ONLINE VIEW SAMPLING FOR ESTIMATING DEPTH FROM LIGHT FIELDS

Changil Kim[⋆†] *Kartic Subr*[⋆] *Kenny Mitchell*[⋆] *Alexander Sorkine-Hornung*[⋆] *Markus Gross*[⋆†]

[⋆]Disney Research Zurich [†]ETH Zurich

ABSTRACT

Geometric information such as depth obtained from light fields finds more applications recently. Where and how to sample images to populate a light field is an important problem to maximize the usability of information gathered for depth reconstruction. We propose a simple analysis model for view sampling and an adaptive, online sampling algorithm tailored to light field depth reconstruction. Our model is based on the trade-off between visibility and depth resolvability for varying sampling locations, and seeks the optimal locations that best balance the two conflicting criteria.

1. INTRODUCTION

The attempts to recover the geometric information such as depth of the scene captured in the light field is gaining more attention. Not only does it play an important role for rendering the light field [1, 2], super-resolving it [3, 4], or finding the focus plane for the post-capture refocus [5, 6], but it also finds its way in the 3D reconstruction and the shape acquisition [7, 8, 9]. Understanding the underlying sampling properties is important to maximize the gain from the effort of acquiring the light field. However, the sampling properties of the light field have been mostly studied in the context of reconstructing the plenoptic function and rendering it by re-sampling process [10, 11, 12]. These are a classical sampling reconstruction problem where radiance is measured at a number of locations in the multi-dimensional domain and the plenoptic function is reconstructed from the sampled values. If each ray $\mathbf{r}$ is seen as a point in 5D space $\mathcal{D} \equiv \mathcal{S}^2 \times \mathbb{R}^3$, then light field reconstruction amounts to reconstructing the height field $\ell(\mathbf{r})$ over this 5D domain.

The problem of depth reconstruction, however, relies on the ill-posed step of finding *correspondences* within this set of light rays. The caliber of depth reconstruction depends crucially on the accuracy of this step, which in turn largely relies on where the rays are sampled—we formulate this as a *view sampling* problem. Although a closely related topic of view selection/planning has been studied in the computer vision and robotics communities, often they are tightly coupled to a specific reconstruction scheme and do not generalize well to others [13, 14] or do not necessarily focus on the specific nature of the currently popular light field acquisition setups [15, 16, 17]. We try to bridge the gap between the light field sampling analysis that has been done regarding rendering and the view sampling that lacks the consideration of the light field. We propose a sampling analysis that is tailored to this particular domain.

We exploit two basic observations: (1) a large displacement of the camera between view samples potentially confuses algorithms that find correspondences since it is possible that locations previously visible are now occluded by other objects in the scene; (2) however, if successive views are "too close" to each other, so that features move by very tiny amounts over image space then it becomes increasingly challenging for correspondence algorithms to resolve the displacement [18]. The right choice of displacement depends on many factors such as the resolution of the image, size of the camera sensor, distance to the scene, the nature and scale of the scene, etc.

Our contributions: Based on these two observations we develop a simple but general *sampling analysis model* and an *online sampling algorithm* based on it to estimate "good" placement of the camera. For the derivation, we assume that the sampling locations are restricted to a line, i.e., $\mathcal{D} \equiv \mathcal{S}^2 \times \mathbb{R}$, and that the analysis is seeded with an inaccurate depth map (obtained using any reconstruction method). Given this, we analyze simple statistics of the scene by trading-off problems due to occlusion with those due to depth resolvability.

Our model considers the very scene being captured and the correspondence algorithm used for reconstruction to gather statistics. Our sampling algorithm uses the model to identify a small set of sampling locations, and successively amasses statistics of the scene, which in turn helps make better view placement in an iterative manner.

1.1. Related work

With alias-free rendering as their goal a substantial body of literature studied sample optimization strategies for light fields. Isaksen et al. [19] and Gortler et al. [1] address how to resample rays from already captured light fields for quality rendering. Chai et al. [20] are one of the first who discussed the optimal sampling rate when provided with constant, approximate, or accurate depth. Durand et al. [21] explore more general physical phenomena regarding light transport and analyze them using Fourier theory.

On the other hand, several previous works in robotics, laser scanning, image-based rendering, and stereo reconstruction

have pointed out the benefits of planning or selecting a *next best view* for improved localization, inspection, and reconstruction quality [22, 23, 24, 25, 16]. These methods achieve considerably improved results by targeting their selection strategies to the specific underlying algorithm. However, they often do not generalize well to other methods and do not always provide an extendable theoretic framework [13, 14, 15, 17].

2. OUR SAMPLING ANALYSIS MODEL

The problem of depth reconstruction relies on determining potential intersections of rays. For this, we define an operator $\mathcal{C} : \mathcal{D} \times \mathcal{D} \rightarrow \{0, 1\}$ that, given two rays $\mathbf{r}_1, \mathbf{r}_2 \in \mathcal{D}$, returns 1 if and only if the two rays originate from the same 3D point. The process of evaluating this operator is known as *correspondence matching* and the implementation of $\mathcal{C}$ has been a long-standing open problem in computer vision. Any algorithm that attempts to be clever with view placement for depth reconstruction must account, in some way, for $\mathcal{C}$.

Conservative sampling interval: Consider a pair of rectified images of an arbitrarily shaped object containing a repetitive texture. If the texture is periodic, then the task of identifying a unique correspondence between pixels is hopeless. However, for a particular pixel, adding a constraint that the camera separation must be small enough to guarantee no occlusion, robustifies the correspondence detection. Formally, we can represent this constraint for each pixel $\mathbf{p}_i$ as a visibility preference function ρ over the sampling position s:

$$\rho_i(s) = \begin{cases} 1 & \text{if } \alpha_i \leq s \leq \beta_i \\ 0 & \text{otherwise} \end{cases} . \tag{1}$$

Here, $[\alpha_i, \beta_i]$ is the interval along s where the scene point projecting to $\mathbf{p}_i$ is guaranteed *not* to be occluded.

Determining visble intervals: Assume that the approximate depth at $\mathbf{p}_i$ is given d_i. Let s_{ij} denote the distance along the baseline where the scene point projecting to $\mathbf{p}_i$ is occluded by a scene point that projects to $\mathbf{p}_j$. Then, using basic trigonometry

$$s_{ij} = r_{ij} \frac{d_i d_j}{d_i - d_j}, \tag{2}$$

where r_{ij} is the image space distance between the pixels (see Figure 1(a)). All distances need to be expressed in the same world units. r_{ij} is related to the pixel disparity by $r_{ij} = (u_j - u_i)/f$ where u_i and u_j are the pixel coordinates of $\mathbf{p}_i$ and $\mathbf{p}_j$, and f is the focal length in pixels. A conservative visibility condition at $\mathbf{p}_i$ guarantees that at least two samples of the scene point projecting to $\mathbf{p}_i$ are visible (and hence can be exactly matched under the Lambertian surface assumption) if the views are within $[\alpha_i, \beta_i]$, where

$$\begin{aligned} \alpha_i &\equiv \max\{s_{ij}\}, \quad \forall j \mid s_{ij} < 0, \\ \beta_i &\equiv \min\{s_{ij}\}, \quad \forall j \mid s_{ij} > 0. \end{aligned} \tag{3}$$

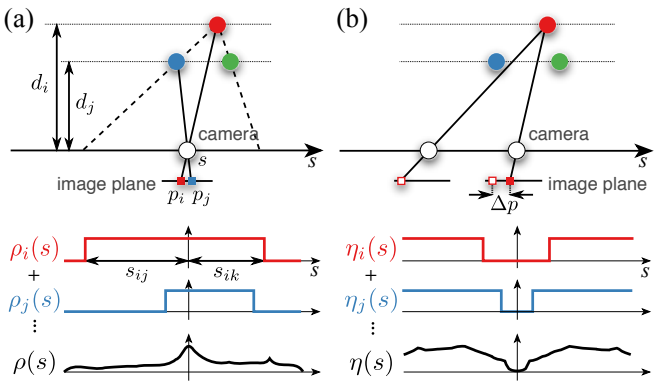

Fig. 1. Determining sampling intervals. **(a)** $\rho_i(s)$ represents an interval of s where the pixel $\mathbf{p}_i$ is visible and thus matching is feasible. **(b)** $\eta_i(s)$ represents an interval where the depth is resolvable for $\mathbf{p}_i$ up to accuracy function $\mathrm{Acc}(\mathcal{C}_{ik})$. The two intervals are each summed over all pixels to form the distributions $\rho(s)$ of non-occluded pixels and $\eta(s)$ of the pixels with resolvable depth, respectively, over s.

Depth resolution of the correspondence algorithm: While a small displacement of the camera along the baseline enjoys the advantage of avoiding occlusion, it introduces the difficulty for the correspondence algorithm to be accurate and reliable. The accuracy of the triangulation of scene points using image features increases with displacement along the baseline [18]. We use a simple measure for estimating the depth resolution, which depends on the accuracy of the correspondence algorithm $\mathcal{C}$. Say that the scene point that projects to $\mathbf{p}_i$ in a view project to $\mathbf{p}_k$ after the camera is translated s units along the baseline (see Figure 1(b)). As for visibility, we define a preference function η for depth resolution over s:

$$\eta_i(s) = \begin{cases} 1 & \text{if } \mathrm{Acc}(\mathcal{C}_{ik}) > \epsilon \\ 0 & \text{otherwise} \end{cases} . \tag{4}$$

where $\mathrm{Acc}(\mathcal{C}_{ik})$ is the accuracy of the operator for the given pixels and ϵ is an arbitrarily chosen threshold. In this paper, we use a constant value $\Delta p = \mathrm{Acc}(\mathcal{C}_{ik})$. See parameter selection in Section 5 for details.

Combining visibility and depth resolution: Clearly depth resolution is better when we use a wide separation distance between views. However, the larger the separation, the more likely that the scene point is occluded at the new location. We perform the trade-off between these two factors by multiplying the two, per pixel, which results in a density γ_i over s, a direct measure of view-location preference for $\mathbf{p}_i$. Accumulating this preference over all pixels yields

$$\gamma(s) = \sum_{\mathbf{p}_i \in I} \gamma_i(s) = \sum_{\mathbf{p}_i \in I} \rho_i(s)\, \eta_i(s). \tag{5}$$

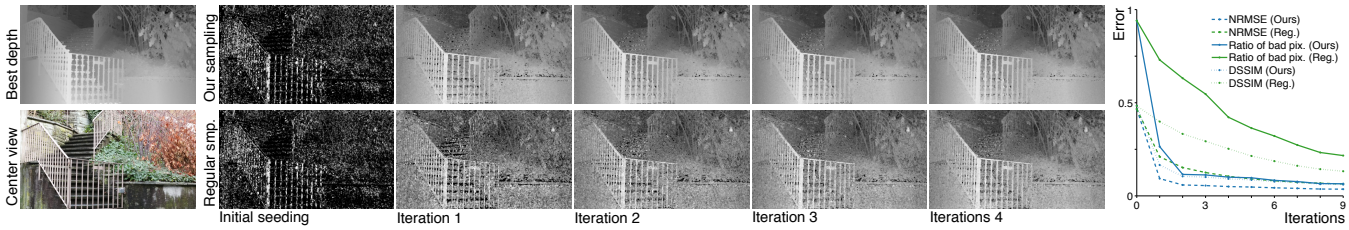

Fig. 2. Resulting depth maps computed after several iterations with $k = 2$. Our sampling strategy *(top row)* is compared against the regular sampling *(bottom row)*. The error plots *(last column; the lower the better)* show the faster convergence of ours towards lower errors.

3. ONLINE VIEW SAMPLING ALGORITHM

One can use the sampling model developed in the previous section to design a new acquisition device provided a rough distribution $\gamma(s)$ is known a priori. Otherwise, our model can guide the acquisition process that best suits the particular scene scanned and the depth reconstruction method used. This section details our adaptive view-sampling algorithm and discusses implementation details.

Initial step: We assume a depth reconstruction method that takes a set of images and produces per-view depth maps, and $k \geq 2$ images that are captured at arbitrary locations $s_1, \ldots, s_k$. Using the given depth reconstruction method, we first compute initial k depth maps, which we use to estimate $\gamma(s)$. With the known initial sampling locations s_i, a set of each estimated local distribution $\gamma_i(s)$ is summed up to form a single global distribution:

$$\gamma(s) = \sum_{i}^{k} \gamma_i(s + s_i). \tag{6}$$

In principle, the local maxima of $\gamma(s)$ can serve as the next sampling locations. Instead of being directly used for view sampling, however, they are all put into the queue which prioritizes the candidates for next steps.

Iterations: At each iteration, at most k sampling locations with the highest priority are dequeued. They indicate the locations where the largest number of pixels fulfill both sampling criteria, and thus where new images should be taken. Upon acquisition of the new images, only those new ones are used to estimate $\gamma(s)$. One can expect better estimation of $\gamma(s)$, when including all images, due to the improved depth computation. However, we found this additional depth accuracy does not bring much advantage in practice, and our book-keeping scheme described shortly handles missing or redundant part of estimated $\gamma(s)$ properly. After obtaining the new distribution covering the new sampling locations, again the local maxima are identified and pushed into the queue. These steps are repeated until either a termination criterion is met or the queue becomes empty.

Termination: At each end of iteration, a depth map is computed from all images captured thus far. The algorithm stops when the improvement achieved by the last iteration becomes negligible. If the depth computation is expensive and should be minimized, a more practical criterion is to stop when the target number of sampling locations are achieved. After the last iteration, all the images captured so far are used for the final depth reconstruction.

Priority queue for sampling locations: The priority queue maintains the sampling location candidates as tuples (s, w), i.e., for an s_i, the queue also stores its associated frequency in the distribution, $w_i = \gamma(s_i)$. Whenever a new tuple is being pushed, the queue first checks if the location s has been already seen before by looking up the directory $\chi(s)$: if it is marked so, the tuple is discarded. If at least one new tuple is added, the queue re-arranges its tuples in the descending order of w_i. Then, for each location s_i from the highest to the lowest priority, the queue contracts all tuples (s_j, w_j) within some distance ζ from s_i, forming a new tuple

$$(s^\star, w^\star) = \Big(\frac{\sum s_j w_j}{\sum w_j},\ \max\{w_j\} \Big),\ \forall (s_j, w_j)\ \big|\ |s_j - s_i| < \zeta. \tag{7}$$

When dequeued, the location s of the tuple is marked in the directory $\chi(s)$ which records the sampling locations dequeued so far and thus prevents duplicate sampling locations from being used again. In our implementation this directory is discretized at the resolution of ζ. The dequeued sampling location s is quantized to the nearest meaningful value if required. For the captured test datasets we set both ζ and the quantization resolution to the step between adjacent images.

4. EXPERIMENTAL RESULTS

We tested our analysis and the online algorithm with the light field depth reconstruction method of Kim et al. [8]. For the computer-generated dataset, we rendered images on demand at the exact sampling locations calculated by our algorithm. The ground truth depth was obtained by an extra depth rendering pass. Two real-world datasets are captured using a consumer digital camera. We captured video clips while the camera moves on a linear path at a constant speed.

Figure 2 shows the resulting depth maps of one of the two real-world datasets over 4 iterations with $k = 2$. The depth

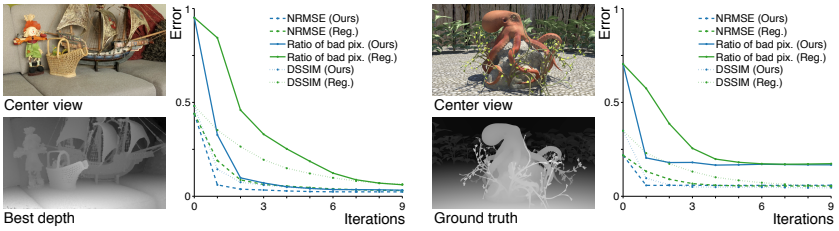

Fig. 3. Quantitative comparisons between our sampling strategy *(blue curves)* and the regular sampling *(green curves)* using real-world *(left)* and synthetic *(right)* datasets. Two strategies are compared against the ground truth or the best possible depth using three error metrics.

maps are computed using increasing number of views whose locations are incrementally determined by our adaptive sampling algorithm. It is compared against the regular sampling, where the same number of consecutive images centered around the reference view. The two plots show the relative error with respect to the best depth map we obtained using the dataset.

Figure 3 shows the errors of the computed depth map against the ground truth (or the best possible) depth map. As in Figure 2 we computed depth maps using our sampling strategy and the regular sampling with the same number of images at each iteration. We used three error metrics: the normalized root-mean-squared errors (NRMSE); the ratio of *bad pixels* whose estimates are different from the truth greater than 5% tolerance; and the structural dissimilarity (DSSIM) that is derived from the structural similarity (SSIM) [26] and defined as $(1 - \text{SSIM})/2$. For all metrics, the lower the better.

5. DISCUSSIONS AND CONCLUSION

Our approach has three properties that will allow for generalization: (1) it considers the statistics of the very scene being captured; (2) it is targeted for the particular depth reconstruction algorithm being used; (3) more constraints (preference functions) may be included for view sampling, besides occlusion and depth resolution (more terms in equation (5)).

Parameter selection: All the results in this paper were generated using the same constant Δp that is set to be one sensor pixel size in world units. We included the parameter in the theory to accommodate various types of correspondence matching algorithms. In principle, Δp must be selected using the appropriate accuracy function $\text{Acc}(\mathcal{C}_{ik})$ in (4) of the particular correspondence algorithm. The second parameter to the algorithm is k, the number of view locations generated at each iteration. We also kept this parameter constant at $k = 2$ for all experiments. We observed, however, that the algorithm converged faster with higher k. In principle, the selection of k depends on the distribution $\gamma(s)$ itself. Selecting an optimal k at each iteration is an open problem for future work.

Seeding: Since our algorithm is online, it requires an approximate depth map to seed the process of view sampling. Theoretically, using a constant function as the seed depth map is sufficient. i.e the algorithm can be seen as having a dummy iteration at the start. In practice, we use the depth from the adjacent k images at center as for the uniform sampling.

Interactive online acquisition: We observed that our algorithm is not sensitive to the spatial resolutions of the intermediate depth. That is, we may use downsampled coarse depth maps (with Δp scaled appropriately) for the estimation of the $\gamma(s)$ distribution and use the highest resolution depth maps only for the final depth computation. This can provide significant speed-up when the depth reconstruction algorithm turns out to be the bottleneck for performance.

Limitations and future work: In general, many reconstruction methods assume Lambertian surfaces and do not properly deal with glossy or specular surfaces. Thus, for such methods, it is desirable to avoid the sampling locations where significant amount of view-dependent effects are observed. To this end, the interval analysis may incorporate the level of inconsistency along the angular axis and guide the sampling locations against problematic areas. Our current algorithm is based on known depth, which in some cases, one may not assume to be viable possibly due to the time and other constraints. In such cases, it would be useful to have some approximate estimation of the sampling distribution. This might be bootstrapped by other types of information, such as monocular depth cues or annotated images. Although we exemplified light fields with a linear camera alignment, the theory does not assume, nor is limited to, such configuration. It would be fruitful to extend the algorithm for 2D camera configurations and even more interesting scenarios such as circular or spherical light fields, or large-scale aerial captures.

6. ACKNOWLEDGMENTS

We wish to thank Maurizio Nitti for creating the synthetic dataset. This work was partly funded by InnovateUK project #101857.

7. REFERENCES

[1] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen, "The Lumigraph," in *ACM SIGGRAPH*, 1996, pp. 43–54.

[2] C. Buehler, M. Bosse, L. McMillan, S.J. Gortler, and M.F. Cohen, "Unstructured Lumigraph rendering," in *ACM SIGGRAPH*, 2001, pp. 425–432.

[3] T.E. Bishop, S. Zanetti, and P. Favaro, "Light field super-resolution," in *IEEE ICCP*, 2009, pp. 1–9.

[4] S. Wanner and B. Goldluecke, "Spatial and angular variational super-resolution of 4D light fields," in *ECCV*, 2012, pp. 608–621.

[5] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," Tech. Rep. CSTR 2005-02, Stanford University, 2005.

[6] K. Venkataraman, D. Lelescu, J. Duparré, A. McMahon, G. Molina, P. Chatterjee, R. Mullis, and S. Nayar, "PiCam: an ultra-thin high performance monolithic camera array," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 166:1–166:13, 2013.

[7] S. Wanner and B. Goldluecke, "Globally consistent depth labeling of 4D light fields," in *IEEE CVPR*, 2012, pp. 41–48.

[8] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross, "Scene reconstruction from high spatio-angular resolution light fields," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 73:1–73:12, 2013.

[9] S. Heber and T. Pock, "Shape from light field meets robust PCA," in *ECCV*, 2014, pp. 751–767.

[10] A. Levin, W.T. Freeman, and F. Durand, "Understanding camera trade-offs through a Bayesian analysis of light field projections," in *ECCV*, 2008, pp. 88–101.

[11] A. Levin and F. Durand, "Linear view synthesis using a dimensionality gap light field prior," in *IEEE CVPR*, 2010, pp. 1831–1838.

[12] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, "Compressive light field photography using overcomplete dictionaries and optimized projections," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 46:1–46:12, 2013.

[13] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S.M. Seitz, "Multi-view stereo for community photo collections," in *IEEE ICCV*, 2007, pp. 1–8.

[14] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys, "Variable baseline/resolution stereo," in *IEEE CVPR*, 2008, pp. 1–8.

[15] G. Olague and R. Mohr, "Optimal camera placement for accurate reconstruction," *Pattern Recognition*, vol. 35, no. 4, pp. 927–944, 2002.

[16] A. Hornung, B. Zeng, and L. Kobbelt, "Image selection for improved multi-view stereo," in *IEEE CVPR*, 2008, pp. 1–8.

[17] Y. Furukawa, B. Curless, S.M. Seitz, and R. Szeliski, "Towards Internet-scale multi-view stereo," in *IEEE CVPR*, 2010, pp. 1434–1441.

[18] R. Szeliski and D. Scharstein, "Sampling the disparity space image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 3, pp. 419–425, 2003.

[19] A. Isaksen, L. McMillan, and S.J. Gortler, "Dynamically reparameterized light fields," in *ACM SIGGRAPH*, 2000, pp. 297–306.

[20] J. Chai, S.-C. Chan, H.-Y. Shum, and X. Tong, "Plenoptic sampling," in *ACM SIGGRAPH*, 2000, pp. 307–318.

[21] F. Durand, N. Holzschuch, C. Soler, E. Chan, and F.X. Sillion, "A frequency analysis of light transport," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1115–1126, 2005.

[22] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 417–433, 1993.

[23] K.N. Kutulakos and C.R. Dyer, "Recovering shape by purposive viewpoint adjustment," *Int. J. Comput. Vision*, vol. 12, no. 2-3, pp. 113–136, 1994.

[24] W.R. Scott, G. Roth, and J.-F. Rivest, "View planning for automated three-dimensional object reconstruction and inspection," *ACM Comput. Surv.*, vol. 35, no. 1, pp. 64–96, 2003.

[25] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich, "Automatic view selection using viewpoint entropy and its applications to image-based modelling," *Comput. Graph. Forum*, vol. 22, no. 4, pp. 689–700, 2003.

[26] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Proc.*, vol. 13, no. 4, pp. 600–612, 2004.