CAD Priors for Accurate and Flexible Instance Reconstruction

Tolga Birdal\textsuperscript{1,2} \quad Slobodan Ilic\textsuperscript{1,2}
\textsuperscript{1} Technische Universität München, Germany
\textsuperscript{2} Siemens AG, Munich, Germany

Abstract

We present an efficient and automatic approach for accurate instance reconstruction of big 3D objects from multiple, unorganized and unstructured point clouds, in presence of dynamic clutter and occlusions. In contrast to conventional scanning, where the background is assumed to be rather static, we aim at handling dynamic clutter where the background drastically changes during object scanning. Currently, it is tedious to solve this problem with available methods unless the object of interest is first segmented out from the rest of the scene. We address the problem by assuming the availability of a prior CAD model, roughly resembling the object to be reconstructed. This assumption almost always holds in applications such as industrial inspection or reverse engineering. With aid of this prior acting as a proxy, we propose a fully enhanced pipeline, capable of automatically detecting and segmenting the object of interest from scenes and creating a pose graph, online, with linear complexity. This allows initial scan alignment to the CAD model space, which is then refined without the CAD constraint to fully recover a high fidelity 3D reconstruction, accurate up to the sensor noise level. We also contribute a novel object detection method, local implicit shape models (LISM) and give a fast verification scheme. We evaluate our method on multiple datasets, demonstrating the ability to accurately reconstruct objects from small sizes up to $125m^3$.

1. Introduction

3D reconstruction involves the recovery of digitized object or scene using a 2D/3D sensor, typically through multiple acquisition steps. From reverse engineering to industrial inspection, its applications are plentiful. Due to such wide use, from the early days of vision, it attracted significant attention of both research community and industry [24, 7]. Despite the huge demand, many marker-free approaches based solely on 3D data either involve acquisition of ordered scans\cite{31, 21}, or follow the de-facto standard pipeline\cite{19} in case of unordered scans. The former suffers from the requirements of redundant depth capture with large overlap and scenes with very little clutter or occlusions. Due to the volumetric nature of scan fusion, such techniques also do not scale well to large objects while retaining high precision. The latter exploits 3D keypoint matching of all scans to one another, alleviating the order constraint. Thanks to 3D descriptors, it could well operate in full 3D. Yet, matching of scans to each other is an $O(N^2)$ problem and prevents the methods from scaling to an arbitrary number of scans. In addition, neither of those can handle scenes with extensive dynamic clutter or occlusions.

Nowadays, with the capability of collecting high quality, large scale and big data, it is critical to offer automated solutions for providing highly accurate reconstructions regardless of the acquisition scenarios. In this paper, we tackle the problem of 3D instance reconstruction from a handful of unorganized point clouds, where the object of interest...
is large in size, texture-less, surrounded by significant dy-
namic background clutter and is viewed under occlusions.
Our method can handle scenes between which no single
transformation exists, i.e. the same objects appear in dif-
f erent locations, such as the one in Fig. 2. We also do
not impose any constraints on the order of acquisition. To
solve all of these problems simultaneously, we make use of
the reasonable assumption that a rough, nominal 3D CAD
model prior of the object-to-reconstruct is available before-
hand and propose a novel reconstruction pipeline. Such
assumption of a nominal prior is valid for many applica-
tions especially in industry, where the objective is to com-
pare the reconstruction to the designed CAD model. Even
for the cases where this model does not exist, one could al-
tways generate a rough, inaccurate mesh model with existing
methods, e.g. KinectFusion [31], to act as a prior. Note that,
due to manufacturing errors, sensor noise, damages and en-
vironmental factors, physical instances deviate significantly
from the CAD models and the end-goal is an automatic al-
grithm to accurately recover the particular instance of the
model. With the introduction of the prior model, we re-
f actor the standard 3D reconstruction procedures via multi-
fold contributions. We replace the scan-to-scan matching
with model-to-scan matching resulting in absolute poses for
each camera. Unlike the case in object instance detection
where false positives (FP) are tolerable, object instance re-
construction is easily jeopardized by the inclusion of a sin-
gle FP. Therefore, one of our goals is to suppress FP, even at
the expense of some true negatives. To achieve this, we con-
tribute a probabilistic Local Implicit Shape Model (LISM)
formulation for the object instance detection and pose esti-
mation, accompanied by a rigorous hypotheses verifi cation
to reject all wrong pose candidates. This matching is fol-
lowed by automatically segmenting out the points belong-
ing to the object and transforming them back onto the model
cordinate frame. Doing this for multiple views results in
roughly aligned partial scans in the CAD space. To fully
recover the exact object, the CAD prior is then discarded
(as it might cause undesired bias) and a global multi-view
refinement is conducted only to optimize the camera poses.

In global scan registration, the creation of a pose graph
indicating which scans can be registered to each other is re-
quired. The typical complexity of obtaining this graph is
$O(N^2)$, where all scans are matched to one another. We
profit from the CAD prior and contribute by automatically
computing this graph, in which only cameras sharing signif-
icant view overlap are linked. This reduces the complexity
to $O(N)$, and robustifies the whole pipeline.

The entire procedure is made ef cient so that large scans
are handled in reasonable time. Exhaustive evaluations
demonstrate high accuracy regardless of the size of the ob-
jects, clutter and occlusions.\footnote{Our suppl. video is under: https://youtu.be/KPA_8BNuOvg}

Figure 2: Dynamic Clutter and Occlusions on Mian Dataset [28]:
There is change in relative locations of the objects between scenes.
This easily fools the modern global registration algorithms [46].
We operate on the object level and circumvents this problem.

2. Prior Art

Arguably, the most wide-spread 3D reconstruction meth-
ods are KinectFusion [31] and its derivatives [9, 21, 37].
These methods have been successful in reconstructing small
isolated objects, but their application is not immediate when
the size increases, or clutter and occlusions are introduced
[32]. Due to extensive usage of signed-distance fields, they
are bound to depth images and a sequential acquisition.

Abundance of works exists in multi-view global align-
ment from multiple 3D unorganized point clouds [2, 10,
13, 12, 43, 47, 38, 17, 40, 22, 8, 12, 46]. These methods
assume the scans to be roughly initialized and reasonably
well-segmented. They, in general, handle slight synthetic
noise well enough, but they do not deal with cluttered and
occluded data. Another track of works try to overcome the
aforementioned constraints by using keypoint detectors and
matching descriptors in 3D scans. These methods operate
on a subset of points during matching. One of the pio-
nering works proposing a feature agnostic, automatic and
constraint-free algorithm is the graph based in-hand scan-
ing from D. Huber and M. Hebert [19]. The authors set a
baseline for this family of methods. Novatnack and Nishino
[33] developed a scale dependent descriptor for better ini-
tialization and fused it with [19] to assess the power of their
descriptor. Yet both of these relied on range image data.
Mian et. al. [28] proposed a tensor feature and a hashing
framework operating on meshes. Fantoni et al. [14] uses
3D keypoint matching as an initial stage of multiview align-
ment to bring the scans to a rough alignment. Zhu et. al.
[47, 49] as well as Liu and Yonghui [27] use genetic algo-
rithms to discover the matching scans and use this in global
alignment context. These stochastic schemes are correspon-
dence free but slow. Similar to [28], Zhu et. al. [48] de-
vises a local-to-global minimum spanning tree method to
align the scans. A majority of these automatic alignment
procedures suffer from increased worst case complexity of
$O(N^2)$, where $N$ is the number of views. Moreover, since
there is no integrated segmentation, the registration proce-
dure cannot handle clutter and occlusions.

Use of CAD models in reconstruction is not novel by
itself. Savarese et. al. [45] enrich the multiview reconstruc-
tion from 2D images with a CAD prior. Guney and Geiger
Figure 3: Proposed 3D reconstruction pipeline: Prior CAD model is trained to create the model representation. Input scenes are then parsed for the model pose. Pose estimates initialize a rough reconstruction, with segmentation and automatically computed pose graph. This is then further refined to the full reconstruction.

[18] use object knowledge to resolve the stereo ambiguities. Birdal et. al. use models in triangulation by registration [4]. Other works [26, 44, 50] use CAD prior to detect generic object classes. Our approach differs from all of those in the sense that we use proxy instance priors for initial alignment of scans and then operate directly on 3D points.

Salas et. al. [36] propose SLAM++ using object priors to constrain a SLAM system. Our work is differentiated from theirs in the sense that we perform instance reconstruction using the CAD prior, and not SLAM. In our setting, the background as well as the object is allowed to change dynamically between different acquisitions.

3. Method

Given a set of unstructured and unordered 3D scenes \( \{S_i\} \in \mathcal{S} \), we seek to find a set of transformations \( \{T_i\} \in SE(3) \) so as to stitch and reconstruct a global model \( \mathcal{S}_G \):

\[
\mathcal{S}_G = \bigcup_{i=1}^{N} (T_i \circ S_i)
\]

(1)

\( T \circ S \) applies transformation \( T \) to the scene \( S \). In this setting, both the transformations \( \{T_i^0\} \in SE(3) \) and the global model \( \mathcal{S}_G \) are unknown and we do not assume known initialization. Due to the lack of a common reference frame and apriori information about \( \{S_i\} \), obtaining the set \( \{T_i^0\} \) typically requires \( O(N^2) \) worst case complexity, where all the scene clouds are matched to one another to obtain the relative transformations aligning them. To better condition the problem and reduce its complexity, we introduce the supervision of a CAD proxy \( \mathcal{M} \) in form of a mesh model and re-write Eq. (1):

\[
\mathcal{S}_G = \bigcup_{i=1}^{N} \left( T_i^M \circ (S_i | \mathcal{M}) \right)
\]

(2)

where \( T_i^M \in SE(3) \) is the transformation from the scene to the model space, such that the segmented scene points \( \{S_i | \mathcal{M}\} \) come to the best agreement. To estimate \( \{T_i^M\} \), we follow a two stage technique. First, a rather approximate estimate \( \{\hat{T}_i^M\} \) is made by matching the CAD model to a single scene. Note that this time, the set \( \{\hat{T}_i^M\} \) can be computed in \( O(N) \), since it only requires CAD to scan alignment. However, because scene clouds suffer from partial visibility, noise and deviations w.r.t. the CAD model, the discovery of the pose of the model in the scans provides only rough initial transformations to the model coordinate frame. For this reason, as the final stage, the CAD prior is disregarded and the scans are globally refined, simultaneously. This lets us reconstruct configurations deviating significantly from the CAD prior.

Our procedure of multi-view refinement is similar to [14], where a global scheme for scan alignment is employed. Let \( S_1, \ldots S_M \) be the set of scans that are to be brought in alignment. To generalize and formalize the notation for registrations of all point clouds to each other, we maintain a directed pose graph in form of an adjacency matrix \( A \in \{0, 1\}^{M \times M} \), such that \( A(h, k) = 1 \) iff cloud \( S_h \) can be registered to cloud \( S_k \). Let \( \theta = (\theta_1, \ldots, \theta_M) \) be the absolute camera poses of each view. The alignment error between two clouds \( S_h \) and \( S_k \) then reads:

\[
E(\theta_h, \theta_k) = A(h, k) \sum_{i=1}^{N_h} \rho \left( \|d(p_i^h \circ p_i^k, \theta_h \circ q_i^k)\| \right)^2
\]

(3)

where \( \{p_i^h \rightarrow q_i^k\} \) are the \( N_h \) closest point correspondences obtained from the clouds \( S_h \) and \( S_k \). The point-to-plane distance \( d(\cdot, \cdot) \) is defined to be:

\[
d(p_i, q_i) = (Rp_i + t - q_i)^T n_i^q
\]

(4)

with \( n_i^q \) referring to the normal associated to point \( q_i \). \( R \) and \( t \) are the components of the pair transformation \( \{R \in SO(3), t \in \mathbb{R}^3\} \). The overall alignment error, which we want to minimize at this stage, is obtained by summing up the contribution of every pair of overlapping views:

\[
E(\theta) = \sum_{h=1}^{M} \sum_{k=1}^{M} A(h, k) \sum_{i=1}^{N_h} \rho \left( \|d(\theta_h \circ p_i^h, \theta_k \circ q_i^k)\| \right)^2
\]

(5)

where \( \rho \) is the robust estimator. The final absolute poses are the result of the minimization \( (\theta_1, ..., \theta_M) = \arg\min_{\theta} E(\theta) \), and align the \( M \) clouds in a least squares sense. In contrast to the pairwise registration error in Eq. (3), which has closed form solution for the relative transformation \( \theta \), there are no closed form solutions in the multiview setting. Therefore, we use a non-linear optimization procedure, Levenberg Marquardt. The rotations are parameterized with angle-axis representation \( (w \in \mathbb{R}^3, \phi) \). We constrain the frame with the highest number of points found on the CAD model to be static and update (solve for) the rest of the poses. In practice, this leads to faster convergence.
Note that, in contrast to the methods that exploit pairwise registration, our poses are absolute and do not suffer from drifts or tracking artifacts. We also do not require a convergence to absolute poses, which are usually obtained by the computation of a minimum spanning tree or shortest paths over the pose graph \([16, 19]\). This property eases the implementation and reduces errors, that are to be encountered in usual heuristics.

Due to the accuracy requirements, unlike \([14]\), we omit using distance transforms at this stage. We rather use speeded up KD-Trees to achieve exact nearest neighbors \([30]\). Since we optimize over the poses, and not over 3D points, the trees are built only once in the beginning and all closest point computations are done in the local coordinate frame of the view of interest. This is important for efficiency. For reasons of accuracy, we use analytical Jacobians. As cloud sizes become large, this optimization exhibits significant computational costs. This is why, a priori sampling plays a huge role, where we use \(20k\) to \(30k\) points per scan, distributed evenly in space.

To summarize, our key contribution lies in obtaining \(\{\mathcal{T}^M_i\}\) in a robust, efficient and accurate manner. We will now show how to compute the rough alignment \(\{\mathcal{T}^M_i\}\) and the pose graph (adjacency matrix) \(A\).

### 3.1. Locally Implicit Models for Estimating \(\{\mathcal{T}^M_i\}\)

While using any method, which is capable of handling 3D points, e.g. \([20, 42, 28, 41, 34]\), is possible, we mainly follow the Geometric Hashing of Drost et al. \([11]\) and Birdal and Ilic \([6]\) due to efficiency and robustness to clutter and occlusions. Yet, we introduce a more effective probabilistic formulation, inspired by the implicit shape models \([23]\).

**Model Description** In the first stage, we generate a pose invariant codebook encoding all possible semi-global structures that could be found on the CAD model. We represent this semi-global geometry via simple point pair features (PPF) of oriented point pairs \((\mathbf{m}_i, \mathbf{m}_j)\):

\[
\mathbf{f}_{ij} = (\|\mathbf{d}\|_2, \angle(\mathbf{n}_i, \mathbf{d}), \angle(\mathbf{n}_j, \mathbf{d}), \angle(\mathbf{n}_i, \mathbf{n}_j))
\]

where \(\mathbf{d} = \mathbf{m}_i - \mathbf{m}_j\), \(\mathbf{n}_i\) and \(\mathbf{n}_j\) are the surface normals at points \(\mathbf{m}_i\) and \(\mathbf{m}_j\). \(\angle(\cdot, \cdot)\) is the angle operator. The complete set of such features \(\mathbf{F} = \{\mathbf{f}_{ij}\}\) for the prior CAD model is collected and quantized to generate the codebook: \(\mathcal{F}\). We use our codebook to relate a feature \(\mathbf{f}\) (key) to a set of oriented references points \(\{(\mathbf{m}_i, \mathbf{m}_j)\}\) (stored in buckets) and build the global model description as an inverted file i.e. a hashtable \(\mathbf{H}\). Thus, each bucket in the codebook contains self-similar point pairs extracted from the CAD model. Whenever a pair from the scene is matched to one in the model, their normals at the reference points are aligned. Then, the full pose of the object can be obtained once the rotation angle \(\alpha\) around the normal is known. This can be done by aligning Local Coordinate Frames (LCF) constructed from matched pairs. Thus, instead of storing the full PPF, we store only this local parameterization \(\{(\mathbf{m}_r, \alpha)\}\) composed of the model reference point \(\mathbf{m}_r\) and rotation angle \(\alpha\). A pair correspondence resolves the full 6DOF pose and what is left is to retrieve the matching pairs \(\{s_r, s_i\}\) and \(\{\mathbf{m}_r, \mathbf{m}_i\}\). We now give a novel way to do this.

**Probabilistic Formulation** During detection, a new point cloud scene \(\mathbf{S}\) is encountered and downsampled to a set of points \(\mathbf{S}_D = \{\mathbf{s}_r\}\), some of which are assumed to lie on the object. The sampling also enforces spatial uniformity (see our suppl. material). We fix a reference point \(\mathbf{s}_r\) and pair it with all the other samples \(\{\mathbf{s}_i\}\). Each pair makes up a PPF \(\mathbf{f}_{ri}\). The original method \([11]\) associates \(\mathbf{f}_{ri}\) to a unique key and can not account for the quantization errors that inevitably happen due to the noise. To circumvent these quantization artifacts, resulting from the hard assignment in \([11]\), we quantize \(\mathbf{f}_{ri}\) to \(K\) different bins \((K > 1)\), activating different codebook entries as in ISM. This soft quantization results in possibly matching buckets \(\mathcal{F}_{ri} = \{\bar{\mathbf{f}}_{ri}, \bar{\mathbf{f}}_{rK}\}\). \(\bar{\mathbf{f}}_{ri}\) indexes the buckets of \(\mathbf{H}\), with weights \(p(\bar{\mathbf{f}}_{ri})\). For each matching bucket, we collect the valid interpretations \(p(\mathbf{m}, \alpha | \bar{\mathbf{f}})\), inversely proportional to the size of the bucket \(N_b\), denoting the probabilities of particular pose configuration, given the quantized feature. Formally:

\[
p(\mathbf{m}, \alpha | \mathbf{s}_r, \mathbf{s}_i) = p(\mathbf{m}, \alpha | \mathbf{f}) = \sum_k p(\mathbf{m}, \alpha | \bar{\mathbf{f}}_k)p(\bar{\mathbf{f}}_k | \mathbf{f})
\]

with \(p(\bar{\mathbf{f}}_k | \mathbf{f}) = \frac{1}{K}\) and \(p(\mathbf{m}, \alpha | \bar{\mathbf{f}}) = \frac{1}{b}\) being uniformly distributed. This probability is actually the prior on the PPF of the particular object and can be computed differently accounting for the nature of the object geometry using a suited distribution. At this point, the gathered pair representations for a particular scene reference point are sufficient to recover for the object pose. However, due to outliers, some of these matches will be erroneous. Therefore, a 2D voting scheme is employed, locally for each scene reference point \(s_r\). The voting space is composed of the alignment of the
The voting space is by no means a guarantee for shared overlap. Therefore, we introduce a rigorous hypothesis verification scheme. The effective verification requires fine registration, while efficient registration requires as few poses as possible. This creates a chicken and egg problem. We address this issue via a multi-level registration approach. In the first stage, sparsely sampled scan points are finely registered to the model using the efficient LM-ICP \[ \text{LM-ICP} \] variant of Iterative Closest Point (ICP) registration \[ \text{ICP} \]. We also build a 3D distance transform for fast nearest neighbor access. Our sparse LM-ICP requires only \( 1 \) ms per hypothesis, allowing us to verify all the hypotheses.

### Hypotheses Verification and Rejection

Devised matching theoretically generates a pose hypothesis for each scene reference point, which is assumed to be found on the model. There are typically \( \sim 400 \) to \( 1000 \) such points, reducing to \( 50 \) poses after the clustering, where the close-by poses are grouped together and averaged. Still, as many hypotheses as the number of clusters remain to be verified and the best pose is expected to be refined. In our problem of instance reconstruction it is critical that no false positive pose hypotheses survives. For this reason we introduce a rigorous hypothesis verification scheme. The effective verification requires fine registration, while efficient registration requires as few poses as possible. This creates a chicken and egg problem. We address this issue via a multi-level registration approach. In the first stage, sparsely sampled scan points are finely registered to the model using the efficient LM-ICP \[ \text{LM-ICP} \] variant of Iterative Closest Point (ICP) registration \[ \text{ICP} \]. We also build a 3D distance transform for fast nearest neighbor access. Our sparse LM-ICP requires only \( 1 \) ms per hypothesis, allowing us to verify all the hypotheses. We define the hypothesis score to be:

\[
\Xi(\theta_t) = \frac{1}{N_M} \sum_{j} \sum_{k} \left\{ \begin{array}{ll} 1, & \| \theta_t^{-1} \circ m_j - s_k(j) \| < \tau \theta \\ 0, & \text{otherwise} \end{array} \right.
\]

where \( \theta_t \) is the pose hypothesis and \( s_k(j) \) the closest sampled scene point to transformed model point \( \theta_t^{-1} \circ m_j \). Intuitively, this score reflects the percentage of visible model points. The surviving poses are then sorted, taken to the next level and densely refined. This coarse to fine scheme is repeated for 3 levels of the pyramid. Finally, a dense registration is performed to accurately obtain the final pose.

Until this stage the surface normals are excluded from the fine registration process. We do this intentionally, to use them as a verification tool. Following registration, we check the surface consistency between the scene and the model.
shows our reconstruction accuracy both prior to and after
for each object. The models
provide PR-curves for LISM
discretization time, a relative model and scene sampling distance of
on the sensor quality. For accurate scanners we use
distance to consider a scene point to be on the model based
The ground truth models (left) and our recon-
stracted part reveals the unscanned region,
pose graph becomes disconnected and therefore, we opti-
mize individually the two sub-components and record the
mean. We also report the number of scans in which the
model is detected and verified. Not every model is visible
every scan. In the end, our mean accuracy is well below a
millimeter, where the used sensor, Minolta Vivid 910 scan-
er, reports an ideal accuracy of \( \sim 0.5 \text{mm} \). We are also not
aware of any other works, reporting reconstruction results
on such datasets. Fig. 6 visualizes our outcome, and Table
1 shows our reconstruction accuracy both prior to and after
the optimization. While our error is quantitatively small, the
qualitative comparison also yields a pleasing result, some-
times being superior even to the original model.

The detection performance of a basic variant of our
method has already been proven to be robust on this dataset
[11]. Figures 7(a) and 7(b) provide PR-curves for LISM
and the hypothesis verification. Note that although LISM
already performs well, our verification clearly improves the
distinction between a match vs false positive. Using a sim-
ple threshold, we could obtain 100\% precision without sac-
ificing the recall. Thus, our score threshold, combined with
the normal consistency check manages to reject all false hy-
potheses, at the expense of rejecting a small amount of TP.

Toy Objects Dataset Since our objective is to assess the
fidelity of the reconstruction to the CAD model, we opt to
use the objects from the 3D printed dataset [37]: Leopard, Teddy and Bunny and Tank (See Fig. 8). The diameters of
objects vary in the range of \( 15 \sim 30 \text{cm} \). The print accuracy

<table>
<thead>
<tr>
<th>Model</th>
<th>w/o Opt.</th>
<th>with Opt.</th>
<th># Scans</th>
<th>Clutter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chef</td>
<td>2.90 ± 2.40</td>
<td>1.07 ± 0.65</td>
<td>22</td>
<td>0.58 ± 0.11</td>
</tr>
<tr>
<td>Chicken</td>
<td>1.71 ± 1.60</td>
<td>0.33 ± 0.24</td>
<td>29</td>
<td>0.61 ± 0.12</td>
</tr>
<tr>
<td>Para.</td>
<td>2.52 ± 2.00</td>
<td>0.41 ± 0.30</td>
<td>12</td>
<td>0.24 ± 0.20</td>
</tr>
<tr>
<td>T-rex</td>
<td>2.36 ± 2.08</td>
<td>0.88 ± 0.62</td>
<td>27</td>
<td>0.14 ± 0.22</td>
</tr>
</tbody>
</table>

**Mian Dataset** We first compare the reconstruction quality on Mian Dataset [28]. This dataset includes 50 laser
scanned point clouds of 4 complete 3D objects, with varying
occlusion and clutter. The objects change locations from
scan to scan, creating dynamic scenes. The clutter and back-
ground also varies as the objects appear together with other

**Live Feedback** Due to the connected-ness of pose graph,
our method is able to keep track of the overlap between all
the point clouds, at all times, informing the user whenever
graph disconnects or overlap is small. The complement of
the already reconstructed part reveals the unscanned region,
which is also fed back to the operator. Incoming scans di-
rectly propagate and form links in the pose graph, allowing
online response to the user’s actions.

## 4. Experimental Evaluation

We evaluate our method against a set of real datasets ac-
cquired by laser scanners and structured light sensors. The
CAD models we work with might contain uneven distribu-
tion of vertices or inner geometry. We always eliminate the
inner structure by thresholding the ambient occlusion val-
ues [29] before the models are re-meshed [25]. At detec-
tion time, a relative model and scene sampling distance of
\( d = \tau \text{diam}(M) \) is used, where \( 0.05 \geq \tau \geq 0.025 \) depend-
ing on the object. We also adjust another threshold on the
distance to consider a scene point to be on the model based
on the sensor quality. For accurate scanners we use 1.5mm,
while for less accurate ones 0.5cm. This does not affect the
segmentation, but the hypothesis verification.

**Table 1:** Reconstruction results on Mian Dataset (in mm). Each object is compared to the model provided by [28] using [1].

\[
\text{Clutter} = 1 - \frac{(\text{Model Surface Area}) \times (1-\text{Occlusion})}{(\text{Scene Surface Area})} \quad (12)
\]
is up to 50µ (micro-meters), well sufficient for consideration as ground truth. To capture the real scenes, a homemade, high accuracy phase-shift sensor, delivering <0.4 mm point accuracy is chosen. We sample up to 10 scans per object, taken out of a 100 frame sequence. To disrupt the acquisition order, we randomly shuffle this subset and apply our reconstruction algorithm. Next, we compute the CAD-to-reconstruction distances in CloudCompare [1]. We do not explicitly register our reconstruction to CAD model, because we already end up on model coordinate frame (having the result in the CAD space is a side benefit of our approach). Moreover, we use the original 100-frame, ordered sequence as an input to standard reconstruction pipelines such as Kinect Fusion [35], Kehl et. al. [21] (also uses color) and Slavcheva et. al. [37] all of which require a temporally ordered set of frames, with a large inter-frame overlap. All of these algorithms take depth image as input, whereas ours uses the unstructured 3D data and the model.

Our results on this dataset are shown in Tab. 2 (Ours) when original CAD is used. We also report the results when KinectFusion (KF) prior is used (Ours-KF). The individual error distribution of the objects are shown in Fig. 7(c).

Table 2: Reconstruction errors on toy objects dataset (mm).

<table>
<thead>
<tr>
<th></th>
<th>Leo</th>
<th>Teddy</th>
<th>Bunny</th>
<th>Tank</th>
</tr>
</thead>
<tbody>
<tr>
<td>KinFU</td>
<td>1.785±1.299</td>
<td>0.998±0.807</td>
<td>0.664±0.654</td>
<td>1.390±1.315</td>
</tr>
<tr>
<td>Kehl</td>
<td>1.018±1.378</td>
<td>1.028±0.892</td>
<td>0.838±0.860</td>
<td>1.573±2.250</td>
</tr>
<tr>
<td>Sdf2Sdf</td>
<td>0.652±0.614</td>
<td>0.910±0.584</td>
<td>0.541±0.436</td>
<td>0.466±0.416</td>
</tr>
<tr>
<td>Ours</td>
<td>0.481±0.519</td>
<td>0.517±0.572</td>
<td>0.415±0.501</td>
<td>0.451±0.322</td>
</tr>
<tr>
<td>Ours-KF</td>
<td>0.536±0.411</td>
<td>0.519±0.582</td>
<td>0.502±0.529</td>
<td>0.468±0.474</td>
</tr>
<tr>
<td>Ours-CO</td>
<td>0.651±0.628</td>
<td>0.544±0.601</td>
<td>0.698±0.506</td>
<td>0.475±0.433</td>
</tr>
</tbody>
</table>

Because our method does not suffer from drift and computes absolute poses all the time, although we use 10 times less scans, we are still 2-4 times more accurate then conventional methods. This also shows that our method could retain the good accuracy of the sensor.

Next, we augment this dataset with further scenes of the same objects, such that clutter and occlusions are present. Some shots are shown in Fig. 8 (mid-row). Our reconstruction accuracy (Ours-CO) is shown in Tab. 2 for different objects. These results are still better than or close to Sdf2Sdf [37]. Due to inclusion of some outliers, our results get slightly worse than the one in no clutter, yet they are still acceptable. However, none of the other approaches can run on this new set due to the existence of significant outliers.

In a further experiment, we gradually decimate the toy models down to a mesh of ≈500 vertices. We exclude tank as the decimation has little effect on the planarities. As shown in Fig. 7(d), even though the CAD prior gets very crude, we are still able to achieve a reasonable reconstruction, as long as the CAD model is still detectable in the scenes. Note that, results of KF prior is plotted in dashes as it is also a form of rough mesh approximation. Furthermore, Fig. 9 visually compares our reconstruction to the state of the art on the tank object. Because we do not use smoothing voxel representations (such as SDF), our method is much better at preserving sharp features at the model edges.
Figure 10: The reconstruction of Turbine(a) in captured cluttered scans(b) is presented in (c). Results in Sofa are shown in (d,e).

Table 3: Object Information, Average reconstruction errors w.r.t. Photogrammetry (in mm) and Timings.

<table>
<thead>
<tr>
<th>Object</th>
<th>Scanner</th>
<th>Scan Res.</th>
<th>Obj. Size</th>
<th>No Scans</th>
<th>No PG Images</th>
<th>PG vs CAD</th>
<th>Surphaser</th>
<th>Our Accuracy</th>
<th>Detect</th>
<th>Verify</th>
<th>Refine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ventil</td>
<td>Surph.</td>
<td>0.3 m</td>
<td>8m³</td>
<td>8</td>
<td>180</td>
<td>1.3 cm</td>
<td>3.6±3.3</td>
<td>2.2±0.4</td>
<td>3.10s</td>
<td>0.27s</td>
<td>112.94s</td>
</tr>
<tr>
<td>Turbine</td>
<td>Surph.</td>
<td>0.4 mm</td>
<td>125m³</td>
<td>10</td>
<td>180</td>
<td>3.4 cm</td>
<td>-</td>
<td>2.5±1.3</td>
<td>3.72s</td>
<td>0.54s</td>
<td>126.13s</td>
</tr>
<tr>
<td>Sofa</td>
<td>Str.Light</td>
<td>1mm</td>
<td>1.7m³</td>
<td>6</td>
<td>68</td>
<td>0.85mm</td>
<td>-</td>
<td>1.4±1.2</td>
<td>1.44s</td>
<td>0.31s</td>
<td>68.82s</td>
</tr>
</tbody>
</table>

Dataset of Large Objects  Finally, we apply our pipeline to quality inspection of real gas turbine casings and large objects. In this real scenario, CAD models come directly from the manufacturer. Due to space constraints, we summarize the data modality in Tab.3. The manufactured parts deviate significantly from the ideal model due to manufacturing and we scan them in the production environment within clutter and occlusions. With such large objects and little resemblance of the CAD prior, obtaining ground truth becomes a challenging task. Thus, we use a photogrammetry (PG) system [5, 4] to collect a sparse set of scene points, by attaching markers on the objects. We capture many images (see Tab.3) from different angles and run Linearis 3D software for bundle adjustment to obtain sparse ground truth. For Ventil object, we also use Surphaser software for reconstruction using external markers. Both Surphaser and our outputs are compared to the PG data in Tab. 3. The mean errors are obtained by CloudCompare [1]. We also provide running times of the individual stages. As seen, our accuracy outperforms an industry standard solution, Surphaser Software, by a margin of 38% on Ventil object. The performance in objects of varying sizes indicate that our reconstruction method is applicable from small to large scale while maintaining repeatability. Fig. 10 presents further qualitative results on our reconstruction of the Turbine and Sofa objects. Please consult the supplementary material for more evaluations.

Limitations  Due to the nature of PPF matching, our approach requires objects with rich geometry. Symmetric objects are also problematic due to ambiguity in pose estimation. Last but not least, currently, there is no mechanism to handle mis-detections. Yet, mis-detections are hardly a problem when the score threshold is reasonably high. This way, we detect in less scenes but avoid mistakes.

5. Conclusions

We proposed reconstruction-via-detection framework, as an alternative perspective to robust 3D instance reconstruction from unconstrained point cloud scans. Our framework integrates probabilistic object detection, hypothesis verification, pose graph construction and multi-view optimization. Such a scheme allowed us to deal with problems of dynamic clutter, occlusion and object segmentation. Moreover, the computational cost is reduced, due to model-to-scan alignment. To the best of our knowledge, this is the first method, capable of reconstructing instances within clutter and occlusions, without explicit segmentation.

As a future direction, we like to take care of confusions stemming from rotational symmetries by optimizing over possible global alignments of the scans. We also plan to extend our method to robotics via next-best view prediction.
References

[1] Cloudcompare - 3d point cloud and mesh processing software. http://www.danielgm.net/cc/, 2016. 6, 7, 8


