BIM-Align: An Automatic Framework for BIM to As-Built Alignment with Multi-View Imagery and 3D Gaussian Splatting

Guangan Chen, Michiel Vlaminck, Gianni Allebosch, Wilfried Philips, Hiep Luong

IPI-URC, imec, Ghent University, Belgium

{guangan.chen, michiel.vlaminck, gianni.allebosch, wilfried.philips, hiep.luong}@ugent.be

Abstract -

Alignment between Building Information Models (BIM) and as-built representations is essential for automated construction progress monitoring. Existing pipelines that rely on images often require manual selection of correspondences or additional site instrumentation. Recent advances in differentiable scene representations, especially 3D Gaussian Splatting (3DGS), open new possibilities for automatic global BIM to as-built alignment through dense supervision from multi-view images. This paper presents *BIM-Align*, an automatic framework for global BIM to as-built alignment with multi-view imagery and 3DGS. The proposed framework initially estimates camera poses using Structure-from-Motion (SfM) and reconstructs a dense point cloud using multi-view stereo (MVS). Subsequently, the target building is extracted from the dense point cloud using DBSCAN clustering, and an initial BIM to as-built alignment is computed with classical geometric methods. The initial alignment is then refined with a coarse-to-fine strategy, where the coarse and fine stages each perform 3DGS optimization with a separate learnable Sim(3) transformation. Experimental results on construction site images show that the proposed approach achieves a mean reprojection RMSE within 2.13 pixels and a mean 3D alignment error within 9.18 cm of a manual baseline, demonstrating the feasibility of automatic BIM to as-built alignment from images with 3DGS.

Keywords -

3D Gaussian Splatting; Construction Site Monitoring; BIM to As-Built Alignment

1 Introduction

Timely and accurate construction progress monitoring is essential for project control and early detection of cost and schedule deviations. Conventional approaches rely on manual site inspections, photographs, and periodic reports, which require substantial labor and time. These limitations motivate automated methods that continuously capture the as-built state and perform comparisons with the as-planned BIM to quantify progress and identify deviations [1, 2, 3, 4, 5].

A common pipeline for automated construction progress monitoring consists of three stages: (i) data acquisition and as-built point cloud reconstruction, (ii) BIM to as-built alignment, and (iii) progress and deviation analysis. Since progress and deviation metrics are computed by comparing BIM and the as-built point cloud, accurate alignment is key to reliable progress monitoring. In practice, the quality and completeness of the as-built point cloud have a significant impact on alignment performance, motivating the selection of appropriate data acquisition modalities. For instance, terrestrial laser scanning and mobile LiDAR can directly capture dense point clouds with high accuracy and have been widely used in construction monitoring [2, 5, 6]. However, LiDAR-based methods require expensive sensors and specialized operation, which limits their practicality for frequent monitoring. In contrast, multi-view 3D reconstruction based on SfM and MVS recovers 3D geometry from standard RGB imagery captured by conventional cameras, which enables data capture at low cost and high frequency [7]. Early work by Golparvar-Fard *et al.* employed a custom implemented SfM pipeline to reconstruct sparse as-built point clouds from unordered construction site imagery for BIM to as-built alignment [8, 9, 10, 11]. The subsequent availability of robust, off-the-shelf SfM-MVS pipeline tools, such as VisualSFM [12], COLMAP [13], and Agisoft [14], has further lowered the barrier to acquiring as-built point clouds, enabling recent studies to generate dense as-built point clouds with improved geometric completeness for BIM to as-built alignment [2, 3, 15].

Despite the advancements in multi-view 3D reconstruction, accurate BIM to as-built alignment remains a major challenge for automated construction progress monitoring. Existing approaches can be categorized into local and global alignment methods, where local methods align individual building elements independently, whereas global methods estimate a unified transformation for the entire building. For local alignment, Omara *et al.* [16] proposed a rigid alignment strategy that aligns individual BIM elements with corresponding point cloud segments. The authors deployed 12 cameras to capture images of a specific region of a building to reconstruct an as-built point cloud. Subsets of the point cloud corresponding to specific structural components, such as reinforced concrete columns, are independently aligned with predefined BIM geometries using rigid transformations with fixed scale. However, in complex building scenes with numerous structural elements, local alignment requires a large number of images for individual elements.

For global alignment, Golparvar-Fard *et al.* [8, 9, 10, 11] proposed a BIM to as-built alignment approach based on a sparse point cloud derived from SfM. In their methods, a set of correspondences between the sparse point cloud and the BIM is manually annotated. A Sim(3) transformation is then computed with the correspondences for BIM to as-built alignment. In practice, SfM-derived sparse point clouds often suffer from incomplete scene coverage. As a result, certain regions present in the BIM lack corresponding points in the sparse point cloud, reducing the reliability of manually selected point correspondences for alignment. To address this limitation, Chen *et al.* [3] employed a dense point cloud generated by MVS for global BIM to as-built alignment using manually selected correspondences. However, these point-picking methods require manual correspondence selection, which is time-consuming and may introduce human error.

Other approaches implicitly achieve global BIM to as-built alignment by enforcing metric consistency during photogrammetric reconstruction, rather than performing an explicit geometric alignment step. Mahami *et al.* [15] placed coded photogrammetric targets on a construction site to fix the datum and absolute scale of point clouds derived from MVS. An as-built mesh was then reconstructed in Autodesk Revit [18] from the scaled point cloud, after which building elements were manually traced. Subsequently, the as-built mesh was compared with a BIM for construction progress monitoring. Similarly, Tuttas *et al.* [19] incorporated photogrammetric ground control points into the bundle adjustment [7], thereby generating point clouds directly in the BIM coordinate system. Masood *et al.* [20] leveraged GPS-based georeferencing to obtain global BIM to as-built alignment by referencing both representations to a common Universal Transverse Mercator (UTM) coordinate system. However, while these approaches demonstrate the feasibility of global BIM to as-built alignment under varying levels of manual intervention during data acquisition, achieving global BIM to as-built alignment from multi-view images without manual correspondences, additional site instrumentation, or external georeferencing remains an open challenge in practical construction environments.

Recent advances in differentiable scene representations, particularly 3D Gaussian Splatting (3DGS) [21], provide a new opportunity to address automatic global BIM to as-built alignment by leveraging dense supervision from multi-view images. 3DGS achieves high quality novel view synthesis by optimizing a set of 3D Gaussians from multi-view images and camera poses with a photometric loss. The Gaussian mean positions are typically initialized from a point cloud obtained by SfM. When the Gaussian mean positions are frozen during 3DGS training, initialization misalignment leads to misalignment in rendered views and increased photometric loss. To mitigate this effect, correcting the initial misalignment during 3DGS training reduces the impact of initialization errors on the photometric loss. In the BIM to as-built alignment setting, we assume that the Gaussian mean positions are initialized from a point cloud sampled from the BIM, while the as-built point cloud defines the reference coordinate frame. Under this formulation, BIM to as-built alignment is equivalent to correcting the misalignment between BIM and the as-built point cloud during 3DGS training. Building on this observation, this study investigates the feasibility of using 3DGS for automatic global BIM to as-built alignment. To this end, we propose an automatic framework. The proposed framework first estimates camera poses and reconstructs a dense point cloud using SfM and MVS, respectively. Subsequently, the target building is extracted from the dense point cloud, serving as the as-built point cloud. An initial BIM to as-built alignment is then estimated using classical geometric registration methods. Afterward, the initial alignment is refined by training 3DGS models initialized from the BIM in a coarse-to-fine manner.

Our main contributions are:

- We propose *BIM-Align*, an automatic framework for global BIM to as-built alignment from multi-view imagery that combines geometric initialization with 3DGS-based refinement.
- We introduce a coarse-to-fine 3DGS optimization strategy, in which BIM to as-built alignment is refined through a global learnable Sim(3) transformation with frozen Gaussian mean positions.
- We provide an evaluation on collected aerial images of a construction site using alignment metrics in image and 3D domains, together with a manual baseline based on annotated correspondences.

2 Method

Given a set of N multi-view images $\mathcal{I} = \{I_i\}_{i=1}^{N}$ of a construction site and a BIM mesh B, our objective is to automatically estimate a Sim(3) transformation that aligns the BIM with the as-built scene reconstructed from the images. The proposed framework consists of three stages and is illustrated in Figure 1. Initially, we estimate the camera poses $\Pi = \{\pi_i\}_{i=1}^{N}$ of the images and reconstruct a dense point cloud of the construction site. Subsequently, an as-built point cloud is extracted from the dense point

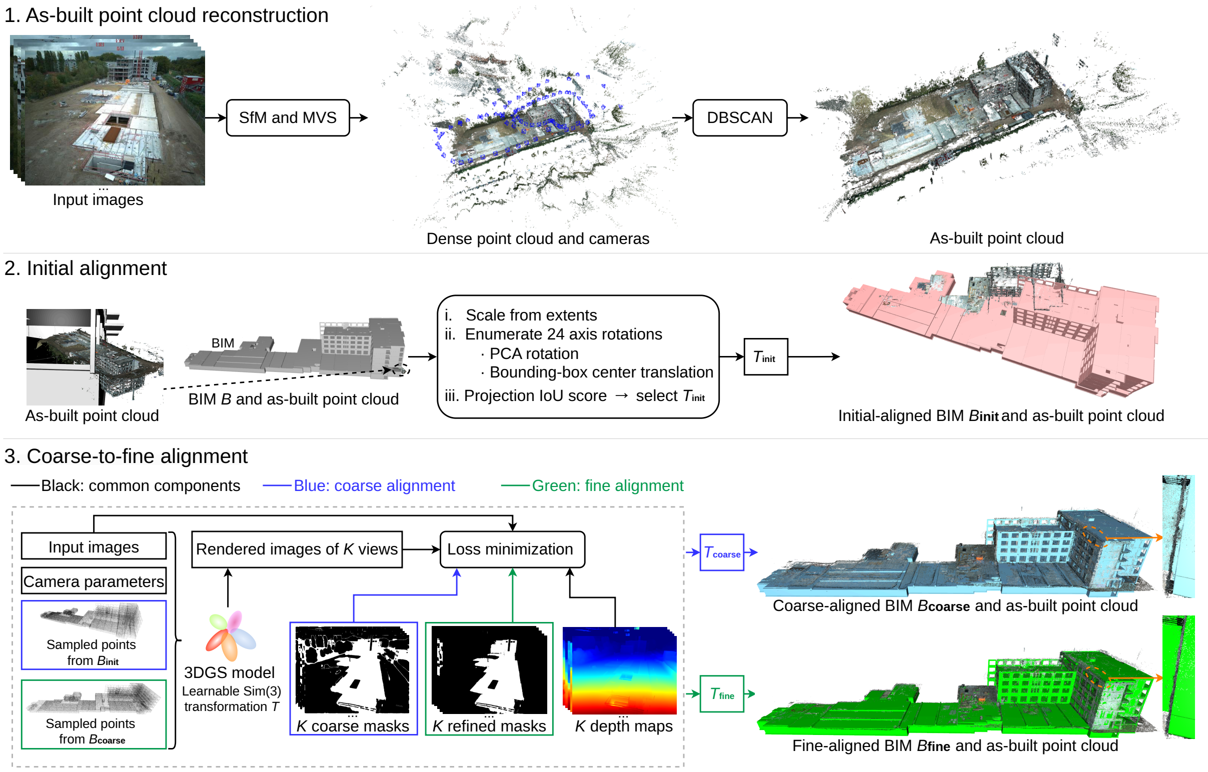

Figure 1. Overview of the proposed *BIM-Align* framework. (1) As-built point cloud reconstruction: SfM and MVS are used to estimate camera poses and reconstruct a dense point cloud, respectively. The as-built point cloud is extracted from the dense point cloud using DBSCAN clustering [17]. (2) Initial alignment: an initial Sim(3) transformation T_{init} is estimated using classical geometric alignment techniques. (3) Coarse-to-fine alignment: starting from the initial alignment, two separate 3DGS optimizations are performed. In the coarse stage, a learnable Sim(3) transform is optimized to estimate T_{coarse}. In the fine stage, the optimization is initialized from the coarse result and a separate Sim(3) transform is optimized to obtain T_{fine}. In addition, depth maps are used during 3DGS training in both phases. Specifically, coarse alignment employs coarse masks, the input point cloud sampled from B_{init}, and $K = 20$ views per iteration, whereas fine alignment employs refined masks, the input point cloud sampled from B_{coarse}, and $K = 1$ view per iteration. For visualization in stages (2) and (3), the as-built point cloud is cropped to retain only the building.

cloud. Next, an initial BIM to as-built alignment is computed using classical geometric alignment techniques, followed by a coarse-to-fine refinement using 3DGS [21].

2.1 Preliminary: 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) [21] is an explicit scene representation proposed for novel view synthesis, which models a scene as a collection of anisotropic 3D Gaussians and supports efficient differentiable rendering. Each Gaussian is parameterized as $G = \{\mu, q, s, o, c, sh\}$. The mean position μ is initialized from an input point cloud and serves as the 3D anchor of each Gaussian. The rotation quaternion q defines the Gaussian orientation, s is a scaling vector that controls spatial extent, o represents opacity, c denotes color, and sh represents spherical harmonic coefficients encoding view-dependent appearance. A scene is represented by a set of Gaussians $\mathcal{G} = \{G_j\}_{j=1}^{H}$, where H denotes the number of Gaussians. Given the image set $\mathcal{I}$ and camera poses Π, the 3DGS model renders each view as $\hat{I}_i = \mathcal{R}(\mathcal{G}, \pi_i)$ using a differentiable renderer $\mathcal{R}(\cdot)$. Scene reconstruction is achieved by optimizing the Gaussian parameters by minimizing a photometric loss $\mathcal{L}$, leading to the following objective:

$$\min_{\mathcal{G}} \sum_{i=1}^{N} \mathcal{L}(\hat{I}_i, I_i) \,.$$

In our study, 3DGS is employed as an optimization backbone for global BIM to as-built alignment rather than for novel view synthesis.

2.2 As-built point cloud reconstruction

Given the image set $\mathcal{I}$, we employ a standard SfM-MVS pipeline implemented in COLMAP [13], where SfM is used to estimate the camera poses Π and reconstruct a sparse point cloud, and MVS is subsequently applied to generate a dense point cloud. Camera intrinsic parameters are estimated implicitly through COLMAP's built-in camera models and bundle adjustment. Subsequently, DBSCAN clustering [17] is applied to the dense point cloud to identify the building of interest. As the input images are captured with a focus on the building, the largest cluster is selected as the approximate as-built point cloud.

2.3 BIM to as-built initial alignment

An initial BIM to as-built alignment provides an appropriate initialization for subsequent refinement using 3DGS. Since the BIM and the COLMAP reconstruction differ in scale and coordinate frame, direct 3DGS optimization is unreliable. We therefore estimate an initial Sim(3) transformation T_{init}.

We first estimate the global scale $s_{\text{init}} \in \mathbb{R}^+$ by matching the overall spatial extents of the BIM and the as-built point cloud, and keep the scale fixed when estimating rotation and translation. To account for ambiguous BIM axis conventions [4], we enumerate 24 axis-aligned rotation hypotheses corresponding to the rotational symmetry group of a cube. For each hypothesis, rotation is further estimated by aligning the dominant principal directions of the BIM and the as-built point cloud using PCA [22]. Translation is computed by aligning their bounding-box centers. This yields a set of candidate Sim(3) transformations $\{T^k_{\text{init}}\}_{k=1}^{24}$.

To select the most plausible candidate, we evaluate each T^k_{init} using an overlap score based on orthographic projections. Specifically, we sample points from the transformed BIM mesh, project both the sampled points and the as-built point cloud onto the XY, XZ, and YZ planes, and rasterize the projections into binary footprint images. The consistency score is computed as the sum of the Intersection-over-Union (IoU) values of the three projection pairs, and the candidate with the highest score is selected as the final initial transformation T_{init}. The initially aligned BIM is denoted as $B_{\text{init}} = T_{\text{init}}(B)$.

2.4 BIM to as-built coarse alignment

We refine the initial BIM to as-built alignment with joint optimization of a BIM-initialized 3DGS model and a learnable global Sim(3) transformation T_{coarse}. Specifically, we uniformly sample P points from the initially aligned BIM B_{init} to construct an input point cloud $B^{\text{pcd}}_{\text{init}}$, which anchors the Gaussian mean positions in 3D space. To preserve the BIM-derived Gaussian structure, Gaussian densification and pruning are disabled throughout training. In addition, each sampled point is initialized with default spherical harmonic coefficients, resulting in a constant initial appearance.

The learnable Sim(3) transformation is parameterized as $T_{\text{coarse}} = (s_{\text{coarse}}, R_{\text{coarse}}, \mathbf{t}_{\text{coarse}})$, where $s_{\text{coarse}} \in \mathbb{R}^+$ denotes the scale factor, $R_{\text{coarse}} \in \text{SO}(3)$ is a rotation matrix, and $\mathbf{t}_{\text{coarse}} \in \mathbb{R}^3$ denotes the translation. To preserve the BIM-derived Gaussian structure, the mean position μ of each Gaussian initialized from $B^{\text{pcd}}_{\text{init}}$ is frozen during 3DGS training. The transformed mean position is obtained by applying the transformation T_{coarse} to the mean: $\mu' = s_{\text{coarse}} R_{\text{coarse}} \mu + \mathbf{t}_{\text{coarse}}$. Given the transformed Gaussian mean positions $\{\mu'\}$, the rendered image for view i is given by $\hat{I}_i = \mathcal{R}(\mathcal{G}(\{\mu'\}), \pi_i)$, where $\mathcal{G}(\{\mu'\})$ denotes the Gaussian set with updated mean positions, while all remaining parameters follow the standard 3DGS formulation [21]. Consequently, the BIM to as-built alignment is refined by jointly optimizing T_{coarse} and the remaining Gaussian parameters to enforce photometric consistency between rendered and observed images across all input views:

$$\min_{\mathcal{G},\, T_{\text{coarse}}} \sum_{i=1}^{N} \mathcal{L}(\mathcal{R}(\mathcal{G}(\{\mu'\}), \pi_i)\,, I_i)\,.$$

To further regularize the optimization, we incorporate depth and mask supervision during 3DGS training. Depth supervision provides additional geometric constraints beyond photometric consistency, which can be ambiguous under weak texture, repetitive patterns, or occlusions. For each image, we predict a monocular inverse depth map using DepthAnything [23] and resolve the scale and shift ambiguity via per-image affine calibration to the SfM-derived point cloud using visible sparse 3D points. Views with abnormal calibration are discarded, and the calibrated inverse depth is used to supervise the rendered inverse depth. In addition, mask supervision suppresses the influence of non-building regions in the photometric loss, which is necessary since $B^{\text{pcd}}_{\text{init}}$ contains only building geometry. To generate masks, we use the prompt-based semantic segmentation model SAM3 [24]. In construction sites, buildings are often incomplete and only partially observed, especially at early stages where only structural slabs may be visible. As a result, extracting complete building masks using generic prompts such as *building* becomes unreliable. We therefore adopt an inverse strategy that segments typical background objects by querying SAM3 with the text prompts *dirt*, *bush*, *tree*, *car*, *sky*, *crane*, *container*, *road*, *fence*, and *metal*. A building mask is obtained by taking the complement of all segmented background regions. As a result, a set of coarse building masks $\mathcal{M}^0 = \{M^0_i\}_{i=1}^N$ for the N input images is created.

The optimization objective for a viewpoint π_i in our

3DGS training is a weighted combination of RGB and depth losses [21]:

$$\mathcal{L}^i = \lambda_{\text{rgb}} \left(\mathcal{L}^i_{\text{rgb}} \odot M^0_i\right) + \lambda_{\text{depth}} \left(\mathcal{L}^i_{\text{depth}} \odot M^0_i\right). \quad (1)$$

Here, the RGB loss $\mathcal{L}^i_{\text{rgb}}$ is a weighted combination of ℓ_1 loss and D-SSIM [25] loss, given by

$$\mathcal{L}^i_{\text{rgb}} = (1 - \lambda_{\text{D-SSIM}})\, \mathcal{L}^i_1 + \lambda_{\text{D-SSIM}}\, \mathcal{L}^i_{\text{D-SSIM}}, \quad (2)$$

where $\mathcal{L}^i_1 = \|\hat{I}_i - I_i\|_1$ and $\mathcal{L}^i_{\text{D-SSIM}} = 1 - \text{SSIM}(\hat{I}_i, I_i)$. The depth loss $\mathcal{L}^i_{\text{depth}}$ is defined as the absolute difference between the rendered and calibrated inverse depth maps. The operator $\odot$ denotes element-wise multiplication. Moreover, the weights λ_{rgb}, λ_{depth}, and $\lambda_{\text{D-SSIM}}$ control the weight of each component.

During coarse alignment, large initial misalignment causes the photometric objective to be dominated by mismatched pixels and incorrect visibilities, resulting in high variance and ambiguous gradients. As a result, single-view supervision is prone to 3DGS training failure. To stabilize the optimization, we compute the mean loss over multiple training views at each iteration. Specifically, we randomly sample $K \geq 1$ distinct indices from the training view index set $\{1, \ldots, N\}$, denoted as $\mathcal{S}$, where $\mathcal{S} \subset \{1, \ldots, N\}$ and $|\mathcal{S}| = K$. Thus, the total loss is given by

$$\mathcal{L}_{\text{all}} = \frac{1}{K} \sum_{i \in \mathcal{S}} \mathcal{L}^i. \quad (3)$$

As a result, the coarse alignment phase yields an optimized transformation T_{coarse} and a coarsely aligned BIM $B_{\text{coarse}} = T_{\text{coarse}}(B_{\text{init}})$.

Despite the segmentation capability of SAM3, the coarse building masks still contain non-building regions, providing insufficient constraints for fine-grained refinement at the coarse phase. We therefore introduce a fine alignment phase.

2.5 BIM to as-built fine alignment

The fine alignment phase leverages refined building masks to further suppress the impact of non-building regions during optimization. Based on the coarse alignment results, a geometric prior for the building region in each image is obtained by back-projecting the coarsely aligned BIM B_{coarse} into each input view, yielding BIM-projected masks $\mathcal{M}^1 = \{M^1_i\}^N_{i=1}$. However, residual misalignment can distort the projected building boundaries, causing boundary erosion and the loss of thin structures in the mask. To address this issue, we refine $\mathcal{M}^1$ using instance masks predicted by SAM3 on the input images. For each input image I_i, let $O_i = \{O^v_i\}^{V_i}_{v=1}$ denote the set of instance masks, where V_i is the number of instances segmented in I_i. We retain an instance mask O^v_i if the fraction of its pixels that lie inside M^1_i exceeds a threshold β. Subsequently, the retained instance masks are merged and combined with the BIM-projected mask M^1_i with set union to produce an intermediate refined mask M^2_i:

$$M^2_i = \left(\bigcup_m O^m_i\right) \cup M^1_i, \quad (4)$$

where O^m_i denotes the m-th retained instance mask in image I_i after overlap filtering. Despite this refinement, M^2_i may still contain clutter tightly adjacent to the building due to residual misalignment. To remove such clutter, we further refine M^2_i by intersecting it with the semantic segmentation mask M^0_i, yielding the final refined mask:

$$M^3_i = M^2_i \cap M^0_i. \quad (5)$$

The set of final refined masks is denoted as $\mathcal{M}^3 = \{M^3_i\}^N_{i=1}$.

In the fine alignment phase, we mostly follow the same optimization procedure as in coarse alignment. In contrast to the coarse phase, we initialize Gaussians using a point cloud sampled from B_{coarse} and compute the loss using the refined masks $\mathcal{M}^3$. Moreover, we calculate the loss using a single randomly sampled view per iteration, with $K = 1$. The fine alignment phase yields a refined Sim(3) transformation T_{fine} and a finely aligned BIM $B_{\text{fine}} = T_{\text{fine}}(B_{\text{coarse}})$.

3 Experiments

3.1 Dataset

We collected a dataset of aerial photographs of a construction site. The dataset contains 87 RGB images (960×720) captured from multiple viewpoints, yielding $N = 87$ input images in our experiments. In addition to the image data, we provide a corresponding as-planned BIM. To ensure consistency between the BIM and the as-built scene for alignment, we incorporate construction progress information and remove all unbuilt elements from the as-planned BIM. The resulting BIM B therefore represents the actual construction state at the time of image acquisition and serves as the BIM input to the subsequent automatic alignment framework in all experiments in our study.

We evaluate BIM to as-built alignment accuracy using reprojection error. For each image, we manually annotate a set of 2D points at distinctive geometric features and select the corresponding 3D reference points from the BIM at the same structural locations to form annotated 2D-3D correspondences. In total, we use 419 correspondence pairs across 87 images for evaluation, with a varying number of correspondences per image.

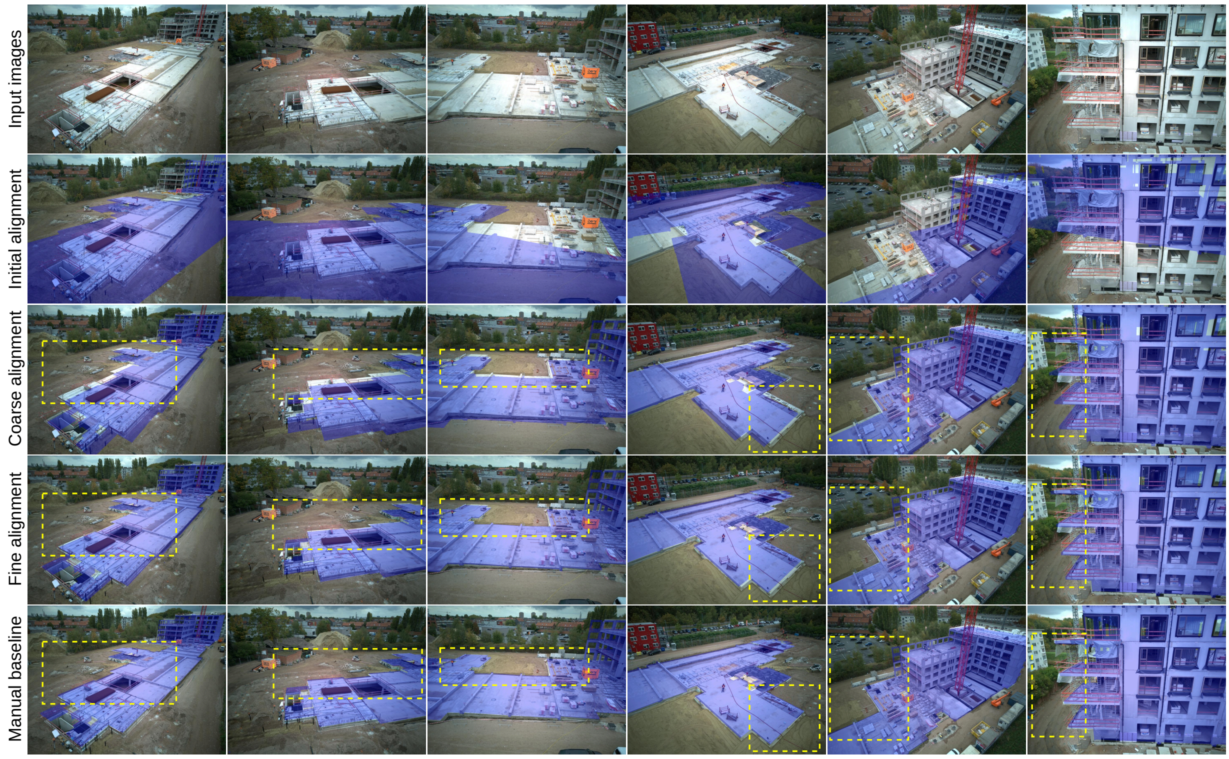

Figure 2. Qualitative comparison of BIM to as-built alignment at different stages. The aligned BIMs are back-projected into input images using the estimated camera poses. The projected BIM geometry is overlaid in blue in the visualized images. From top to bottom, the figure shows input images, initial alignment, coarse alignment, fine alignment, and the manual baseline. For visualization, BIM elements corresponding to an underground parking garage, which are not observable in the input images, are removed prior to projection.

3.2 Metrics

Alignment accuracy is evaluated in both image space and 3D space. Given an annotated 2D point p in image I_i with corresponding 3D point q on the BIM B, let T denote the estimated transformation at a given alignment stage. For the initial, coarse, and fine stages, T is defined as T_{init}, $T_{\text{coarse}}T_{\text{init}}$, and $T_{\text{fine}}T_{\text{coarse}}T_{\text{init}}$, respectively. In image space, q is transformed by T and projected onto I_i using camera pose π_i, yielding a 2D point p'. The 2D alignment error is defined as the Euclidean distance between the annotated point p and the projected point p'. In 3D space, the annotated point p is backprojected into the coordinate frame of BIM B, yielding a 3D point q''. The 3D alignment error is defined as the Euclidean distance between the corresponding BIM point q and the backprojected point q''. Since the BIM coordinate system is not expressed in physical units, the 3D distance is converted to centimeters using a scale ratio α. For each image, point-wise errors are aggregated using RMSE, and the overall performance across all images is reported using the mean ($\text{RMSE}_{\text{mean}}$), median ($\text{RMSE}_{\text{med}}$), and 95th percentile ($\text{RMSE}_{\text{p95}}$).

3.3 Implementation

In both the coarse and fine alignment phases, we uniformly sample $P = 500{,}000$ points from B_{init} and B_{coarse}, respectively, to construct the input point clouds for 3DGS training. The learnable Sim(3) transformation is optimized in a continuous parameter space. To ensure positivity, the scale is optimized in log-space and the rotation is parameterized and optimized as a unit quaternion during training. During the coarse alignment phase, the learning rates for the transformation parameters are initialized to 10^{-1} and exponentially decayed to 10^{-2} over 500 iterations [21]. The number of views used for loss computation per iteration is set to $K = 20$, balancing training stability and GPU memory constraints. In the fine alignment phase, the overlap threshold β for refined building mask generation is set to 0.9. The learning rates for the transformation parameters are initialized to 10^{-3} and exponentially decayed to 10^{-4} over 30,000 iterations. During fine alignment, a single view is used to compute the photometric loss per iteration, with $K = 1$. For metric evaluation in 3D space, we use a scale ratio of $\alpha = 99.36$ to convert distances in the BIM B coordinate system to centimeters.

3.4 Manual baseline

We compare our method against a manual point picking baseline. Specifically, we manually annotate 30 corresponding points distributed over the BIM B and the as-built point cloud, and compute the Sim(3) transformation using [26].

3.5 Results

Table 1 reports a quantitative comparison of different alignment stages using 2D reprojection error in pixel units and 3D reprojection error in metric units (cm). For evaluation, the coarse alignment is run five times. Each coarse result is then used to initialize one corresponding fine-alignment run, yielding five paired coarse-to-fine trials. We report the mean and standard deviation over these five trials.

Table 1. Quantitative results of different alignment methods measured by 2D reprojection error (px) and 3D reprojection error (cm). Lower values indicate better performance.

Method	2D reprojection error (px)			3D reprojection error (cm)		
	$\text{RMSE}_{\text{mean}} \downarrow$	$\text{RMSE}_{\text{med}} \downarrow$	$\text{RMSE}_{\text{p95}} \downarrow$	$\text{RMSE}_{\text{mean}} \downarrow$	$\text{RMSE}_{\text{med}} \downarrow$	$\text{RMSE}_{\text{p95}} \downarrow$
Ours (initial alignment)	332.52	272.46	824.43	1275.61	1131.08	2829.79
Ours (coarse alignment)	21.22 ± 3.75	18.97 ± 3.02	39.62 ± 7.03	77.85 ± 13.53	77.40 ± 13.00	138.23 ± 28.02
Ours (fine alignment)	7.11 ± 0.72	6.46 ± 0.76	15.23 ± 1.27	28.61 ± 2.83	21.16 ± 2.44	62.67 ± 5.08
Manual baseline	**4.98**	**4.46**	**10.08**	**19.43**	**15.69**	**38.37**

As shown in Table 1, the initial alignment shows severe global misalignment. Nevertheless, it provides a feasible starting point for subsequent refinement with 3DGS. From the initial to the coarse and fine stages, both the 2D and 3D reprojection errors consistently decrease, demonstrating the effectiveness of the proposed framework in progressively minimizing misalignment. Compared with the manual baseline, the fine alignment achieves competitive performance in both 2D and 3D metrics. In particular, the fine alignment reaches a 2D $\text{RMSE}_{\text{mean}}$ of 7.11 px, which is 2.13 px higher than the manual baseline, and a 3D $\text{RMSE}_{\text{mean}}$ of 28.61 cm, which is 9.18 cm higher than the manual baseline. The larger difference in 3D RMSE_{p95} suggests that the 3D metric is more sensitive to residual misalignment, where small 2D deviations can lead to larger spatial errors after backprojection. These quantitative results demonstrate the feasibility of using 3DGS-based coarse-to-fine refinement for global BIM to as-built alignment, as evidenced by the consistent reduction in both 2D and 3D errors and the small gap to the manual baseline.

Qualitative results are shown in Figure 2, where aligned BIMs are back-projected into input images using the estimated camera poses. For visualization, we show the qualitative results of the paired trial that achieved the lowest mean reprojection error after fine alignment. A clear coarse-to-fine improvement is observed, and the fine alignment produces projections that closely approach the manual baseline. These visualizations further show that the proposed framework produces visually consistent projections across viewpoints, indicating the effectiveness of 3DGS for global BIM to as-built alignment.

3.6 Ablation study

We conducted ablation studies to analyze the impact of key design parameters in our method and report the average performance over five runs for each experiment. In this section, only 2D reprojection error in pixel units is reported.

Table 2. Ablation study on the number of views K used to compute the photometric loss during the coarse alignment stage, evaluated using 2D reprojection error in pixel units only. Results for $K = 5$ are reported as N/A due to training failure.

K	$\text{RMSE}_{\text{mean}} \downarrow$	$\text{RMSE}_{\text{med}} \downarrow$	$\text{RMSE}_{\text{p95}} \downarrow$
5	N/A	N/A	N/A
10	30.51	28.44	55.71
20	**21.22**	**18.97**	**39.62**

Effect of the number of views in coarse alignment. Table 2 evaluates the impact of the number of views K used to compute the photometric loss during the coarse alignment stage, evaluated using 2D reprojection error in pixel units. When $K = 5$, the combination of large initial misalignment and insufficient multi-view constraints leads to 3DGS training failure. This setting is therefore reported as N/A. Increasing K improves both 3DGS training stability and alignment accuracy, as more views provide more robust multi-view photometric constraints. In particular, K is evaluated up to 20 due to GPU memory constraints, with $K = 20$ achieving the lowest errors across all metrics.

Table 3. Ablation study on the mask type used in the fine alignment stage, evaluated using 2D reprojection error in pixel units only. $\mathcal{M}^1$: BIM-projected masks; $\mathcal{M}^3$: refined masks. Fine alignment is initialized from the best coarse-alignment run with $K = 20$.

Masks	$\text{RMSE}_{\text{mean}} \downarrow$	$\text{RMSE}_{\text{med}} \downarrow$	$\text{RMSE}_{\text{p95}} \downarrow$
$\mathcal{M}^1$	13.87	13.50	25.68
$\mathcal{M}^3$	**6.68**	**6.21**	**14.27**

Effect of mask quality in fine alignment. Among five coarse-alignment runs with $K = 20$, the run with the lowest $\text{RMSE}_{\text{mean}}$ is used to initialize the fine alignment in this ablation study. Table 3 studies the impact of different mask types in the fine alignment stage using 2D reprojection error in pixel units. Using BIM-projected masks $\mathcal{M}^1$ yields limited improvement due to residual misalignment. The proposed refined masks $\mathcal{M}^3$, obtained by further refining $\mathcal{M}^1$ with instance segmentation masks and semantic segmentation masks, improve performance further and achieve the lowest mean, median, and tail errors. These results highlight the importance of accurate masks for fine alignment.

4 Limitations

The proposed framework has several limitations. First, the coarse-to-fine refinement depends on a sufficiently accurate initial alignment, particularly in orientation. The initialization may become unreliable when DBSCAN extracts only a partial building point cloud or when the target building exhibits strong geometric symmetry. Future work may improve this stage using unmanned aerial vehicle (UAV) GPS priors or semantic structural cues. Second, the refinement stage assumes relatively clean and visible building surfaces. Scaffolding, safety nets, and other temporary structures may introduce mismatches between the images and the BIM geometry, which can affect the optimization. Third, the proposed framework assumes as input a BIM that is already consistent with the observed construction state. Thus, the automation applies to the alignment process under this input condition, and extending the method to operate directly on complete BIMs remains future work. Finally, the current evaluation is limited to a single construction site, a specific construction stage, and UAV imagery. Accordingly, the robustness and transferability of the framework across different construction completion levels, capture strategies, and site layouts remain to be evaluated.

5 Conclusion

This study investigated the feasibility of using 3DGS for automatic global BIM to as-built alignment from multi-view imagery, given a BIM that is consistent with the observed construction state. The proposed *BIM-Align* framework combines classical geometric initialization with coarse-to-fine 3DGS refinement under the supervision of refined building masks. Experimental results show that our method achieves satisfactory alignment and approaches a manual baseline in a fully automatic manner, indicating the effectiveness of 3DGS for global BIM to as-built alignment. Further validation under more diverse and challenging construction scenarios is needed in future work.

6 Acknowledgment

This work was financially supported by the VLAIO COOCK-project 'AIDA' (HBC.2024.0591), a collaboration between imec, Buildwise, and Embuild. We gratefully acknowledge all project partners for their valuable contributions to technical discussions and data collection. We also thank the ibens project team of the Hoboken site for granting access to the construction project and for their practical support during data acquisition. In addition, this research received partial funding from the Flemish Government under the 'Onderzoeksprogramma Artificiele Intelligentie (AI) Vlaanderen' program.

References

[1] Mani Golparvar-Fard, Feniosky Peña-Mora, and Silvio Savarese. D^{4}AR–A 4-dimensional augmented reality model for automating construction progress monitoring data collection, processing and communication. *Electronic Journal of Information Technology in Construction*, 14(13):129–153, 2009.

[2] Muhammad Sami Ur Rehman, Muhammad Tariq Shafiq, and Fahim Ullah. Automated computer vision-based construction progress monitoring: a systematic review. *Buildings*, 12(7):1037, 2022. doi:10.3390/buildings12071037.

[3] Guangan Chen, Michiel Vlaminck, Wilfried Philips, and Hiep Luong. Multi-view 3D reconstruction for construction site monitoring. In *19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pages 469–476. SCITEPRESS, 2024. doi:10.5220/0012308300003660.

[4] Alexander Braun, Sebastian Tuttas, Uwe Stilla, and André Borrmann. BIM-based progress monitoring. In *Building Information Modeling: Technology Foundations and Industry Practice*, pages 463–476. Springer, 2018. doi:10.1007/978-3-319-92862-3_28.

[5] Miguel Arturo Vega Torres, Anna Ribic, Borja García de Soto, and André Borrmann. BIMCaP: BIM-based AI-supported LiDAR-camera pose refinement. *arXiv preprint arXiv:2412.03434*, 2024. doi:10.48550/arXiv.2412.03434.

[6] Slávek Zbirovskỳ and Václav Nežerka. Open-source automatic pipeline for efficient conversion of large-scale point clouds to IFC format. *Automation in Construction*, 177:106303, 2025. doi:10.1016/j.autcon.2025.106303.

[7] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: a tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. doi:10.1561/0600000052.

[8] Mani Golparvar-Fard, Feniosky Peña-Mora, Carlos A Arboleda, and SangHyun Lee. Visualization of construction progress monitoring with 4D simulation model overlaid on time-lapsed photographs. *Journal of computing in civil engineering*, 23(6):391–404, 2009. doi:10.1061/(ASCE)0887-3801(2009)23:6(391).

[9] Mani Golparvar-Fard, Feniosky Peña-Mora, and Silvio Savarese. Integrated sequential as-built and as-planned representation with D^{4}AR tools in support of decision-making tasks in the AEC/FM industry. *Journal of Construction Engineering and Management*, 137(12):1099–1116, 2011. doi:10.1061/(ASCE)CO.1943-7862.0000371.

[10] Mani Golparvar-Fard, Feniosky Pena-Mora, and Silvio Savarese. Monitoring changes of 3D building elements from unordered photo collections. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 249–256. IEEE, 2011. doi:10.1109/ICCVW.2011.6130250.

[11] Mani Golparvar-Fard, Feniosky Pena-Mora, and Silvio Savarese. Automated progress monitoring using unordered daily construction photographs and IFC-based building information models. *Journal of Computing in Civil Engineering*, 29(1):04014025, 2015. doi:10.1061/(ASCE)CP.1943-5487.0000205.

[12] Changchang Wu. VisualSFM: A Visual Structure from Motion System. `http://ccwu.me/vsfm/index.html`, 2011. Software, accessed 2025-12-24.

[13] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. doi:10.1109/CVPR.2016.445.

[14] Agisoft LLC. Agisoft photogrammetry software. `https://www.agisoft.com`, 2025. Commercial software, accessed 2025-12-24.

[15] Hadi Mahami, Farnad Nasirzadeh, Ali Hosseininaveh Ahmadabadian, and Saeid Nahavandi. Automated progress controlling and monitoring using daily site images and building information modelling. *Buildings*, 9(3):70, 2019. doi:10.3390/buildings9030070.

[16] Hany Omar, Lamine Mahdjoubi, and Gamal Kheder. Towards an automated photogrammetry-based approach for monitoring and controlling construction site activities. *Computers in Industry*, 98:172–182, 2018. doi:10.1016/j.compind.2018.03.012.

[17] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[18] Autodesk, Inc. Autodesk revit. `https://www.autodesk.com/products/revit/`, 2025. BIM authoring software, accessed 2025-12-24.

[19] Sebastian Tuttas, Alexander Braun, André Borrmann, and Uwe Stilla. Acquisition and consecutive registration of photogrammetric point clouds for construction progress monitoring using a 4D BIM. *PFG–journal of photogrammetry, remote sensing and geoinformation science*, 85(1):3–15, 2017. doi:10.1007/s41064-016-0002-z.

[20] Mustafa K Masood, Antti Aikala, Olli Seppänen, and Vishal Singh. Multi-building extraction and alignment for as-built point clouds: a case study with crane cameras. *Frontiers in Built Environment*, 6:581295, 2020. doi:10.3389/fbuil.2020.581295.

[21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139–1, 2023. doi:10.1145/3592433.

[22] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010. doi:10.1002/wics.101.

[23] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything: unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. doi:10.1109/CVPR52733.2024.00987.

[24] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, et al. SAM 3: segment anything with concepts. 2025. doi:10.48550/arXiv.2511.16719.

[25] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi:10.1109/TIP.2003.819861.

[26] Berthold KP Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the optical society of America A*, 4(4):629–642, 1987. doi:10.1364/JOSAA.4.000629.