

Accurate Point Measurement in 3DGS - A New Alternative to Traditional Stereoscopic-View Based Measurements

Deyan Deng¹, Rongjun Qin^{1,2*}

¹ Dept. of Civil, Environmental and Geodetic Engineering, The Ohio State University, Columbus OH, US

² Dept. of Electrical and Computer Engineering, The Ohio State University, Columbus OH, US
{deng.1069, qin.324}@osu.edu

Keywords: 3D Gaussian Splatting, Point Measurement, Stereoscopic Measurement, Photogrammetry, Spatial Intersection, Web Application.

Abstract

3D Gaussian Splatting (3DGS) has revolutionized real-time rendering with state-of-the-art novel view synthesis, but its applicability to accurate geometric measurement remains limited. Compared with multi-view stereo (MVS)-based point clouds or mesh models, 3DGS provides superior visual quality and completeness, while existing measurement approaches still rely on stereoscopic workstations or direct measurements on incomplete and inaccurate reconstructed geometry. As a novel view synthesizer, 3DGS reproduces source views and smoothly interpolates intermediate viewpoints, enabling users to intuitively identify congruent points across multiple views. By triangulating these correspondences, accurate 3D point measurements can be obtained. Inspired by traditional stereoscopic measurement, the proposed approach removes the need for stereo workstations and biological stereoscopic capability, while naturally supporting multi-view measurements for improved accuracy. We implement a web-based application to demonstrate this proof of concept using UAV-based aerial datasets. Experimental results show that the proposed method achieves measurement accuracy comparable to or better than traditional stereoscopic measurement approaches while operating entirely on non-stereo workstations. In particular, the proposed method consistently outperforms direct mesh-based measurements, achieving RMSEs of 1–2 cm on well-defined points. On challenging thin structures, the proposed method reduces RMSE from 0.062 m to 0.037 m, and successfully measures sharp corners where mesh-based methods fail entirely. The source code and documentation are open-source and available at: https://github.com/GDAOSU/3dgs_measurement_tool.

1. Introduction

3D point coordinate measurement is a core process in the surveying, mapping, and construction industries. Traditionally, this was dominated by *in-situ* methods such as total stations and Global Navigation Satellite Systems (GNSS), which provide high-accuracy coordinates through direct field observation of discrete points. The evolution toward non-contact, remote sensing techniques—particularly photogrammetry and LiDAR—marked a significant paradigm shift (Luhmann et al., 2013, Wehr and Lohr, 1999, Vosselman and Maas, 2010, Tong et al., 2025).

The current practice in generating 3D models is largely dominated by automated dense image matching and multi-view stereo (MVS) algorithms (Hirschmüller, 2008, Furukawa and Ponce, 2010). While this paradigm has significantly advanced applications such as large-scale surface modeling and true-orthophoto rectification, the resulting *explicit geometric representations* (e.g., dense point clouds and 3D meshes) still suffer from measurement inaccuracies. The reliability of querying coordinates from these models is fundamentally constrained by the geometric fidelity and surface smoothing inherent in the reconstruction pipeline. Consequently, for applications demanding high-precision point measurements, the industry still often relies on traditional manual or stereoscopic approaches, which require specialized hardware and operator expertise (Luhmann et al., 2013, McGlone, 2013).

Recently, 3D Gaussian Splatting (3DGS) has emerged as a disruptive technology in both research and industry (Kerbl

et al., 2023, Li et al., 2025, Huang et al., 2025, Zhong et al., 2025). Building on breakthroughs in neural rendering, including NeRF-based scene representations (Mildenhall et al., 2020) and neural point-based rendering (Aliev et al., 2020), 3DGS provides high-fidelity, photorealistic renderings in real-time. However, unlike a mesh or point cloud, 3DGS does not encode an explicit surface; it represents a scene as a set of optimized 3D Gaussians designed for view synthesis. This presents a challenge for metrology, as a simple “pick” on a rendered image does not correspond to a well-defined geometric surface.

To bridge this gap, this study proposes a new alternative by developing and validating a web-based application that enables precise 3D point measurement from 3DGS renderings. Our system is built on the Cesium platform (Cesium Team, 2025), streaming 3DGS results converted to 3D Tiles (Open Geospatial Consortium, 2018). We reintroduce the classical principle of stereo measurement—spatial intersection—but implement it in a flexible, modern visualization environment. The method computes a 3D coordinate as the least-squares intersection of multiple rays, following standard multi-view geometry formulations (Hartley and Zisserman, 2003). Figure 1 shows the user interface of the developed web application.

As an alternative to traditional stereoscopic-view based systems, our web-based implementation of spatial intersection offers several significant advantages over the classic methods it modernizes. First, it democratizes the measurement process by removing the reliance on specialized hardware, such as stereo glasses or 3D mice. Consequently, it no longer requires the operator to possess biological stereo viewing capabilities,

* Corresponding author: qin.324@osu.edu

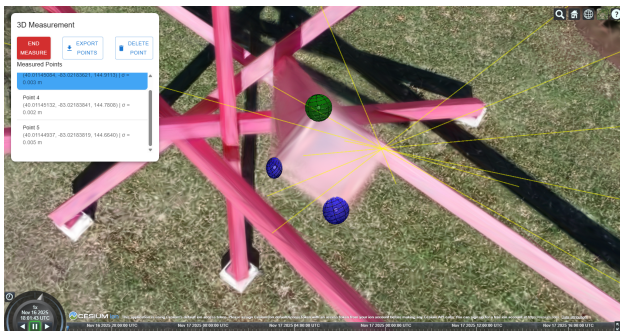


Figure 1. User interface of the web-based measurement application. The panel (left) lists the computed 3D points and their corresponding RMS error (σ). In the 3D view, multiple user-aimed rays (yellow lines) are used to compute the 3D point via spatial intersection. The final point is visualized as a 3D error ellipsoid (blue/green), providing immediate feedback on the measurement's precision.

making it accessible to a wider range of users. Second, it simplifies the workflow by directly leveraging the camera pose from the rendering engine (e.g., Cesium), eliminating the need for complex camera model calibration and pose format conversions between different software packages. Finally, our method is not limited to the two views of a stereo pair; it inherently supports a multi-ray ($N > 2$) intersection, which allows for a more robust least-squares adjustment (as discussed in Sec. 2.1) and can deliver higher precision than traditional two-ray stereo triangulation.

To validate the accuracy and advantages of this approach, we compare our results against (1) direct measurements on a photogrammetric mesh model and (2) high-precision ground truth coordinates derived from tie-point triangulation. This analysis verifies that our web-based tool is an accurate and reliable method, combining the rendering fidelity of 3DGS with the metrological rigor of photogrammetry.

This work presents a measurement-oriented framework for 3DGS that enables accurate 3D point measurement within a modern, web-based visualization environment. By integrating classical forward intersection principles with 3DGS rendering, the proposed approach extends the use of 3DGS from visualization to quantitative measurement. The main contributions of this paper are threefold:

- A web-based multi-ray spatial intersection method that enables accurate 3D point measurement directly from 3DGS renderings;
- An intuitive uncertainty-aware user interface that provides immediate feedback on measurement precision;
- A quantitative evaluation demonstrating accuracy comparable to established photogrammetric approaches and superior reliability relative to mesh-based measurements.

2. Methodology

Our methodology leverages the unique strength of 3DGS as a high-fidelity view exploration tool. Unlike static mesh models, 3DGS provides seamless, photorealistic novel view synthesis, allowing an operator to intuitively navigate a scene and inspect

a target of interest from multiple perspectives. This smooth view transition, which replicates moving between the original dense source images, makes it straightforward for a user to identify the same semantic point, referred to as a congruent point, across several different views.

The core of our method is to then apply the classic, robust photogrammetric principle of spatial intersection to these user-identified congruent points. By collecting the viewing rays from multiple perspectives, we can compute a highly accurate 3D coordinate for the target, bypassing the geometric inaccuracies of an underlying mesh.

2.1 Core Principle: 3D Point from Spatial Intersection

In traditional photogrammetry, spatial intersection (or forward triangulation) computes the 3D coordinate of a point by finding the intersection of at least two 3D rays. Each ray is defined by an image measurement and its corresponding camera's pose. Our method applies this exact principle, using the 3DGS rendering as a high-fidelity guide for aiming these rays.

2.1.1 Geometric and Algebraic Formulation A single 3D ray, \mathbf{R}_i , is defined by its origin $\mathbf{C}_i = (X_i, Y_i, Z_i)^T$ (the camera center) and a 3D unit direction vector $\mathbf{d}_i = (u_i, v_i, w_i)^T$. We seek to find the unknown 3D point $\mathbf{P} = (x, y, z)^T$ that lies at the intersection of N such rays (where $N \geq 2$).

Geometrically, if the point \mathbf{P} lies perfectly on the ray \mathbf{R}_i , then the vector from the camera center to the point, $(\mathbf{P} - \mathbf{C}_i)$, must be parallel to the ray's direction vector \mathbf{d}_i . In a perfect, noise-free system, the cross product of two parallel vectors is the zero vector:

$$(\mathbf{P} - \mathbf{C}_i) \times \mathbf{d}_i = \mathbf{0} \quad (1)$$

Let $\Delta \mathbf{P}_i = \mathbf{P} - \mathbf{C}_i = (x - X_i, y - Y_i, z - Z_i)^T$. Expanding the cross product yields three equations:

$$\begin{cases} (y - Y_i)w_i - (z - Z_i)v_i = 0 \\ (z - Z_i)u_i - (x - X_i)w_i = 0 \\ (x - X_i)v_i - (y - Y_i)u_i = 0 \end{cases} \quad (2)$$

Although there are three equations, only two are linearly independent. We can select any two to form a linear system for that ray. Choosing the first two and rearranging them to isolate the unknown parameters (x, y, z) on one side, we get:

$$\begin{cases} (0)x + (w_i)y + (-v_i)z = w_iY_i - v_iZ_i \\ (-w_i)x + (0)y + (u_i)z = -w_iX_i + u_iZ_i \end{cases} \quad (3)$$

2.1.2 Least-Squares Solution This system is for a single ray. In a real-world scenario, due to small aiming errors, the N rays will not intersect at a single perfect point. We therefore formulate a linear least-squares problem, $\mathbf{Ax} = \mathbf{b}$, to find the point \mathbf{P} that minimizes the sum of squared distances to all N rays.

For each ray i , we define a 2×3 matrix \mathbf{A}_i and a 2×1 vector \mathbf{b}_i :

$$\mathbf{A}_i = \begin{bmatrix} 0 & w_i & -v_i \\ -w_i & 0 & u_i \end{bmatrix}, \quad \mathbf{b}_i = \begin{bmatrix} w_i Y_i - v_i Z_i \\ -w_i X_i + u_i Z_i \end{bmatrix} \quad (4)$$

The full system for N rays is constructed by stacking these individual matrices:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_N \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_N \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (5)$$

Here, \mathbf{A} is a $2N \times 3$ matrix, \mathbf{b} is a $2N \times 1$ vector, and \mathbf{x} is the 3×1 vector of the unknown 3D coordinates. This overdetermined system is robustly solved for \mathbf{x} using the normal equation:

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (6)$$

2.1.3 Uncertainty Assessment After solving for \mathbf{x} , the quality of the intersection can be assessed. The vector of residuals \mathbf{v} is computed as $\mathbf{v} = \mathbf{A}\mathbf{x} - \mathbf{b}$. The *a posteriori* variance of unit weight, $\hat{\sigma}_0^2$, is then given by:

$$\hat{\sigma}_0^2 = \frac{\mathbf{v}^T \mathbf{v}}{r} \quad (7)$$

where r is the redundancy of the system (degrees of freedom), calculated as $r = 2N - 3$ (for N rays, each contributing 2 equations, and 3 unknown parameters).

The Root Mean Square (RMS) of the residuals, which represents the average geometric distance of the rays from the computed point, is $\text{RMS} = \sqrt{\hat{\sigma}_0^2}$. This RMS value serves as a crucial uncertainty indicator for each measurement. Furthermore, the covariance matrix for the estimated point \mathbf{P} is:

$$\mathbf{Q}_{\hat{\mathbf{x}}\hat{\mathbf{x}}} = \hat{\sigma}_0^2 (\mathbf{A}^T \mathbf{A})^{-1} \quad (8)$$

The diagonal elements of $\mathbf{Q}_{\hat{\mathbf{x}}\hat{\mathbf{x}}}$ provide the variances ($\sigma_x^2, \sigma_y^2, \sigma_z^2$) of the computed coordinates, allowing for a rigorous, quantitative assessment of the measurement's precision.

Although an arbitrary number of synthetic observations can be generated, they do not increase the effective degrees of freedom beyond those provided by the original images. Increasing the number of sampled points primarily improves numerical stability rather than introducing new independent information.

2.2 Algorithm Flow for 3DGS-based Measurement

The 3DGS model provides the two components necessary for this method: (1) a high-fidelity rendered image and (2) the precise camera pose for that rendering. Our method overcomes the lack of an explicit surface by using the 3DGS rendering purely as a high-fidelity *visual reference* for aiming the measurement rays. The algorithm flow for a single point measurement is as follows:

Algorithm 1: Multi-Ray Spatial Intersection from 3DGS

1. **Initialize:** Create an empty list L_{rays} to store measurement rays.
2. **Loop** for $i = 1$ to N (where $N \geq 2$ is the number of views):
 - Operator navigates the 3D viewer to a desired viewpoint V_i .
 - Operator selects a pixel p_i corresponding to the congruent point.
 - Get current camera pose C_i (origin) and direction D_i (from C_i through p_i) from the rendering engine.
 - Define ray $\mathbf{R}_i = (\mathbf{C}_i, \mathbf{d}_i)$, where \mathbf{C}_i is the camera origin and \mathbf{d}_i is the unit direction vector.
 - Add \mathbf{R}_i to L_{rays} .
3. **Compute:** Construct the \mathbf{A} matrix and \mathbf{b} vector by stacking the components ($\mathbf{A}_i, \mathbf{b}_i$) from each ray $\mathbf{R}_i \in L_{rays}$ (as defined in Sec. 2.1.2).
4. **Solve:** Calculate the 3D point $\mathbf{x} = (x, y, z)^T$ using the least-squares solution: $\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$.
5. **Assess:** (Optional) Compute residuals $\mathbf{v} = \mathbf{A}\mathbf{x} - \mathbf{b}$ and uncertainty metrics as defined in Sec. 2.1.3.
6. **Return:** The 3D coordinate \mathbf{x} , standard deviation $\hat{\sigma}_0$ and covariance matrix $\mathbf{Q}_{\hat{\mathbf{x}}\hat{\mathbf{x}}}$.

The resulting 3D coordinate \mathbf{x} , along with its computed uncertainty (e.g., the full covariance matrix $\mathbf{Q}_{\hat{\mathbf{x}}\hat{\mathbf{x}}}$ and the derived standard deviations), is then recorded in a list of all measured points for subsequent use.

2.3 Web-Based PoC Implementation

To validate this methodology, we implemented the procedure described in Algorithm 1 as a web-based proof-of-concept (PoC) application. This implementation allows operators to perform measurements from any standard web browser. The client-side application is built directly upon the **Cesium.js platform (Cesium Team, 2025)**, which provides the core 3D globe, all camera control APIs, and the high-fidelity 3DGS rendering engine. The system's technical architecture, which handles secure data streaming and computation, is briefly summarized in Table 1.

3. Experiments and Analysis

Our primary hypothesis is that 3DGS-based models, when queried with our proposed multi-ray intersection method, yield more accurate and robust 3D measurements than direct measurements on conventional mesh models. We further hypothesize that this advantage is most pronounced in challenging geometric areas—such as thin structures and sharp edges—where mesh representations are notoriously prone to smoothing or failure.

To test this, we designed a quantitative experiment to assess absolute accuracy against a high-precision ground truth. We

Table 1. System Implementation Architecture

Component	Technology & Role
Data Storage	Amazon S3. Hosts 3DGS models converted to OGC 3D Tiles.
Backend	Node.js / Express.js. Serves the client application and acts as a secure gateway to stream 3D Tiles data from S3.
Web Client	Cesium.js / React. Cesium.js provides the core 3DGS rendering, 3D globe, and camera APIs. React manages the UI overlays.
Computation	JavaScript. Implements the client-side least-squares solution (Eq. 6) within the client application.

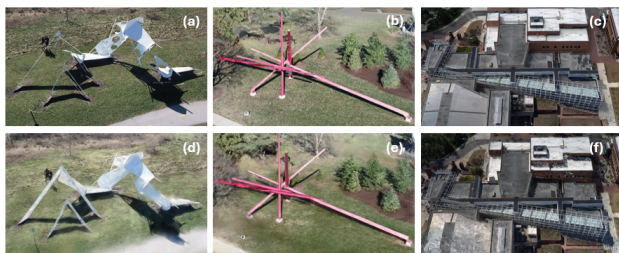


Figure 2. Datasets used for experiments. (a - c): mesh models for Dataset 1 - 3; (d - f): 3DGS models for Datasets 1 - 3.

selected three distinct, real-world UAV datasets (Fig. 2) that embody these common challenges (e.g., reflective surfaces, thin poles, complex trusses). Success on these varied scenes provides strong evidence for the general applicability of our method.

3.1 Experimental Setup

3.1.1 Datasets We tested this idea on three datasets captured with UAV imagery and processed using Bentley iTwin Capture Modeler. This software was used to generate both the 3DGS scenes and the conventional 3D textured meshes.

- **Dataset 1 (Steel Sculpture¹):** A white steel statue with complex, smooth, and reflective surfaces. It includes several thin, pole-like structures (e.g., traffic light poles, instruction poles). This dataset was used for accuracy and the advantage analysis.
- **Dataset 2 (Wood Sculpture²):** A structure made of red wood and steel. It features sharp, rectangular corners at its base. This dataset was also used for accuracy and the advantage analysis.
- **Dataset 3 (Campus Building):** A building on campus featuring complex steel structures, fine trusses, and thin railings. This dataset was used for the accuracy assessment.

Figure 2 shows mesh and 3DGS models for the three datasets used in this study.

¹ The artwork is titled 'Out of Bounds'.

² The artwork is titled 'PIVOT'.

3.1.2 Ground Truth Generation To establish a high-precision ground truth, we defined a set of Validation Points (VPs) for each scene. These VPs are high-accuracy 3D coordinates derived from multi-ray photogrammetric triangulation, and they serve as our "gold standard" for error analysis. This process was conducted *after* a successful aerial triangulation (AT) of the entire image block using the tools within iTwin Capture Modeler.

For each of the 20 desired VPs, we used the "Survey" tools to "Add Tie Point." This initiated a multi-view measurement workflow. The operator manually identified the target feature (e.g., a building corner) on one image and then selected it on multiple corresponding images. To ensure the highest accuracy, the operator zoomed in significantly on each image to achieve precise, sub-pixel pointing.

Each VP was measured in a minimum of five images (and often more, to increase geometric redundancy). Once measured, iTwin Capture Modeler computes the 3D coordinate of the point via a robust spatial intersection based on the known camera poses from the AT. As shown in Figure 3, this multi-image method ensures a rigorous and reliable 3D position, which served as our ground truth for all subsequent error analysis.

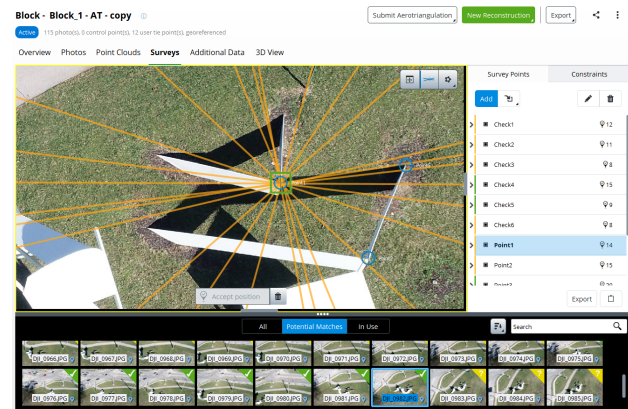


Figure 3. Ground truth generation in iTwin Capture Modeler (Bentley Systems, 2024). A single validation point (VP) is precisely measured across multiple source images. The system then computes its 3D coordinate via rigorous spatial intersection, which serves as the high-accuracy ground truth for our validation.

The ground-truth measurements are themselves subject to operator-dependent uncertainty. However, since the same operator and protocol were used consistently across all methods, this uncertainty is expected to affect all methods similarly and does not bias the relative comparison.

3.1.3 Comparison Method For a baseline comparison, we performed standard point measurements on mesh. These measurements were conducted in the iTwin Capture Desktop Viewer (Bentley Systems, 2024), which provides tools to view the mesh model and measure points, lines, and surfaces by clicking directly on the 3D mesh geometry.

3.2 Accuracy Assessment

An operator measured the 20 ground truth VPs (Figure 4) on all three datasets using two methods:

1. **Mesh:** A single-click measurement on the 3D mesh model in the iTwin Capture viewer.



Figure 4. Distribution of the 20 validation points (VPs) used for the accuracy assessment, shown on the 3DGS rendering of the three datasets. (Top Left) Dataset 1: steel sculpture; (Bottom Left) Dataset 2: wood sculpture. (Right) Dataset3: campus building.

2. **3DGS** ($N = 5$): Our web-based method, using $N = 5$ rays aimed at the target in the 3DGS rendering.

The 3D coordinate error was computed for each VP by comparing the measured coordinate to the ground truth coordinate. The Root Mean Square Error (RMSE) was then calculated for each method.

The results, summarized in Table 2, clearly demonstrate that the 3DGS spatial intersection method achieved a consistently higher accuracy (lower RMSE) and better precision (lower standard deviation) than the conventional mesh-picking method across all three datasets.

This superior performance is attributable to two primary factors that directly address the limitations of mesh-picking.

First, our method improves statistical stability through redundancy. By computing a least-squares solution from $N > 2$ rays, it inherently averages out and mitigates the small, random human aiming errors from any single viewpoint. The mesh-picking method, in contrast, is an $N = 1$ measurement; its accuracy is entirely and precariously dependent on the perfection of a single click from a single line of sight.

Second, the 3DGS rendering provides a higher-fidelity target for aiming. Mesh models are, by nature, an *approximation* of the real-world geometry, subject to surface smoothing and texture interpolation that can blur or slightly displace sharp features. The 3DGS rendering, however, provides a photorealistic visual reference that is closer to the original source imagery, allowing the operator to aim at a sharper, more defined visual target.

The "ambiguity" of the single-click method, as noted, is its fundamental flaw. A user *thinks* they are clicking a feature's apex, but they are actually just intersecting a ray with a pre-existing mesh triangle. Our 3DGS method avoids this entirely: the user identifies the *visual feature* from multiple angles, and the algorithm *computes* its 3D coordinate. This process inherently resolves parallax ambiguity and is not constrained by a pre-computed, and potentially smoothed, mesh surface.

While the accuracy gains on these well-defined validation points are in the 1-2 cm range, this test highlights a critical weakness in mesh-based metrology. As we will demonstrate in the following section, this weakness escalates from a minor inaccuracy to a complete measurement failure when applied to difficult or thin structures.

Table 2. Accuracy assessment (in meters) comparing mesh and 3DGS measurements against 20 validation points (VPs) per dataset. **ME** = Mean Error (Bias), **Std** = Standard Deviation (Precision).

Dataset	Method	N	RMSE	ME	Std
1 (Steel)	Mesh	20	0.030	0.028	0.015
	Ours	20	0.022	0.022	0.003
2 (Wood)	Mesh	20	0.034	0.033	0.005
	Ours	20	0.021	0.019	0.006
3 (Building)	Mesh	20	0.031	0.028	0.024
	Ours	20	0.019	0.014	0.008

3.3 Examples of measuring complex objects

This test highlights the primary advantage of 3DGS in capturing detail where meshes fail, particularly with common UAV capture scenarios. We selected target points on two types of 'difficult' features:

- Thin Structures (Dataset 1):** Thin, pole-like structures (e.g., traffic light poles, instruction poles) on the Statue dataset. These are classic failure cases for meshing algorithms, which tend to reconstruct them as fragmented or discontinuous.
- Low-Angle/Sharp Corners (Dataset 2):** Geometrically sharp, rectangular corners at the base of the "Red Wood" statue. These features were captured from high-angle UAV-only imagery. This common capture scenario resulted in poor viewing angles and insufficient image data near the ground, causing the mesh reconstruction algorithm to fail.

An operator attempted to measure a key point on these features using both the mesh-pick method and our 3DGS method.

The results from this test highlight the practical limitations of mesh-based metrology and the distinct advantages of our 3DGS-based method.

For **thin structures (Dataset 1)**, five feature points were selected on pole-like objects. The mesh-based method could only produce a valid measurement for three of them, as the mesh geometry for the other two points was fragmented and discontinuous, as shown in Figure 5 Col. 2. The proposed 3DGS-based method, however, successfully measured all five points because the 3DGS rendering remained visually coherent (Figure 5 Col. 3). Furthermore, for the three measurable points, the mesh method yielded an RMSE of **0.062 m**, while the 3DGS method achieved a significantly better RMSE of **0.037 m**. This demonstrates a clear improvement in both measurement completeness and accuracy.

For **low-angle sharp corners (Dataset 2)**, five feature points were selected. The mesh reconstruction completely failed to recover these features due to the high-angle (nadir-only) UAV imagery, which lacked oblique views of the vertical faces (Figure 6 Col. 2). This resulted in **zero measurable points** for the mesh-picking method. In stark contrast, the 3DGS rendering, optimized for view synthesis, clearly depicted the corner (Figure 6 Col. 3). This allowed the 3DGS method to successfully measure all five points with a high-accuracy

Point	Mesh detail	3DGS detail	Mesh measurement	3DGS measurement
1			✓	✓
2			✓	✓
3			✓	✓
4			✗	✓
5			✗	✓

Figure 5. **Visual analysis of measurement feasibility on thin structures (Dataset 1).** This table compares measurement attempts on five different thin, pole-like features. The *Mesh detail* column (Col. 2) shows the mesh geometry, which appears missing or discontinuous for points 4 and 5 (circled in yellow), leading to measurement failure. The *3DGS detail* column (Col. 3) shows the visually coherent 3DGS rendering, which enabled successful and precise measurements (indicated by the blue error ellipsoid) for all five points.

RMSE of **0.013 m**, demonstrating its robustness even with challenging capture geometries.

These two failure cases are directly linked to the fundamental assumptions of their respective technologies. Mesh reconstruction pipelines are optimized for continuous surfaces and are highly dependent on capture geometry; they often fail when faced with thin objects (which violate surface patch assumptions) or poor viewing angles (which create data gaps). Our method is immune to these geometric reconstruction failures. 3DGS is optimized for **view synthesis**, not **geometric explicitness**. As long as a feature is visible in the source images, 3DGS can typically produce a photorealistic rendering of it. Our system leverages this: it does not "pick" a model, but rather uses the high-fidelity rendering as a visual guide to perform a robust photogrammetric spatial intersection. This effectively bypasses the entire mesh-generation problem, allowing operators to measure what they can *see*, not just what the meshing algorithm managed to *build*.

4. Conclusions

This paper introduced and validated a proof-of-concept (PoC) web application that bridges the gap between the high-fidelity rendering of 3D Gaussian Splatting (3DGS) and the critical industry need for precise 3D metrology. We demonstrated that by re-implementing the classic photogrammetric principle of spatial intersection, a 3DGS scene can serve as a robust visual reference for accurate coordinate measurement.

Point	Mesh detail	3DGS detail	Mesh measurement	3DGS measurement
1			✗	✓
2			✗	✓
3			✗	✓
4			✗	✓
5			✗	✓

Figure 6. **Measurement feasibility on sharp corners (Dataset 2).** The mesh model (column 2) fails to reconstruct the sharp corners due to poor UAV viewing angles, making measurement impossible ("✗"). The 3DGS rendering (column 3) clearly depicts the corners, enabling successful measurement ("✓").

Our experiments confirm our primary hypothesis. First, on well-defined validation points, our multi-ray least-squares method achieves a high measurement accuracy (RMSE of 1-2 cm) that is superior to direct measurements on derivative mesh models. Second, and more significantly, our method excels where mesh-based approaches fail. The 3DGS renderings retain fine details of thin structures and sharp corners that are lost in mesh reconstruction, allowing our tool to provide accurate measurements where mesh-based methods are either highly inaccurate or entirely impossible.

This PoC highlights several key advantages of this "measurement-on-rendering" approach: (1) it democratizes metrology by removing the need for specialized stereo workstations; (2) it is accessible to all operators, not just those with biological stereo-viewing capabilities; (3) it simplifies workflows by using camera poses directly from the rendering engine; and (4) it naturally supports robust, multi-ray ($N > 2$) intersections for higher precision. We acknowledge that traditional stereoscopic measurement still holds advantages, particularly in textured-poor regions where human operators can leverage structural context. Our method is presented as a powerful and accessible alternative for a wide range of applications, transforming 3DGS from a passive visualization tool into an active platform for quantitative analysis.

Future work will focus on enhancing automation. While the current method relies on manual aiming from multiple views, a clear next step is to integrate semi-automated point correspondence by leveraging feature matching techniques across renderings. Further development will also involve expanding the toolset to support other essential metrology tasks, such as measuring distances, areas, and volumes. A significant research direction will be applying this to LoD2 building model extraction, specifically by developing methods to automatically

generate topological relationships (e.g., building footprints, rooflines) from a sequence of related point measurements.

Acknowledgements

This work is supported by the Office of Naval Research (Award No. N000142312670) and Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number 140D0423C0075.

References

- Aliiev, K. et al., 2020. Neural point-based graphics. *ECCV*.
- Bentley Systems, 2024. itwin capture modeler. <https://www.bentley.com/software/itwin-capture-modeler/>. [Software].
- Cesium Team, 2025. Cesiumjs: An open-source javascript library for 3d globes and maps. <https://cesium.com/platform/cesiumjs/>. Accessed: 2025-11-13.
- Furukawa, Y., Ponce, J., 2010. Accurate, dense, and robust multiview stereopsis. *IEEE Conference on Computer Vision and Pattern Recognition*, 1362–1370.
- Hartley, R., Zisserman, A., 2003. *Multiple View Geometry in Computer Vision*. 2 edn, Cambridge University Press.
- Hirschmüller, H., 2008. Stereo Processing by Semiglobal Matching and Mutual Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2), 328–341.
- Huang, D., Liu, H., Xu, N., Qin, R., 2025. Dynamic urban scene modeling with 3D gaussian splatting from UAV full motion videos. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Geospatial Week)*.
- Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics (SIGGRAPH 2023)*, 42(4), 114.
- Li, Z., Yao, S., Wu, T., Yue, Y., Zhao, W., Qin, R., García-Fernández, A. F., Levers, A., Ralph, J., Zhu, X., 2025. ULSR-GS: Urban large-scale surface reconstruction Gaussian Splatting with multi-view geometric consistency. *ISPRS Journal of Photogrammetry and Remote Sensing*, 230, 861–880.
- Luhmann, T., Robson, S., Kyle, S., Boehm, J., 2013. *Close-Range Photogrammetry and 3D Imaging*. De Gruyter.
- McGlone, C. (ed.), 2013. *Manual of Photogrammetry*. 6 edn, ASPRS.
- Mildenhall, B. et al., 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. *ECCV*.
- Open Geospatial Consortium, 2018. Ogc 3d tiles specification 1.0, ogc 18-053r1. <https://docs.ogc.org/cs/18-053r2/18-053r2.html>.
- Tong, X., Gao, S., Ye, Z., Chen, P., Liu, S., Xie, H., Liu, X., Hong, Z., Zhang, D., Yang, J., Song, Y., Lv, H., Cao, D., 2025. Crack Parameter Determination Based on Multiple-Window Matching Strategy and Crack Grid Geometry Analysis. *Photogrammetric Engineering & Remote Sensing*, 91(9), 551–562.
- Vosselman, G., Maas, H.-G., 2010. *Airborne and Terrestrial Laser Scanning*. CRC Press.
- Wehr, A., Lohr, U., 1999. *Airborne Laser Scanning—An Introduction and Overview*. 54.
- Zhong, J., Li, M., Gruen, A., Liao, X., Qin, J., Wang, B., 2025. AHigh-Quality Underwater 3D Reconstruction Solution for Coral Reef Environments Leveraging Advanced Photogrammetric Computer Vision Techniques. *Photogrammetric Engineering & Remote Sensing*, 91(6).