Reciprocal Visibility for Guided Occlusion Removal with Drones

Rakesh John Amala Arokia Nathan, Sigrid Strand, Dmitriy Shutin, and Oliver Bimber

***Abstract*—In this paper, a guidance strategy is proposed to optimize synthetic aperture sampling for occlusion removal with drones based on point-cloud representation of occluders. Pre-recorded LiDAR scans are utilized to compute the visibility of fixed inspection regions on the ground that are intended for recurrent monitoring from the air. By utilizing Helmholtz reciprocity, the drone-collected LiDAR scans are used to computationally obtain a reciprocal visibility of potential drone positions in the air from points of interest on the ground. This visibility forms a basis for a novel navigation strategy. This strategy was shown to drive drones to optimal aerial monitoring positions, thus reducing occlusion and consequently the sampling time. Compared to previous unguided sampling, we achieve a 5-20% higher visibility with 9-17 times less samples in our experiments.**

***Index Terms*—synthetic aperture sampling, aerial imaging, occlusion removal, drones.**

I. Introduction

For many applications that apply aerial imaging, occlusion caused by dense vegetation (such as forest) represents a fundamental problem. Synthetic aperture sensing approaches, such as Airborne Optical Sectioning (AOS) [1] (which essentially is unstructured light-field capturing [2] extended to aerial imaging), remove occlusion in drone recordings computationally in real-time. As illustrated in Fig. 1a, AOS registers and integrates (i.e., averages) multiple images captured with conventional camera optics at different drone positions to an integral image that mimics a wide (several meters) synthetic camera aperture. The area a in which these images are sampled is the size of the synthetic aperture. Registration and integration depend on the camera poses and on a defined focal surface F (e.g., a given plane or a registered digital elevation map of the ground surface at distance h). The computed integral images (Fig. 1b) have an extremely shallow depth of field. This causes all targets located on the focal surface at distance F to appear sharp and unoccluded while the blur signal of the occluders at distance o from F (i.e., not on the focal surface) is spread widely (b) over the integral image (which suppresses their contribution). The main advantages of AOS over alternatives, such as light detecting and ranging (LiDAR) or synthetic aperture radar (SAR), are high spatial resolution, real-time processing capabilities, and wavelength independents. Since it is applicable to images captured in all spectral ranges and processing is in the range of milliseconds, AOS has been demonstrated previously for use-cases, such as search and rescue, wildlife observation, wildfire detection, surveillance, forest ecology, and archaeology.

A main challenge of synthetic aperture imaging approaches, such as AOS, is optimal sampling (i.e., the question where to capture images within the synthetic aperture to achieve highest integrated visibility with the smallest number of samples and, hence, shortest sampling and flight time). Several approaches [3], ranging from blind sampling strategies through predefined waypoint flights (1D lines and 2D grids) with single drones, over parallel sampling with camera arrays, to situation adaptive particle swarm optimization with drone swarms have been investigated. Yet a more systematic (and formal) approach for selection of appropriate sampling locations is still needed.

Our goal in this article is to propose a novel approach toward identification of optimal aerial sampling locations that will result in a better exposure of locations on the ground. The proposed strategy is shown to improve AOS by reducing required sampling and flight times; however, it is not restricted to AOS and can also be used for other applications, like aerial surveillance or inspection. In its essence, the proposed method utilizes a prior LiDAR scanning of a forest to identify locations of ground visibility. Let us stress that such (possibly sparse) point-cloud data is usually not suitable for occlusion removal due to a low resolution and inappropriate wavelengths; it is therefore not used in AOS. Yet the contained depth informa-

R.J.A.A. Nathan and O.Bimber are with the Department of Computer Science, Johannes Kepler University, 4040 Linz, Austria, e-mail: rakesh.amala_arokia_nathan@jku.at, oliver.bimber@jku.at. S. Strand and D. Shutin are with the Institute of Communications and Navigation Communications Systems, German Aerospace Center, 82234 Oberpfaffenhofen-Wessling, e-mail: sigrid.strand@dlr.de, dmitriy.shutin@dlr.de.

Manuscript received XXXMONTHXXX XXDAYXXX, XXXYEARXXX.

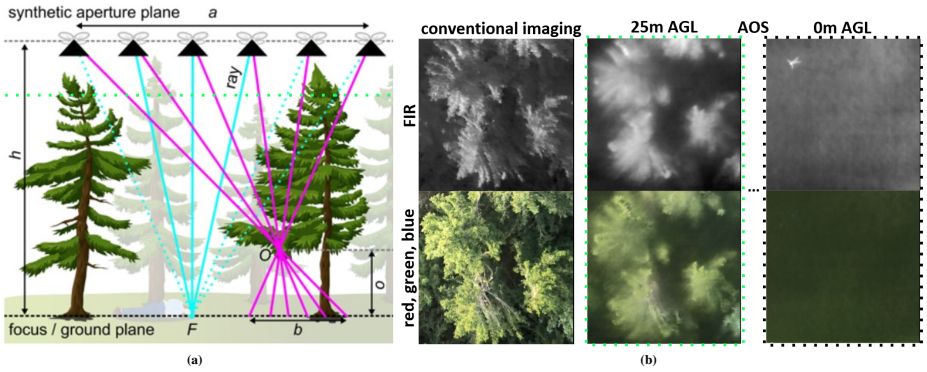

Fig. 1. Synthetic aperture imaging and integration principle (a). Examples of conventional aerial images and AOS integral images of forest at two different synthetic focal distances and captured at various spectral ranges (visible red/green/blue and far-infrared (FIR)/thermal)(b). This example illustrates a search and rescue use-case where in particular thermal measurements are relevant. The person on the ground becomes visible in the FIR channel after occlusion removal (i.e., by focusing computationally on the ground). This data has been recorded with a single drone in a 30m ×30m waypoint grid and at an altitude of 35m above ground level (AGL) [5].

tion in LiDAR scans can be utilized to directly compute a sparse mask of visible ground points for each potential drone perspective, and thus identify visibility of locations on the ground.

By building upon Helmholtz reciprocity [4], we use LiDAR scans to calculate reciprocal visibility (RV). RV reflects the visibility of prospective sampling locations in the air concerning designated ground points of interest. These points can represent specific inspection zones like roads, paths, or distinct areas targeted for periodic monitoring to detect people, vehicles, animals, wildfires, etc. The result is a visibility map that encodes which point on the ground is visible with which quality from any position in the air. Note that while Helmholtz reciprocity principle is primarily used in ray optics, it has been also applied in computational imaging for dual photography [6], surface reconstruction [7], radiometric compensation [8], and computational relighting [9]. It is also fundamental to classical rendering techniques, such as ray-tracing and the bidirectional distribution function. Here we use it for guided aerial imaging and efficient occlusion removal.

By utilizing RV, we then propose a novel greedy navigation strategy that computes optimal sampling positions. Note, that all of these steps can be pre-computed offline (assuming constant forest structure and inspection regions); The resulting sampling positions can then be covered by a single drone sequentially or multiple drones in parallel during periodic online monitoring to determine a widely occlusion free integral image of the inspection regions. For our discussion in the following, we will apply the same procedural forest model as in [3] to simulate 3D data, which would normally result from LiDAR scans. We show the potential advantage of RV-guided sampling over previous unguided sampling, such as that in [3].

II. Theory

Let us consider the synthetic aperture imaging and integration process as described previously (see also Fig. 1a). Let us further consider N image acquisitions, giving N binary visibility masks $\boldsymbol{V}_n$, $n = 1, \dots, N$, at the resolution $M = R_v \times R_v$ pixels per mask, which indicate visibility or occlusion of the ground from all N drone positions over the synthetic aperture plane. The resulting and 1D vectorized $R_v \times R_v$ pixel integral visibility map $\boldsymbol{i}$ can then be formed by a convex combination of the collected single visibility masks as follows:

$$\boldsymbol{i} = \frac{1}{Z} \sum_{n=1}^{N} \boldsymbol{v}_n p_n = \boldsymbol{V}\boldsymbol{p}, \tag{1}$$

where $\boldsymbol{p} = [p_1, \dots, p_N]^\top$ represents the synthetic aperture by a 1D *binary* vector that indicates which visibility maps are included into $\boldsymbol{i}$, $Z = \|\boldsymbol{p}\|^2$ is a normalization constant, and $\boldsymbol{V} = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_N]$ is a so-called visibility matrix. Its columns are defined as vectorized representations of the registered visibility maps $\boldsymbol{v}_n = \mathrm{vec}\{\boldsymbol{V}_n\}$. Note that $\boldsymbol{V}$ is a $M \times N$ matrix; also, w.l.o.g. we absorb the scalar Z into $\boldsymbol{V}$ to simplify further notation.

The structure of $\boldsymbol{V}$ bears the key to understanding the reciprocal visibility principle. The element $v_n^m = [\boldsymbol{V}]_{m,n}$, i.e., an entry of $\boldsymbol{V}$ at the nth column and the mth row, corresponds to the visibility from the nth position on the synthetic aperture plane at the direction m. Thus, while a single column $\boldsymbol{v}_n$ of $\boldsymbol{V}$ encodes the visibility at a fixed aperture plane position n, a single row encodes visibility along a single (spatial) direction m over different aperture positions (see also Fig. 2 as an example for 2D case). Note that in this way the 2D structure

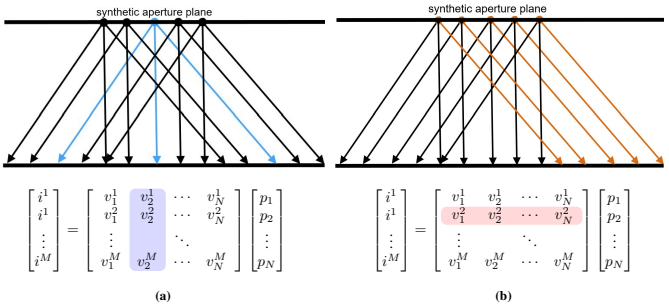

Fig. 2. 2D illustration of the 4D visibility field in $\boldsymbol{V}$. While same columns of $\boldsymbol{V}$ represent same positions on the synthetic aperture plane (a), same rows of $\boldsymbol{V}$ represent same directions (b). Note, that each ray has a unique position-direction encoding.

of $\boldsymbol{V}$ effectively encodes an actual 4D visibility field.

Now, given a desired visibility map $\boldsymbol{i}$, we could identify an optimal sampling pattern $\boldsymbol{p}$ on the synthetic aperture plane, e.g., as a solution with respect to $\boldsymbol{p}$ of an optimization[1] $\min_{\boldsymbol{p}} \|\boldsymbol{i} - \boldsymbol{V}\boldsymbol{p}\|$ with respect to some preferred norm $\|\cdot\|$. Unfortunately, such an optimization problem is ill-posed and computationally very challenging for realistic processing times and memory space for typical values of M and N.

A computationally more attractive approach in this respect relies on the principle of Helmholtz reciprocity. This principle states that the transport of light which passes through a passive medium is dual with respect to its travel direction: an emitter and a receiver can be swapped, while ensuring the same light-transport. Helmholtz reciprocity applies to all passive media, and covers reflection, refraction, absorption, as well as occlusion. In our setting this principle can be applied as follows:

Assume a reverse scenario, where the synthetic aperture plane is now located on the ground, and the drone camera is looking upwards. In this case (1) can be transformed as

$$\boldsymbol{i}_\uparrow = \boldsymbol{V}^\top \boldsymbol{p}_\uparrow, \tag{2}$$

where we used a subscript $(\cdot)_\uparrow$ to explicitly indicate that here the integrated visibility map is of the "sky" obtained by "looking" upwards from the synthetic aperture plane on the ground. Thus, $\boldsymbol{p}_\uparrow$ is a binary aperture selection vector again, and $\boldsymbol{i}_\uparrow$ is the "sky" integral visibility.[2] Thus, in (1) $\boldsymbol{V}$ describes the visibility field in top-down direction. Its transpose in (2) describes the dual visibility field in bottom-up direction.

Note that a "reciprocal" approach effectively casts an inverse problem of recovering $\boldsymbol{p}$ from (1) as a forward problem (2). Indeed, pixels in a 2D transformed vector $\boldsymbol{i}_\uparrow$ will directly tell us the locations in the air from which ground points encoded by $\boldsymbol{p}_\uparrow$ are visible.

[1] Note that an estimation of $\boldsymbol{p}$ is in general an inverse problem.

[2] Again here an integral visibility $\boldsymbol{i}_\uparrow$ is a convex combination of columns in $\boldsymbol{V}^\top$, implying a normalization constant $1/Z = 1/\|\boldsymbol{p}_\uparrow\|^2$.

To better understand this concept in a relevant application setting consider a realization of (1) and (2) in Fig. 3. Here we

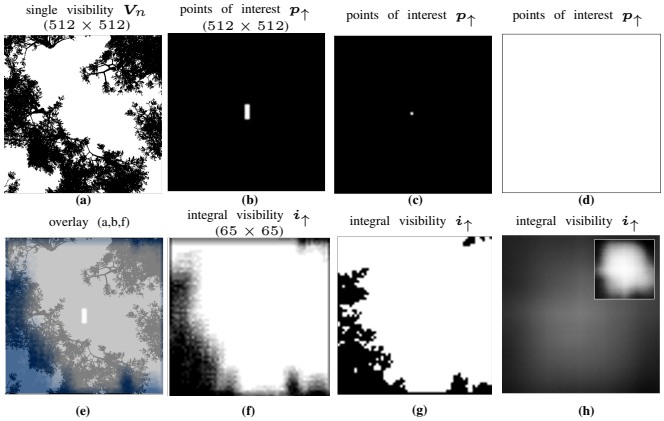

Fig. 3. 2D visualizations of components in (2) using procedural forest simulation. Single visibility masks $\boldsymbol{V_n}$ (a) indicate visibility (1) and occlusion (0) as seen top-down from the synthetic aperture plane to the ground. For selected points of interest on the ground $\boldsymbol{p}_\uparrow$ (b,c,d), the integrated visibility maps $\boldsymbol{i}_\uparrow$ (f,g,h) indicate the visibility of points in $\boldsymbol{p}_\uparrow$ from points at the synthetic aperture plane, with larger values corresponding to higher visibility. The inlay in (h) is a contrast enhanced version of visibility. An overlay of (a,b,f) illustrates the correct spatial alignment of resulting integrated visibility with the occluders in the forest: darker blue indicates lower visibility of the selected ground points. Here, the average tree height is 20m and altitude of the synthetic aperture plane is 35m above ground level. Simulated density: 100trees/ha (birch) on a $32 \times 32\text{m}^2$ area.

show a single visibility mask $\boldsymbol{V}_n$ with $M = 512 \times 512$ pixels of a simulated procedural birch forest as seen top-down, from the center perspective of the synthetic aperture at 35m altitude above ground level, covering a 32x32m^2 area with a 50° field of view camera, resulting in 6.25x6.25cm^2-sized ground pixels (Fig. 3a). Assume also that a total of $N = 65^2 = 4.225$ such visibility maps is available (at 65×65 synthetic aperture grid points), and consider a binary pattern of ground points of interest $\boldsymbol{p}_\uparrow$ of resolution 512×512 (Fig. 3b). Naturally, the integral visibility $\boldsymbol{i}_\uparrow$ (Fig. 3f) will have N pixels. If $\boldsymbol{p}_\uparrow$ contains only a single point of interest on the ground (see Fig. 3c), then, according to (2), $\boldsymbol{i}_\uparrow$ is a pinhole-view of the "sky", as seen from this point upwards (see Fig. 3g). In case $\boldsymbol{p}_\uparrow = \mathbf{1}$, i.e., all ground points are of interest (Fig. 3d), the resulting $\boldsymbol{i}_\uparrow$ is shown in Fig. 3h. Higher values indicate higher visibility of the selected ground points. Note that since typically $\boldsymbol{V}$ is a tall matrix, i.e., $M \geq N$, in reciprocal visibility an integral visibility map $\boldsymbol{i}_\uparrow$ will typically have a lower resolution of N pixels. Unfortunately, the complexity of the theoretical approach presented in this section is clearly not practical. Even in the considered example with $N = 4.225$, the visibility matrix $\boldsymbol{V}$ will consist of $\sim 10^9$ coefficients: the corresponding sampling time would be massive, yet it will lead to a rather low resolution of the resulting integrated visibility: N pixels of integrated visibility versus M pixels of acquired top-down visibility masks. In Sec. III we discuss how this theoretical principle can be implemented efficiently in practice and how the resolution of the integral visibility can be improved.

III. Practical Reciprocal Visibility Implementation

Let us outline the practical steps to reciprocal visibility usage in the considered context. Our aim is to compute an integral visibility map that would tell us which points on the ground will be visible at different synthetic aperture positions in the air. This information can then be re-used to optimally navigate camera-equipped drones for occlusion removal while reducing sampling and flight time.

To this end, let us now assume that we are given (or we can design) a pattern $\boldsymbol{p}_\uparrow$ that represents points of interest on the ground that we wish to monitor. The reverse imaging would imply an availability of the corresponding visibility matrix describing visibility from bottom upwards, following (2). The latter is unfortunately physically unavailable. Instead, consider a scanning drone, equipped with e.g., a LIDAR that can generate a (possibly sparse) point cloud representation of the forest. These collected scans, although sparse, are in fact top-down "proxies" of $\boldsymbol{V}_n$ used in (1). Although point clouds cannot be used directly for occlusion removal, they can be 3D projected to compute a high-resolution, e.g., 512×512, equivalent representation of a bottom-up visibility map $\boldsymbol{U}_\uparrow \approx \boldsymbol{V}^\top$. The latter is the key to a practical realization of the method – instead of using acquired *areal* masks $\boldsymbol{V}$, we utilize LiDAR data to *numerically* compute the reciprocal version $\boldsymbol{U}_\uparrow$ of bottom-up visibility at the desired resolution. In fact, any depth perceiving visual sensor can be used instead of LiDAR. As the result, the visibility from positions on the ground surface upward can be *computationally* reconstructed for a "forward" computation of the ground visibility. This can then be utilized by AOS for selecting better, more optimal sampling locations for synthetic aperture imaging. This approach is illustrated in Fig. 4.

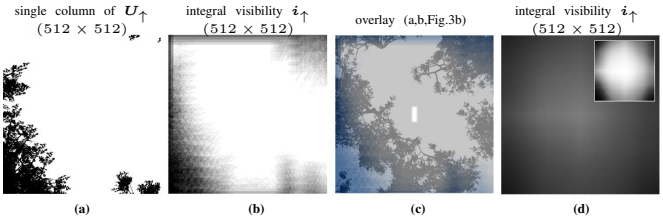

Fig. 4. A single bottom-up visibility mask (one column of $\boldsymbol{U}_\uparrow$) computed from a ground point of interest – center ground position in this example – (a) is the high-resolution counter-part of the integrated visibility map $i_\uparrow$ for the sample point in (2) - see Fig. 3g. Corresponding high-resolution integrated visibility maps (b for Fig. 3f, d for Fig. 3h) and the overlay visualization (c) that corresponds to Fig. 3e. Here, the integrated visibility map in (b,c) has been computed for only 21 ground points of interest that approximates the shape of our rectangular pattern shown in (c). Simulated density: $100trees/ha$ on 32x32m^2.

Here we consider $\boldsymbol{p}_\uparrow$ as shown in Fig. 3b, i.e., a rectangular inspection area of 21 points on the ground. A visibility image corresponding to one of these points, i.e., to a single column of $\boldsymbol{U}_\uparrow$, is exemplary shown in Fig. 4a. Again, we use a simulated data here, but in practice a computation reconstruction of the visibility image from LiDAR data should be used. Note that $\boldsymbol{U}_\uparrow$ is a tall matrix, leading to a high-resolution integral image $\boldsymbol{i}_\uparrow = \boldsymbol{U}_\uparrow \boldsymbol{p}_\uparrow$ in Fig. 4b. In case $\boldsymbol{p}_\uparrow = \mathbf{1}$, i.e., all points on the ground are considered, we obtain in Fig. 4d a high-resolution version of integral visibility Fig. 3h.

Let us point out that, despite its simplicity, the computed integral visibility $\boldsymbol{i}_\uparrow$ completely "obscures" which pixel of $\boldsymbol{p}_\uparrow$ is visible. In other words, we obtain an integrated visibility

of all points on the ground that are selected in $\boldsymbol{p}_\uparrow$. There

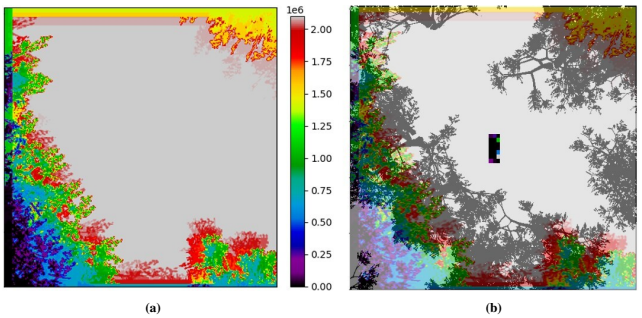

Fig. 5. Coded integrated visibility map $\boldsymbol{i}_\uparrow$ for the example shown in Fig. 4b (a) and the overlay (b) that corresponds to Fig. 4d. In this case, unique binary codes for $K = 21$ ground points of interest (individual codes are now visualized in our target area) are integrated. The possible $2^{21} = 2.097.152$ different integrated combinations in $\boldsymbol{i}_\uparrow$ are color coded. Simulated density: 100trees/ha on a $32 \times 32\text{m}^2$ area.

is however a possibility to encode this information in a way that will allow to extract visibilities of each individual point from the integral $\boldsymbol{i}_\uparrow$. The solution we propose here relies on a single assumption that the visibility matrix $\boldsymbol{U}_\uparrow$ is binary. Now, consider $K \in \mathbb{N}$ as the number of points of interest on the ground. For each ground point of interest in $\boldsymbol{p}_\uparrow$ we now assign a non-binary code – a unique positive number 2^k, $k = \{1, \ldots, K\}$. Such a number, say $2^4 = 16$ in a binary representation will have a digit 1 at one unique position – at the 4th digit in this example. As such, the binary visibility map associated with this bit can then be "decoded" from the integral $\boldsymbol{i}_\uparrow$. One can think of these numbers as orthogonal codes in binary domain. This will allow de-coupling their contributions from a finite sum $\boldsymbol{i}_\uparrow$. Thus, by converting the integral values in $\boldsymbol{i}_\uparrow$ to a binary representation, one can "decode" the visibility of individual ground point of interest. A sparse example for such a (coded) integrated visibility map for our $K = 21$ ground points of interest is illustrated in Fig. 5.

Note that the coded coefficients in $\boldsymbol{i}_\uparrow$ must not be normalized. Although formally $\boldsymbol{i}_\uparrow$ is no longer a convex combination of columns of $\boldsymbol{U}_\uparrow$, after decoding the integral visibility it will correspond exactly to a single ground point in $\boldsymbol{p}_\uparrow$.

The main drawback of this encoding approach is that K depends on the bit-depth of the processing hardware. Yet for $K > 32$, multiple sequential passes can be used allowing higher number of bits to the visibility map $\boldsymbol{i}_\uparrow$.

IV. Sampling with RV Constraints

Given a coded integrated visibility map $\boldsymbol{i}_\uparrow$ (that represents visibility of ground points of interest $\boldsymbol{p}_\uparrow$ as seen from the synthetic aperture plane), we can now constrain the sampling at the the synthetic aperture plane to optimize visibility. Considering all possible sampling constellations for an N-dimensional $\boldsymbol{i}_\uparrow$ would require evaluating $\sum_{k=1}^{N} k!$ sampling possibilities – an optimal, but computationally intractable undertaking. Instead, a simpler, greedy approximation can find a solution from a given starting position s quickly by maximizing visibility while ensuring sufficient uniformity among all reconstructed $\boldsymbol{p}_\uparrow$,as we detail in Algorithm 1.

Initially, we store the starting position s and its visibility code $\boldsymbol{i}_\uparrow(s)$ in sampling set S (line 1). For each iteration, we then compute the bit-wise averages ($\boldsymbol{I_a}$ and $\boldsymbol{I_b}$) of all

Algorithm 1. **GreedySampling**($\boldsymbol{i}_\uparrow, s$)

1: $S = \{[s, \boldsymbol{i}_\uparrow(s)]\}$
2: **while** exit condition not reached **do**
3: $\quad \boldsymbol{I_a} = BitwiseAverage(S)$
4: $\quad$ **for** all $c \in \boldsymbol{i}_\uparrow$ **do**
5: $\quad\quad \boldsymbol{I_b}(c) = BitwiseAverage(S \cup \{c\})$
6: $\quad$ **end for**
7: $\quad C = \{c \in \boldsymbol{i}_\uparrow \quad | \quad \|\boldsymbol{I_b(c)}\| > \|\boldsymbol{I_a}\|\}$
8: $\quad c = Closest(\{c \in C \quad | \quad max\, \|\boldsymbol{I_b(c)} - \boldsymbol{I_a}\|\}, s)$
9: $\quad S = S \cup \{[c, \boldsymbol{i}_\uparrow(c)]\}$
10: $\quad$ s=c
11: **end while**
12: **return** (S)

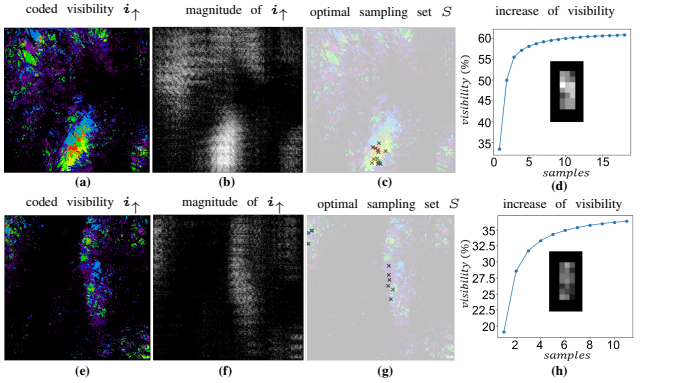

Fig. 6. Greedy sampling for different forest densities (a-d: 400trees/ha, e-h: 500trees/ha). Coded integrated visibility maps $\boldsymbol{i}_\uparrow$ (a,e) and their uncoded magnitudes (b,f) reveal clear patches of better and worse visibility (for ground points $\boldsymbol{p}_\uparrow$ and their encodings as shown in Fig. 5b). The determined set of optimal sampling points S for 50 random start positions and a variance threshold of $T = 33\%$ (c,g). Results in (d,h) show the increase of visibility over sequential sampling steps in S. The inlay shows the final reconstruction of the occluded target pattern by integrating the samples S. Simulated area for both cases: on $32 \times 32\text{m}^2$.

codes in S without a possible candidate c in $\boldsymbol{i}_\uparrow$ (line 3) and with it (line 5). We then compute a new set C (line 7) of candidates c for which the visibility increases if considered ($\|\boldsymbol{I_b(c)}\| > \|\boldsymbol{I_a}\|$). From this set, we then select the candidate with maximal visibility gain, and if multiple equal candidates exist, the one closest (in terms of a Euclidian distance) to the current sampling position s (line 8). This candidate is finally added to the sampling set S (line 9) and becomes the new current sampling position (line 10). This procedure is repeated until an exit condition is reached (line 2) – for example if C remains empty for two subsequent iterations, or a maximum number of iterations is completed. Note that the bit-wise average of visibility codes represents results we achieve when computing the integral image by averaging corresponding single images at the selected sample positions, as explained in Sect. I. Note also that although this method is deterministic for the same start position s, the local greedy decision in each iteration does not guarantee finding a global optimal solution. Therefore we repeat Algorithm 1 for several random starting positions and choose the sampling set S with maximal visibility ($\max(\|\boldsymbol{I_a}\|)$) for which the visibility over all $\boldsymbol{p}_\uparrow$ is sufficiently uniform ($\text{var}(\boldsymbol{I_a}) \leq T$, for a given variance threshold T). By thresholding the visibility variance

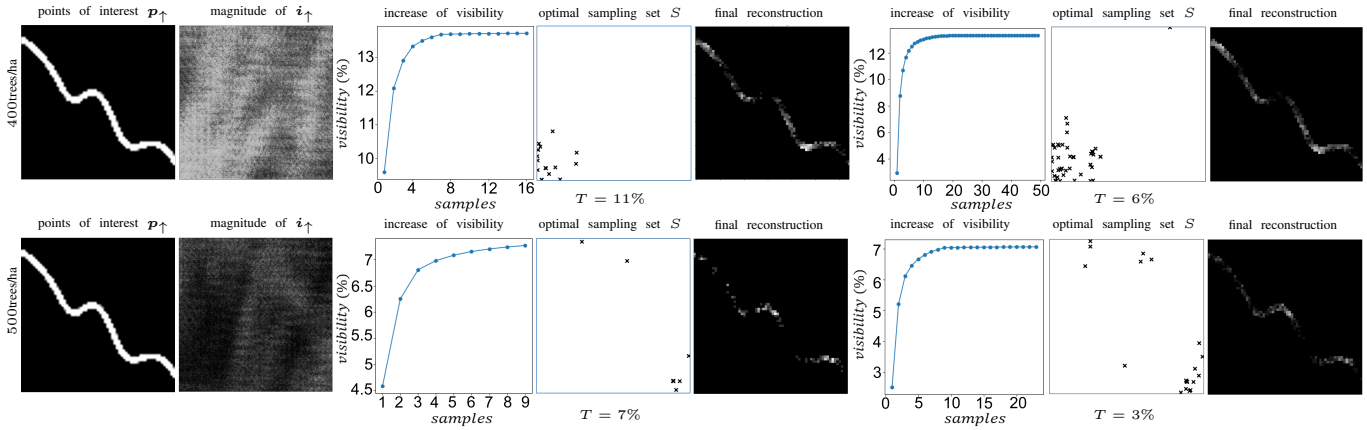

Fig. 7. Visibility magnitudes (scaled for better visibility) of a forest path covered by 400trees/ha and 500trees/ha. A lower variance threshold T leads to more uniform reconstructions with more samples required and slightly smaller overall visibility (for 400trees/ha: 13.72% visibility and 16 samples for $T = 11\%$ variance vs. 13.32% visibility with 49 samples for $T = 6\%$; for 500 trees/ha: 7.27% visibility with 9 samples for $T = 7\%$ variance vs. 7.07% visibility with 23 samples for $T = 3\%$). Simulated area for both cases: $32 \times 32\text{m}^2$. For all cases, 50 random start positions were used.

we restrict solutions with an extremely high visibility for some points in $\boldsymbol{p}_\uparrow$ while others have an extremely low visibility. Note, that T correlates to the degree of occlusion, and has been set to the minimum variance of I_a over all sets S to maximize uniformity in visibility. Note that the presented sampling scheme is a greedy approach; as such it is not guaranteed to provide an optimal solution. Moreover, due to the structure of the algorithm it is difficult at this stage to make any theoretical claims about its reliability, as the corresponding statistics will depend on empirically set algorithm parameters, e.g., threshold T. Nonetheless, it does lead to increased visibility, as we will show in the following. The examples illustrated in Fig. 6 show that even with a small number of samples (18 for 400trees/ha and 11 for 500trees/ha) relatively high target visibilities (60.8% for 400trees/ha and 36.4% for 500trees/ha) can be achieved. For comparison, the unguided particle swarm optimization (POS) presented in [3] achieves (for targets of a similar size), 42% with 160 samples for 400trees/ha and 31% with 190 samples for 500trees/ha. Once all sampling positions are determined, they can be covered either sequentially by a single drone (here, a shortest-path algorithm can be applied for path-planning [10]), or in sequential batches of N parallel samples captured by a swarm of N drones. In the latter case closest samples in subsequent batches can be determined with minimum cost bipartite matching [11]).

While in all previous examples $\boldsymbol{p}_\uparrow$ was a simple rectangular region of interest on the ground, the example in Fig. 7 covers a more sophisticated shape of a forest path. Since the path is sampled with 240 points, we compute 240 bits of its integrated visibility code, as explained in Sec. III.

V. Conclusion and Outlook

In this article we show by simulation that pre-scanned point-clouds can be utilized to guide real-time synthetic aperture sampling. Our approach is scalable with respect to the covered area. For real-time monitored static ground regions and widely unchanged forests the visibility map is static and can be pre-computed. We plan to validate the approach with real LiDAR scans to determine $\boldsymbol{i}_\uparrow$, also accounting for localization and pose estimation uncertainties, typical for, e.g., GPS, IMU, or compass. Here, an additional stochastic sampling as one proposed in [3] within the proximity of the RV-constrained sampling positions could be added. The simple greedy choice sampling strategy presented above is not deterministic, as it depends on random seed points. Our visibility variance threshold could be better derived directly from an estimated degree of occlusion instead of by maximizing uniformity in visibility over all points of interest. Furthermore, $\boldsymbol{i}_\uparrow$ can initially be down-sampled or segmented to ensure visibility clusters of minimal size that better reflect imprecision. The investigation of extensions to better cope with real data is part of our future work.

Acknowledgment

Thanks to Otto von Guericke University for support in the simulation. This research was funded by the Austrian Science Fund (FWF) and German Research Foundation (DFG) under grant numbers P32185-NBL and I 6046-N, as well as by the State of Upper Austria and the Austrian Federal Ministry of Education, Science and Research via the LIT-Linz Institute of Technology under grant number LIT2019-8-SEE114.

References

[1] I. Kurmi, D. Schedl, and O. Bimber, Airborne Optical Sectioning, J. Imaging, vol. 4, no. 102, 2018.
[2] Davis, A., Levoy, M. and Durand, F. (2012), Unstructured Light Fields. Computer Graphics Forum, 31: 305-314.
[3] R. J. A. A. Nathan, I. Kurmi and O. Bimber, Drone swarm strategy for the detection and tracking of occluded targets in complex environments, Nature Communications Engineering 2, 55, 2023.
[4] B. Hapke, Theory of Reflectance and Emittance Spectroscopy, Cambridge University Press, Cambridge, UK, 1993
[5] D. C. Schedl, I. Kurmi and O. Bimber, Search and rescue with airborne optical sectioning, Nature Machine Intelligence, 2, 783–790, 2020.
[6] P. Sen, et al., Dual photography, Proc. ACM SIGGRAPH, pp. 745–755, 2005. pp. 745–755, 2005.
[7] Y. Ding, et al., "Polarimetric Helmholtz Stereopsis", IEEE Trans. Pattern Anal. Mach. Intell., vol., no., pp. 1-15, 2024, doi: 10.1109/TPAMI.2024.3357100.
[8] Wetzstein, G. and Bimber, O., "Radiometric Compensation through Inverse Light Transport", Proc. Pacific Graphics, pp. 391-399, 2007.
[9] P. Debevec, Virtual Cinematography: Relighting through Computation, in Computer, vol. 39, no. 8, pp. 57-65, Aug. 2006.
[10] A. Madkour, W. G. Aref, F. U. Rehman, M. A. Rahman, and S. Basalamah, A survey of shortest-path algorithms, arXiv preprint arXiv:1705.02044, 2017.
[11] M. K. Asathulla, S. Khanna, N. Lahn, and S. Raghvendra, A Faster Algorithm for Minimum-cost Bipartite Perfect Matching in Planar Graphs, ACM Trans. on Algorithms, vol. 16, issue 1, article no. 2, pp 1–30, 2019.