

# Semi-Automatic Registration of a Robotic Total Station and a CAD Model Without Control Points

Christoph Klug  
 VRVIS Research Center  
 Vienna, Austria  
 klug@vrvis.at

Clemens Arth  
 Graz University of Technology  
 Graz, Austria  
 arth@icg.tugraz.at

Dieter Schmalstieg  
 Graz University of Technology  
 Graz, Austria  
 schmalstieg@icg.tugraz.at

Thomas Gloor  
 Hilti Corp.  
 Schaan, Liechtenstein  
 thomas.gloor@hilti.com

**Abstract**— The accurate registration of a robotic total station with respect to a given CAD model is a crucial task in the construction industry. Common registration techniques rely on a reference network of control points in the CAD model. One must establish correspondences between control points in the CAD model and measured points in the field. Usually physical markers or natural points of interest are selected as control points. We present a user-guided algorithm for simple and efficient registration of a robotic total station with a CAD model in indoor environments without the need for control points. The user interaction is reduced to selecting a local Manhattan-like corner structure for initial model alignment; accurate registration of the device is carried out automatically. Our algorithm relies on angle and distance measurements only and, therefore, is not limited to vision based robotic total stations. In particular, we propose a new algorithm for robust Manhattan corner extraction.

**Index Terms**—Mechatronics and Robotics, Motion and Servo Control

## I. INTRODUCTION

A robotic total station (RTS) is used for accurate measurements of angles, distances and 3D points [1]; a modern RTS combines an electronic theodolite, an electronic distance meter (EDM) and one or more cameras. The simplified geometric model with one camera and no parallax effects between the components is shown in Fig. 1.

An RTS provides measurements in a local coordinate frame. Measurements from multiple positions must be linked together by registering the measurements with respect to a common coordinate system. Common registration techniques rely on correspondences between control points. Usually, physical markers or natural points of interest are selected as control points. Physical markers can be reflective targets with known measurement properties or non-reflective targets like geodetic marks or cast metal disks. Common natural measurement targets are building corners, edges or steeples. They have a high recall value and avoid physical installation of targets.

A common task of as-built and indoor surveying is the registration of an RTS with respect to a CAD model. Especially when planning or documenting installations, planning reconstructions or placing physical markers for assembly drills, fast RTS registration without dedicated control points is desired.

This work was funded through the COMET program No. 843272 by the Austrian FFG.

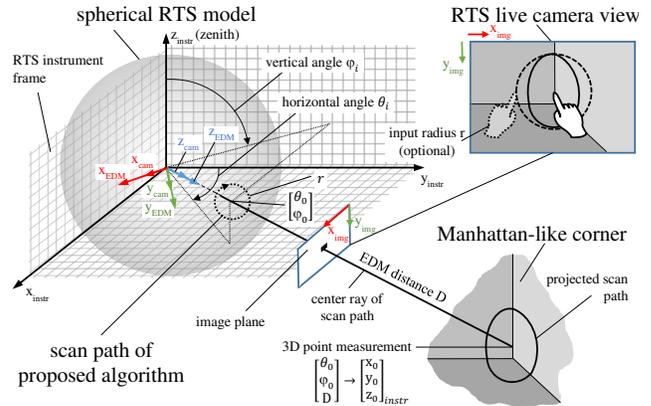


Fig. 1: Idealized geometric model of a calibrated RTS as used by Klug *et al.* [2]. The spherical coordinate frame of the RTS measurements, the EDM frame and the camera coordinate frame are aligned in this particular model.

In this work, we address the problem of semi-automatic reflectorless registration of an RTS with respect to a polygonal 3D model for indoor environments. The registration is performed with a sparse 3D point cloud scan to keep the measurement duration to an acceptable level.

Our algorithm reduces the required user interaction, but retains an accurate registration without reflective targets. The proposed approach simplifies the registration and relaxes the accuracy constraints for initial measurements. We avoid computationally expensive calculations to allow real-time implementation on embedded systems and mobile devices.

## II. RELATED WORK

Without the claim of completeness, we provide a review of published work about reflectorless RTS and model registration.

Uren and Price [1] provide an introduction to basic surveying methods and devices, including the design and handling of RTS, error sources and error correction methods. Coaker [3] analyzes reflectorless measurement methods of RTS in terms of accuracy, precision and reliability.

Klug *et al.* [2] apply line and plane approximations to measure man-made corner structures manually. However, RTS registration is not addressed in their work.

The iterative closest point (ICP) algorithm and its variants are widely used practical solutions to the object registration problem. The algorithm was first introduced by Chen and Medioni

[4], and Besl and McKay [5]. Mehdi [6] provides a comprehensive study of rigid body registration and error evaluation using the ICP algorithm. Tam *et al.* [7] and Bellekens *et al.* [8] provide a general introduction to point cloud registration; Salvi *et al.* [9] provide a review of range image registration methods. However, to the best of our knowledge, no complete system for semi-automatic reflectorless registration of an RTS with respect to a polygonal model has been proposed so far.

We describe robust initialization and ICP sample selection for the proposed problem and an optimization for the measurement order. In this work, we use plane correspondences from local Manhattan-like structures to initialize the registration flow. Hulik *et al.* [10] present a robust plane extraction algorithm based on the Hough transform [11]; Schnabel *et al.* [12] use the random sample consensus (RANSAC) framework for robust shape extraction. Bueno *et al.* [13] use geometric keypoint descriptors for the coarse registration of dense point clouds. Bosché [14], [15] uses point and plane correspondences in combination with ICP for the registration of terrestrial laser scans and CAD models. However, the low measurement rates of RTS require robust algorithms for sparse point clouds, whereas all methods discussed above are designed for dense point clouds and do not perform well with the desired point cloud sparsity.

Our system is closely related to the system proposed by Bosché [14], [15], but tailored towards RTS registration. The initialization of our algorithm is based on the plane estimation algorithm proposed by Nguyen *et al.* [16] to reduce the required user interaction and number of measurements; Nguyen *et al.* combine a mobile EDM with a real-time SLAM system to extract 3D models from sparse 3D point measurements. In particular, the authors apply graph-based point segmentation and expectation maximization (EM) plane fitting. We modified the proposed methods for RTS and realistic indoor scenarios, for which image-based segmentation is not applicable due to the frequently occurring low contrast regions.

### III. RTS REGISTRATION

Registering an RTS with respect to CAD model can be reduced to a generic 3D registration problem. Traditional 3D registration algorithms use three or more point correspondences to estimate the pose between two models [17]. The ICP algorithm and its variants are widely used for automatic object registration, but require carefully selected sample points and good initialization, especially when using sparse point clouds.

Our algorithm can determine the registration without dedicated control points by using local geometry for initial pose estimation and an ICP based sampling and refinement step. Alternatively, the initial pose can be roughly defined by the operator using the CAD model.

Fig. 2 compares the manual registration workflow with our assisted approach, which uses local geometry instead of control points. When using manual registration, the user measures points with the RTS and specifies correspondences in the CAD model. When using the assisted method, the user positions the RTS, roughly specifies the initial pose in the CAD model,

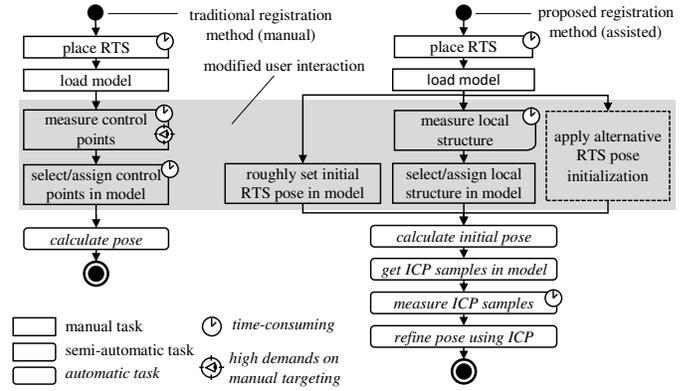


Fig. 2: Manual registration workflow (left) compared to the workflow of the proposed CAD based registration (right). The manual workflow requires visible and CAD-registered control points, whereas control points are optional for the proposed method.

and triggers the registration; measurement samples are taken automatically to refine the initial registration.

#### A. Manual RTS Registration

When using manual RTS registration, the user is responsible for measuring and assigning control points. The steps for this approach include: a) place the total station in the environment, b) load the model, c) create control points in the model, d) measure the control points, e) assign measured and model control points, and f) calculate the current pose of the total station with respect to the model frame. In the simplest case, this problem can be mathematically formalized as point cloud fitting problem with known point correspondences, using a six degree of freedom (DOF) Euclidean transformation.

Let  $\mathbf{x}'_i$  be a CAD point,  $\mathbf{x}_i$  be the corresponding measured 3D point, and  $\{\mathbf{R}, \mathbf{t}\}$  be the relative Euclidean transformation between the measurement frame and the CAD frame. Then, the relative transformation between the two frames is given by

$$\mathbf{x}'_i = \mathbf{R} \cdot \mathbf{x}_i + \mathbf{t} \quad \{\mathbf{R}, \mathbf{t}\} = \operatorname{argmin}_{\mathbf{R}, \mathbf{t}} \sum_{i=1}^m \|\mathbf{R} \cdot \mathbf{x}_i + \mathbf{t} - \mathbf{x}'_i\|^2 \quad (1)$$

with the point correspondence count  $m \geq 3$ . The problem is known as *best rigid body transformation in least square sense* with a well-established closed form solution by Horn [18].

#### B. Assisted RTS Registration

The proposed assisted RTS registration is an iterative method, which requires good initial values for the rotation  $\mathbf{R}$  and translation  $\mathbf{t}$  for fast and robust registration. Our method supports the user in defining an initial pose and then refines the result automatically. The most simple, yet robust method is defining the initial pose manually in the CAD model.

Another pose initialization method is measuring a local corner structure and matching its pose with respect to a selected CAD region. An RTS provides accurate single point measurements with a low sampling rate. Therefore, we explicitly avoid dense point cloud scans to keep the registration reasonable fast. While the ICP is an established solution for the registration of point clouds and meshes, the sparsity of the

---

**Algorithm 1:** Peak detection algorithm with non-maximum suppression of successive points. The non-maximum suppression in this algorithm is limited to a single subsequent edge. The angle threshold is set to  $\alpha_{thr} = \frac{\pi}{4}$  (experimentally determined).

---

```

1 function peakfind
  Input :  $angles := \{\tilde{\alpha}_i\}$ 
  Output:  $peak\_indices$ 
2 state  $s := NO\_PLATEAU$ 
3 foreach  $i$  in  $\{1, N_s\}$  do
4   if  $s = NO\_PLATEAU$  AND  $\tilde{\alpha}_i > \alpha_{thr}$  then
5     APPEND  $i$  to  $peak\_indices$ 
6      $s := PLATEAU$ 
7   else
8      $s := NO\_PLATEAU$ 
9 return  $peak\_indices$ 

```

---

measurements is critical for ICP based pose estimation without providing further user input<sup>1</sup>. Multiple initial pose candidates would be required for a robust pose estimation, which comes at the expense of additional computation time.

Our algorithm is based on the work of Nguyen *et al.* [16], who use sparse 3D point clouds, graph based segmentation and EM for robust plane detection. However, our setup requires certain modifications: 1) The mobile EDM is replaced by an RTS. Hence, no SLAM system is required for connecting multiple measurements. 2) 3D points can be measured with higher accuracy and with a predefined scan order. This reduces the optimization complexity and allows for a more robust algorithm. 3) Realistic indoor scenarios and usage on construction sites demand high stability with low contrast images. Hence, the point segmentation proposed by the authors cannot be used. 4) Pose estimation of a mesh and a point cloud is not part of the original algorithm and must be addressed separately.

In the following, we provide modifications to the method proposed by Nguyen *et al.*, which lead to a simple, yet robust method for Manhattan-like corner pose estimation for RTS in low contrast scenarios. The RTS camera is only used to steer the RTS to the area of interest, but not for segmentation. The camera can be removed completely for setups with different hardware control interfaces.

### C. Local Manhattan-Like Corner Estimation With Ordered Sparse Point Clouds

First, the proposed algorithm defines a measurement path for the RTS for creating a set of ordered 3D point measurements. Using the measured point set, plane detection is applied to classify the sparse scan of a local Manhattan-like structure into following three categories: single wall, 3D edge with two visible planes, or 3D corner with three visible planes. In the current work, only the latter is accepted as valid measurement. The complete plane extraction algorithm consists of following steps: 1) Automatically measure the local area of a target using

<sup>1</sup>We assume an EDM update rate of 0.5...20 Hz for repeated measurements of the same target, but lower update rates for different targets [1], [19]. We assume an acceptable sample count of  $N_s \leq 40$  points (experimentally determined, indoor).

---

**Algorithm 2:** Simple greedy non-maximum suppression based on the size of the point groups. List boundary handling is omitted for reasons of readability.

---

```

1 function nonmaximumsup
  Input :  $peak\_indices$ 
  Output:  $filtered\_peak\_indices$ 
2 state  $s = NO\_PLATEAU$ 
3 var  $idx_{prev} = 0$ 
4 foreach  $idx$  in  $peak\_indices$  do
5   if  $(idx - idx_{prev}) > 3$  then
6     APPEND  $idx$  to  $filtered\_peak\_indices$ 
7      $idx_{prev} = idx$ 
8 return  $filtered\_peak\_indices$ 

```

---

a circular movement, 2) create initial point set groups, 3) refine groups using EM, and 4) extract planes using the RANSAC framework for each point group. Fig. 1 shows the circular scan path, the projected scan path at the measured corner, and an exemplary RTS control user interface (UI) for the algorithm.

Discrete sample points define a sparse point cloud around the current target. The points are recorded by an RTS using a circular EDM motion. The set of RTS angle control parameters  $[\theta_i \ \varphi_i]^T$  is defined by

$$\begin{bmatrix} \theta_i \\ \varphi_i \end{bmatrix} = \begin{bmatrix} \theta_0 \\ \varphi_0 \end{bmatrix} + r \cdot \begin{bmatrix} \cos(\omega_i) \\ -\sin(\omega_i) \end{bmatrix}, \quad i = 1 \dots N_s, \quad \omega_i = i \frac{2\pi}{N_s + 1} \quad (2)$$

where  $\theta_0$  and  $\varphi_0$  are the horizontal and vertical angle of the center of the measurement,  $\theta_i$  and  $\varphi_i$  are the spherical coordinates of sample point  $\mathbf{x}_i$ , and  $N_s$  is the sample count. By steering the RTS approximately to the corner, the measurement center angles  $\theta_0$  and  $\varphi_0$  are implicitly defined by the current RTS pose. The radius  $r$  defines the swing of the spherical RTS control parameters for the local measurement. When defined visually, the angle between the center ray and the back-projected ray of a user defined image point is used. However,  $r$  can also be derived from a predefined spherical bounding box at the corner, using an initial measured EDM distance, which further reduces the required user input.

The proposed classification accepts simple Manhattan-like structures with one, two or three main planar areas.

The point cloud, consisting of  $N_s$  measured points  $\mathbf{x}_i$ , is interpreted as closed polygon  $K = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{N_s}, \mathbf{x}_1\}$  for pre-processing and initial point grouping. Besides image-based segmentation, Nguyen *et al.* use EDM distances for initial point grouping. In contrast, RTS provide not only distances, but 3D point measurements, which we use for segmentation.

Let the interior angle  $\alpha_i$  be the angle between two successive measured points, interpreted as 3D polygon vertices. Then,  $\alpha_i$  is given by

$$\alpha_i = \arccos(\mathbf{v}_i \cdot \mathbf{v}_{i-1}) \quad \mathbf{v}_i = \frac{\mathbf{x}_{i+1} - \mathbf{x}_i}{\|\mathbf{x}_{i+1} - \mathbf{x}_i\|} \quad (3)$$

where  $\mathbf{v}_i$  is the direction from vertex  $\mathbf{x}_i$  to vertex  $\mathbf{x}_{i+1}$ . To identify initial coplanar point groups, we subtract a systematic angle  $\alpha_s$  according to

$$\tilde{\alpha}_i = \max(\{|\alpha_i| - \alpha_s, 0\}) \quad \alpha_s = \frac{2 \cdot \pi}{N_s + 1} \quad (4)$$

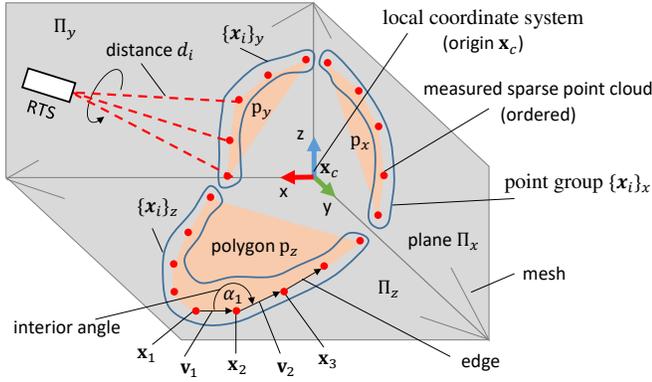


Fig. 3: Manhattan-like structure detection. Once detected, the initial RTS registration with respect to the CAD model is done by aligning a local coordinate system of the measured and selected structure. The local coordinate axes are automatically extracted from both, the CAD model and the measured point cloud.

The systematic angle  $\alpha_s$  describes the angle between two consecutive edges for a perpendicularly measured, planar target. We perform peak detection of the absolute values of the remaining angles with downstream non-maximum suppression of the detected plateaus. The peak detection algorithm is shown in Alg. 1. After detection, groups with less than three points are joined using the greedy iterative algorithm shown in Alg. 2. EM optimization is applied for robust plane detection. In the Expectation step, measurements are assigned to coplanar point groups. However, the sorted input allows for k-means clustering in the Expectation step rather than using Gaussian Mixture Models as proposed by Nguyen *et al.* [16]. Let the plane  $\Pi_j$  be the estimated geometric model of point group  $j$ . The initial likelihood  $p$  of a homogeneous measurement point  $\tilde{\mathbf{x}}_i$  to belong to the plane  $\Pi_j$  is obtained from Alg. 2. In the subsequent iterations, the likelihood is updated using

$$p(\tilde{\mathbf{x}}_i \in \Pi_j | \Pi_j) = \begin{cases} 1, & j = \underset{k}{\operatorname{argmin}}(\Pi_k^T \cdot \tilde{\mathbf{x}}_i), |j - m| \leq 1 \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where  $\Pi_k$  is a  $4 \times 1$  vector which contains the plane parameters for the co-planar point group  $k$ . In the previous iteration, point  $\mathbf{x}_i$  was assigned to plane  $\Pi_m$ . In the Maximization step, plane fitting and merging of similar planes is applied. However, in all algorithm steps, only connected groups are considered. This reduces the effects of invalid inter-class point assignments and invalid plane merging. The benefits of the ordered point clouds can be compared to the benefits of *organized point clouds* over *unorganized point clouds*, as discussed by Trevor *et al.* [20]. Fig. 3 shows an exemplary measurement. Different representations of the Manhattan-like structure are used for the individual algorithm steps. Hence, the local structure is represented as single mesh, connected planes, polygons, or grouped points.

From the proposed plane detection, three planes are selected and classified as a corner structure. First, the most prominent plane  $\Pi_i$  is selected, using the inlier count  $N(i)$  of the plane estimation. Then, the two remaining planes  $\Pi_j$  and  $\Pi_k$  are

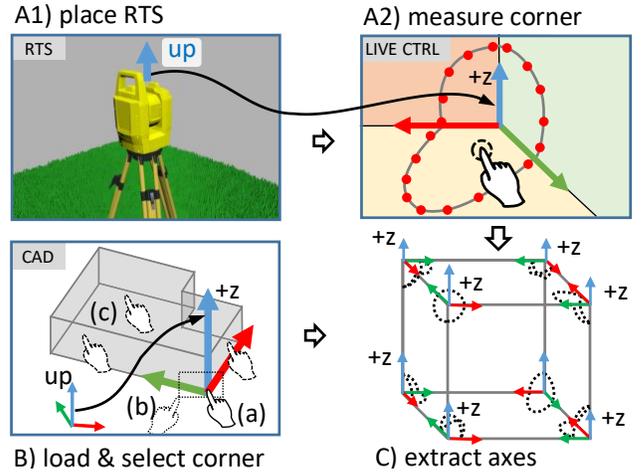


Fig. 4: Local coordinate axes extraction and assignment. The up-vector of the scene and the up-vector of the RTS are used to resolve the ambiguity of the direction and label of the z-axis. Selection methods: (a) select corner by vertex; (b) select corner area; (c) select three surfaces. The task flow A1-A2-C can be executed parallel to the task flow B-C to reduce the idle time of the user.

selected by maximizing the span and inlier count of the three planes. Hence, the optimization problem is given by:

$$\left. \begin{aligned} i &= \underset{i}{\operatorname{argmax}}(N(i)), \{j, k\} = \underset{j, k}{\operatorname{argmax}}(C_c), \text{ subject to:} \\ C_c &= c_1 C_1 + c_2 C_2 \quad 0 \leq \{c_1, c_2\} \leq 1 \\ C_1 &= (\mathbf{n}_i \times \mathbf{n}_j) \cdot \mathbf{n}_k \quad C_2 = \frac{N(i) + N(j) + N(k)}{N_s} \\ i &\neq j \neq k \quad 3 \leq \{N(i), N(j), N(k)\} \leq N_s \end{aligned} \right\} \quad (6)$$

where  $N(l)$  is the inlier count for some plane  $l$ ;  $C_c$  is the fitness function, defined as weighted sum of  $C_1$  and  $C_2$ ;  $C_1$  maximizes the span, and  $C_2$  maximizes the inlier count of the planes  $\Pi_i$ ,  $\Pi_j$  and  $\Pi_k$ . The weights  $c_1$  and  $c_2$  were experimentally determined and were set to  $c_1 = c_2 = 0.5$ .

For subsequent alignment, a right-handed local coordinate system as shown in Fig. 4 is defined, which is called *local corner frame* in the following. The origin  $\mathbf{x}_c$  is given by the intersection of the planes, the axes are given by orthonormalizing the plane normals  $\mathbf{n}_i$ ,  $\mathbf{n}_j$  and  $\mathbf{n}_k$ . For the economy of notation, the plane normals and the related orthonormal vectors are denoted by the same variables<sup>2</sup>.

The initial RTS pose can be estimated by relating the local corner frame of the selected CAD model and the measured corner. At this state, only the orientations of the axes are known, but not the labels and directions. Hence, a pose ambiguity with 24 possibilities must be resolved. The correspondences between axis labels and the extracted plane normals can either be selected manually or assigned automatically. Automatic assignment requires additional constraints; the up-vectors for

<sup>2</sup>While the interpretation of the plane intersection and normals as local coordinate system enforces orthonormalization, the latter is not mandatory for the proposed algorithm.



Fig. 5: Sampling problems. Sample distribution of standard uniform sampling method distribution highly depends on initial registration position.

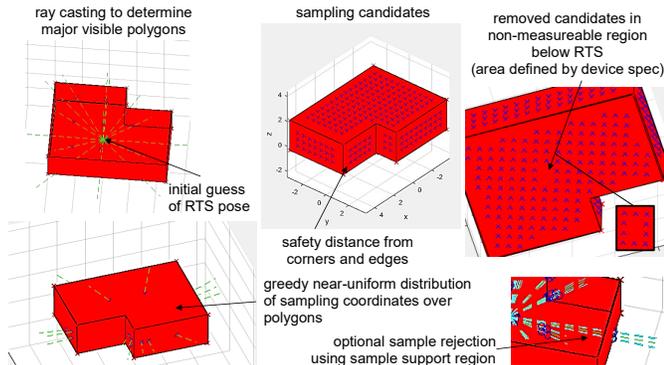


Fig. 6: Robust sample extraction using initial pose and visible CAD polygons, which are used to control the RTS.

both, the CAD model and the RTS must be approximately known. As RTS require upright operation, and surveying models usually contain the information about the up-vector, no additional user input is required. Without loss of generality, we assume that the up-directions of the CAD model and the RTS are both aligned with the z-axis of the particular coordinate system. In the proposed setup, Manhattan-like corners with three planes usually have a prominent z-plane. Thus, the z-plane can be selected from the previous detected corner planes:

$$i_z = \underset{l}{\operatorname{argmax}} |\arccos(\mathbf{z} \cdot \mathbf{n}_l)| \quad k \in \{i, j, k\} \quad (7)$$

where  $i_z$  is the index of the z-plane, and  $\mathbf{n}_l$  is the plane normal of the extracted planes  $\Pi_i$ ,  $\Pi_j$  or  $\Pi_k$ . Eqn. 7 can be applied to both, the selected CAD corner and the measured corner. This reduces the remaining alignment ambiguity to four possibilities as shown in Fig. 4.

The measured samples are mainly distributed in one octant of the local corner frame. By enforcing that the majority of the measured samples have to be in the positive half spaces of the local corner frame, the remaining ambiguity can be resolved for the measured corner. Similarly, the alignment of the local corner frame of the selected CAD area can be fixed, using each vertex of the selection as vote for a half space. In case of ambiguous votes, additional user input is required.

Finally, the measured and selected local corner frame can be aligned using Eqn. 1. ICP can be applied to the selected corner and the measured samples to refine the initial RTS pose.

#### D. Automatic Pose Refinement Using Additional Samples

The estimated RTS pose of the proposed initialization method relies on local measurements only. Hence, measurement un-

certainties and discrepancies between model and reality may lead to an unacceptable registration error. In the following, we discuss the individual steps of an iterative pose refinement method, which uses the previous results as initialization. In general, the proposed method automatically 1) selects robust and well-distributed sample points for further pose refinement, 2) calculates the minimum movement for the RTS to measure the sample points, 3) measures the selected samples, and 4) refines the registration pose.

#### E. Robust Sampling

For pose refinement, additional sampling points are required. The sampling points have to be visible from the current RTS position. At this state, only the RTS pose from the initial registration can be used to pick sampling candidates from potential visible regions of the CAD model. Without prior knowledge about the scene, the following sampling strategies could be applied: a) random or uniform sampling within the bounding box of the CAD model, b) random or uniform sampling on a unit sphere around the initial RTS pose, c) random or uniform sampling of points of the surface of the CAD model. Random sampling without guidance would lead to an unacceptable measurement time, even with a modern RTS. Furthermore, without incorporating the initial pose, the spatial distribution of random or uniform distributed samples highly depends on the scene setup, as shown in Fig. 5. In this section, we provide a better sampling strategy by incorporating the CAD model and the initial RTS pose. First, we create a potentially visibility set of polygons. This step extracts polygons of the CAD model that are visible from the initial RTS pose. Connected, co-planar polygons are joined together for further sampling.

Then, we define sampling candidates as uniformly distributed point sets for each polygon. Indoor environments usually have joints, skirting boards and similar details, which are not presented in CAD models. Juretzko [21] describes problems with edge and corner measurements in reflectorless RTS mode. Klug *et al.* [2] substantiate the statements of Juretzko by analyzing measurement errors of corners. Therefore, sampling candidates near polygon edges are excluded explicitly. Fig. 6 shows sampling candidates, which are calculated, filtered and measured by the pose refinement step.

Then, the proposed sampling method uses a greedy point-picking algorithm, which iterates over polygons to select well-distributed samples. The procedure terminates after a predefined number of samples. Alg. 3 shows the pseudo-code for picking sample points.

Finally, the extracted sampling points are converted to spherical coordinate angles for controlling the RTS.

#### F. Efficient Sample Measurement Order

Measuring random samples with an RTS without ordering would lead to unnecessary RTS movements. Proper definition of the scan order decreases the power consumption of the device, increases laser safety on construction side, and lowers the overall measurement time. Smooth measurement movements also increase the aesthetic aspect of the RTS operation.

**Algorithm 3:** Greedy point-picking algorithm. Points are selected from each polygon group. Connected co-planar CAD areas are joined before point-picking.

```

1 function select_sample_points;
  Input : sample count threshold  $sample\_cnt$  and sample
           candidates with reference to polygons  $p_{cand}$ 
  Output: selected sample points  $p_{sel}$ 
2 list  $p_{sel} := \{\}$ 
3 foreach  $poly$  in  $polygons$  do
4    $p := RANDOM\_SAMPLE(poly, p_{cand})$ 
5   if  $p = \{\}$  then
6     // no more candidates for  $poly$ 
7     REMOVE  $poly$  from  $polygons$ 
8   else
9     APPEND  $p$  to  $p_{sel}$ 
10    REMOVE  $p$  from  $p_{cand}$ 
11    if  $COUNT(p_{sel}) = sample\_cnt$  then
12      BREAK
13 return  $p_{sel}$ 

```

Finding the minimum distance between multiple stops is known as the travelling salesman problem (TSP). One common formalization is based on linear integer programming (ILP)

$$x_{ij} = \begin{cases} 1 & \text{connection between point } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where  $x_{ij}$  describes the movement from point  $i$  to point  $j$ . The shortest distance between two points on a sphere can be expressed in spherical coordinates by the *great-circle distance*

$$\Delta\alpha = \arccos(\sin\theta_i \sin\theta_j + \cos\theta_i \cos\theta_j \cos(\varphi_i - \varphi_j)) \quad (9)$$

$$d_{ij} = r \cdot \Delta\alpha \quad r = 1 \quad (10)$$

where  $\Delta\alpha$  is the central angle between points  $i$  and  $j$ , and  $d_{ij}$  is the shortest distance between them and  $r$  the sphere radius.

The ILP for  $N_s$  measurement points can be written as:

$$\left. \begin{aligned} & \min_{x_{ij}} \sum_1^{N_s} \sum_1^{N_s} d_{ij} x_{ij}, \text{ subject to:} \\ & x_{ij} \in \{0, 1\} \quad 1 \leq \{i, j\} \leq N_s \\ & \sum_{i=1, i \neq j}^{N_s} x_{ij} = 1 \quad \sum_{j=1, j \neq i}^{N_s} x_{ij} = 1 \end{aligned} \right\} \quad (11)$$

which leads to a smooth sampling path and decreases the overall sampling time [22]. Note that sub-tours are explicitly allowed for performance reasons. Fig. 7 visualizes the optimization of an exemplary set of points to measure with and without sub-routines.

### G. ICP Pose Refinement

The problem of refining the initial RTS pose with respect to the CAD model is a *pairwise surface-based registration* problem. A well-studied solution is the ICP algorithm. In this work, we used the ICP algorithm with measured points, model surfaces, and the point-plane distance metric as described by Mehdi [6].

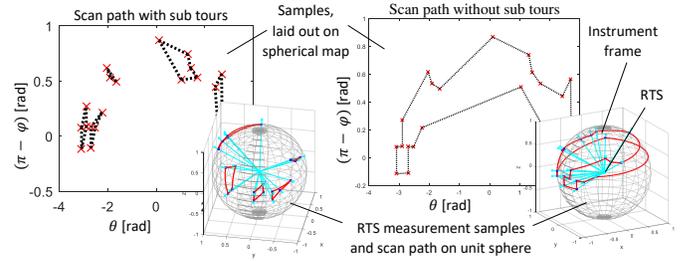


Fig. 7: Scan path optimization of an exemplary set of samples, with and without sub-routines.

## IV. EXPERIMENTS

The proposed method is mainly designed for indoor scenarios, where Manhattan-like corners are present and measurable. Physical evaluation setups contain different error sources, which contribute to the overall measurement uncertainty for both, the reference result and the algorithm under test. In general, the uncertainty in the result of a measurement or calculation that consists of multiple components is known as *combined standard uncertainty*. The estimation, propagation and assessment of the measured physical quantities and the related uncertainties are described in the JCGM 100:2008 Guide to the Expression of Uncertainty (GUM) [23].

Without controlling the properties of the physical setup, the uncertainty of the result of a measurement or calculation cannot be separated reliably into effects of different error sources. This renders a physical evaluation approach critical for the in-depth analysis of the proposed algorithm. For this reason, we used a simulated RTS setup in this work<sup>3</sup>. Hence, effects of sensor noise and as-built discrepancies with respect to the registration results could be analyzed.

In the following, we provide the description of the test data and the test coverage for the proposed algorithm, the simulated RTS measurement setup and the variations of the uncertainty sources for the RTS sensors and the CAD models.

### A. Test Setup and Test Coverage

We altered several test parameters for better test coverage as shown in the test taxonomy in Tab. I. In particular, we compared different registration methods, different levels of as-built discrepancies, and different sensor noise settings. We report all registration results with and without subsequent ICP refinement for better comparison of intermediate results. For ICP, we used 15 point-plane correspondences.

The RTS control parameters and the reference result for the registration were automatically generated for each test. The CAD models and their variants were generated semi-automatically. The model variants were used to simulate as-built discrepancies. We compared the reference RTS registration with the manual registration method and with the assisted registration method. For manual registration, we used three point correspondences for each test. We also applied the ICP approach to the manual registration for better comparison;

<sup>3</sup>An early version of the Unity3D-based RTS simulator was successfully used by Klug *et al.* [2] for the design of interactive RTS algorithms.

TABLE I: Test set taxonomy and label encoding; test set label format:  $\{P\_D\_N\}$ . Example:  $c\_d00\_n0$  denotes the test set that uses the corner estimation method as initialization, no as-built discrepancies and no sensor noise.

Name	Variable	Value	Description
Initialization method	P	p c	3 points Corner
Refinement method			None ICP
As-built discrepancy	D	d00 d15 d30 dco (cmp)	None 15mm Curvature 30mm Curvature Cutouts (combined test results)
Noise	N	n0 n1	None Measurement uncertainty

TABLE II: Sensor noise settings, modeled as measurement uncertainties, enumerated by the test set label format variable  $N$ .

Label	Description	$u_1(d)$	$u_2(d)$	$u_1(\alpha)$	$u_2(\alpha)$
n0	Without noise	0	0	0	0
n1	With noise	$0.75E-3m$	$10E-6m$	$\frac{5}{3600}, \frac{2\pi}{360}$ rad	0

here, we showed that the corner estimation of the workflow proposed in Fig. 2 can easily be replaced by any other initialization method, like using three point correspondences.

The individual test sets analyze different as-built discrepancies. We analyzed 1) no as-built discrepancies, 2) as-built discrepancies at corner with 15mm and 30mm curvature radius, and 3) as-built discrepancies at faces with missing cutouts. In case of as-built discrepancies, the CAD model did not have the same detail level as the reality. We simulated the scenarios with detailed CAD models, but used the simplified CAD model in our registration algorithm. Furthermore, we simulated measurement uncertainties for sensors and targeting, using the settings given in Tab. II. The EDM measurement uncertainty  $u_c(d)$  and the angle and targeting uncertainty  $u_c(\alpha)$  were simulated by

$$u_c(x) \approx \sqrt{u_1(x) + (xu_2(x))^2} \quad x \in \{d, \alpha\} \quad (12)$$

where  $u_c(x)$  is the simulated standard uncertainty with normal distribution and zero mean;  $u_1(x)$  is the additive and  $u_2(x)$  is the proportional uncertainty component;  $u_c(\alpha)$  is the angle sensor and targeting uncertainty, which is assumed to be equal for horizontal and vertical angle.

The steps required for test data generation are shown in Fig. 8. During evaluation, the registration algorithm and the reference simulation can use different variants of the same CAD model for analyzing effects of as-built discrepancies. We used detailed models for simulation and models with as-built discrepancies as input for the discussed algorithms. The simulated as-built discrepancies include missing CAD model details, such as edge curvature, windows and door cutouts.

The error metric used in this work is given as follows: Let the registration error of the RTS with respect to the CAD model be the difference in angle and position between the true and the calculated RTS frame. Then, the position error  $e_t$

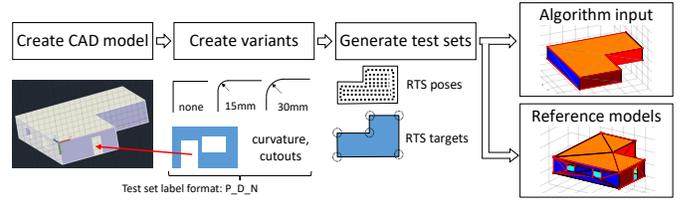


Fig. 8: Test data generation, using single-source for different CAD model variants. As-built variants include fillets and curvature at edges and missing details like door and window cutouts.

and the rotation error  $e_r$  for test set  $j$  are given by

$$e_t = \sqrt{\sum_{i=1}^N \frac{\|\mathbf{y}_i - \mathbf{y}'_i\|^2}{N}} \quad e_r = \sqrt{\sum_{i=1}^N \frac{(\Delta\alpha_i)^2}{N}} \quad (13)$$

where  $N$  is the test count of test set  $j$ ;  $i$  is the test index within the test set;  $\mathbf{y}_i$  and  $\mathbf{y}'_i$  are the true and the estimated RTS position, respectively;  $\Delta\alpha_i$  is the minimum rotation between the true and the estimated RTS pose. The minimum rotation is given by the relative rotation between the calculated and the true RTS frame, using the rotational part of the corresponding axis-angle representation. In addition, we calculated the standard uncertainties  $u_c(e_t)$  and  $u_c(e_r)$  according to GUM [23].

RTS sensor simulation with Unity3D uses single-precision floating-point arithmetic and causes outliers in the test results due to round-off effects and EDM ray-casting misses. We used the median absolute deviation to reduce the influence of outliers [2], [24]. In particular, we applied the Matlab function *isoutlier* with default parameters.

Each test set contains 130 tests. On average, we removed  $\approx 11\%$  outliers from each set. Test set labels are defined in Tab. I. In addition, we defined combined test sets for the as-built discrepancies for better comparison. In particular, we applied Eqn. 13 on the collected values for  $\|\mathbf{y}_i - \mathbf{y}'_i\|$  and  $\Delta\alpha_i$  of the different as-built discrepancy test sets.

## B. Registration Results

The results presented in Tab. III and Tab. IV show the benefits of using ICP as a downstream step for RTS registration algorithms. In all cases, we observed an decrease in error and uncertainty when using ICP. In addition, the proposed initialization method reduces the required user interaction for registration. For special setups, camera-based targeting can be avoided at all. However, the majority of the test sets result in a better initial registration when using the manual method, which was to be expected. In particular, we assumed an idealized setup, for which the targeting uncertainty was reduced to the sensor noise of the RTS. User dependent variations have not been analyzed. All cases are affected by the numeric precision of the Unity3D-based sensor simulation, which is limited to single-precision floating-point arithmetic. As-built discrepancies of CAD models have high influence on the initialization methods. Hence, expert knowledge and experience in measuring natural targets cannot be replaced. However, currently, the traditional method does not include ICP, while it is an integral step for our proposed method.

TABLE III: Evaluation results without simulated sensor noise ( $n0$ ). Test set labels are defined in Tab. I.

Test Set	Initialization		ICP	
	$e_r$ [m]	$e_r$ [rad]	$e_r$ [m]	$e_r$ [rad]
p_n0_d00	8.89E-06 ± 7.30E-06	1.49E-06 ± 1.48E-06	1.03E-06 ± 6.98E-07	3.76E-07 ± 2.40E-07
c_n0_d00	3.56E-05 ± 2.86E-05	5.12E-06 ± 3.74E-06	1.27E-06 ± 8.66E-07	4.37E-07 ± 2.82E-07
p_n0_d15	1.83E-02 ± 8.58E-03	8.35E-04 ± 4.97E-04	1.37E-06 ± 1.01E-06	3.48E-07 ± 2.12E-07
c_n0_d15	1.45E-04 ± 2.94E-04	2.68E-04 ± 4.92E-04	1.30E-06 ± 8.94E-07	3.74E-07 ± 2.69E-07
p_n0_d30	3.63E-02 ± 2.06E-02	1.68E-03 ± 1.29E-03	1.20E-06 ± 7.12E-07	3.57E-07 ± 1.91E-07
c_n0_d30	7.22E-05 ± 7.87E-05	9.58E-06 ± 1.02E-05	1.36E-06 ± 9.02E-07	5.60E-07 ± 3.69E-07
p_n0_dco	1.01E-05 ± 9.28E-06	1.54E-06 ± 1.54E-06	1.11E-06 ± 6.66E-07	2.89E-07 ± 1.53E-07
c_n0_dco	3.35E-05 ± 2.38E-05	4.82E-06 ± 3.14E-06	1.02E-06 ± 6.17E-07	2.72E-07 ± 1.34E-07
p_n0_cmp	3.35E-03 ± 5.93E-03	1.30E-05 ± 3.53E-05	1.18E-06 ± 7.79E-07	3.36E-07 ± 1.94E-07
c_n0_cmp	4.73E-05 ± 4.46E-05	6.93E-06 ± 5.98E-06	1.23E-06 ± 8.12E-07	3.99E-07 ± 2.62E-07

TABLE IV: Evaluation results with simulated sensor noise ( $n1$ ).

Test Set	Initialization		ICP	
	$e_r$ [m]	$e_r$ [rad]	$e_r$ [m]	$e_r$ [rad]
p_n1_d00	1.40E-03 ± 1.02E-03	1.95E-04 ± 1.38E-04	5.96E-04 ± 2.88E-04	1.48E-04 ± 1.06E-04
c_n1_d00	2.22E-02 ± 1.59E-02	3.43E-03 ± 2.45E-03	6.85E-04 ± 4.02E-04	1.81E-04 ± 1.30E-04
p_n1_d15	1.83E-02 ± 8.91E-03	8.72E-04 ± 5.09E-04	7.14E-04 ± 3.95E-04	1.08E-04 ± 5.44E-05
c_n1_d15	4.67E-02 ± 3.11E-02	6.70E-03 ± 4.96E-03	6.54E-04 ± 3.47E-04	1.58E-04 ± 1.15E-04
p_n1_d30	3.88E-02 ± 2.25E-02	1.72E-03 ± 1.33E-03	6.51E-04 ± 3.21E-04	1.17E-04 ± 6.32E-05
c_n1_d30	3.23E-02 ± 2.55E-02	4.51E-03 ± 3.58E-03	8.23E-04 ± 4.98E-04	2.04E-04 ± 1.77E-04
p_n1_dco	1.71E-03 ± 1.19E-03	2.54E-04 ± 1.72E-04	5.81E-04 ± 3.06E-04	1.15E-04 ± 5.93E-05
c_n1_dco	2.39E-02 ± 1.47E-02	3.47E-03 ± 2.24E-03	6.54E-04 ± 3.48E-04	1.23E-04 ± 8.99E-05
p_n1_cmp	7.43E-03 ± 8.66E-03	4.60E-04 ± 4.22E-04	6.31E-04 ± 3.25E-04	1.16E-04 ± 6.37E-05
c_n1_cmp	2.96E-02 ± 2.12E-02	4.11E-03 ± 3.01E-03	6.94E-04 ± 3.92E-04	1.54E-04 ± 1.12E-04

## V. CONCLUSION

In this work, we introduced a semi-automatic method for registration of RTS with respect to a CAD model. By using different methods for initial registration, we show the flexibility of the proposed algorithm. We explicitly avoided computationally complex methods to limit the power consumption and runtime to an acceptable level.

We proposed a robust Manhattan-like corner estimation method for ICP initialization. Alternative methods, which differ in the amount of user interaction and reliability of the initial registration, like single wall initialization or Manhattan-like structures with two walls, could be developed. Shape-priors could be extracted from the CAD selection. While this might be more robust in terms of detection, the added causal dependency would increase the overall duration of the registration workflow. In the current work, the ordered point cloud of the initial measurements was interpreted as 3D polygon, and the interior angles of the polygon edges were used for clustering. Alternative methods could be analyzed regarding the reliability of the initial clustering.

We used three point correspondences as reference registration method. One next step is to generalize point correspondences to point-and-direction correspondences, which covers a broader range of model registration methods. Furthermore, effects from systematic measurement errors like scale differences between current measurements and the CAD model can be analyzed or compensated.

The analysis of computer vision-algorithms and user dependent variations are beyond the scope of this work. User studies with a physical setup could be used in addition to the proposed analysis for additional insights and a more practical comparison of the proposed algorithms.

The proposed sample candidate-picking algorithm highly relies on the level of detail of the CAD model. In-cooperating further information, such as saliency regions could further decrease the uncertainty of the RTS registration. In particular, different sampling strategies on a spherical saliency map could be applied.

## REFERENCES

- [1] J. Uren and B. Price, *Surveying for Engineers*. Basingstoke England New York: Palgrave Macmillan, 2010.
- [2] C. Klug, D. Schmalstieg, and C. Arth, "Measuring human-made corner structures with a robotic total station using support points, lines and planes," in *VISAPP*. SciTePress, 2017, pp. 17–27.
- [3] L. H. Coaker, "Reflectorless total station measurements and their accuracy, precision and reliability," B.S. Thesis, University of Southern Queensland, 2009.
- [4] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145–155, Apr 1992.
- [5] P. J. Besl and H. D. McKay, "A method for registration of 3-d shapes," *PAMI*, vol. 14, no. 2, pp. 239–256, Feb 1992.
- [6] H. M. Mehdi, "New algorithms in rigid-body registration and estimation of registration accuracy," *Department of Electrical and Computer Engineering*, 2008.
- [7] G. K. L. Tam, Z. Cheng, Y. Lai, F. C. Langbein, Y. Liu, D. Marshall, R. R. Martin, X. Sun, and P. L. Rosin, "Registration of 3D point clouds and meshes: A survey from rigid to nonrigid," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 7, pp. 1199–1217, 2013.
- [8] B. Bellekens, V. Spruyt, and M. Weyn, "A survey of rigid 3D point-cloud registration algorithms," *Ambient 2014, Fourth Int. Conf. Ambient Comput. Appl. Serv. Technol. 2014.*, no. c, pp. 8–13, 2014.
- [9] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, "A review of recent range image registration methods with accuracy evaluation," *Image Vis. Comput.*, vol. 25, pp. 578–596, 2007.
- [10] R. Hulik, M. Spanel, P. Smrz, and Z. Materna, "Continuous plane detection in point-cloud data based on 3D hough transform," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 86–97, 2014.
- [11] P. V. C. Hough, "Method and means for recognizing complex patterns," 1962, uS Patent 3,069,654.
- [12] R. Schnabel, R. Wahl, and R. Klein, "Efficient ransac for point-cloud shape detection," *Computer Graphics Forum*, vol. 26, no. 2, pp. 214–226, 2007.
- [13] M. Bueno, H. González-Jorge, J. Martínez-Sánchez, and H. Lorenzo, "Automatic point cloud coarse registration using geometric keypoint descriptors for indoor scenes," *Autom. Constr.*, vol. 81, pp. 134–148, 2017.
- [14] F. Bosché, "Automated recognition of 3D cad model objects in laser scans and calculation of as-built dimensions for dimensional compliance control in construction," *Adv. Eng. Informatics*, vol. 24, pp. 107–118, 2010.
- [15] F. Bosché, "Plane-based registration of construction laser scans with 3D/4D building models," *Adv. Eng. Informatics*, vol. 26, pp. 90–102, 2012.
- [16] T. Nguyen, R. Grasset, D. Schmalstieg, and G. Reitmayr, "Interactive syntactic modeling with a single-point laser range finder and camera," *ISMAR*, pp. 107–116, 2013.
- [17] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solutions of absolute orientation using orthonormal matrices," *J. Opt. Soc.*, vol. 5, pp. 1127–1135, 1988.
- [18] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America A*, vol. 4, no. 4, p. 629, 1987.
- [19] W. Lienhart, M. Ehrhart, and M. Grick, "High frequent total station measurements for the monitoring of bridge vibrations," *J. Appl. Geod.*, vol. 11, no. 1, pp. 1–8, 2017.
- [20] A. J. B. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," *Semant. Percept. Mapp. Explor.*, 2013.
- [21] M. Juretzko, "Reflektorlose Video-Tachymetrie - ein integrales Verfahren zur Erfassung geometrischer und visueller Informationen," Ph.D. dissertation, Ruhr-Universitt Bochum, 2004.
- [22] R. Vanderbei, *Linear Programming: Foundations and Extensions*, ser. International Series in Operations Research & Management Science. Springer, 2008.
- [23] "JCGM 100:2008 - Evaluation of Measurement Data - Guide to the Expression of Uncertainty in Measurement," Int. Organ. Stand. Geneva ISBN, Standard, 2008.
- [24] C. Leys, C. Ley, O. Klein, P. Bernard, and L. Licata, "Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median," *J. Exp. Soc. Psychol.*, vol. 49, no. 4, pp. 764–766, 2013.