Construction Robotics (2026) 10:10
https://doi.org/10.1007/s41693-026-00183-3

RESEARCH

Fusion of fiducial marker and point cloud data for reliable multi-camera tracking in robotic assembly of construction elements

Sam Wilcock[1] · **Ornella Iuorio**[1]

Received: 24 October 2025 / Revised: 23 February 2026 / Accepted: 27 March 2026

Abstract

Accurate state estimation of building components across large workspaces is crucial for automated construction assembly, yet construction environments are inherently indeterminate due to material variability, structural deflection and changing site conditions. This paper addresses the resulting tension between precision and adaptability by presenting a dual-camera and point-cloud fusion framework for robotic assembly of timber panels. A tripod-mounted global RGB-D camera provides continuous workspace coverage, while an eye-in-hand camera supplies high-precision local measurements when components enter its field of view. AprilTag fiducial markers on panels and the robot base support continuous self-calibration of the global camera relative to the robot, and statistical fusion combines pose estimates through distance-based confidence weighting, inverse-variance weighting and temporal outlier rejection. These fused poses are optionally refined using point cloud registration via a coarse-to-fine Iterative Closest Point (ICP) scheme. The system is embedded in a ROS-based architecture that links real-time sensing to a parametric design environment and an impedance-controlled Kuka iiwa manipulator. Experiments on the insertion of interlocking timber panels show that dual-camera fusion substantially improves consistency over global-only sensing and enables successful assembly where manual calibration or single-sensor feedback fails. The results demonstrate how balancing precision sensing with tolerance to uncertainty can support robust, adaptive robotic construction workflows.

Keywords Fiducial marker · Point cloud · Digital twin · Sensor fusion

1 Introduction

Reliable state estimation of building components is a cornerstone of automated construction assembly. Accurate knowledge of component poses is critical if autonomous tasks are to be performed precisely. While dual-camera fusion and multi-sensor setups have been extensively studied in general robotics, their specific application to the automated assembly of architectural elements - such as timber panels - remains underexplored. Construction environments present unique intricacies, including frequent occlusions, dynamic lighting, and the need to manipulate large, fragile components with high dimensional variability (Shen et al. 2024). In such settings, implementations relying on single sensors often struggle to maintain both the broad workspace coverage required for tracking multiple elements and the high local precision necessary for tight-tolerance insertion tasks.

Consequently, there is a significant research gap in adapting and integrating these multi-camera fusion techniques specifically for the demands of robotic construction automation. By combining multiple image sources from a global workspace camera and a hand-in-eye camera mounted on the manipulator, it is possible to leverage each sensor within its optimum working range. This multi-camera approach ensures continuous monitoring of the assembly scene while providing the necessary precision when the robot interacts with a component. Using AprilTag fiducial markers (Olson 2011), materials can be tagged to act as reliable reference points for pose estimation, whilst similar markers on the robot base enable continuous self-calibration of the global camera relative to the robot arm. Furthermore, by incorporating an RGB-D camera to capture point cloud data during assembly operations, there is additional potential to augment pose estimates using point cloud registration techniques. This integration develops capabilities for both online motion planning

✉ Sam Wilcock
sam.wilcock@polimi.it

Ornella Iuorio
ornella.iuorio@polimi.it

[1] Department of Architecture, Built Environment, and Construction Engineering, Politecnico di Milano, Milan, Italy

Published online: 09 May 2026

in construction environments and the generation of live digital twins for process feedback (Abdelrahman et al. 2025). By addressing the specific challenges of sensor calibration, data fusion, and their direct application to timber panel assembly, this research contributes to advancing the state-of-the-art in robotic construction automation.

2 Related work

2.1 Fiducial markers for camera feedback

Fiducial markers, such as AprilTags (Olson 2011) or ArUco (Garrido-Jurado et al. 2014) are widely used for pose estimation in robotics and construction. High-contrast patterns allow cameras to quickly identify individual components and compute their position and orientation when the cameras are calibrated (Kunic et al. 2024; Kalaitzakis et al. 2021). This rapid recognition enables online monitoring and feedback for robotic manipulators, supporting scene understanding, localisation, and alignment checks of structures in dynamic, cluttered construction environments (Iturralde et al. 2023; Song et al. 2021). Markers can also be combined with other sensors, such as Inertial Measurement Units, to further improve state estimates (Kayhani et al. 2022).

Such markers have been previously applied for the tracking of excavator vehicles on construction sites, demonstrating high accuracy in pose estimation for multiple machine components optically (Lundeen et al. 2016), or for the identification and pose estimation of connectors on a building facade for measurement of the building (Iturralde et al. 2023). However, the use of a single camera source was noted to limit the amount of tags that could be seen and required constant manual recalibration of the camera pose.

2.2 Multi-camera systems for state measurement

Using multiple markers and cameras improves pose estimation robustness and accuracy. Arranging markers in known configurations increases redundancy and reduces reprojection errors via the perspective-n-point problem (Yoon et al. 2006; Malyuta 2018). If some markers are occluded or partially detected, their poses can be inferred from visible markers, and ambiguities in orientation due to lighting or corner uncertainty can be mitigated by combining observations from multiple viewpoints (Guan et al. 2024; Barros et al. 2025). Multi-camera setups allow each camera to be localised relative to shared tags, enabling broader workspace coverage while minimising manual calibration (Muñoz Salinas et al. 2018).

In robotic manipulation, multi-camera systems often combine a fixed global camera with a mobile eye-in-hand camera to leverage their respective strengths (Popescu et al. 2020). Global cameras provide continuous, broad monitoring of the workspace, essential for tracking multiple components and avoiding collisions, but suffer from reduced resolution and accuracy at a distance. Conversely, eye-in-hand cameras offer high-precision, close-range measurements crucial for delicate tasks like grasping and insertion, but have a severely limited field of view (Zarei et al. 2025). Fusing these data streams allows a system to maintain global context while opportunistically exploiting local precision when the end-effector approaches a target. Together, these strategies allow reliable pose estimation of components even in cluttered, dynamic construction environments.

2.3 Point cloud registration for pose refinement

While fiducial markers provide structured, high-speed pose estimates, they require unobstructed, planar surfaces and do not capture the full 3D geometry of the object. In timber assembly, where components may warp or deviate from their design dimensions, relying solely on markers can lead to alignment errors during insertion. Point cloud data, captured via depth-sensing cameras, offers dense 3D information of the environment, enabling object detection, segmentation, and scene understanding (Li et al. 2021; Kang and Kim 2024). These techniques, whether classical (e.g., surface matching) or learning-based, support as-built model generation and comparison to design data (Tang et al. 2010; Settimi et al. 2025).

When object geometry is known, point clouds can be registered to CAD models to refine pose estimates. Registration techniques, such as Iterative Closest Point (ICP) align scans by minimising distances between points (Rusinkiewicz and Levoy 2001; Ma et al. 2020). However, ICP is prone to local minima and requires a strong initial alignment. In this work, fiducial markers provide this robust initial estimate, allowing ICP to act as a fine-tuning step. This hybrid approach leverages the speed and global tracking of AprilTags with the geometric verification of point cloud registration, improving robustness in cases of partial occlusion or complex assembly setups (Barone et al. 2012; Yang and Allen 1998). By integrating these methods, the system can verify the geometric integrity of the assembled structure and correct for the dimensional variability inherent in timber construction.

2.4 Research gap

Despite the maturity of multi-camera systems and sensor fusion in general robotics, their direct application to the automated assembly of architectural panels remains underexplored. Existing studies often focus on either broad site monitoring, such as tracking excavators (Lundeen et al. 2016), or isolated, small-scale manipulation tasks. However, the robotic assembly of timber structures requires a system

capable of both tracking multiple large components across a wide workspace and providing sub-centimeter precision during the physical insertion of joints. Current state estimation methods in construction often rely on single-frame observations or single-sensor setups (Iturralde et al. 2023), which fail to resolve the fundamental coverage-precision trade-off. Therefore, a significant research gap exists in developing an integrated, multi-sensor framework that combines global monitoring, local precision, and point cloud verification specifically for the dynamic and cluttered environments typical of construction automation.

2.5 Contribution

Whilst existing work demonstrates the use of markers, part localisation, and multiple cameras for robot localisation in construction, there is little research combining all three aspects for robot localisation and object manipulation within a single integrated system specifically tailored for the assembly of architectural panels. Furthermore, current state estimation methods often rely solely on single-frame observations without leveraging the complementary strengths of global and local sensors during the physical assembly process.

Our primary contribution addresses the coverage-precision trade-off in construction robotics through a hybrid sensing approach applied to timber panel assembly. We develop a dual-camera system that maintains continuous workspace monitoring via a global camera while opportunistically leveraging high-precision measurements from a hand-mounted camera during manipulation. Key innovations include: (1) statistical fusion algorithms that weight pose estimates based on camera-specific performance characteristics, (2) continuous self-calibration using robot base markers to maintain global camera registration, and (3) validation showing improved tracking consistency over global-only approaches while preserving full workspace coverage during a physical assembly task. This hybrid strategy enables robust pose estimation across large construction workspaces without sacrificing the precision benefits of close-range sensing when available, directly facilitating the automated construction of complex timber structures.

The core objectives of this research are to:

1. Demonstrate continuous self-calibration of a global camera system by utilising fixed base markers on the robot, thereby improving the long-term localisation of structural elements relative to the robot (Sect. 3.2).
2. Design and implement an algorithm for statistically combining pose estimates from both end-effector and global cameras, applying weights based on calibrated camera performance and incorporating outlier filtering (Sect. 3.4)
3. Investigate the potential for refining state estimates through the incorporation of point cloud data using iterative closest point (ICP) registration (Sect. 3.6).
4. Establish the efficacy of laser-engraved AprilTags in timber as a robust alternative to conventional paper markers (Sect. 4.1).
5. Apply the developed vision system to an initial workflow for the automatic insertion assembly of digitally fabricated timber panels and evaluate the resulting improvements in physical robotic assembly (Sect. 4.5).

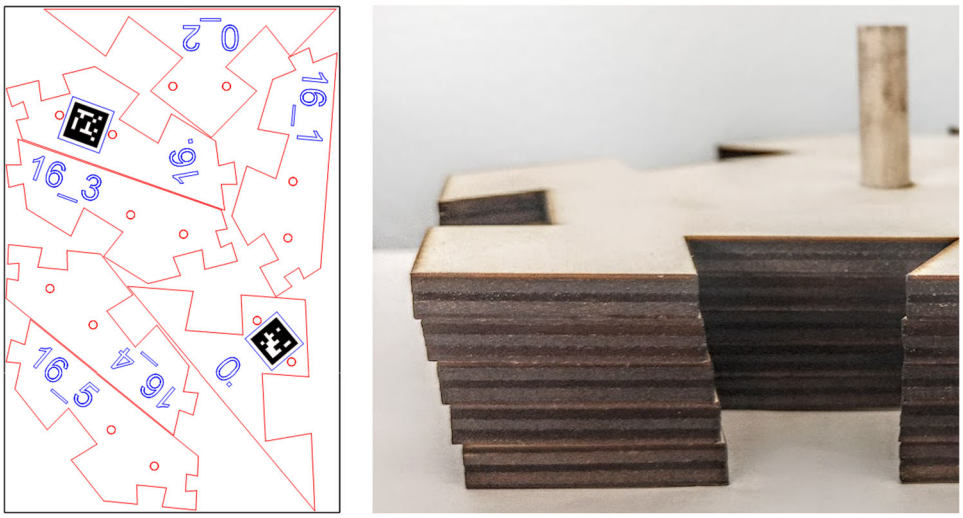
3 Methodology

3.1 System setup

A shell panel structure was selected as a test-bed for manipulation assembly with joints which will self-support during cantilevering. Using such a system allows for testing of the sensing system which is developed, particularly due to the inherent displacement in the structural system from clearances which introduces uncertainties compared to the design (Rogeau et al. 2022). The shell geometry is form-found to be near-funicular and is realized as planar panels with integral dovetail-style interlocking joints to provide positional constraint during assembly (Wilcock et al. 2024; Robeller and Weinand 2015). Panels are built from stacked 3 mm poplar sheets (nominal total thickness 18 mm). A subset of the designed structure that fits within a Kuka iiwa7 reach volume was selected for manufacture and testing; panel dimensions and joint geometry were exported from CAD for fabrication and for later use in pose-estimation and registration routines.

Panel outlines were discretized into stacked contours (see Fig. 1) and laser cut, with inbuilt laser engraved AprilTags on the surface of the panels. After assembly, the 3 mm sheet panels were glued and dowelled to form the test-bed used for sensing and robotic experiments.

Two cameras were used: a low-cost webcam mounted eye-in-hand on the manipulator gripper, and an Orbbec Astra RGB-D depth camera on a tripod providing a more global view of the scene (Fig. 2). A bundle of four 36h11 AprilTags (16 cm) was fixed on the manipulator table with known relative poses. Camera intrinsics and distortion were estimated with standard checkerboard procedures; extrinsics were computed via observed tag poses and robot kinematics to relate camera, tag and robot frames. Because the global camera pose is continuously estimated from these fixed base markers, the tripod can be repositioned during an experiment and automatically re-registered to the robot and workpiece without manual recalibration, effectively enlarging the robot's functional workspace beyond its static reach. Full extrinsic calibration details are found in the Appendix A.



(a) (b)

Fig. 1 Slicing discretization of the panels. **a** The cutting layout, packed using OpenNest (Vestartas 2021). **b** The stacked and glued result. Reducing the thickness of the stacked contour layers results in a more accurate approximation of the intended curved surface

The proposed system architecture employs a hybrid, distributed approach centred around a Robot Operating System (ROS Kinetic) master node running on an Ubuntu Linux desktop, where ROS is used over the newer ROS2 due to the availability and maturity of the iiwa_stack control interface (Hennersperger et al. 2017) This central core acts as the primary hub for high-level process control, state management, and direct communication with the Kuka iiwa7 manipulator. Rather than relying on a monolithic structure, the architecture strategically distributes specialized workloads across environments to leverage the specific strengths of each platform. Parametric design integration, geometric visualization and structural "as-designed" transformations/pose generation are offloaded to an external Windows machine running Grasshopper. This external computer-aided design (CAD) and planning environment communicates with the centralized ROS core over a local network via roslibpy websocket interface (Casas 2019), avoiding processing bottlenecks and so ensuring that robust real-time robotic control remains uninterrupted by computationally heavy geometric functions and visualisation.

Within the internal ROS ecosystem, the data processing pipeline is further compartmentalized into specialized nodes to optimise computational performance. To prevent bottlenecks and latency in the real-time control loop, computationally intensive tasks are dealt with in dedicated compiled C++ ROS nodes, while instead less intensive tasks are in parts controlled through Python scripts (for example for the initial conversion of camera data to image topics). Sensor data processing is managed by bespoke services written in C++. These services variously handle robot control and motion planning (Hennersperger et al. 2017; Coleman et al. 2014), marker detection and processing (using AprilTag Olson 2011 and OpenCV Bradski 2000) point cloud capture (primarily utilising Open3D Zhou et al. 2018) and sensor data fusion, subsequently publishing the structured sensor data as ROS topics. The processed transformation frames from the sensor fusion process can then be acted on by high-level yet less time-critical control signals from the external machine, e.g. the next panel to pick and the next calculated insertion pose, calculated geometrically to adapt the as-designed plan to the as-built real environment. By clearly delineating the external CAD environment, which can consume available topic data as they are produced from the internal processing and robot control pipelines, the architecture achieves an efficient control framework for robotic structural assembly.

3.2 Sensing of panels for process feedback

Figure 3 describes the workflow of the proposed approach to combine AprilTag pose estimates from multiple sources, processing of data, and combination with point cloud data for generating refined state estimates for panel elements. On a legacy CPU Linux laptop, dual-camera AprilTag tracking runs (on a separate thread from camera image processing) at 25–35 Hz, and time-stamp synchronisation between cameras works within a 50 ms window to accomodate this in tag history windows; fusion and filtering add ¡20 ms latency thus dropping end-to-end tracking-to-pose publication to around 20–25 Hz; optional ICP using pointcloud data can significantly increase sampling by around 600 ms depending on downsampling and correspondence radius, thus acting as the bottleneck which could be improved on by offloading to dedicated hardware or upgrading to a GPU.

For accurate estimation of panel poses based on the location of their fiducial markers within the camera image, a number of calibration and configuration steps first had to be carried out. The global camera was calibrated at a distance of

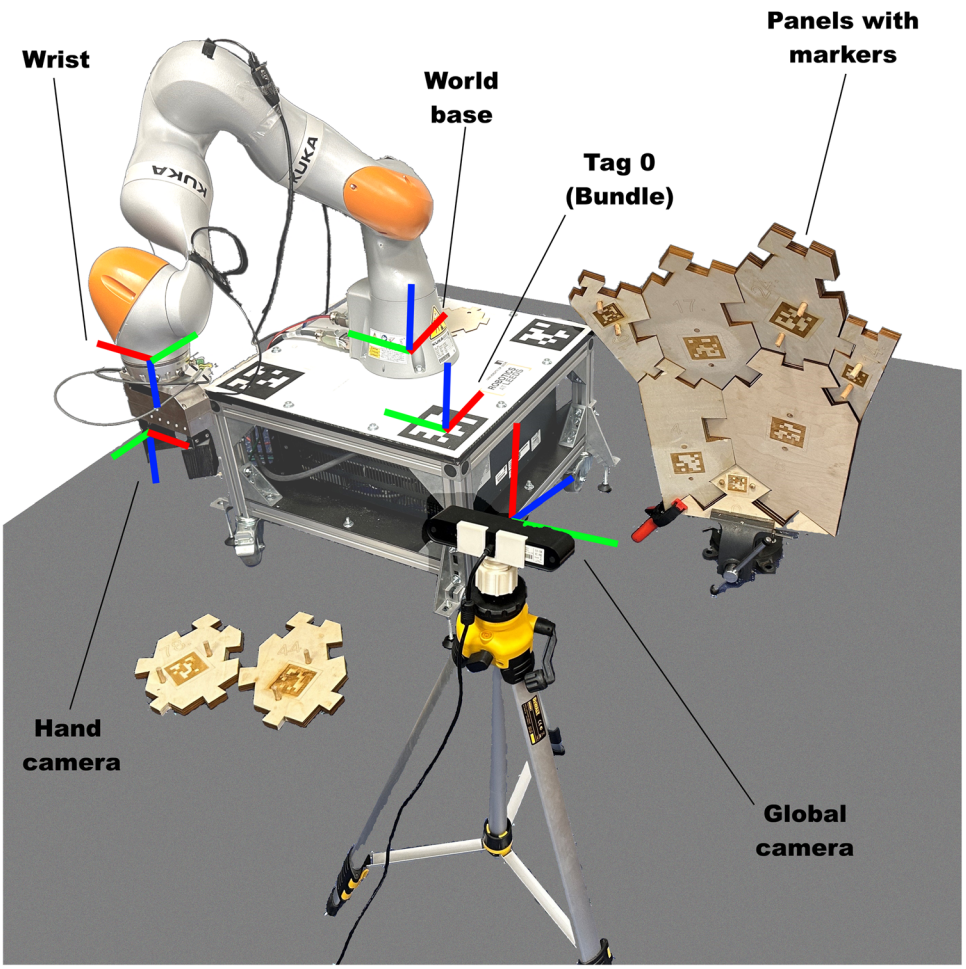


Fig. 2 Tag detection, global and hand cameras and reference frames. Orthogonal reference frames use convention of red/green/blue to represent *X*/*Y*/*Z* axes

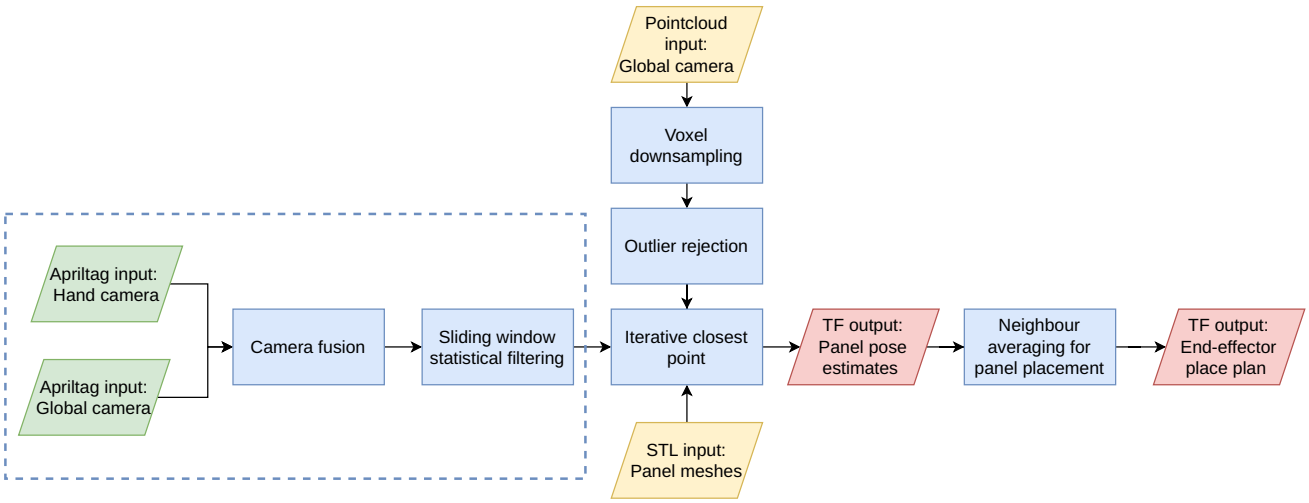


Fig. 3 A system diagram of the main components for panel pose estimation used. TF denotes homogenous transformation frames, i.e. poses. Boxed region is further elaborated in Fig. 4

3 m using an A3 checkerboard, in order to gain the camera's distortion and intrinsic matrices respectively for combating distortion and to translate from points in the camera space to points in 3D space. A smaller, high resolution "web-cam" camera was similarly calibrated using a smaller A4 checkerboard at a distance of ∼ 30 cm. The panel meshes, AprilTag sizes and orientation were exported to a JSON format file for easy reading, as were the relative locations of the paper bundle's markers. Based on earlier works (Malyuta 2017, 2018), the perspective-n-point (PNP) algorithm (Fischler and Bolles 1981) was utilised through OpenCV (Bradski 2000) to recognise the base point of bundles within the camera's field of view using all of the visible tags. This provides the benefit that the bundle can be recognised with partial occlusion, i.e. when some or even most of the bundle's markers are obstructed from the camera.

To measure the difference between a pose estimate and the ground truth values, it would be possible to simply take distance measurements from the base points of the panels. This does not however take into account the rotation of panels; for this, the average distance metric ADD-6d (Hinterstoisser et al. 2013) is used:

$$\text{ADD}^p = \frac{1}{M}\sum_{i=1}^{j} \min_{0<k<M} \left\| (\mathbf{R}\mathbf{x}_i + \mathbf{t}) - \left(\bar{\mathbf{R}}\mathbf{x}_k + \bar{\mathbf{t}}\right) \right\|, \tag{1}$$

where pose $p = \mathbf{R}|\mathbf{t}$ is the ground truth pose, made up of a rotation matrix $\mathbf{R}$ and a column vector of translations $\mathbf{t}$, and similarly $\bar{p} = \bar{\mathbf{R}}|\bar{\mathbf{t}}$ is the pose estimate. M is the number of points in the test set. The ADD-6d metric defines the average distance error between each pose point i on the panel mesh and its closest counterpart point k_{min}, for which the mesh geometry is populated with $10 \times 10 \times 10$ points. This allows the comparing of poses given by AprilTags both to expected poses using an already-assembled panel as a reference (where the as-designed transformations between panels can be used to provide expected poses), and to point cloud data which can be used to refine the knowledge of the assembly scene state.

3.3 Averaging homogenous transformations

Homogenous transformations can be decomposed into rigid transformation components, comprising translation and rotation, through the general form of the transformation matrix from frame i to frame j

$$\mathbf{T}_j^i = \begin{bmatrix} \mathbf{R}_{3\times3} & \mathbf{t}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{2}$$

where $\mathbf{R}$ is the rotation matrix, which can also be converted into quaternions using e.g. Shepperd's method (Shepperd 1978), and $\mathbf{t}$ is the translation vector. In order to take a (weighted) average of multiple transformations, the translation vectors and quaternions are dealt with separately. Averaging of translation vectors can be performed by taking the vector average of the x, y, z components for each neighbour translation vector $\mathbf{t}_i$ with weight w_i to give a new 3×1 translation vector

$$\mathbf{t}_{\text{ave}} = \frac{\sum_{i=1}^{n} \mathbf{t}_i w_i}{\sum_{i=1}^{n} w^i} \tag{3}$$

To get the average quaternion rotation whilst avoiding quaternion flip according to Markley et al. (2007), it is possible to minimise the squared Frobenius norm of the difference between rotation matrices. Hence, given a list of n quaternion vectors of form $q_i = q_{xi}\mathbf{i} + q_{yi}\mathbf{j} + q_{zi}\mathbf{k} + q_{wi}$ with corresponding weights w_i, a $4 \times n$ matrix containing them all as column vectors is constructed as follows:

$$\begin{aligned} \mathbf{Q}_{4\times n} &= [\mathbf{q}_1 w_1, \mathbf{q}_2 w_2, \ldots, \mathbf{q}_n w_n] \\ &= \begin{bmatrix} q_{x1}w_1 & q_{x2}w_2 & \ldots & q_{xn}w_n \\ q_{y1}w_1 & q_{y2}w_2 & \ldots & q_{yn}w_n \\ q_{z1}w_1 & q_{z2}w_2 & \ldots & q_{zn}w_n \\ q_{w1}w_1 & q_{w2}w_2 & \ldots & q_{wn}w_n \end{bmatrix} \end{aligned} \tag{4}$$

which is multiplied by its transpose to give $\mathbf{M}_{4\times4} = \mathbf{Q} \cdot \mathbf{Q}^T$. By the eigendecomposition of the vector M, its largest real eigenvalue and the corresponding eigenvector is found. Normalising this eigenvector gives an appropriate mean quaternion $\mathbf{q}_{\text{ave}}$. The average transformation can then be found by a simple combination of the two rigid transformations $\mathbf{t}_{\text{ave}}$ and $\mathbf{Q}_{\text{ave}}$, built by converting the quaternion back to a rotation matrix (Shoemake 1985), and to a transformation matrix using the form from (2).

3.4 Combining AprilTag estimates and filtering

The two data streams of AprilTag estimates from the separate cameras can give complimentary, yet slightly differing estimates of panel poses. Additionally, changing lighting conditions can cause uncertainty in pose estimates from the AprilTag software; in particular, orientation estimates can suffer from "flips" where the positions for corner points of markers are ambiguous based on a 2D image. In order to smooth and filter the AprilTag data, C++ method has been developed to discard statistical outliers and give a weighted average for position and orientation estimates.

The majority of the work on transformation frames is delegated to a class which stores a limited amount of historic transformation frame data for each panel relative to the world coordinates. When an AprilTag pose estimate is provided by either camera, the time is logged, and one of three cases can happen. Either both cameras have received data on a particular marker within a pertinent recent timeframe, in which

case a weighted average of the two cameras is pushed to the historic transformation stack using the previously described transformation averaging; or, there is only recent data from one of the two cameras, in which case that is pushed to the stack without averaging. Further processing is performed on the transformation frame data through statistical outlier rejection at a set frequency. Using a sliding window of the previous n transformations for a given panel's marker, in a First In, First Out (FIFO) queue data structure, the transformations are again split into rigid translation and rotation components, and the averages taken. The distance of each historic transformation's rigid component from their respective mean is found and stored. Translation distance is defined as the Euclidean norm of the vector difference between the component and its mean, and the quaternion distance is defined as the angle between the rotation attitude and the mean attitude. The standard deviation σ of both transformation sets is calculated as

$$\sigma = \sqrt{\frac{\Sigma(d(x_i, \mu))^2}{N}} \tag{5}$$

by definition, where x_i is the ith transformation component, μ is the average, $d(a, b)$ is the distance function for the transformation type (translation or rotation), and N is the number of samples in the sliding window. Using the standard deviation, sigma clipping (Steinwolf 2010) is applied: the transformation component distances are compared to standard deviation. If they are further than 1 standard deviation from the mean, they are discarded, meaning that under a Gaussian error distribution the central 68% of points are retained. The new mean combined transformation is reconstructed as averages from the inliers using Eqs. (2) to (4). The full workflow for the AprilTag fusion process, detailing the logic gates for camera selection and the subsequent FIFO buffering, is illustrated in Fig. 4.

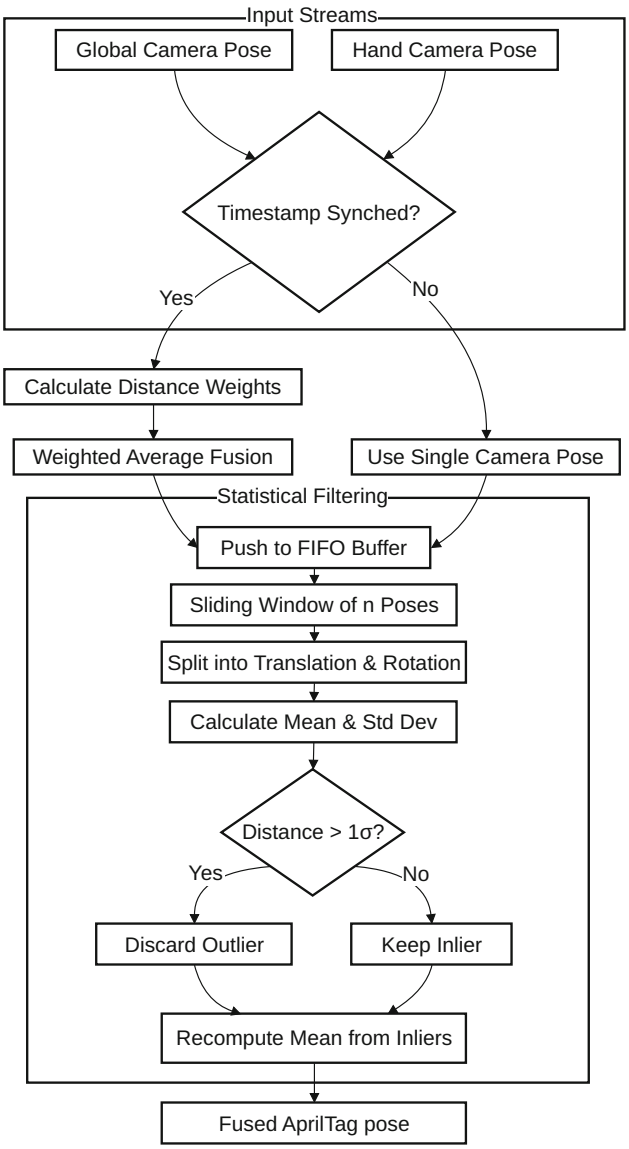


Fig. 4 Schematic data flow for the fusion and filtering of AprilTag poses. Incoming streams are synchronised, weighted by distance reliability, and buffered (with a separate FIFO buffer per tag/panel). A sliding window filter then removes statistical outliers before the final pose estimate is computed

3.5 Storage of homogenous transforms for and calculation of required poses

From the digital design model, the position and quaternion orientation transformations of each panel with respect to a designated base panel are determined. These transformations, along with relevant marker sizes and the rotation angle of each panel's AprilTag relative to its mesh reference frame, are stored in a structured format for later use.

Given the pose of any reference panel in the world frame, the required pose of another panel in the world frame can be estimated (as a homogeneous transformation) by

$$\mathbf{T}i^{\text{world}} = \mathbf{T}j^{\text{world}} \cdot \left(\mathbf{T}j^{0}\right)^{-1} \cdot \mathbf{T}i^{0}, \tag{6}$$

where i is the index of the panel to be located, j is the index of the reference panel, and 0 denotes the base panel to which all stored transforms are relative. From this relationship, it follows that, provided an estimated pose of a reference panel j is known, and the as-designed transforms between the base panel and both the reference panel and locating panel i are available, the as-designed pose of the new panel can be reconstructed.

In cases where several neighbouring panels have already been assembled within the structure, multiple reference panels may be available. To improve insertion pose accuracy, the average of the transformations estimated from each available neighbour can be taken.

3.6 Refining AprilTag state estimates with point cloud data

3.6.1 Preprocessing point cloud data

When receiving point cloud data from the Orbbec camera, the initial input is unsuitable for use and requires some preprocessing steps. The points of the cloud are often irregularly spaced, giving regions different densities. This can be an issue for the use of point cloud registration techniques later as it makes 1 to 1 point matching difficult. Additionally, noise appears within the point clouds which needs to be removed. This preprocessing stage is integrated as a blocking call within the ROS service structure; when a refinement request is triggered, the latest point cloud frame is frozen and processed. While statistical outlier removal, voxel grid downsampling and point cloud registration are computationally intensive (later shown to be > 600 ms), the operation is performed only during static verification phases (pre-grasp or pre-insertion), thus avoiding latency in the real-time control loop.

To clean this data, points beyond a certain distance away from the camera are first discarded, to focus only on the region of interest. A statistical outlier discarding algorithm is used to remove the spurious noise, and voxel downsampling is applied to make the cloud more regular. This also has the beneficial effect of reducing the number of points in the cloud, making registration faster.

3.6.2 Point cloud registration with iterative closest point

The generalised iterative closest point algorithm was applied to get a better estimate of the panel pose based on the initial estimate from the AprilTag data. The iterative closest point (ICP) algorithm describes a popular technique of registering a geometry to a matching model geometry, and finding the correct transformation to best fit them together. It works by minimizing the distance between corresponding points in two sets of points, which are matched somehow. A baseline ICP method is as follows Rusinkiewicz and Levoy (2001):

1. Sample a number of random points on the target geometry.
2. Match the points to the closest corresponding point on the model (source) geometry.
3. Minimize the least squares error between the sets of corresponding points by a rigid transformation.
4. Reiterate process until error is below a required value.

While this original technique works well in many cases, the optimisation problem is generally non-convex and thus it can fall into local minima. To solve this, the two meshes should first be roughly well aligned, which would make ICP appear to be less useful. This initial global registration step can be done in a number of different ways, including Random Sample Consensus (RANSAC) (Raguram et al. 2008) and feature based methods (Zhou et al. 2016). In this case, however, the AprilTag pose estimates provide this initial rough registration.

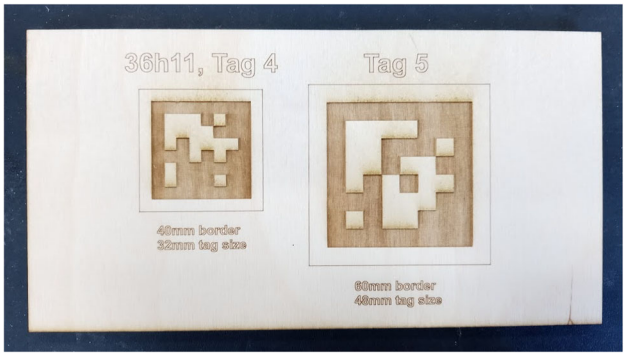


Fig. 5 The AprilTag test board laser cut in wood

Alternative forms of ICP exist, notably the point-to-plane ICP, the point-to-projection and generalised ICP (Pomerleau et al. 2013), which are modifications of the method by which the corresponding point is found.

Here, the point-to-plane implementation of generalized ICP provided in Open3D is used. The function accepts parameters, including maximum correspondence distance (i.e. the maximum distance allowed to recognise that two points may be associated). A custom service was built to handle ICP requests, calling ICP either once or twice. For the single call, input parameters include the correspondence distance, the voxel radius (for voxel downsampling of the point cloud data) and a maximum number of iterations for registration. To allow better registration avoiding local minima, a coarse-to-fine scheme is utilised where the maximum distance in which points are searched is incrementally reduced. Full simulation code is provided for viewing online and running without a robot, using Gazebo to simulate a noisy point cloud source (Wilcock 2025).

4 Experiments and validation

4.1 AprilTags in wood

To test the viability of using AprilTags engraved into wood, a trial laser cut sheet was made with two different marker sizes, one 40 mm across and one 60 mm (Fig. 5). A copy of the test sheet was printed with black ink on white paper, and the markers were moved away from a webcam on a tripod progressively, recording pose estimates. The furthest distance

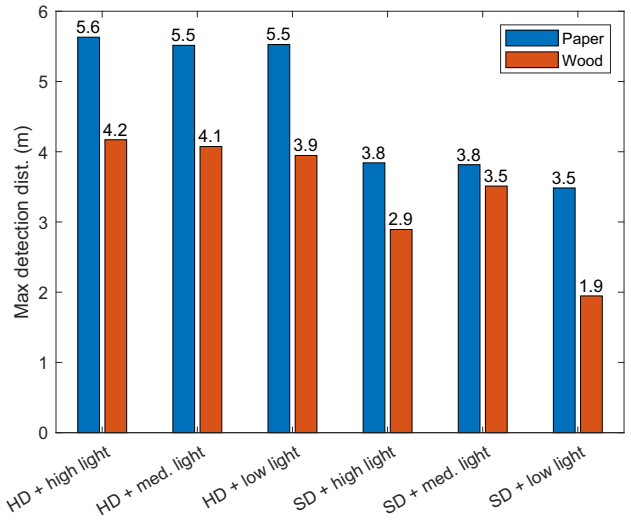


Fig. 6 Graph of maximum detection distances for the smaller 40 mm test marker

from the camera registered was taken to be a maximum detection distance. Two aspects of using AprilTags which were also desirable to explore were the effects of light levels and camera resolution on detection distance. Lighting levels are often variable in a changing environment, such as one with a moving robot arm. Additionally, the cheap cameras available had either a high-definition (1920 × 1080 pixels) mode or standard definition (640 × 480 pixels) mode. While the HD mode will clearly be better at detecting at greater distances due to a higher density of pixels, it is also significantly slower to perform tag detection on particularly when using multiple cameras, and so it was hoped that the pose detection would still work with the SD mode. As is seen in Fig. 6, the use of laser cut wood does reduce detection distance by at least 1 m in every case. Interestingly however, the markers are still detectable nearly 2 m away in standard definition mode, well within the reach of the arm and so suitable for use here. With the smaller 40 mm markers, 3.62 m detection was observed in the best case, and 1.41 m in the worst (not shown in bar graph).

4.2 Combining dual-camera AprilTag data to improve pose estimates

4.2.1 Dual streams

A tag of 10 cm size was placed at a known pose on the ground, halfway along the robot base between the two base tags #0 and #1 on the ground. The global camera on the tripod was set up at a height of 1.5 m, with an adjustable angle below horizontal pointing directly at the marker centre, so that the marker was in the direct centre of the camera frame for each measurement. The gripper with its camera was additionally positioned exactly vertically over the centre of the marker with the camera facing down.

In order to test the accuracy of the pose measurements, the two cameras were then separately moved towards and away from the marker. The gripper camera position was measured through the kinematics of the robot arm, based on the joint angles; the position of the camera was measured from the base point under the tripod centre. For the gripper camera, the Z-distance measured in the AprilTag software between the marker and the camera was suitable, whilst for the global camera, the horizontal distance was calculated from the Z-distance (straight from the camera to the marker) using trigonometry, based on the height of the camera, and the height of the robot base. Both cameras were moved in 10 cm intervals. The results of this test are shown in Fig. 7a.

Since the distance that should be found is known, the data was also adjusted by subtracting the known (measured) data from the unknown (calculated from AprilTags) to better display the position estimation error (Fig. 7b). As can be seen from the graph, the data for both cameras is reasonably accurate. For the hand camera difference data, the mean bias of the data is + 14 mm, while for the global camera difference data there is a calculated bias of + 69 mm. Outliers more than 1 standard deviation away from the mean are highlighted.

As can be seen, the global camera couldn't get readings below 50 cm, due to the inclination angle being unachievable with the tripod. Readings for the low-resolution hand camera stopped after ∼ 1.6 m, primarily because the gripper would not move directly vertically any higher whilst facing down.

Both cameras exhibited a positive bias in the distance measurements. This could be due to a miscalculation in the size of the marker—if the size of the marker has been input as larger than it actually is, the estimates of its position will be further away than the actual position. On measurement, however, the AprilTag size is correct. The bias error is much larger in the case of the global camera than for the hand camera, and so it would appear that there is a minor experimental setup issue for the tripod camera, which would be explained by a backwards tilt of 2.61° on the tripod, since distance measurements were taken from the bottom of the tripod and such a tilt is practically unnoticeable to the eye. Additionally, this bias error would be corrected in relation to the robot for picking if the position of the camera with relation to the robot was better known through observation of the base markers on the robot.

The hand camera exhibits its best estimation accuracy at small values, while the global camera gives best estimation between 1.5 and 2.5 m. This aligns with the expectations from the calibration of the respective cameras, since the hand camera was calibrated at a range of 0.3 m and the global camera was calibrated from 3 m. The outlier errors in the global camera coincided with the marker being more difficult to register in the lower resolution image, particularly with the chang-

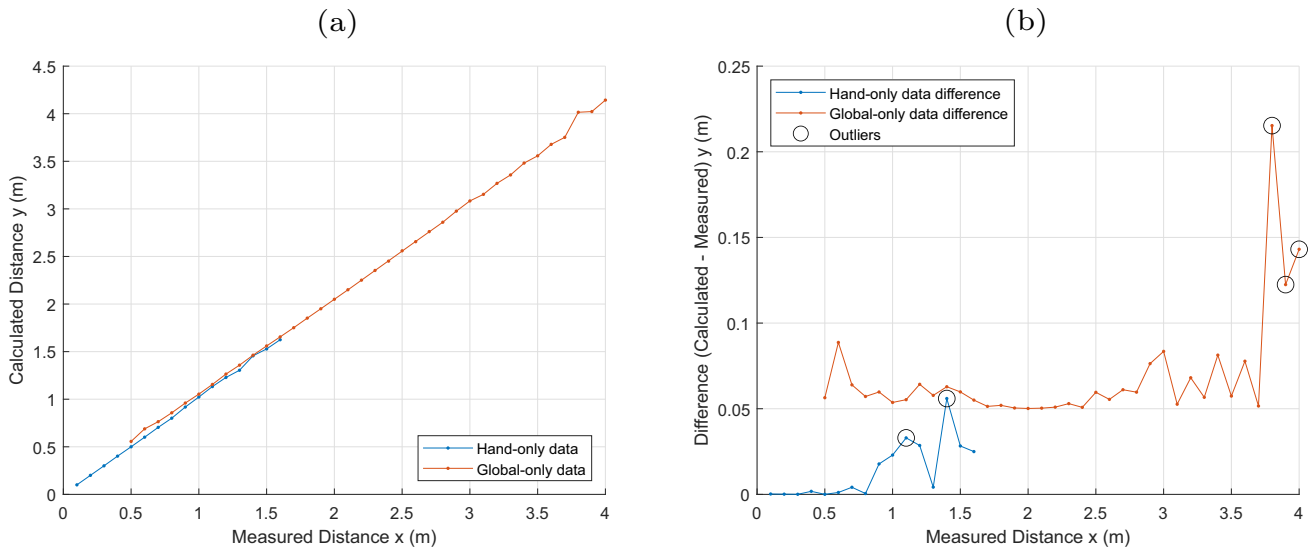


Fig. 7 Calculated distance of the markers from each camera, with the camera at known distances. **a** Plot of calculated distance (from AprilTag software) vs measured distance. **b** Difference between calculated and measured distance

ing light levels in the ground-level lab space with windows facing a busy road. For the calibration however, outside of the outliers, the standard deviations of 17 mm and 32 mm, respectively, for the two cameras represent reasonable accuracy under imperfect experimental conditions.

To make best use of both cameras, the state estimator for marker poses was set to apply a weighted average to the respective camera data. From observation of the data in Fig. 7, the hand camera is most accurate between 10 and 80 cm, whilst the global camera is most accurate between 1 and 3 m. Algorithm 1 describes how, at each sampling timestep and for each marker, pose data can be combined from the two cameras.

Poses are input as transforms (T) which can be split into translation and rotation components. Note that at a particular timestep, either one or both of the cameras may not have a particular marker within its FOV, in which case that camera's contribution to the position estimate is null.

In Algorithm 1, the frobeniusAverage function corresponds to Shepperd's method described in Eq. (4), while the calculateWeight function computes a distance-based confidence ($\in [0, 1]$) using a Gaussian-shaped window:

$$\text{distConf} = \begin{cases} 0, & \text{if outOfRange} \\ \exp\left(-\left|\frac{\text{distance}-\text{center}}{\text{scaleFactor}\times\text{rangeWidth}}\right|^{\text{decayFactor}}\right), & \text{otherwise.} \end{cases} \tag{7}$$

where $\text{center} = \frac{\text{minRange}+\text{maxRange}}{2}$ and $\text{rangeWidth} = \frac{\text{maxRange}-\text{minRange}}{2}$. The boolean outOfRange is defined as

$$\begin{aligned} \text{outOfRange} &= (\text{distance} < \text{center} - \text{scaleFactor} \times \text{rangeWidth}) \\ &\vee (\text{distance} > \text{center} + \text{scaleFactor} \times \text{rangeWidth}) \end{aligned} \tag{8}$$

Algorithm 1 Estimating pose using weighted averaging based on distance.

Input: distanceA (Float), poseA (T), distanceB (Float), poseB (T)
Output: estimatedPose (T or None), totalWeight (Float or None)
Initialisation :
minRangeA, maxRangeA = 0.1, 0.8
minRangeB, maxRangeB = 1.0, 3.0
Calculate Weights :
weightA = calculateWeight(distanceA, minRangeA, maxRangeA)
weightB = calculateWeight(distanceB, minRangeB, maxRangeB)
Estimate Pose using Weighted Average :
if weightA + weightB == 0 **then**
  return None
end if
totalWeight = weightA + weightB
estimatedRotation = frobeniusAverage(rotationA, weightA, rotationB, weightB)
estimatedTranslation = (translationA * weightA + translationB * weightB) / totalWeight
estimatedPose = combineRotationTranslation(estimatedRotation, estimatedTranslation)
return estimatedPose, totalWeight

In the implemented fusion, the final per-camera fusion weight used to combine pose estimates is the product of this distance confidence and an inverse-variance term derived from each sensor's calibrated measurement uncertainty:

$$w_i = \text{distConf}_i \times \frac{1}{\sigma_i^2}, \tag{9}$$

where σ_i is the calibrated per-camera positional standard deviation (in metres). This formulation ensures that sensors

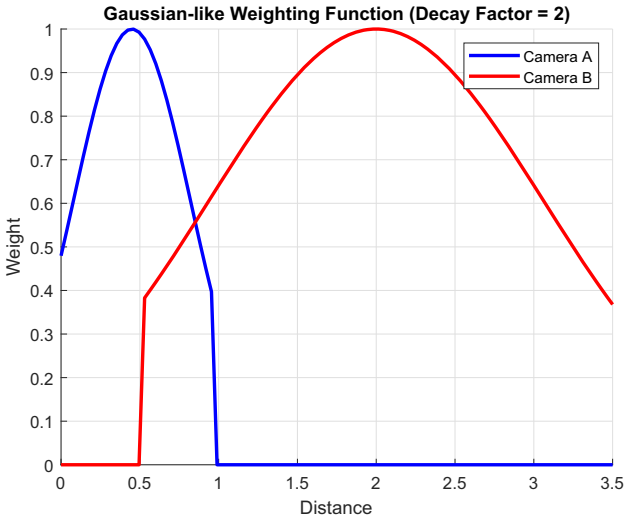


Fig. 8 Camera weights at various distances, with a scaling factor of 1.5 and decay of 2 (prior to multiplying by inverse of variance for the combined case). Camera A is hand, Camera B is global

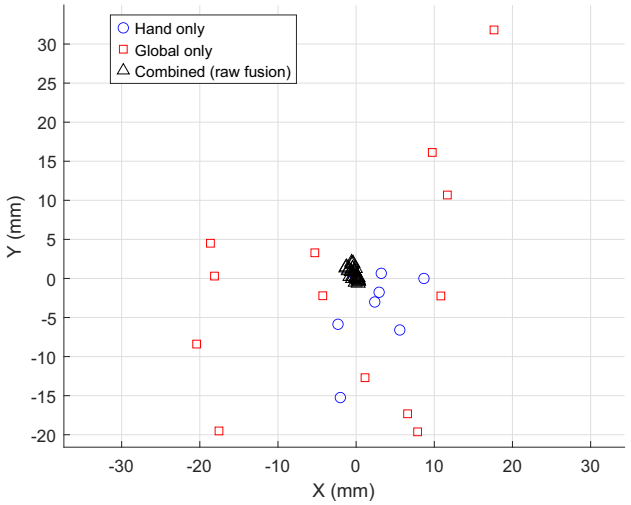


Fig. 9 Readings of marker position from the dual cameras in the world XY plane. Readings from the two cameras and combination of data using Algorithm 1/Eq. (9)

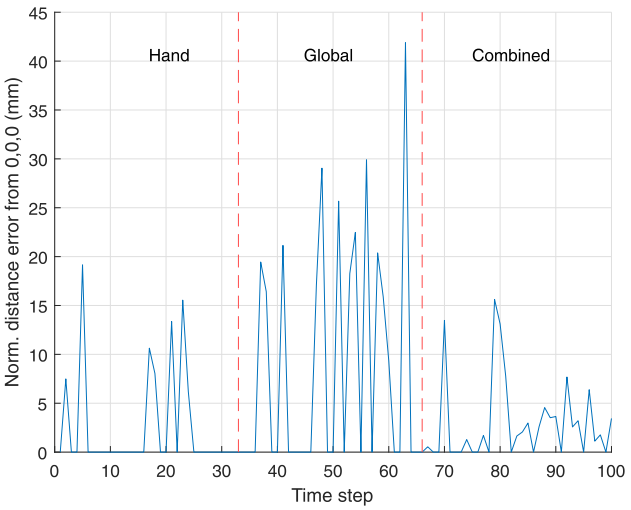


Fig. 10 Normal translation error at various timesteps, for different camera combination setups

with lower measurement variance (higher precision) dominate the fused estimate when they are in-range and confident, while the distance-shaped term gates sensors outside their useful ranges. The statistical results reported in Sect. 4.3 use this combined weighting strategy (hand-camera $\sigma \ll$ global-camera σ), which explains the significant improvement of the combined system over the global-only baseline.

This design is consistent with classical estimation: for unbiased Gaussian noise, inverse-variance weighting is the BLUE/MLE fusion rule (Gelb et al. 1974; Bar-Shalom 2001; Särkkä 2013). The Gaussian-shaped distance term acts as a smooth, range-dependent prior (soft gating), reflecting that pose uncertainty increases with distance and decreases with tag image footprint in vision/photogrammetry (Hartley 2004).

The effect of this weighting is demonstrated in Fig. 8 (displaying the distConf values prior to multiplying by the inverse-variance). Note from the graph that the scaling factor has been selected to provide a cross-over region between the two cameras, so that there is no completely uncertain space between them, although it is assigned less confidence through a lower total weight. Then, at each timestep that the cameras are checked for visible AprilTags, if one is visible, the estimated pose for a particular marker is saved based on Algorithm 1 and Eqs. (7)–(9), as is the totalWeight. The total weighted values then give respective confidence values of each observation frame for a particular marker.

In order to test this, a known environment was set up. A larger 20 cm size AprilTag was placed at a location designated as [0, 0, 0]. The hand camera was positioned at a position [0.5, 0.5] away in the XY plane facing the marker, whilst the global camera was placed at [2.0, 2.0]. The relative positions were roughly confirmed by checking AprilTag readings.

The light levels in the room were then varied over 99 timesteps of taking pose readings. For the first 33 steps, only the hand camera is used; for the next 33 steps, only the global camera is used; and for the final 33 timesteps, both cameras are activated. Figure 9 shows the effect of combining the data readings using Algorithm 1, demonstrating the precision improvement given by combining the two streams. Figure 10 shows the position estimation error that develops over time for the 3 cases of camera combinations.

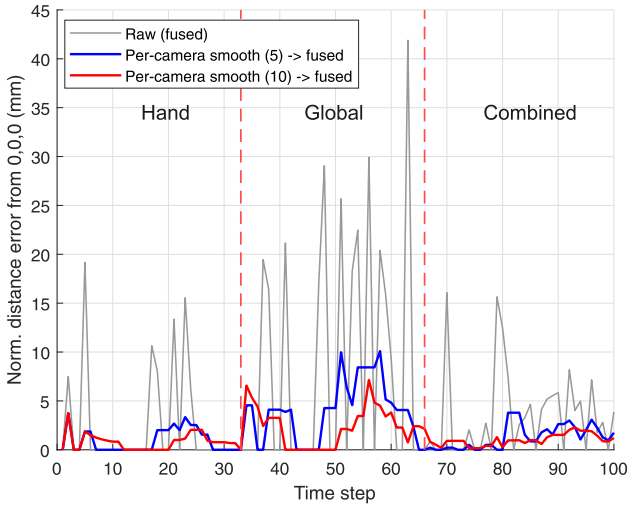


Fig. 11 Moving average filter applied to the distance data, with two different window sizes

4.2.2 Smoothing and filtering

The latest n historic pose estimates are combined by weighted averaging and statistical outlier filtering as described in Sect. 3.4. The total weighted values from Algorithm 1 then give respective confidence values of each observation frame for a particular marker. Note that when a panel is picked by the gripper, the history for its respective marker is cleared until it is placed again.

For each step, the previous n poses are collected as transforms and split into rotational and translational components, which are then filtered for statistical outliers within one standard deviation before the inliers are averaged (see Sect. 3.4). As can be seen in Fig. 11, increasing the sliding window size reduces the error of the translational norm estimate caused by outlier data, showing a significant improvement over the raw data - the maximum error is reduced from 42 mm to just 10 mm in the 5 time step case, and 7 mm in the 10 time step case. Note that for automated usage in the controller the sliding window size is increased to 50 as the sampling rate is far higher than this manually collected data.

4.3 Statistical validation of dual-camera performance

To quantify fusion benefits, the 33 timesteps of each phase are analysed. Three series are distinguished: (i) raw–instantaneous Euclidean position-error norm per timestep including stream combining phase; (ii) causal moving averages of the raw series' (10 timestep lookback specific to camera); and (iii) filtered additional outlier rejection. Unless noted, statistics and hypothesis tests use the final filtered series.

Table 1 Parameter settings and descriptions for ICP parametric study

Parameter	Values
Voxel radius	[0.005 m, 0.01 m]
Fine corr. dist	[3, 5, 10, 20, 35, 50] × voxel_rad
Coarse corr. dist	2 × max_fine_dist
Max fine iterations	[4, 40, 100]
Max coarse iterations	[0, 4, 10]
Noise (see Sect. 4.4)	[LOW, HIGH]

The primary objective was to demonstrate whether augmenting the global camera (which provides broad workspace coverage) with occasional hand camera readings could improve overall system performance. The hand camera, while highly accurate at close range (0.62 ± 1.14 mm mean error), has severely limited coverage and cannot observe most panels in the workspace. The global camera provides essential workspace coverage but with reduced accuracy (2.31 ± 2.95 mm mean error).

Results (filtered; mean $\pm$ SD, median, n): Hand $= 0.79 \pm 0.86$ mm (0.77, $n = 33$); Global $= 2.35 \pm 2.02$ mm (2.28, $n = 33$); Combined $= 1.02 \pm 0.58$ mm (0.94, $n = 34$).

One-sided two-sample t-tests (alternative: single-camera error > Combined) show: Hand vs Combined, $p = 0.8976$ (Cohen's $d = -0.313$, no significant difference); Global vs Combined, $p = 0.0002$ (Cohen's $d = 0.904$, Combined significantly better).

For context, the Global raw series has mean 8.70 mm; comparing Global (raw) to Combined (filtered) yields an 88.3% reduction ($8.70 \rightarrow 1.02$ mm). Causal moving averages (MA5/MA10) reduce short-term variance but do not remove outliers or bias; the filtered fusion achieves larger mean and variance reductions via outlier rejection and inverse-variance weighting, while preserving full workspace coverage.

4.4 Using point cloud data for ICP to augment fiducial pose estimate

Initial parametric testing was carried out to find suitable parameters for ICP through simulation within Gazebo. A model of the Orbbec Astra camera was set up with artificial noise from a known position, to have the effect of transforming the pose estimate provided by the simulated AprilTag. Noisy point cloud data was additionally simulated by perturbation of the mesh model within the simulation environment. Noise was defined as high or low. High noise introduces 8 cm of translation in every direction and 40° of rotation along every axis, whilst low introduces 3 cm and 5° of rotation.

A parametric study was applied to the simulation using the parameters shown in Table 1. Both a single-shot (fine only, Fig. 12), and a refinement test (coarse-fine, Fig. 13)

was run for each parameter set, as shown by the 0 or 4 max_coarse_iters in Table 1. The parameters were iterated through combinatorially by changing, sequentially, fine correspondence distance, number of coarse iterations then maximum number of fine iterations, then noise level, producing 72 parameter sets. For the fine only case, no coarse iterations were run, giving 36 parameter sets. The parameter sets were then tested for 2 different voxel radii each, giving 216 different parameter sets in total.

For the parameter set, reducing the voxel size makes the point cloud data more dense after downsampling, and increases the computation time in all cases. The fine-only, single shot case of Fig. 12 displays some parameter sets which perform well for the low-noise data—achieving ADD − 6d < 10% (= 0.046 m) of panel width in below 1 s—but for the high noise cases there are no such highly performant parameter sets for either the 1 cm or 5 mm voxelised data. For the coarse-fine data in Fig. 13, for the majority of high noise parameters, except for those with only 4 fine iterations, the total duration of the ICP is at least 2 s (on an i5 processor running Linux in a virtual machine). Of interest is the fact that, even with a low number of 4 fine ICP iterations, the ADD-6d error value tends to be improved significantly, always within 10% except in the 4 fine iterations cases.

For the coarse-fine ICP scheme shown in Fig. 13, there are now some parameter sets found which manage to get ADD-6d under 10% of the panel width (= 0.046 m) within a short time for both the high and low noise states. These parameter sets are extracted for closer inspection in Fig. 14. The best performing parameter set is where the correspondence distance for fine ICP is 10 times the voxel radius, with a lowest maximum ICP time at the minimum ADD-6d value found of 0.028 m, or 6% of the panel width. The maximum number of fine ICP iterations was 40 for every parameter set displayed in Fig. 14. Note that for the low-noise data, there are minimal improvements in ADD-6d scoring using the fine-tuning, and the duration of ICP tends to be increased. Additionally, there appears to be little benefit in terms of ADD-6d scoring in having a higher maximum amount of fine-tuning iterations, as the process stops early when it converges. From this data, it appears that 1 cm voxel size with 4 max coarse iterations and 40 max fine iterations is suitable here for correcting discrepancies in pose data for panels using their point clouds.

The effect of the selected parameter set for ICP is shown acting on real-world data in Fig. 15a acting on panel #8, which is shown with low noise being applied to the AprilTag provided pose and used to generate the mesh data (red). The actual panel cloud being read is shown in grey, blue represents the data after coarse ICP refinement and blue represents the cloud after fine ICP refinement. This demonstrates the correction and improvement that the coarse/fine ICP scheme applies to the real data with potentially noisy AprilTag input. Also shown is the effect of adding additional panels to the system (Fig. 15b, c), which pulls the registration towards the volumetric centroid of the entire set of panels; which can however be mitigated by selecting a region for cropping based on the initial guess.

4.5 Robotic insertion experiments

In the absence of a vacuum gripper, custom gripper fingers were developed to grasp the manufacturing reference dowels on the face of the panels, allowing the panels to reach the furthest extents of the reach space. The triangular shape of the fingers provided a funneling effect of sorts; in closing, the dowels would naturally be pushed to the apex of the fingers, allowing the gripper to correct for slight errors in picking (Fig. 16).

Taking panel geometry and as-designed relative poses between neighbours as input, a pick and placepart insertion pipeline was conceived (Fig. 17). The arm was set to use impedance control during linear Cartesian insertion movements. This helped to mitigate potential damage caused by incorrect state estimates, by adding a spring/damper effect to the end-effector along its Z axis.

The robot end effector were manually moved through a scanning motion of the semi-built panel structure to collection data for camera fusion. Panels were placed into the end-effector fingers, and then the robot was commanded to place them into the structure at the correct pose using translational insertion vectors found from the geometry. The panel placements were tested in 4 cases with 3 repetions each:

1. With no sensor feedback based on manual measurement of base location
2. With AprilTag only feedback (only global camera)
3. With combined and smoothed AprilTag feedback (both cameras)
4. With point cloud registration for fine tuning

Panel placement was tested primarily for a single panel, with either only 1 neighbour in the structure, or 2 neighbours, with the base panel at a calibrated known pose. Results for success of placement using the previously found optimised parameters were recorded for each case, and are shown in Table 3. As can be seen, the robot was unable to assemble in the basic, non-sensor feedback case because the deflection in the structure was too significant, and the panel was pushed into the surface of the neighbours instead of inserted properly.

The assembly process highlights the structural fragility of the shell during the intermediate stages. While the final geometry is self-supporting, the partially assembled structure becomes increasingly susceptible to elastic deformation as height increases (Fig. 18). To mitigate collapse or joint damage during the physical insertion of these thin timber

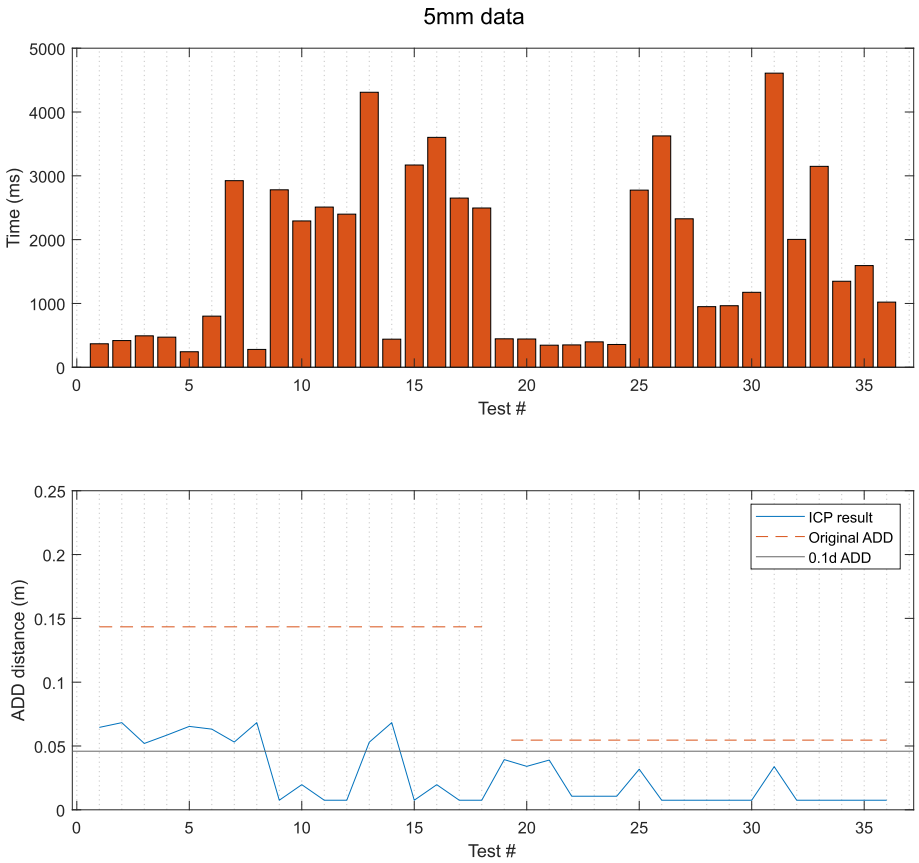


Fig. 12 Efficiency vs. accuracy trade-off for single-shot ICP (0.5 cm voxel). The left region shows low initial noise, the right high noise. Each vertical bar represents the total processing time. The overlaid line plot (red dots) indicates the resulting ADD-6d pose error (m). Optimal parameters minimise both bar height (time) and ADD-6d pose error

panels, the robotic insertion strategy incorporates impedance control (stiffness damping) along the insertion vector. This allows the arm to act as a virtual spring, complying with minor misalignments rather than rigidly forcing the fragile timber joints together. This compliance, coupled with the refined pose estimation, was essential for keeping the panels from falling or breaking during the assembly process without the need for external shoring. Particularly, improvements to apply more rigid joints might be suitable for future research (and other works already assess the use of additional arms for temporary scaffolding during assembly (Bruun et al. 2024)).

While using only the global camera on the tripod for AprilTag feedback, the insertion into a single neighbour was successful. However, for two neighbours, the estimated insertion pose was not accurate enough, and the dovetail for the second panel hit the face of the neighbour. This was corrected when adding the hand camera and smoothing the data. The initial inspection of the neighbours allowed for a better state estimate for the insertion pose, providing an average pose to insert it into, based on the designed pose transforms from both neighbours. Finally, for the insertion of panels making use of point cloud data, the panel could not be inserted with 2 neighbours despite multiple repeat attempts. The point cloud data was being registered to an incorrect area for the dual panel case, in between the two neighbours, suggesting that the cropping region aspect should be refined in future work. Finally, the rest of the panels were then inserted, using only the combined and smoothed AprilTag feedback mechanism, as shown in Fig. 18.

To systematically evaluate the stability of the robot state recognition and the physical assembly task across these multiple experimental runs, a Failure Mode and Effects Analysis (FMEA) was conducted (Table 2). By observing the repeated assembly attempts detailed in Table 3, specific failure modes were identified, ranging from perception errors to physical jamming. The FMEA framework not only categorises how each assembly step works or fails, but also outlines the implemented controls that enabled the successful insertions, alongside recommended improvements for future industrial scaling.

The primary failure modes observed during the repeated trials were: (1) gross pose estimation errors leading to collision, particularly when relying on a single distant camera; (2) structural deflection of the partially built shell causing joint misalignment; (3) tag occlusion during the final approach phase; and (4) point cloud registration failures in multi-panel contexts. As demonstrated in Table 3, the implemented mitigations—specifically the dual-camera fusion, temporal smoothing, and impedance control—significantly increased

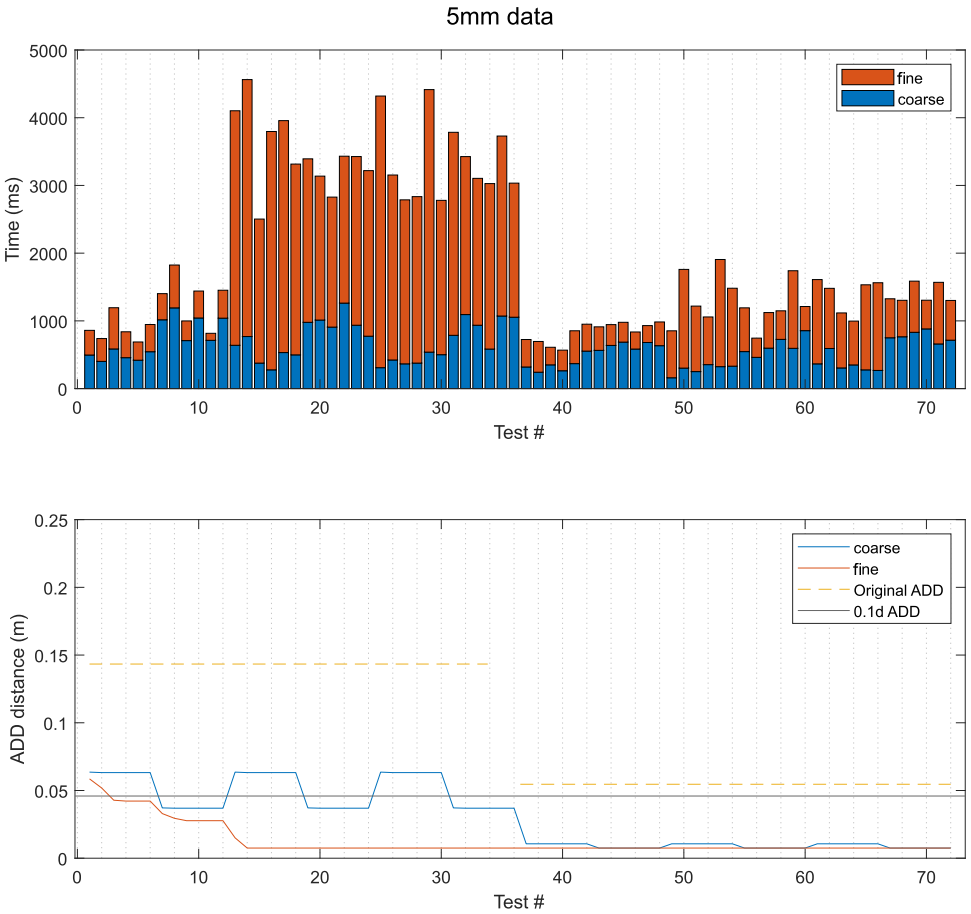


Fig. 13 Efficiency and accuracy trade-off for coarse-to-fine ICP (0.5 cm voxel). The left region shows low initial noise, the right high noise. Each vertical bar represents the total processing time (stacked: blue = coarse ICP, orange = fine ICP). The overlaid line plot (red dots) indicates the resulting ADD-6d pose error (m). Optimal parameters minimise both bar height (time) and ADD-6d pose error

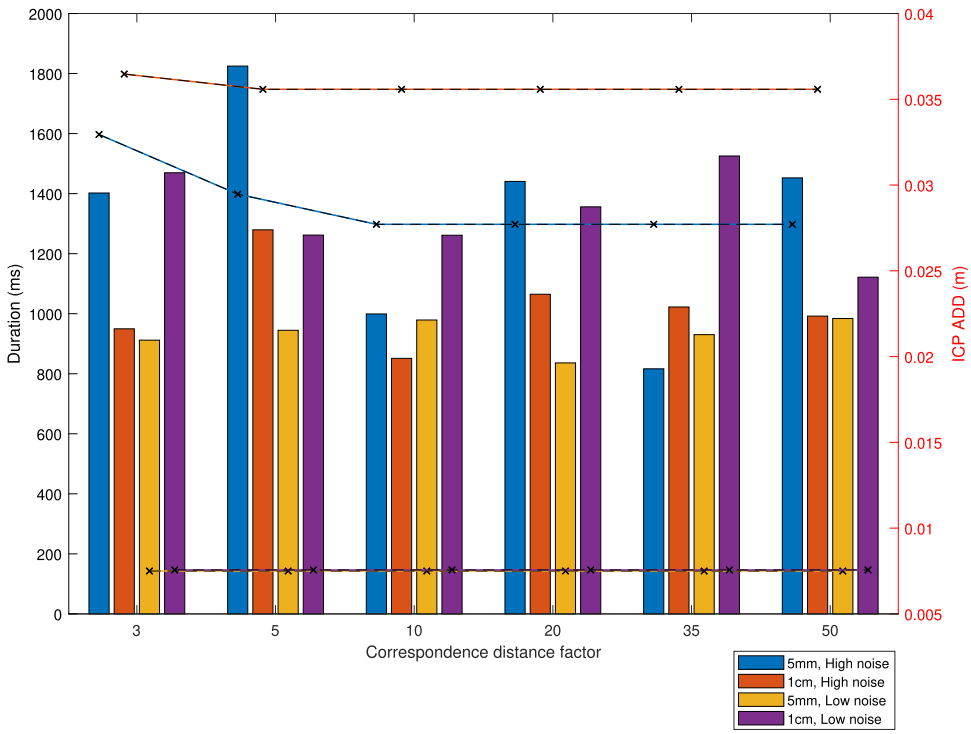
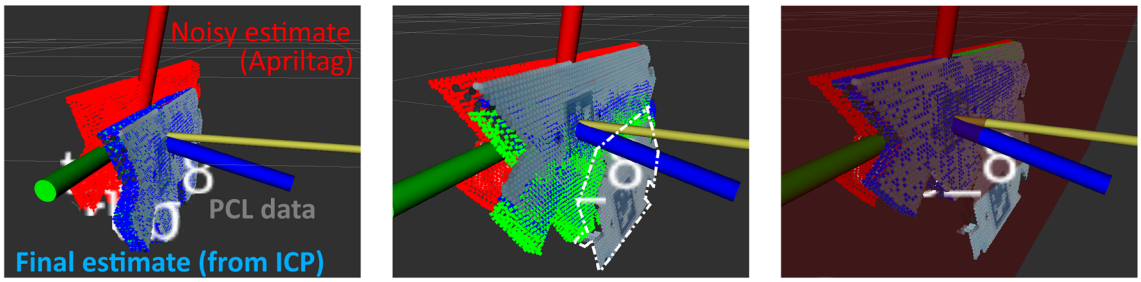
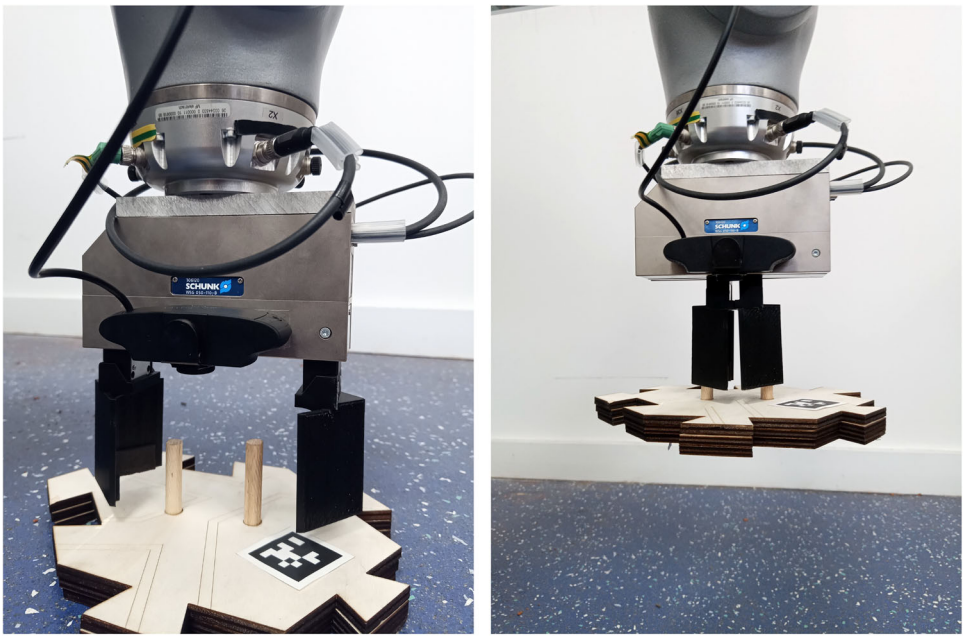


Fig. 14 Closer inspection of ADD-6d error and duration for the coarse/fine ICP scheme. Bars indicate full duration, lines indicate ADD-6d error measurement



(a) (b) (c)

Fig. 15 Coarse/fine ICP acting on an artificially noisy AprilTag pose, in red. In grey is the point cloud data measured for the panel, green is the coarse estimate, and blue represents the final pose estimated by the point-to-plane ICP scheme. **a** A single panel in the scene demonstrates accurate pose refinement with ICP. **b** Now with a neighbour added (highlighted dashed region), the ICP procedure attempts to place the point cloud between the panel and its neighbour, shown by the misaligned green estimate. **c** The effect of cropping the point cloud to a region of interest (red box) dictated by the initial AprilTag estimate can be seen, which allows ICP to correctly locate the cloud without being pulled into the neighbour panels



(a) (b)

Fig. 16 3D printed V-shaped gripper configuration for grasping dowels, in an early test which utilised paper tags. **a** Gripper open around dowels. **b** Gripper closed. Dowels have been funnelled into a known position

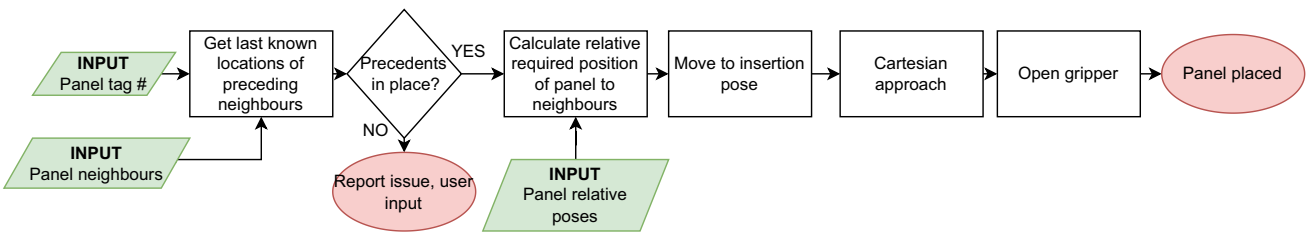


Fig. 17 Insertion pipeline within ROS, using AprilTag pose estimates for positioning

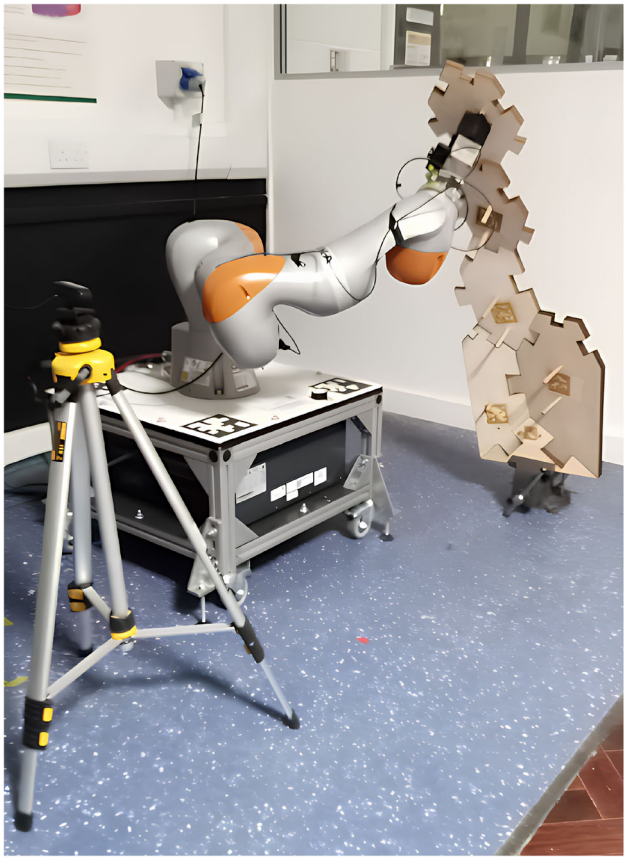


Fig. 18 Final assembly of shell structural panels with the manipulator arm using dual camera AprilTag pose estimation. The first 3 base panels were manually assembled and the arm was used for assembly of the other 4

the stability and success rate of the assembly task compared to deterministic or single-sensor baselines.

5 Comparison with the state of the art

State-of-the-art fiducial pipelines typically rely on a single global camera observing AprilTags/ArUco markers (Olson 2011; Muñoz Salinas et al. 2018), or an eye-in-hand camera offering high precision in a limited workspace (Kalaitzakis et al. 2021). Multi-camera tag systems improve coverage and robustness by viewing shared tags or boards and solving PnP across visible markers (Yoon et al. 2006; Malyuta 2018; Guan et al. 2024; Muñoz Salinas et al. 2018), but are often calibrated offline and can drift when extrinsics change. Marker-seeded point cloud registration is also common, using fiducials to initialise ICP or related variants (Yang and Allen 1998; Barone et al. 2012; Rusinkiewicz and Levoy 2001; Zhou et al. 2016); this boosts local accuracy yet can be slow or unreliable without careful region-of-interest selection in clutter.

Table 2 Failure mode and effects analysis (FMEA) summarising observed faults, implemented controls, and future improvements for the robotic assembly process

Process step	Potential failure mode	Potential effects	Implemented controls (current)	Recommended improvements (future)
State recognition (global)	Gross pose estimation error or tag "flip" due to lighting/distance	Incorrect target coordinate calculated; Robot collides with neighbour panel	Multi-camera fusion with inverse-variance weighting; Temporal outlier rejection (sigma-clipping)	Integrate active lighting; Upgrade to high-resolution industrial PoE cameras
State recognition (local)	Tag occlusion during final end-effector approach	Loss of tracking immediately prior to insertion; Blind movement	Dual-camera redundancy ensures global camera maintains tracking when hand camera is occluded	Implement markerless visual-inertial odometry (VIO) for continuous tracking during occlusion
Point cloud registration	ICP converges to local minima between multiple panels	Severe pose distortion; Complete failure of insertion (Case 4, 2-neighbours)	Initial AprilTag pose used to crop Point Cloud to a specific Region of Interest (ROI)	Implement semantic segmentation or deep-learning based bounding box generation prior to ICP
Physical insertion	Joint misalignment due to structural deflection of the shell	Dovetail joint jams; Risk of fracturing the fragile timber panels	Impedance control (stiffness damping) allows robot compliance; 2-neighbour pose averaging	Integrate force-torque sensor feedback for active search-and-insert routines

Table 3 Success of placing panel under different feedback conditions, with varying numbers of existing neighbour panels in structure

Case	Neighbours	
	1	2
1 (Deterministic)	XXX	XXX
2 (Single camera)	✓✓ X	XXX
3 (Combined cams)	✓✓✓	X✓✓
4 (Cams & PCL)	✓✓✓	✓ XX

Compared to these, our method combines: (i) continuous self-calibration of the global camera using fixed robot-base tags, (ii) statistical fusion of dual-camera tag poses with distance-aware, inverse-variance weighting and robust temporal sigma-clipping, and (iii) optional coarse-to-fine G-ICP refinement. Empirically, the fused stream (1.02 ± 0.58 mm) significantly outperforms a global-only filtered baseline (2.35 ± 2.02 mm; $p = 0.0002$), while preserving full coverage; the hand camera remains most precise at close range (0.79 ± 0.86 mm) but lacks coverage. ICP provides additional local refinement when a focused ROI is available, at the cost of higher runtime (Table 4).

6 Conclusion

This work presented a hybrid sensing framework that fuses a global RGB-D camera with an eye-in-hand camera for reliable, workspace-wide tracking of construction elements. The global stream maintains coverage and self-calibrates via robot base markers, the hand stream provides close-range precision, and optional ICP refines final poses. Because the global camera is tripod-mounted and continuously re-registered to the base tags, both the sensor and the workpiece can be repositioned while preserving a consistent robot-centric frame, effectively expanding the functional workspace without fixed, survey-grade infrastructure. In experiments, distance-aware, inverse-variance fusion reduced mean position error from 8.70 mm (global raw) to 1.02 mm (fused, filtered) and significantly outperformed a filtered global-only baseline ($p = 0.0002$). A causal sliding window with sigma-clipping further suppressed outliers, lowering maximum translation error from about 4 cm (unfiltered) to roughly 0.9–1.2 cm depending on window size. Laser-engraved AprilTags in timber were reliably detected with low-cost cameras at practical ranges ($\approx$ 2 m in SD mode), and the fused, smoothed feedback enabled successful insertions where deterministic or single-camera strategies failed, particularly in multi-neighbour placements.

Performance remains sensitive to extrinsic calibration (e.g., hand camera calibration inaccuracies) and lighting-driven tag ambiguities. Furthermore, iterative closest point

Table 4 Summary comparison with representative approaches

Method	Strengths	Limitations
Single global camera + tags (Olson 2011; Muñoz Salinas et al. 2018)	Wide coverage; low cost; simple deployment	Accuracy degrades with distance; sensitive to lighting/occlusion; typically fixed extrinsics; no range-aware fusion or robust temporal filtering
Eye-in-hand only (Kalaitzakis et al. 2021)	High precision near contact; good for local tasks	Poor global coverage; frequent occlusions; no global consistency
Multi-camera tags, PnP/boards (Yoon et al. 2006; Malyuta 2018; Guan et al. 2024; Muñoz Salinas et al. 2018)	Redundancy and robustness to occlusion; broader coverage	Often offline/static calibration; limited self-calibration during operation; fusion rarely distance/variance aware; basic outlier handling
Marker-seeded registration (ICP/FGR) (Yang and Allen 1998; Barone et al. 2012; Rusinkiewicz and Levoy 2001; Zhou et al. 2016)	Refines local pose; robust with good ROI and initialisation	Runtime overhead; sensitive to clutter/ROI; can drift without good priors
Proposed dual-camera fusion + self-calibration + robust filtering (+ optional ICP)	Preserves coverage while approaching close-range precision; continuous self-calibration to robot markers; inverse-variance and distance-aware fusion; temporal sigma-clipping; validated insertion success	Fusion assumes calibrated uncertainties; residual bias if extrinsics change without checks; ICP adds latency and needs ROI tuning

(ICP) accuracy and CPU runtimes—currently exceeding 1–2 s—depend heavily on region of interest (ROI) selection and scene clutter, limiting online use in highly dynamic scenes, but remaining acceptable for semi-static construction environments such as presented here. To address these limitations and broaden the system's applicability, future work might explore integrating modern markerless pose estimation methods. Visual-inertial odometry (Mitterberger et al. 2020) and feature-based tracking offer promising alternatives for maintaining continuous pose estimates during fiducial occlusion. Similarly, deep learning-based frameworks (e.g., PoseCNN, PVNet) eliminate the need for physical markers entirely. While these methods currently demand significantly higher computational resources and extensive application-specific training datasets (Molaei et al. 2024), they could powerfully complement our system by providing coarse initial alignments or dynamic ROI generation to mitigate ICP bottlenecks.

Future work should focus on more robust online calibration and uncertainty propagation; making ICP reliably real-time via interface-focused ROI cropping, feature/segment-based gating, GPU acceleration, or hand-mounted depth sensing; and extending to multi-camera networks with Bayesian temporal fusion. Crucially, to translate this study into a blueprint for real-world construction automation, future research must bridge the gap between the current laboratory-scale, laser-cut timber panels and full-scale industrial applications. A specific industrial scenario would involve the automated prefabrication of large-scale timber structures, such as Cross-Laminated Timber (CLT) or glulam assemblies, where components are manufactured via industrial CNC milling, rather than laser cutting which was used in this study to match the scale of work and resources. In such a setting, the proposed dual-camera framework would be scaled up by replacing the low-cost webcam with an industrial-grade PoE camera network and integrating the ROS-based control backend into standard industrial PLC environments via standardised sensor formats (e.g. OPC UA). The fiducial markers, currently laser-engraved, could be seamlessly integrated into the CNC milling process as shallow routed pockets or printed tags applied during the factory quality-control phase. By utilising the developed FMEA to design automated fault-recovery behaviours, this blueprint demonstrates how the proposed fusion strategy offers a practical, scalable path to precise state estimation for full-scale robotic construction, delivering the precision of close-range sensing without sacrificing the global coverage required for large architectural elements.

Appendix A Camera extrinsic calibration

Two cameras were set up and intrinsic calibration performed with a standard checkerboard calibration, with one low-cost camera affixed to the manipulator gripper to work eye-in-hand, and one Orbbec Astra RGB-D camera on a tripod, to give a more global view of the scene (Fig. 2). For extrinsic calibration, using the multiple markers on the robot base, an accurate ensemble result can be found and the pose of the global camera (gCam) relative to the world frame is

$$\mathbf{T}^{\text{world}}_{\text{gCam}} = \left(\mathbf{T}^{\text{gCam}}_{\text{tag0}}\right)^{-1} \cdot \mathbf{T}^{\text{world}}_{\text{tag0}} \tag{A1}$$

where the homogenous transformation from the global camera frame to tag 0 $\left(\mathbf{T}^{\text{gCam}}_{\text{tag0}}\right)$ is given by the AprilTag library, and from world to tag 0 $\left(\mathbf{T}^{\text{world}}_{\text{tag0}}\right)$ is known by the placement of the marker at the corner of the table.

Additionally, an accurate pose estimate of the hand camera with relation to the wrist itself can be found. The hand camera was set up facing tag 0. Note that $\mathbf{T}^{\text{world}}_{\text{wrist}}$ is known from arm kinematics, $\mathbf{T}^{\text{world}}_{\text{tag0}}$ is known by the placement of the marker, and $\mathbf{T}^{\text{hCam}}_{\text{tag0}}$ is provided by the pose estimate of the tag in the camera frame. Then the transformation from the wrist to the hand camera is given by

$$\mathbf{T}^{\text{wrist}}_{\text{hCam}} = \left(\mathbf{T}^{\text{world}}_{\text{wrist}}\right)^{-1} \cdot \mathbf{T}^{\text{world}}_{\text{tag0}} \cdot \mathbf{T}^{\text{tag0}}_{\text{hCam}} \tag{A2}$$

By taking a series of results for this transformation whilst moving the end-effector and taking an average of the results, taking care with averaging the quaternions, a reasonable estimate of the camera pose relative to the wrist could be calibrated. Note that while the hand camera can be calibrated in this way, it is impractical to recalibrate during the operation of the arm for assembly, and so the camera should not be allowed to move with respect to the end-effector after initial calibration. Conversely, the global camera on its tripod is constantly self-calibrating during the arm operation when it can see one of the four markers on the arm. In this way, the tripod can be moved to get a better view of regions of focus. This could, of course, be extended to additional cameras to give better sensor coverage and redundancy. Additionally, since there are multiple markers on the base with known relative poses comprising a tag bundle, a degree of redundancy is provided since only one of the 4 markers needs to be in view of a camera to estimate the pose of tag 0.

Funding Open access funding provided by Politecnico di Milano within the CRUI-CARE Agreement. This work was supported by the Engineering and Physical Sciences Research Council [Grant number EP/T517860/1] and by the AS Lab at Politecnico di Milano.

Data availability Data available on request.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Abdelrahman M, Macatulad E, Lei B et al (2025) What is a digital twin anyway? Deriving the definition for the built environment from over 15,000 scientific publications. Build Environ 274:112748. https://doi.org/10.1016/j.buildenv.2025.112748

Barone S, Paoli A, Razionale AV (2012) Three-dimensional point cloud alignment detecting fiducial markers by structured light stereo imaging. Mach Vis Appl 23(2):217–229. https://doi.org/10.1007/s00138-011-0340-1

Barros I, Bezerra R, Assabumrungrat R et al (2025) Improving indoor localization: a low-cost, multi-marker and multi-camera system for robot tracking. In: 2025 IEEESICE international symposium on system integration (SII). IEEE, Munich, Germany, pp 1083–1089. https://doi.org/10.1109/SII59315.2025.10870879

Bar-Shalom Y (2001) Estimation with applications to tracking and navigation. Wiley-Interscience, New York (A Wiley-Interscience publication. Includes bibliographical references and index) (Print version record)

Bradski G (2000) The opencv library. Dr Dobb's J Softw Tools

Bruun EPG, Parascho S, Adriaenssens S (2024) Cooperative robotic fabrication for a circular economy. In: De Wolf C, Çetin S, Bocken NMP (eds) A circular built environment in the digital age. Circular economy and sustainability. Springer International Publishing, Cham, pp 129–149. https://doi.org/10.1007/978-3-031-39675-5_8

Casas G (2019) roslibpy: Python ROS bridge library. Type: Computer Program

Coleman D, Şucan I, Chitta S et al (2014) Reducing the barrier to entry of complex robotic software: a MoveIt! Case study. J Softw Eng Robot 5(1):3–16. https://doi.org/10.6092/joser_2014_05_01_p3

Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 24(6):381–3. https://doi.org/10.1145/358669.358692

Garrido-Jurado S, Muñoz Salinas R, Madrid-Cuevas FJ et al (2014) Automatic generation and detection of highly reliable fiducial markers under occlusion. Pattern Recognit 47(6):2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005

Gelb A, Kasper JF, Nash RA et al (eds) (1974) Applied optimal estimation. MIT Press, Cambridge

Guan J, Hao Y, Wu Q et al (2024) A survey of 6DoF object pose estimation methods for different application scenarios. Sensors 24(4):1076. https://doi.org/10.3390/s24041076

Hartley R (2004) Multiple view geometry in computer vision, 2nd edn. Cambridge University Press, Cambridge (title from publisher's bibliographic system) (viewed on 05 Oct 2015)

Hennersperger C, Fuerst B, Virga S et al (2017) Towards MRI-based autonomous robotic us acquisitions: a first feasibility study. IEEE Trans Med Imaging 36(2):538–548

Hinterstoisser S, Lepetit V, Ilic S et al (2013) Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In: Lee KM, Matsushita Y, Rehg JM et al (eds) Computer vision—ACCV 2012. Springer, Berlin, pp 548–562. https://doi.org/10.1007/978-3-642-37331-2_42

Iturralde K, Shen J, Bock T (2023) AprilTag detection for building measurement. In: 40th international symposium on automation and robotics in construction, vol 2023 Proceedings of the 40th ISARC, Chennai, India. IAARC, Chennai, India, pp 589–592. https://doi.org/10.22260/ISARC2023/0079

Kalaitzakis M, Cain B, Carroll S et al (2021) Fiducial markers for pose estimation: overview, applications and experimental comparison of the ARTag, AprilTag, ArUco and stag markers. J Intell Robot Syst 101(4):71. https://doi.org/10.1007/s10846-020-01307-9

Kang CH, Kim SY (2024) 3D-modeling techniques for object recognition based on point cloud data. JMST Adv 6(3):329–340. https://doi.org/10.1007/s42791-024-00086-w

Kayhani N, Zhao W, McCabe B et al (2022) Tag-based visual-inertial localization of unmanned aerial vehicles in indoor construction environments using an on-manifold extended Kalman filter. Autom Constr 135:104112. https://doi.org/10.1016/j.autcon.2021.104112

Kunic A, Cognoli R, Naboni R (2024) RE:thinking timber architecture. Enhancing design and construction circularity through material digital twin. In: Thomsen MR, Ratti C, Tamke M (eds) Design for rethinking resources. Springer International Publishing, Cham, pp 409–422. https://doi.org/10.1007/978-3-031-36554-6_26

Li W, Cheng H, Zhang X (2021) Efficient 3D object recognition from cluttered point cloud. Sensors 21(17):5850. https://doi.org/10.3390/s21175850

Lundeen KM, Dong S, Fredricks N et al (2016) Optical marker-based end effector pose estimation for articulated excavators. Autom Constr 65:51–64. https://doi.org/10.1016/j.autcon.2016.02.003

Ma JW, Czerniawski T, Leite F (2020) Semantic segmentation of point clouds of building interiors with deep learning: augmenting training datasets with synthetic BIM-based point clouds. Autom Constr 113:103144. https://doi.org/10.1016/j.autcon.2020.103144

Malyuta D (2017) Apriltag_ros. Github. https://github.com/AprilRobotics/apriltag_ros

Malyuta D (2018) Guidance, navigation, control and mission logic for quadrotor full-cycle autonomy. Master's thesis ETH Zurich. https://doi.org/10.3929/ethz-b-000248154

Markley FL, Cheng Y, Crassidis JL et al (2007) Averaging quaternions. J Guid Control Dyn 30(4):1193–1197. https://doi.org/10.2514/1.28949

Mitterberger D, Dörfler K, Sandy T et al (2020) Augmented bricklaying: human–machine interaction for in situ assembly of complex brickwork using object-aware augmented reality. Constr Robot 4(3–4):151–161. https://doi.org/10.1007/s41693-020-00035-8

Molaei A, Kolu A, Lahtinen K et al (2024) Automatic recognition of excavator working cycles using supervised learning and motion data obtained from inertial measurement units (IMUs). Constr Robot 8(2):14. https://doi.org/10.1007/s41693-024-00130-0

Muñoz Salinas R, Marín-Jimenez MJ, Yeguas-Bolivar E et al (2018) Mapping and localization from planar markers. Pattern Recognit 73:158–171. https://doi.org/10.1016/j.patcog.2017.08.010

Olson E (2011) AprilTag: a robust and flexible visual fiducial system. In: 2011 IEEE international conference on robotics and automation. IEEE. https://doi.org/10.1109/icra.2011.5979561

Pomerleau F, Colas F, Siegwart R et al (2013) Comparing ICP variants on real-world data sets. Auton Robot 34(3):133–148. https://doi.org/10.1007/s10514-013-9327-2

Popescu DC, Dumitrache I, Caramihai SI et al (2020) High precision positioning with multi-camera setups: adaptive Kalman fusion algorithm for fiducial markers. Sensors 20(9):2746. https://doi.org/10.3390/s20092746

Raguram R, Frahm JM, Pollefeys M (2008) A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus. Springer, Berlin, pp 500–513. https://doi.org/10.1007/978-3-540-88688-4_37

Robeller C, Weinand Y (2015) Interlocking folded plate—integral mechanical attachment for structural wood panels. Int J Space Struct 30(2):111–122. https://doi.org/10.1260/0266-3511.30.2.111

Rogeau N, Gamerro J, Latteur P et al (2022) Design considerations for robotically assembled through-tenon timber joints. Constr Robot 6(3):293–304. https://doi.org/10.1007/s41693-022-00080-5

Rusinkiewicz S, Levoy M (2001) Efficient variants of the ICP algorithm. In: Proceedings third international conference on 3-D digital imaging and modeling. IEEE Comput. Soc, IM-01, pp 145–152. https://doi.org/10.1109/im.2001.924423

Särkkä S (2013) Bayesian filtering and smoothing. No. 3 in Institute of Mathematical Statistics textbooks. Cambridge University Press, Cambridge

Settimi A, Gilliard D, Skevaki E et al (2025) DiffCheck: a scan-CAD evaluation tool for digital manufacturing and assembly processes in timber construction. CRC Press, pp 687–695. https://doi.org/10.1201/9781003658641-83

Shen J, Huang Z, Jiao L (2024) Self-supervised monocular depth estimation on construction sites in low-light conditions and dynamic scenes. Autom Constr 168:105848. https://doi.org/10.1016/j.autcon.2024.105848

Shepperd SW (1978) Quaternion from rotation matrix. J Guid Control 1(3):223–224. https://doi.org/10.2514/3.55767b

Shoemake K (1985). Animating rotation with quaternion curves. https://doi.org/10.1145/325334.325242

Song Y, Koeck R, Luo S (2021) Review and analysis of augmented reality (AR) literature for digital fabrication in architecture. Autom Constr 128:103762. https://doi.org/10.1016/j.autcon.2021.103762

Steinwolf A (2010) Shaker random testing with low kurtosis: review of the methods and application for sigma limiting. Shock Vib 17(3):219–231. https://doi.org/10.1155/2010/502829

Tang P, Huber D, Akinci B et al (2010) Automatic reconstruction of as-built building information models from laser-scanned point clouds: a review of related techniques. Autom Constr 19(7):829–843. https://doi.org/10.1016/j.autcon.2010.06.007

Vestartas P (2021) OpenNest—2D polyline packing for fabrication such as laser or CNC cutting. https://www.food4rhino.com/en/app/opennest

Wilcock S, Fang H, Dogar MR et al (2024) Integrating R-funicularity, local stability and inter-panel constraint assessment for discrete timber shell construction design. Structures 64:106592. https://doi.org/10.1016/j.istruc.2024.106592

Wilcock S (2025) ros_icp_simulation. Zenodo. https://doi.org/10.5281/zenodo.1479784

Yang R, Allen PK (1998) Registering, integrating, and building cad models from range data. In: Proceedings. 1998 IEEE international conference on robotics and automation (Cat. No.98CH36146), vol 4. IEEE, Leuven, Belgium, pp 3115–3120. https://doi.org/10.1109/ROBOT.1998.680904

Yoon JH, Park JS, Kim C (2006) Increasing camera pose estimation accuracy using multiple markers. In: Pan Z, Cheok A, Haller M et al (eds) Advances in artificial reality and tele-existence, vol 4282. Springer, Berlin, pp 239–248. https://doi.org/10.1007/11941354_25

Zarei M, Chhabra R, Janabi-Sharifi F (2025) Correlation-aware dual-view pose and velocity estimation for dynamic robotic manipulation. https://doi.org/10.48550/ARXIV.2510.05536. https://arxiv.org/abs/2510.05536. Version number: 1

Zhou QY, Park J, Koltun V (2016) Fast global registration. Springer International Publishing, Berlin, pp 766–782. https://doi.org/10.1007/978-3-319-46475-6_47

Zhou QY, Park J, Koltun V (2018) Open3D: a modern library for 3D data processing.https://doi.org/10.48550/ARXIV.1801.09847. arxiv:1801.09847. Version number: 1