo-consistency fails to hold at occluded pixels, where some view rays will hit the occluder.

clusion. A common assumption is that, when refocused to the correct depth (the depth of the center view), angular pixels corresponding to a single spatial pixel represent viewpoints that converge to one point in the scene. If we collect these pixels into an *angular patch* (Eq. 6), they exhibit photo-consistency for Lambertian surfaces, which means they all share the same color (Fig. 2a). However, this assumption is not true when occlusions occur at a pixel; photo-consistency no longer holds (Fig. 2b). Enforcing photo-consistency on these pixels will often lead to incorrect depth results, causing smooth transitions around sharp

occlusion boundaries.

In this paper, we explicitly model occlusions, by developing a modified version of the photo-consistency condition on angular pixels. Our main contributions are:

1. An occlusion prediction framework on light-field images that uses a modified angular photo-consistency.
2. A robust depth estimation algorithm which explicitly takes occlusions into account.

We show (Sec. 3) that around occlusion edges, the angular patch can be divided into two regions, where only one of them obeys photo-consistency. A key insight (Fig. 3) is that the line separating the two regions in the *angular domain* (correct depth vs. occluder) has the same orientation as the occlusion edge does in the *spatial domain*. This observation is specific to light-fields, which have a dense set of views from a planar camera array or set of sub-apertures. Standard stereo image pairs (nor general multi-view stereo configurations) do not directly satisfy the model.

We use the modified photo-consistency condition, and the means/variances in the two regions, to estimate initial occlusion-aware depth (Sec. 4). We also compute a predictor for the occlusion boundaries, that can be used as an input to determine the final regularized depth (Sec. 5). These occlusion boundaries could also be used for other applications like segmentation or recognition. As seen in Fig. 1, our depth estimates are more accurate in scenes with complex occlusions (previous results smooth object boundaries like the holes in the basket). In Sec. 6, we present extensive results on both synthetic data (Figs. 9, 10), and on real scenes captured with the consumer Lytro Illum camera (Fig. 11), demonstrating higher-quality depth recovery than previous work [8, 20, 22, 26].

2. Related Work

(Multi-View) Stereo with Occlusions: Multi-view stereo matching has a long history, with some efforts to handle occlusions. For example, the graph-cut framework [12] used an occlusion term to ensure visibility constraints while assigning depth labels. Woodford et al. [25] imposed an additional second order smoothness term in the optimization, and solved it using Quadratic Pseudo-Boolean Optimization [19]. Based on this, Bleyer et al. [5] assumed a scene is composed of a number of smooth surfaces and proposed a soft segmentation method to apply the asymmetric occlusion model [24]. However, significant occlusions still remain difficult to address even with a large number of views.

Depth from Light-Field Cameras: Perwass and Wietzke [18] proposed using correspondence techniques to estimate depth from light-field cameras. Tao et al. [20] combined correspondence and defocus cues in the 4D Epipolar Image (EPI) to complement the disadvantages of each other. Neither method explicitly models occlusions. McCloskey [16] proposed a method to remove partial occlusion

in color images, which does not estimate depth. Wanner and Goldluecke [22] proposed a globally consistent framework by applying structure tensors to estimate the directions of feature pixels in the 2D EPI. Yu et al. [26] explored geometric structures of 3D lines in ray space and encoded the line constraints to further improve the reconstruction quality. However, both methods are vulnerable to heavy occlusion: the tensor field becomes too random to estimate, and 3D lines are partitioned into small, incoherent segments. Kim et al. [11] adopted a fine-to-coarse framework to ensure smooth reconstructions in homogeneous areas using dense light-fields. We build on the method by Tao et al. [20], which works with consumer light-field cameras, to improve depth estimation by taking occlusions into account.

Chen et al. [8] proposed a new bilateral metric on angular pixel patches to measure the probability of occlusions by their similarity to the central pixel. However, as noted in their discussion, their method is biased towards the central view as it uses the color of the central pixel as the mean of the bilateral filter. Therefore, the bilateral metric becomes unreliable once the input images get noisy. In contrast, our method uses the mean of about half the pixels as the reference, and is thus more robust when the input images are noisy, as shown in our result section.

3. Light-Field Occlusion Theory

We first develop our new light-field occlusion model, based on the physical image formation. We show that at occlusions, some of the angular patch remains photo-consistent, while the other part comes from occluders and exhibits no photo consistency. By treating these two regions separately, occlusions can be better handled.

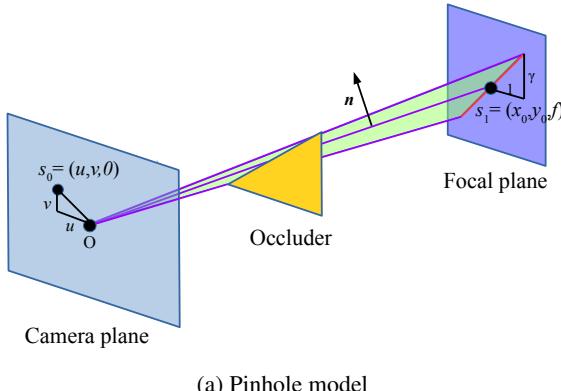
For each pixel on an occlusion edge, we assume it is occluded by only one occluder among all views. We also assume that we are looking at a spatial patch small enough, so that the occlusion edge around that pixel can be approximated by a line. We show that if we refocus to the occluded plane, the angular patch will still have photo-consistency in a subset of the pixels (unoccluded). Moreover, the edge separating the unoccluded and occluded pixels in the angular patch has the same orientation as the occlusion edge in the spatial domain (Fig. 3). In Secs. 4 and 5, we use this idea to develop a depth estimation and regularization algorithm.

Consider a pixel at (x_0, y_0, f) on the imaging focal plane (the plane in focus), as shown in Fig. 3a. An edge in the central pinhole image with 2D slope γ corresponds to a plane P in 3D space (the green plane in Fig. 3a). The normal \mathbf{n} to this plane can be obtained by taking the cross-product,

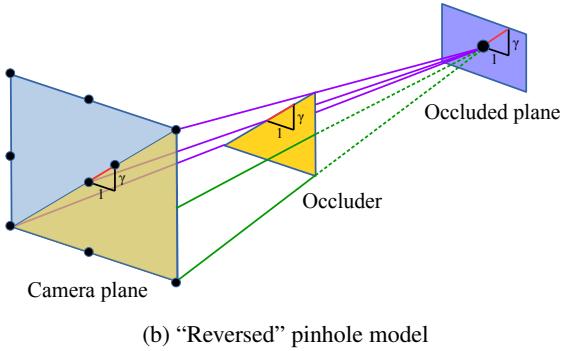
$$\mathbf{n} = (x_0, y_0, f) \times (x_0 + 1, y_0 + \gamma, f) = (-\gamma f, f, \gamma x_0 - y_0). \quad (1)$$

Note that we do not need to normalize the vector. The plane equation is $P(x, y, z) \equiv \mathbf{n} \cdot (x_0 - x, y_0 - y, f - z) = 0$,

$$P(x, y, z) \equiv \gamma f(x - x_0) - f(y - y_0) + (y_0 - \gamma x_0)(z - f) = 0. \quad (2)$$



(a) Pinhole model



(b) ‘Reversed’ pinhole model

Figure 3: Light-field occlusion model. (a) Pinhole model for central camera image formation. An occlusion edge on the imaging plane corresponds to an occluding plane in the 3D space. (b) The ‘reversed’ pinhole model for light-field formation. It can be seen that when we refocus to the occluded plane, we get a projection of the occluder on the camera plane, forming a reversed pinhole camera model.

In our case, one can verify that $\mathbf{n} \cdot (x_0, y_0, f) = 0$ so a further simplification to $\mathbf{n} \cdot (x, y, z) = 0$ is possible,

$$P(x, y, z) \equiv \gamma f x - f y + (y_0 - \gamma x_0) z = 0. \quad (3)$$

Now consider the occluder (yellow triangle in Fig. 3a). The occluder intersects $P(x, y, z)$ with $z \in (0, f)$ and lies on one side of that plane. Without loss of generality, we can assume it lies in the half-space $P(x, y, z) \geq 0$. Now consider a point $(u, v, 0)$ on the camera plane (the plane where the camera array lies on). To avoid being shadowed by the occluder, the line segment connecting this point and the pixel (x_0, y_0, f) on the image must not hit the occluder,

$$P(\mathbf{s}_0 + (\mathbf{s}_1 - \mathbf{s}_0)t) \leq 0 \quad \forall t \in [0, 1], \quad (4)$$

where $\mathbf{s}_0 = (u, v, 0)$ and $\mathbf{s}_1 = (x_0, y_0, f)$. When $t = 1$, $P(\mathbf{s}_1) = 0$. When $t = 0$,

$$P(\mathbf{s}_0) \equiv \gamma f u - f v \leq 0. \quad (5)$$

The last inequality is satisfied if $v \geq \gamma u$, i.e., the critical slope on the angular patch $v/u = \gamma$ is the same as the edge orientation in the spatial domain. If the inequality above is



(a) Occlusion in central view (b) Occlusion in other views

Figure 4: Occlusions in different views. The insets are the angular patches of the red pixels when refocused to the correct depth. At the occlusion edge in the central view, the angular patch can be divided evenly into two regions, one with photo-consistency and one without. However, for pixels around the occlusion edge, although the central view is not occluded, some other views will still get occluded. Hence, the angular patch will not be photo-consistent, and will be unevenly divided into occluded and visible regions.

satisfied, both endpoints of the line segment lie on the other side of the plane, and hence the entire segment lies on that side as well. Thus, the light ray will not be occluded.

We also give an intuitive explanation of the above proof. Consider a plane being occluded by an occluder, as shown in Fig. 3b. Consider a simple 3×3 camera array. When we refocus to the occluded plane, we can see that some views are occluded by the occluder. Moreover, the occluded cameras on the camera plane are the projection of the occluder on the camera plane. Thus, we obtain a ‘reversed’ pinhole camera model, where the original imaging plane is replaced by the camera plane, and the original pinhole becomes the pixel we are looking at. When we collect pixels from different cameras to form an angular patch, the edge separating the two regions will correspond to the same edge the occluder has in the spatial domain.

Therefore, we can predict the edge orientation in the angular domain using the edge in the spatial image. Once we divide the patch into two regions, we know photo consistency holds in one of them since they all come from the same (assumed to be Lambertian) spatial pixel.

4. Occlusion-Aware Initial Depth Estimation

In this section, we show how to modify the initial depth estimation from Tao et al. [20], based on the theory above. First, we apply edge detection on the central view image. Then for each edge pixel, we compute initial depths using a modified photo-consistency constraint. The next section will discuss computation of refined occlusion predictors and regularization to generate the final depth map.

Edge detection: We first apply Canny edge detection on the central view (pinhole) image. Then an edge orientation predictor is applied on the obtained edges to get the orientation angles at each edge pixel. These pixels are candidate

occlusion pixels in the central view. However, some pixels are not occluded in the central view, but are occluded in other views, as shown in Fig. 4, and we want to mark these as candidate occlusions as well. We identify these pixels by dilating the edges detected in the center view.

Depth Estimation: For each pixel, we refocus to various depths using a 4D shearing of the light-field data [17],

$$L_\alpha(x, y, u, v) = L\left(x + u\left(1 - \frac{1}{\alpha}\right), y + v\left(1 - \frac{1}{\alpha}\right), u, v\right), \quad (6)$$

where L is the input light-field image, α is the ratio of the refocused depth to the currently focused depth, L_α is the refocused light-field image, (x, y) are the spatial coordinates, and (u, v) are the angular coordinates. The central viewpoint is located at $(u, v) = (0, 0)$. This gives us an angular patch for each depth, which can be averaged to give a refocused pixel.

When an occlusion is not present at the pixel, the obtained angular patch will have photo-consistency, and hence exhibits small variance and high similarity to the central view. For pixels that are not occlusion candidates, we can simply compute the variance and the mean of this patch to obtain the correspondence and defocus cues, similar to the method by Tao et al. [20].

However, if an occlusion occurs, photo-consistency will no longer hold. Instead of dealing with the entire angular patch, we divide the patch into two regions. The angular edge orientation separating the two regions is the same as in the spatial domain, as proven in Sec. 3. Since at least half the angular pixels come from the occluded plane (otherwise it will not be seen in the central view), we place the edge passing through the central pixel, dividing the patch evenly. Note that only one region, corresponding to the partially occluded plane focused to the correct depth, exhibits photo-consistency. The other region contains angular pixels that come from the occluder, which is not focused at the proper depth, and might also contain some pixels from the occluded plane. We therefore replace the original patch with the region that has the minimum variance to compute the correspondence and defocus cues.

To be specific, let (u_1, v_1) and (u_2, v_2) be the angular coordinates in the two regions, respectively. We first compute the means and the variances of the two regions,

$$\bar{L}_{\alpha,j}(x, y) = \frac{1}{N_j} \sum_{u_j, v_j} L_\alpha(x, y, u_j, v_j), \quad j = 1, 2 \quad (7)$$

$$V_{\alpha,j}(x, y) = \frac{1}{N_j - 1} \sum_{u_j, v_j} (L_\alpha(x, y, u_j, v_j) - \bar{L}_{\alpha,j}(x, y))^2, \quad (8)$$

where N_j is the number of pixels in region j . Let

$$i = \arg \min_{j=1,2} \{V_{\alpha,j}(x, y)\} \quad (9)$$

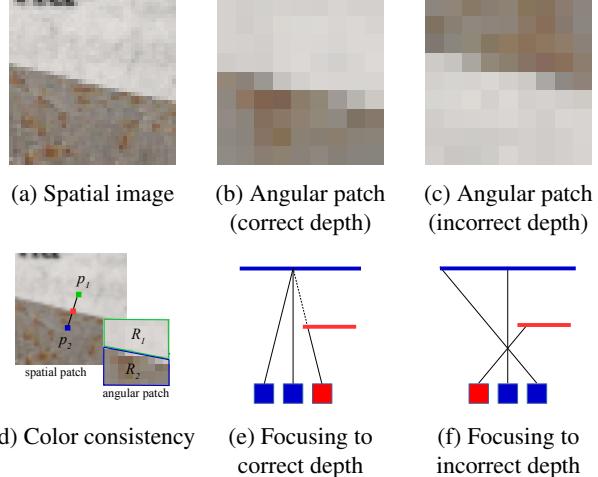


Figure 5: *Color consistency constraint.* (b)(e) We can see that when we refocus to the correct depth, we get low variance in half the angular patch. However, in (c)(f) although we refocused to an incorrect depth, it still gives low variance response since the occluded plane is very textureless, so we get a “reversed” angular patch. To address this, we add another constraint that p_1 and p_2 should be similar to the averages of R_1 and R_2 in (d), respectively.

be the index of the region that exhibits smaller variance. Then the correspondence response is given by

$$C_\alpha(x, y) = V_{\alpha,i}(x, y) \quad (10)$$

Similarly, the defocus response is given by

$$D_\alpha(x, y) = (\bar{L}_{\alpha,i}(x, y) - L(x, y, 0, 0))^2 \quad (11)$$

Finally, the optimal depth is determined as

$$\alpha^*(x, y) = \arg \min_{\alpha} \{C_\alpha(x, y) + D_\alpha(x, y)\} \quad (12)$$

Color Consistency Constraint: When we divide the angular patch into two regions, it is sometimes possible to obtain a “reversed” patch when we refocus to an incorrect depth, as shown in Fig. 5. If the occluded plane is very textureless, this depth might also give a very low variance response, even though it is obviously incorrect. To address this, we add a color consistency constraint that the averages of the two regions should have a similar relationship with respect to the current pixel as they have in the spatial domain. Mathematically,

$$|\bar{L}_{\alpha,1} - p_1| + |\bar{L}_{\alpha,2} - p_2| < |\bar{L}_{\alpha,2} - p_1| + |\bar{L}_{\alpha,1} - p_2| + \delta, \quad (13)$$

where p_1 and p_2 are the values of the pixels shown in Fig. 5d, and δ is a small value (threshold) to increase robustness. If refocusing to a depth violates this constraint, this depth is considered invalid, and is automatically excluded in the depth estimation process.

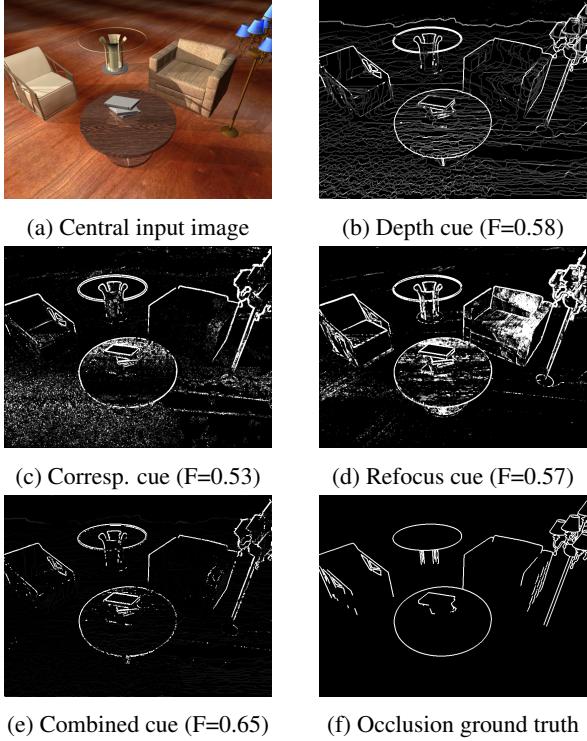


Figure 6: *Occlusion Predictor (Synthetic Scene)*. The intensities are adjusted for better contrast. F-measure is the harmonic mean of precision and recall compared to the ground truth. By combining three cues from depth, correspondence and refocus, we can obtain a better prediction of occlusions.

5. Occlusion-Aware Depth Regularization

After the initial local depth estimation phase, we refine the results with global regularization using a smoothness term. We improve on previous methods by reducing the effect of the smoothness/regularization term in occlusion regions. Our occlusion predictor, discussed below, may also be useful independently for other vision applications.

Occlusion Predictor Computation: We compute a predictor P_{occ} for whether a particular pixel is occluded, by combining cues from depth, correspondence and refocus.

1. Depth Cues: First, by taking the gradient of the initial depth, we can obtain an initial occlusion boundary,

$$P_{\text{occ}}^d = f(\nabla d_{\text{ini}}/d_{\text{ini}}) \quad (14)$$

where d_{ini} is the initial depth, and $f(\cdot)$ is a robust clipping function that saturates the response above some threshold. We divide the gradient by d_{ini} to increase robustness since for the same normal, the depth change across pixels becomes larger as the depth gets larger.

2. Correspondence Cues: In occlusion regions, we have already seen that photo-consistency will only be valid

in approximately half the angular patch, with a small variance in that region. On the other hand, the pixels in the other region come from different points on the occluding object, and thus exhibit much higher variance. By computing the ratio between the two variances, we can obtain an estimate of how likely the current pixel is to be at an occlusion,

$$P_{\text{occ}}^{\text{var}} = f\left(\max\left\{\frac{V_{\alpha^*,1}}{V_{\alpha^*,2}}, \frac{V_{\alpha^*,2}}{V_{\alpha^*,1}}\right\}\right). \quad (15)$$

where α^* is the initial depth we get.

3. Refocus Cues: Finally, note that the variances in both the regions will be small if the occluder is textureless. To address this issue, we also compute the means of both regions. Since the two regions come from different objects, their colors should be different, so a large difference between the two means also indicates a possible occlusion occurrence. In other words,

$$P_{\text{occ}}^{\text{avg}} = f(|\bar{L}_{\alpha^*,1} - \bar{L}_{\alpha^*,2}|) \quad (16)$$

Finally, we compute the combined occlusion response or prediction by the product of these three cues,

$$P_{\text{occ}} = \mathcal{N}(P_{\text{occ}}^d) \cdot \mathcal{N}(P_{\text{occ}}^{\text{var}}) \cdot \mathcal{N}(P_{\text{occ}}^{\text{avg}}) \quad (17)$$

where $\mathcal{N}(\cdot)$ is a normalization function that subtracts the mean and divides by the standard deviation.

Depth Regularization: Finally, given initial depth and occlusion cues, we regularize with a Markov Random Field (MRF) for a final depth map. We minimize the energy:

$$E = \sum_p E_{\text{unary}}(p, d(p)) + \sum_{p,q} E_{\text{binary}}(p, q, d(p), d(q)). \quad (18)$$

where d is the final depth and p, q are neighboring pixels. We adopt the unary term similar to Tao et al. [20]. The binary energy term is defined as

$$\begin{aligned} E_{\text{binary}}(p, q, d(p), d(q)) = & \\ & \exp\left[-(d(p) - d(q))^2/(2\sigma^2)\right] \\ & /(|\nabla I(p) - \nabla I(q)| + k|P_{\text{occ}}(p) - P_{\text{occ}}(q)|) \end{aligned} \quad (19)$$

where ∇I is the gradient of the central pinhole image, and k is a weighting factor. The numerator encodes the smoothness constraint, while the denominator reduces the strength of the constraint if two pixels are very different or an occlusion is likely to be between them. The minimization is solved using a standard graph cut algorithm [6, 7, 13]. We can then apply the occlusion prediction procedure again on this regularized depth map. A sample result is shown in Fig. 6. In this example, the F-measure (harmonic mean of precision and recall compared to ground truth) increased from 0.58 (depth cue), 0.53 (correspondence cue), and 0.57 (refocus cue), to 0.65 (combined cue).

6. Results

We compare our results to the methods by Wanner et al. [22], Tao et al. [20], Yu et al. [26], and Chen et al. [8]. For Chen et al., since code is not available, we used our own implementation. We perform extensive tests using the synthetic dataset created by Wanner et al. [23] as well as new scenes modeled by us. Our dataset is generated from 3ds-Max [1] using models from the Stanford Computer Graphics Laboratory [9, 14, 21] and models freely available online [2]. Upon publication of this work, the dataset will be available online. While the dataset by [23] only provides ground truth depth, ours provides ground truth depth, normals, specularity, lighting, etc, which we believe will be useful for a wider variety of applications. In addition to synthetic datasets, we also validate our algorithm on real-world scenes taken by the Lytro Illum camera.

Occlusion Boundaries: For each synthetic scene, we compute the occlusion boundaries from the depth maps generated by each algorithm, and report their precision-recall curves. For our method, the occlusions are computed using only the depth cue instead of the combined cue in Sec. 5, to compare the depth quality only. The results on both synthetic datasets are shown in Figs. 8a,8b. Our algorithm achieves better performance than current state-of-the-art methods. Next, we validate the robustness of our system by adding noise to a test image, and report the F-measure values of each algorithm, as shown in Fig. 8c. Although Chen et al. [8] performs very well in the absence of noise, their quality quickly degrades as the noise level is increased. In contrast, our algorithm is more tolerant to noise.

Depth Maps for Synthetic Scenes: Figure 9 shows the recovered depths on the synthetic dataset by Wanner et al. [23]. It can be seen that our results show fewer artifacts in heavily occluded areas. We obtain the correct shape of the door and window in the top row, and accurate boundaries along the twig and leaf in the bottom row. Other methods smooth the object boundaries and are noisy in some regions. Figure 10 shows the results on our synthetic dataset. Notice that we capture the boundaries of the leaves, and fine structures like the lamp and holes in the chair. Other methods smooth over these occlusions or generate thicker structures. The RMSE of the depth maps are shown in Table 1.

Depth Maps for Real Scenes: Figures 1 and 11 compare results on real scenes with fine structures and occlusions, captured with Lytro Illum light-field camera. Our method performs better around occlusion boundaries, especially for thin objects. Ours is the only method that captures the basket holes in Fig. 1. In Fig. 11, our method properly captures the thin structure in the top row, reproduces the spokes of the wheel (second row) without over-smoothing, captures the fine structure of the flower (third row), and reproduces the complicate shape of the chair (last row).



(a) Small area occlusion (b) Multi-occluder occlusion

Figure 7: *Limitations.* The upper insets show close-ups of the red rectangle, while the lower insets show the angular patches of the green (central) pixels when refocused to the correct depth. If (a) the occluded area is very small, or (b) more than one occluder is present, there is no simple line that can separate the angular patch into two regions.

Wanner et al.	Tao et al.	Yu et al.	Chen et al.	Ours
0.0470	0.0453	0.0513	0.0375	0.0355
0.1104	0.1098	0.0919	0.0925	0.0848

Table 1: *Depth RMSE on synthetic scenes.* The first row is the result on dataset by Wanner et al., and the second row is on our dataset. Our method achieves lowest RMSE on both datasets. Note that RMSE is not the best metric for the improvements on thin structures provided by our method.

Limitations and Future Work: Our algorithm cannot handle situations where the occluded plane is very small, or if the single occluder assumption fails to hold (Fig. 7). If the occluded area is very small, there is no simple line that can separate the angular patch into two regions. If we have multiple edges intersecting at a point, its angular patch needs to be divided into more than two regions to achieve photo consistency. This may be addressed by inspecting the spatial patch around the current pixel instead of just looking at the edges. Finally, our algorithm cannot perform well in textureless regions just like other methods.

7. Conclusion

In this paper, we propose an occlusion-aware depth estimation algorithm. We show that although pixels around occlusions do not exhibit photo-consistency in the angular patch when refocused to the correct depth, they are still photo-consistent for part of the patch. Moreover, the line separating the two regions in the angular domain has the same orientation as the edge in the spatial domain. Utilizing this information, the depth estimation process can be improved in two ways. *First*, we can enforce photo-consistency on only the region that is coherent. *Second*, by exploiting depth, correspondence and refocus cues, we can perform occlusion prediction, so smoothing over these boundaries can be avoided in the regularization. We demonstrate the benefits of our algorithm on various synthetic datasets as well as real-world images with fine structures.

Acknowledgement

We acknowledge support from ONR grant N00014-15-1-2013, a Berkeley Fellowship, Intel, and funding from Sony to the UC San Diego Center for Visual Computing.

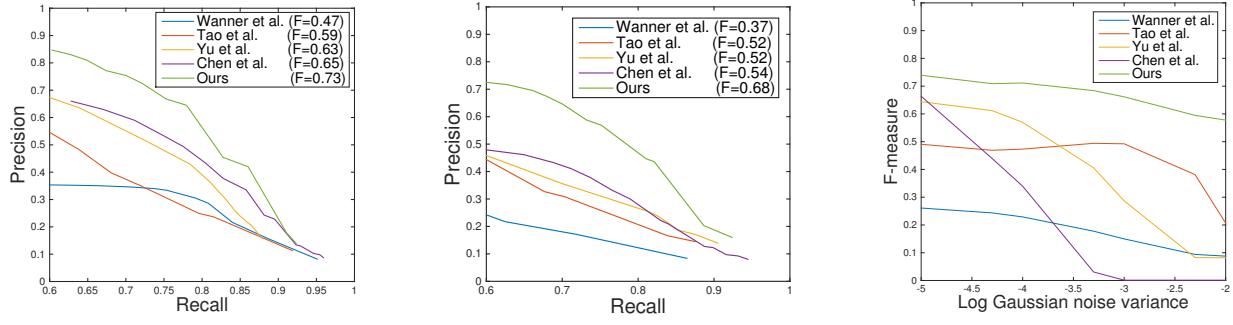


Figure 8: (a) PR-curve of occlusion boundaries on dataset of Wanner et al. [23] (b) PR-curve on our dataset. (c) F-measure vs. noise level. Our method achieves better results than current state-of-the-art methods, and is robust to noise.

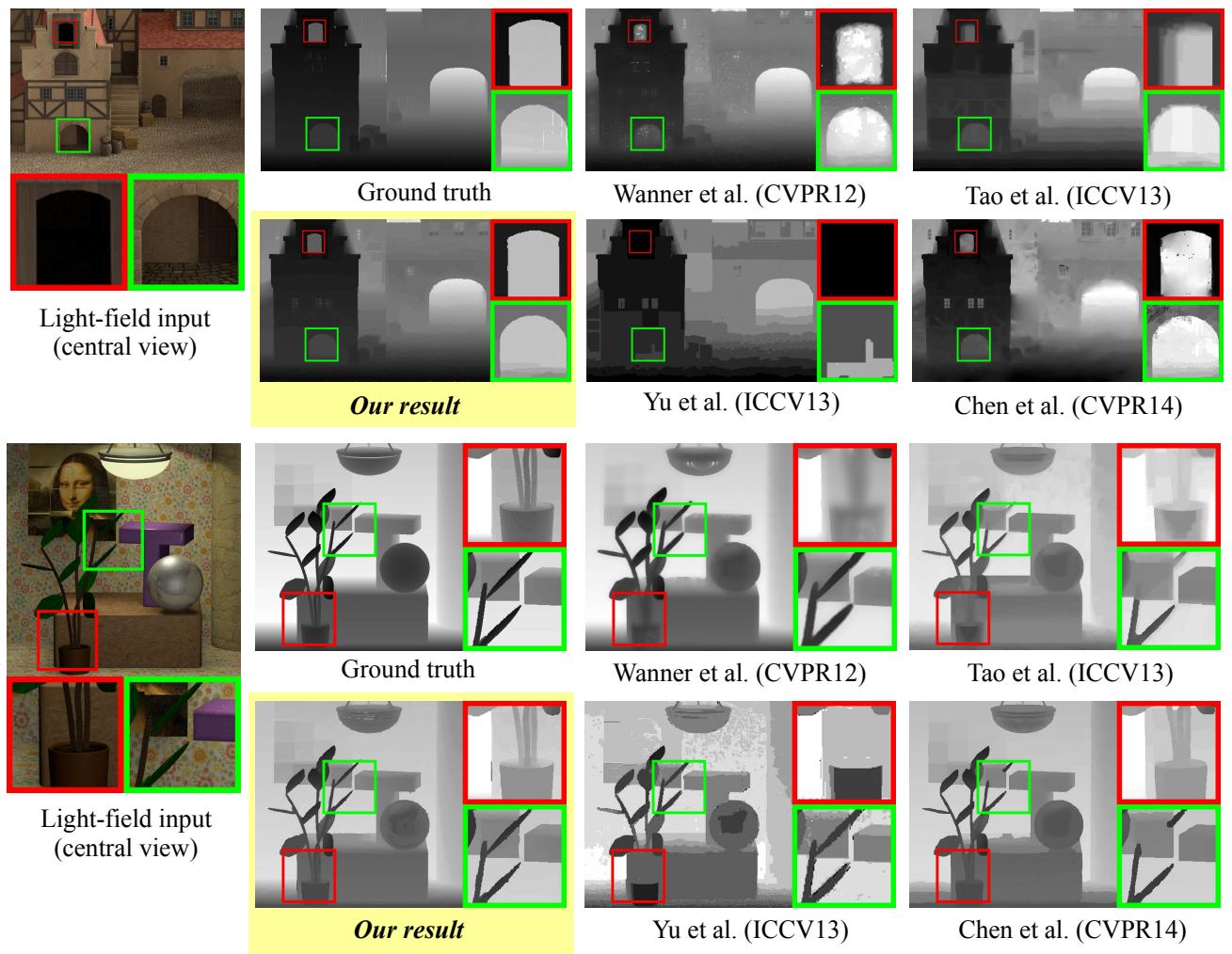


Figure 9: Depth estimation results on synthetic data by Wanner et al. [23]. Some intensities in the insets are adjusted for better contrast. In the first example, note that our method correctly captures the shape of the door/window, while all other algorithms fail and produce smooth transitions. Similarly, in the second example our method reproduces accurate boundaries along the twig/leaf, while other algorithms generate smoothed results or fail to capture the details, and have artifacts.

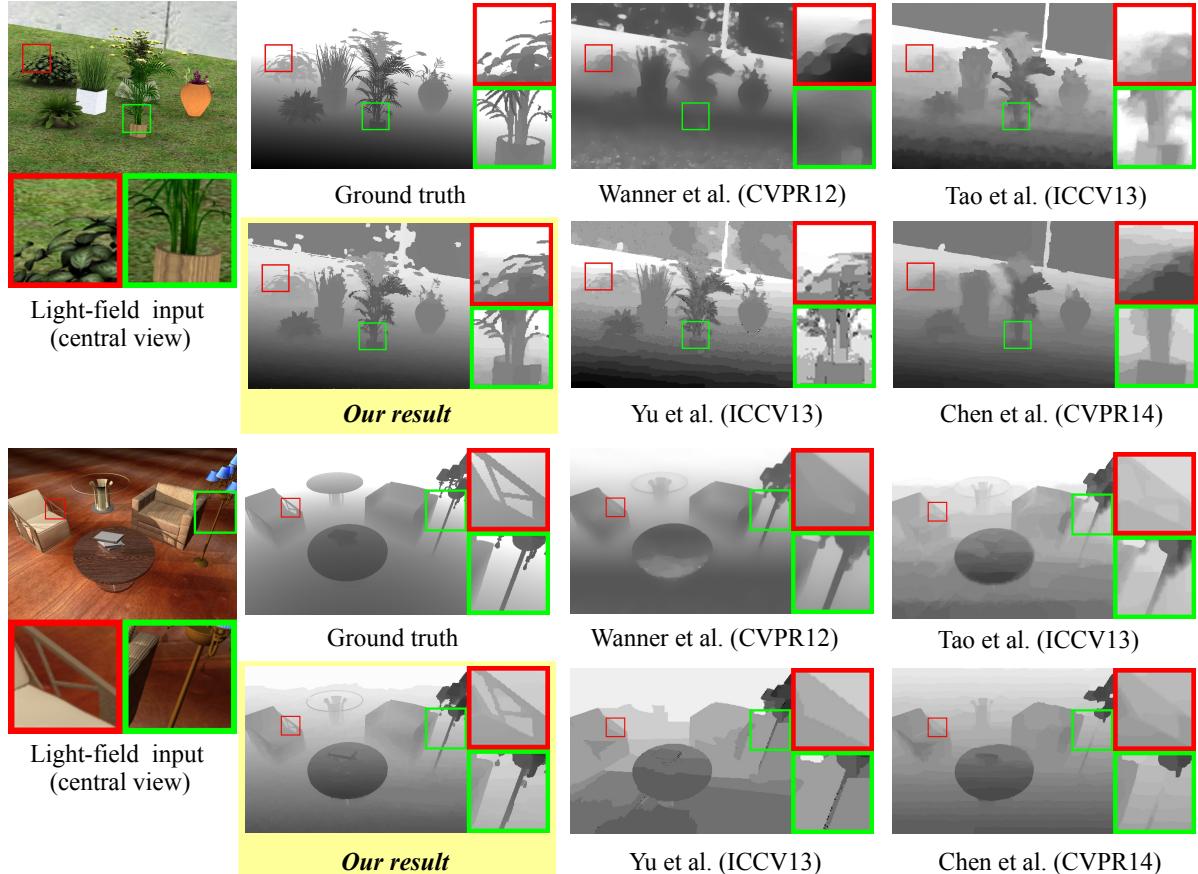


Figure 10: *Depth estimation results on our synthetic dataset. Some intensities in the insets are adjusted for better contrast. In the first example, our method successfully captures the shapes of the leaves, while all other methods generate smoothed results. In the second example, our method captures the holes in the chair as well as the thin structure of the lamp, while other methods obtain smoothed or thicker structures.*

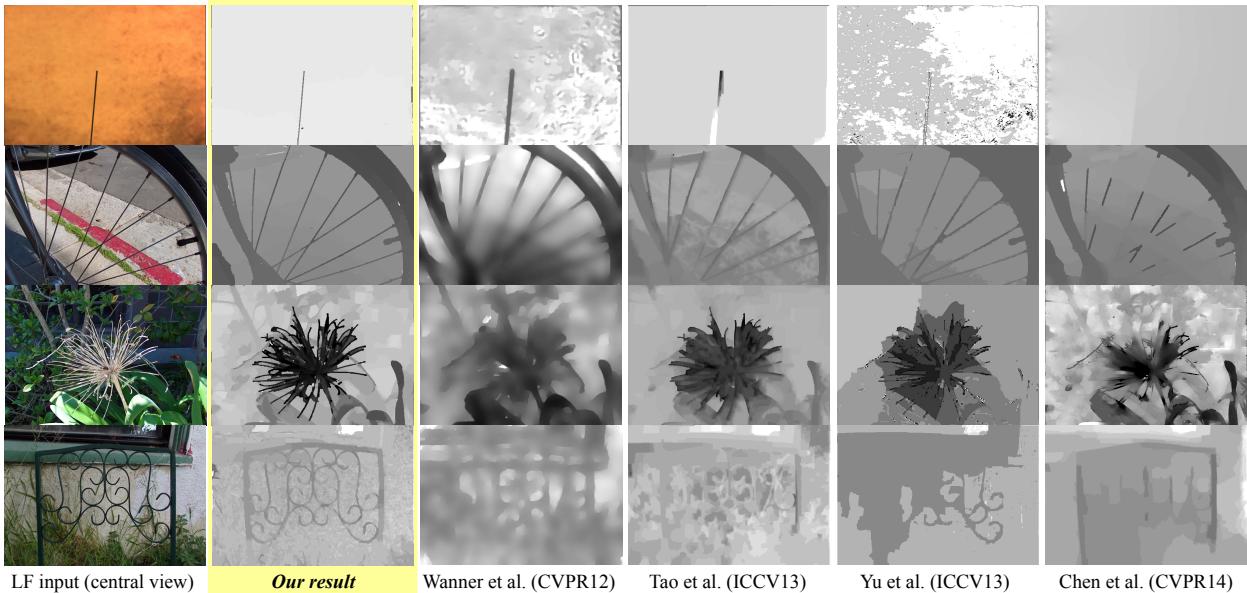


Figure 11: *Depth estimation results on real data taken by the Lytro Illum light-field camera. It can be seen that our method realistically captures the thin structures and occlusion boundaries, while other methods fail, or generate dilated structures.*

References

- [1] 3d modeling, animation, and rendering software. <http://www.autodesk.com/products/3ds-max>. ⁶
- [2] Free 3ds models. <http://www.free-3ds-models.com>. ⁶
- [3] Lytro redefines photography with light field cameras. press release, june 2011. <http://www.lytro.com>. ¹
- [4] E. H. Adelson and J. Y. A. Wang. Single lens stereo with a plenoptic camera. *IEEE transactions on pattern analysis and machine intelligence*, 14(2):99–106, 1992. ¹
- [5] M. Bleyer, C. Rother, and P. Kohli. Surface stereo with soft segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1570–1577. IEEE, 2010. ²
- [6] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, 2004. ⁵
- [7] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001. ⁵
- [8] C. Chen, H. Lin, Z. Yu, S. B. Kang, and J. Yu. Light field stereo matching using bilateral statistics of surface cameras. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1518–1525. IEEE, 2014. ^{2, 6}
- [9] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312. ACM, 1996. ⁶
- [10] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 43–54. ACM, 1996. ¹
- [11] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. H. Gross. Scene reconstruction from high spatio-angular resolution light fields. *ACM Trans. Graph.*, 32(4):73, 2013. ²
- [12] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Computer Vision ECCV 2002*, pages 82–96. Springer, 2002. ²
- [13] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159, 2004. ⁵
- [14] V. Krishnamurthy and M. Levoy. Fitting smooth surfaces to dense polygon meshes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 313–324. ACM, 1996. ⁶
- [15] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42. ACM, 1996. ¹
- [16] S. McCloskey. Masking light fields to remove partial occlusion. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 2053–2058. IEEE, 2014. ²
- [17] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a handheld plenoptic camera. *Computer Science Technical Report CSTR*, 2(11), 2005. ⁴
- [18] C. Perwass and L. Wietzke. Single lens 3d-camera with extended depth-of-field. In *IS&T/SPIE Electronic Imaging*, pages 829108–829108. International Society for Optics and Photonics, 2012. ^{1, 2}
- [19] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. ²
- [20] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi. Depth from combining defocus and correspondence using light-field cameras. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 673–680. IEEE, 2013. ^{1, 2, 3, 4, 5, 6}
- [21] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 311–318. ACM, 1994. ⁶
- [22] S. Wanner and B. Goldluecke. Globally consistent depth labeling of 4d light fields. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 41–48. IEEE, 2012. ^{2, 6}
- [23] S. Wanner, S. Meister, and B. Goldlücke. Datasets and benchmarks for densely sampled 4d light fields. In *Annual Workshop on Vision, Modeling and Visualization: VMV*, pages 225–226, 2013. ^{6, 7}
- [24] Y. Wei and L. Quan. Asymmetrical occlusion handling using graph cut for multi-view stereo. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 902–909. IEEE, 2005. ²
- [25] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon. Global stereo reconstruction under second-order smoothness priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2115–2128, 2009. ²
- [26] Z. Yu, X. Guo, H. Ling, A. Lumsdaine, and J. Yu. Line assisted light field triangulation and stereo matching. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2792–2799. IEEE, 2013. ^{2, 6}