Hybrid ICP

Kamil Dreczkowski and Edward Johns

Abstract—ICP algorithms typically involve a fixed choice of data association method and a fixed choice of error metric. In this paper, we propose Hybrid ICP, a novel and flexible ICP variant which dynamically optimises both the data association method and error metric based on the live image of an object and the current ICP estimate. We show that when used for object pose estimation, Hybrid ICP is more accurate and more robust to noise than other commonly used ICP variants. We also consider the setting where ICP is applied sequentially with a moving camera, and we study the trade-off between the accuracy of each ICP estimate and the number of ICP estimates available within a fixed amount of time.

I. INTRODUCTION

Precise object localisation is essential for robots to interact with their surroundings effectively. The 2020 BOP Challenge [1] revealed the importance of the Iterative Closest Point (ICP) algorithm for accurate pose estimation by showing that 7 out of 10 of the top-performing methods relied on ICP for pose estimation refinement. For example, ICP increased the performance of CosyPose [2] by 6.1%, taking it from 3rd to 1st place in the challenge, and that of Pix2Pose [3] by 24.9%, taking it from 22nd to 4th place in the challenge.

Of all the design choices in an ICP implementation, there are two which affect the accuracy and convergence rate the most [4]. The first is the data association method used to find correspondences between two point clouds. The second is the error metric minimised when comparing distances between these correspondences. Existing algorithms that use a fixed data association method and fixed error metric commonly fail to estimate object poses robustly, as they are not flexible enough to cope with a range of geometries and ICP initialisations. In this paper, we propose Hybrid ICP, a flexible algorithm that chooses the most appropriate data association method and error metric based on the live image of an object and the current ICP estimate.

The contributions of this paper are summarised as follows. For our first contribution, we benchmark some of the most commonly used ICP variants on object pose estimation. This reveals that, and explains why, existing ICP variants fail to estimate the pose of individual objects robustly. For our second contribution, we propose Dynamic Switching, a method that uses an estimate of the Visible Surface Discrepancy (VSD) [5] calculated at the current ICP estimate, which we call the mean VSD estimate (MVE), to optimise the choice of the data association method. For our third contribution, we propose Cascading ICP, which sequentially minimises the point-to-point and then the point-to-plane error metric, to overcome the individual limitations of these error metrics. For our fourth contribution, we introduce the full Hybrid ICP, which combines Dynamic Switching, Cascading ICP, and point-to-point ICP. A single iteration of Hybrid ICP is shown in this figure. First, the mean VSD estimate (MVE) is calculated using the live image of an object and the current ICP estimate. Then, the MVE is used by Dynamic Switching to optimise the choice of the data association method and error metric. Finally, ICP is used for N iterations to update the current estimate before moving to the next Hybrid ICP iteration.

II. RELATED WORK

The ICP algorithm consists of 5 independent components: 1) point selection, 2) data association, 3) correspondence weighting, 4) outlier rejection, and 5) the error metric. In the past three decades, researchers have proposed many methods for each of these components. For example, [4], [8] propose methods for point selection, [8], [9] for outlier rejection, and [4] outlines methods for weighing correspondences. The main data association methods are nearest neighbour (NN) [10], normal shooting [11], and projective [12] data association. The main ICP error metrics are the point-to-point
the point-to-plane [11] error metrics. Rusinkiewicz et al. [4] benchmark many existing ICP variants on three synthetic scenes. As they illustrate, not all components of ICP affect its performance. They found that only the data association method and the choice of the error metric have a large impact, while the remaining components have a data-dependent and/or a negligible impact on performance. Therefore, in contrast to related work, in this paper, we do not propose a novel method for one of the components of ICP, but instead, we propose to use the visible geometry of an object and the current ICP estimate to decide which method to use. The majority of papers that introduce novel ICP variants such as [13] and [14], benchmark these variants on either unrealistic or indoor and outdoor scenes. The geometry of such scenes is usually rich in features and is not representative of the geometry of everyday objects. Therefore, even if a variant is shown to perform well on such scenes, it can fail to estimate individual object poses due to the lack of geometric constraints. In this paper, however, we specifically study ICP for the task of individual object pose estimation.

## III. Background

### A. Iterative Closest Point (ICP)

The ICP algorithm [10], [11] is commonly used for geometric alignment of point clouds when an initial estimate of their transformation is known. When applied to pose estimation, the first point cloud is an object model \( O = \{ o_i, m_i \} \), and the second point cloud \( C = \{ c_j, n_j \} \) is computed from a segmented depth image using the camera intrinsic matrix, where each point \( o_i \) and \( c_j \) is associated with a normal \( m_i \) and \( n_j \). This paper assumes that segmentation masks are provided by a segmentation method such as the U-Net [15].

#### Algorithm 1: Generic ICP Algorithm

**Input:** Point clouds \( C = \{ c_j, n_j \} \) and \( O = \{ o_i, m_i \} \)  
An initial transformation \( T^o_{CO} = [R^o_{CO} | t^o_{CO}] \)  
Maximum number of iterations \( max\_iter \)  

**Output:** A refined transformation \( T^k_{CO} \)

for \( k = 1, ..., max\_iter \) do

1. \( c_i, n_i \leftarrow \text{DataAssociation}(T^k_{CO}, o_i, C) \) for all \( i \)
2. \( (o_i, m_i) \leftarrow (T^k_{CO} o_i, R^k_{CO} m_i) \) for all \( i \)
3. \( T^k \leftarrow \arg \min \sum \text{ErrorMetric}(c_i, n_i, o_i, m_i, T) \)
4. \( T^{k+1}_{CO} \leftarrow T^k T^k_{CO} \)

Pseudocode of the generic ICP algorithm is shown in Algorithm 1. This algorithm only highlights 2 of the 5 components of ICP as the remaining components have been shown to have little effect on its performance [4]. The algorithm is initialised with two point clouds \( C \) and \( O \), an initial estimate of the transformation between them, \( T^0_{CO} \), and with the maximum number of iterations \( max\_iter \). In this work, we assume that ICP initialisation is provided by a global pose estimator such as CosyPose [2]. During each iteration of the algorithm, a data association method, **DataAssociation**, assigns points \( o_i \in O \) to points \( c_j \in C \). All points in \( O \) are then transformed using the current ICP estimate. All correspondences \( (c_j, n_j, o_i, m_i) \) are then used to minimise an error metric, **ErrorMetric**, to solve for an incremental transformation that is used to update the current ICP estimate.

1) **Data Association:** The two most popular data association methods are NN [10] and projective [12] data association. NN data association uses \( T^o_{CO} \) to transform each point \( o_i \in O \) into the frame of reference of \( C \), and associates each point with its closest neighbour in \( C \).

Algorithms that use projective data association store point clouds \( C \) and \( O \) in vertex maps \( V_C, V_O \in \mathbb{R}^{H \times W \times 3} \) [16]. At each pixel location \((v, u)\), a vertex map stores the \((x, y, z)\) coordinates of the 3D point that was projected to that same pixel in the depth image used to compute the vertex map. \( V_O \) is obtained by rendering \( O \) at \( T^1_{CO} \). Projective data association first uses \( T^0_{CO} \) to transform each point \( o_i \in O \) into the frame of reference of \( C \). It then projects that point onto the image plane of \( V_C \) using the camera intrinsic matrix. The point \( c_j \) stored at the pixel to which \( o_i \) is projected is then associated with \( o_i \).

ICP assumes that each point \( o_i \in O \) has a corresponding point \( c_j \in C \). As this assumption is often violated, researchers have proposed to only associate points that are no further than \( \tau_{\text{max}} \) apart [8], and that have an angle between normals no larger than \( \theta_{\text{max}} \) [9].

2) **Error metric:** The two most common error functions are the point-to-point [10] and point-to-plane [11] functions. The point-to-point error function is defined as:

\[
E_{\text{p2p}}(c, n, o, m, T_{CO}) = \| T_{CO} o - c \|^2
\]

and computes squared distances between correspondences in \( O \) and \( C \). The point-to-plane error function is defined as:

\[
E_{\text{p2p}}(c, n, o, m, T_{CO}) = \| n^T (T_{CO} o - c) \|^2
\]

and computes squared distance between points in \( O \) and linear surface approximations in \( C \).

### B. Visible Surface Discrepancy (VSD)

Evaluating pose estimators that make predictions from single images is challenging, as often more than one object pose is consistent with an input image, and all such poses are equally valid. The VSD [1], [5] is a pose error function that is invariant to this ambiguity and is defined as:

\[
e_{\text{VSD}}(D_{\text{est}}, D_{\text{gt}}, M_{\text{est}}, M_{\text{gt}}, \tau) = \frac{\sum_{p \in M_{\text{est}} \cap M_{\text{gt}}} e(p, D_{\text{est}}, D_{\text{gt}}, M_{\text{est}}, M_{\text{gt}}, \tau)}{\| M_{\text{est}} \|}
\]

where the cost function is defined as:

\[
e(p, D_{\text{est}}, D_{\text{gt}}, M_{\text{est}}, M_{\text{gt}}, \tau) = \begin{cases} 0 & \text{if } p \in M_{\text{est}} \cap M_{\text{gt}} \land |D_{\text{est}}(p) - D_{\text{gt}}(p)| < \tau \\ 1 & \text{otherwise} \end{cases}
\]
and $M_{gt}$ are visibility masks for $D_{ext}$ and $D_{gt}$, and $\tau$ is a misalignment tolerance.

IV. HYBRID ICP

Hybrid ICP consists of two novel methods: Dynamic Switching, which optimises the data association method, and Cascading ICP, which combines point-to-point and point-to-plane ICP.

A. Data Association

1) Limitations of NN and projective data association: NN data association relies on distances between points to associate them. For this reason, it is robust to initialisation noise and can nearly always bring two surfaces into better alignment. This is because every point has some corresponding closest point. However, NN algorithms are computationally intensive and are not robust to scenes with surfaces that are very close to each other. To see why this is, consider two neighbouring points on a single surface of a thin object. Given slight misregistration, these points can be matched to points that lie on opposite sides of the object.

On the other hand, projective data association is performed in constant time and is an order of magnitude faster [4]. Also, as points are associated based on geometry and the pinhole camera model [17], geometric relationships hold between correspondences, resulting in consistent and robust matching. However, for some ICP initialisations, projective data association can fail to find any correspondences altogether. This happens when all points $o \in V_C$ are projected outside of $V_C$ or onto its regions that do not contain the object.

2) Our solution, Dynamic Switching: We now introduce Dynamic Switching, a novel method that optimises the data association method based on an estimate of the VSD error calculated for the current ICP estimate. This error is an informative proxy for the percentage of points for which projective data association would fail to find a correspondence for. If it is low, we know that many points would be associated as there is good alignment between the input image and an image rendered at the current ICP estimate. In this case, it is best to use projective data association as it yields geometrically consistent correspondences and is faster than NN data association [4]. However, if the VSD error is large, we can expect that projective data association would find few matches. In this case, it is best to use NN data association as it is more robust to initialisation noise and is more likely to find correspondences.

Dynamic Switching is a simple algorithm that can be integrated into the generic ICP algorithm by using it to optimise the data association method before pose estimation or every $N$ ICP iterations. In essence, Dynamic Switching computes an estimate of the VSD error for the current ICP estimate, and based on its magnitude, decides which data association method to use. If the VSD estimate is more than or equal to some threshold $\alpha$, NN data association is used. And if the estimate is smaller than $\alpha$, projective data association is used. In this project, we have found that $\alpha = 0.4$ is a suitable threshold for object pose estimation with noiseless models and images that enable generalisation to unseen objects (see section III of our supplementary material).

From Equation 1, we observe that the VSD error only requires an object’s ground truth pose to render a distance map and to compute a visibility mask at that pose. We propose to approximate the distance map and visibility mask with a distance map and segmentation mask of the input depth image. The VSD error estimate calculated using these two approximations is identical to the VSD error in the absence of noise. Similarly to the 2020 BOP Challenge [1] evaluation methodology, we calculate the mean of VSD error estimates for misalignment tolerances ranging from $5\% - 50\%$ of an object’s diameter, increasing in steps of $5\%$, and refer to this as the mean VSD estimate (MVE) hereafter.

B. Error Metrics

1) Limitations of point-to-point and point-to-plane ICP: In point-to-point ICP, each point is associated with a discrete pointwise approximation of the other surface. In point-to-plane ICP, each point is associated with a continuous linear approximation of the other surface. The implications of this are twofold. (1) More accurate results can be often obtained with point-to-plane ICP [9]. When ICP is used for pose estimation, points in $O$ are commonly uniformly distributed across an object’s surface, while those in $C$ have a non-uniform distribution as they are computed from an image. Consequently, it is often impossible to precisely align all points from $O$ with those in $C$, while it is possible to align them with continuous linear approximations of the object’s surface at these points. (2) Point-to-plane ICP is less robust to simple geometries. The point-to-plane error metric does not penalise points for moving along tangent planes of their correspondences, allowing two surfaces to effectively “slide” past each other. If the region of overlap between two surfaces being aligned has almost uniform curvature, the two surfaces can diverge starting from a slight misregistration [9].

2) Our solution, Cascading ICP: We now introduce Cascading ICP, a two-stage algorithm that combines point-to-point and point-to-plane ICP. During the first stage, Cascading ICP registers two input point clouds with point-to-point ICP. In the second stage, Cascading ICP refines the output of the first stage with point-to-plane ICP. During both stages, if during any ICP iteration:

1) the number of correspondences is 0
2) the number of correspondences is smaller than the number of correspondences during the first or the previous iteration by $5\%$
3) the mean loss increases compared to the previous iteration

Cascading ICP retrieves the last estimate before the algorithm began diverging and moves to the next stage or outputs the result. This way, if point-to-plane ICP diverges, Cascading ICP returns the output of stage 1. And if both stages diverge, Cascading ICP simply returns ICP initialisation. Both stages are also terminated early if the ICP loss converges.
The reason for refining ICP initialisation with point-to-point ICP before registering two point clouds with point-to-plane ICP is that everyday objects commonly have simple and symmetrical geometries. Refining ICP initialisation decreases the probability of point-to-plane ICP diverging when the two point clouds have few geometric features while enabling Cascading ICP to still benefit from the potentially more accurate result of point-to-point ICP.

In preliminary pose estimation experiments, we have found that using Cascading ICP only offers a boost in performance when combined with projective data association. When NN data association is used, Cascading ICP performs consistently worse than point-to-point ICP. This is because everyday objects typically have surfaces that are very close to each other, and NN data association yields many incorrect correspondences for such surfaces. Incorrect correspondences provide misleading constraints for point-to-plane ICP that prevent it from converging to the global minimum. For this reason, Hybrid ICP conditions its choice of error metric on the data association method chosen by Dynamic Switching.

C. Hybrid ICP - Combining Dynamic Switching, Cascading ICP, and point-to-point ICP

Hybrid ICP is an iterative algorithm that encapsulates the generic ICP algorithm. A single iteration of Hybrid ICP is shown in figure 1. During each iteration, Hybrid ICP begins by using Dynamic Switching to optimise the choice of the data association method. If projective data association is chosen, Hybrid ICP aligns the two point clouds with Cascading ICP for \( N \) iterations. If NN data association is chosen, it aligns the two point clouds with point-to-point ICP for \( N \) iterations.

The intuition behind Hybrid ICP is as follows. During any iteration, if Dynamic Switching chooses projective data association (i.e. the pre-ICP MVE is low), this implies that the two point clouds are already well-aligned. Cascading ICP is the most suitable algorithm in this regime as it is more robust to various geometries than point-to-point or point-to-plane ICP alone (see section III of our supplementary material). If Dynamic Switching chooses NN data association (i.e. the pre-ICP MVE is high), this implies that the two point clouds are not well-aligned and that many found correspondences will be incorrect. In this case, it does not make sense to register the two point clouds with Cascading ICP because normals from incorrect correspondences will provide misleading constraints for the point-to-plane error metric preventing it from aligning the two point clouds. For this reason, we instead use point-to-point ICP to bring the two surfaces into better alignment by insisting that the distance between all correspondences should be minimised. If NN point-to-point ICP successfully aligns the two point clouds, Dynamic Switching will likely choose projective data association in the subsequent Hybrid ICP iteration. In this work, we propose to fix the number of Hybrid ICP iterations to 2 and to allow ICP to converge within each Hybrid ICP iteration.

V. Sequential Pose Estimation

Consider a camera moving into an static object while sequentially re-estimating its pose. At each time step \( t = 1, ..., T \), the object is at the pose \( T_{O,C} \) in the camera’s frame, the camera captures an image of the object, and ICP estimates its pose while the camera moves by \( T_{C+1,C} \), where \( T_{C+2,O} = T_{C+1,C} T_{C,O} \). If the camera is mounted to a robot’s end-effector, its motion can be accurately determined from robot kinematics and extrinsic camera calibration.

When transitioning from time \( t \) to \( t+1 \), all previous estimates must be transformed by the camera’s motion to account for the movement of its frame of reference, i.e. each previous estimate \( \tilde{T}^{l}_{C,O} \) must be transformed to the new camera frame according to \( \tilde{T}^{l}_{C+1,O} = T_{C+1,C} \tilde{T}^{l}_{C,O} \), where the superscript \( l \) is the time step at which the estimate was made. Given this, at every time step \( t \), we have a list of estimates of the object’s pose in the current camera frame, possibly with associated uncertainties i.e. \( \{ \tilde{T}^{1}_{C,O}, \Sigma^{1} \}, ..., \{ \tilde{T}^{T}_{C,O}, \Sigma^{T-1} \} \), where \( \Sigma^{i} \) is the uncertainty associated with the estimate \( \tilde{T}^{i}_{C,O} \). In this work, we propose to use the MVE, \( e_{MVE} \), calculated for an ICP estimate as an uncertainty proxy. Although not theoretically grounded, the MVE is relatively cheap to compute and broadly captures the accuracy of an estimate. In the next section, we describe 6 methods which can be used to initialise ICP at every time step of a trajectory, and to calculate a final pose estimate given multiple estimates made at previous time steps. The Average, Weighted average, Filtering and Filtering with constant weights methods all use the algorithm proposed by Barfoot et al. [18] for fusing SE(3) estimates (see section I of our supplementary material for details). Out of these four methods, the weighted average and filtering methods require an estimate of the covariance associated with each pose estimate. We propose to use \( e_{MVE} \) multiplied by the identity matrix \( I \in \mathbb{R}^{6\times6} \) as a proxy for this covariance.

A. Methods for Sequential ICP Initialisation

Last Estimate: When queried at time step \( t \), this method returns \( \tilde{T}^{t-1}_{C,O} \).

Average: When queried for an estimate, this method takes a deterministic average of all previous estimates and returns the result.

Weighted Average: When queried for an estimate, this method fuses all previous estimates using their associated uncertainties and returns the result.

Filtering with constant weights: After ICP makes an estimate \( \tilde{T}^{t}_{C,O} \), this method fuses \( (\tilde{T}^{t}_{C,O}, I) \) with the previous estimate \( (\tilde{T}^{t-1}_{C,O}, \Sigma^{t-1}) \) to obtain a new estimate \( (\tilde{T}^{t}_{C,O}, \Sigma^{t}) \).

It then sets \( (\tilde{T}^{t}_{C,O}, \Sigma^{t}) \leftarrow (\tilde{T}^{t}_{C,O}, \Sigma^{t}) \). When queried for an estimate at time step \( t \), this method returns \( \tilde{T}^{t}_{C,O} \).

Filtering: This is identical to the method described above, with the exception that after ICP makes an estimate, it fuses \( (\tilde{T}^{t}_{C,O}, e_{MVE} I) \) with \( (\tilde{T}^{t-1}_{C,O}, \Sigma^{t-1}) \) to obtain \( (\tilde{T}^{t}_{C,O}, \Sigma^{t}) \).

Most Confident: This method stores all previous measurements with their associated uncertainties. When queried for
an estimate, it returns the estimate with the lowest uncertainty.

VI. EXPERIMENTS

A. Experimental Procedure

This section only briefly outlines the experimental procedure adopted in our simulation experiments. For further details, see section II of our supplementary material.

1) Objects: We consider 20 ShapeNet [19] objects from 20 different object categories. These objects are shown in figure 2. We use the 10 objects shown in the top row of this figure to tune the Dynamic Switching threshold (see section III of our supplementary material), and the remaining 10 to benchmark Hybrid ICP against other algorithms.

2) Evaluation Metric: In line with the evaluation methodology of the 2020 BOP Challenge [1], we compute the VSD pose error for misalignment tolerances ranging from 5%–50% of an object’s diameter, increasing in steps of 5%, and report the average of these errors marginalised across objects, object poses, and initialisation poses.

3) Sampling Object Poses: In single-image experiments (section VI-B), we sample object poses such that the distance between the object and the camera is uniformly distributed between \( [d_{obj}, 0.6] \) m, where \( d_{obj} \) is the object’s diameter. In trajectory-based experiments (section VI-C), we fix the distance between the object and the camera to 1 m. In all experiments, we set the camera’s orientation so that its optical axis intersects with a point in the proximity of the object’s centre. This ensures that an object is never rendered exactly at the centre of an image.

4) Sampling Initialisation Poses: Given a magnitude of the translation and rotation perturbation, \( \delta_t \) and \( \delta_R \), we first uniformly sample two vectors \( v \) and \( \eta \) from the surface of the unit sphere. The translation perturbation is then defined as \( t_\delta = \delta_t v \). The rotation perturbation \( R_\delta \in SO(3) \) is obtained by converting the axis-angle vector \((\delta_\theta, \eta)\) to a matrix. We define ICP initialisation as \( T_{CO} = [R_{CO} R_\delta t_{CO} t_\delta] \), where \( T_{CO} = [R_{CO} t_{CO}] \) is an object’s ground truth pose.

5) Relationship between translation and rotation errors of a sampling-based pose estimator: We first estimated the relationship between the magnitudes of translation and rotation errors of a sampling-based pose estimator to generate realistic initialisations. From preliminary experiments, we have found that on average, estimated poses had 1.92° of rotation error per 1 mm of translation error. We have also found that the mean translation and rotation error was 6.2 mm and 9.5° respectively.

B. Single Image Pose Estimation Refinement

We now describe three experiments that were carried out to test the robustness of Hybrid ICP to 1) initialisation noise, 2) depth image noise, and 3) model noise. All main results are illustrated in Figure 3, where each row corresponds to each of the three experiments. The first column benchmarks NN variants, the second projective variants, and the final benchmarks Hybrid ICP against the most robust NN and projective variant. As baselines, we implement point-to-point, point-to-plane and Cascading ICP, all with both NN and projective data association. For completeness, we also implement a Cascading ICP variant that first does point-to-plane ICP and then cascades to point-to-point ICP. Hereafter, we refer to the original variant (see section IV-B.2) as cascading point-plane ICP, and to this variant as cascading plane-point ICP.

1) Robustness to Initialisation Noise: We quantify initialisation noise as the mean VSD error calculated for ICP initialisation (pre-ICP VSD). In this experiment, we sample translation perturbations from a uniform distribution, \( \delta_t \sim U([0, 0.15]) \) m, and calculate the corresponding rotation perturbations \( \delta_R \) according to the ratio described in section VI-A.5. We use rejection sampling to ensure a uniform distribution of pre-ICP VSD errors across 10 equally spaced bins. In total, we sample 1000 object poses per object. All results for this experiment are shown in the top row of Figure 3.

Observing the first two figures, we notice that variants that use a fixed data association method and error metric fail to estimate object poses reliably. This is because these variants are not flexible enough to cope with simple geometries with few features and all possible initialisations. The two figures also reveal that both variants fail in the high pre-ICP VSD regime and that projective variants tend to perform best in the low pre-ICP VSD regime while NN point-to-point ICP in the high pre-ICP VSD regime. This is because when the pre-ICP error is high, projective variants fail to find correspondences, while NN variants do not have this problem. As the rightmost graph illustrates, Hybrid ICP is the most robust variant across all pre-ICP VSD intervals. This figure also shows that Hybrid ICP outperforms projective cascading point-plane ICP even in the low pre-ICP VSD regime. This is because false correspondences found by projective data association can result in an update that worsens the registration. If the region of overlap between the surfaces being aligned does not decrease and the mean loss does not increase, Cascading ICP would continue as normal. In contrast to this, Hybrid ICP would detect this as the MVE would increase, and would switch to NN data association to coarsely re-align the two surfaces.

2) Robustness to Depth Image Noise: In this experiment, we assume Gaussian noise in depth images with a stan-
Fig. 3: The first column benchmarks NN ICP variants, the second benchmarks projective ICP variants, and the last benchmarks Hybrid ICP against the most robust NN and projective variant. The top row investigates robustness to initialisation noise, the second row robustness to depth image noise, and the final row robustness to model noise. Pre-ICP VSD is the mean VSD calculated for ICP initialisation. Post-ICP VSD is the mean VSD calculated for the output of ICP. Note that the lower the mean VSD, the more accurate a method is.

Observing the first two figures, we notice that projective ICP variants are less susceptible to noise in depth images. We hypothesise that this is because projective variants yield geometrically consistent correspondences while NN variants are sensitive to noise. Another interesting observation is that the performance of NN and projective point-to-plane, Cascading point-plane, and Cascading plane-point increases after the standard deviation of the noise is increased from 0% to 1%. Although counter-intuitive, we hypothesise that the reason for this is that all test objects have simple geometries with few features, while all of these variants rely on point-to-plane ICP. In the absence of noise, these variants are brittle and susceptible to the surfaces “sliding” past each other during ICP. However, in the presence of noise, noisy normals help to “lock” the surfaces in place, limiting their ability to “slide” and diverge. Finally, examining the rightmost figure reveals that Hybrid ICP is the most robust variant to noise in depth images.

3) Robustness to Model Noise: We model noise in object models by creating 4 reconstructions of each of the objects with varying levels of noise with TSDF integration [20]. We create each reconstruction by integrating 100 randomly sampled images. Before integrating images to obtain reconstruction \( i \in [1, 2, 3, 4] \), we:

- Add \((i - 1)/2\) pixels to all camera intrinsic parameters,
- Add Gaussian noise to all depth images with a standard deviation of \( x\% \) of a pixel’s depth value, for \( x \in [0, 1, 2, 3, 4, 5] \). To evaluate all methods, we sample 100 poses per object and render the objects at these poses. We then perturb each ground truth pose by a translation and rotation perturbation \( \delta_t = 6.2 \text{ mm} \) and \( \delta_\theta = 9.5^\circ \) (i.e. by the mean error of the sampling-based pose estimator) to obtain ICP initialisation. Finally, we evaluated all methods using identical depth images and initialisations by adding different noise levels to the original images. Results are shown in the middle row of Figure 3.
deviation equal to \((i-1)\%\) of a pixels depth value,

- Perturb all ground truth object poses by a translation and rotation perturbation \(\delta_t = (i-1)\) mm and \(\delta_\theta = (i-1)^\circ\).

Figure 4 illustrates triangular meshes of all objects for all five noise levels. We sample 100 object poses per object to evaluate all variants and render the objects at these poses. We then perturb each ground truth pose by a translation and rotation perturbation \(\delta_t = 6.2\) mm and \(\delta_\theta = 9.5^\circ\) (i.e. by the mean error of the sampling-based pose estimator). Finally, we evaluated all methods using identical depth images and initialisations while using object models with different noise levels. All results for this experiment are shown in the bottom row of Figure 3.

Once again, the first two figures reveal that projective ICP variants are more robust to model noise than NN variants. Again, we hypothesise that this is because projective variants yield geometrically consistent correspondences. Also, from the rightmost figure, we can see that Hybrid ICP is the most robust variant to model noise.

4) Speed of Convergence: Table I shows the mean and standard deviation of inference times for all evaluated algorithms, averaged across all three single-image experiments. As these results illustrate, projective variants are significantly faster than Hybrid ICP and NN variants.

**TABLE I: Inference time of all ICP variants (fastest on top).**

| Method                  | Execution time (s) |
|-------------------------|--------------------|
| Projective point-to-plane | 0.061 ± 0.023     |
| Projective point-to-plane | 0.085 ± 0.038     |
| Projective cascading plane-point | 0.094 ± 0.035     |
| Projective cascading plane-point | 0.101 ± 0.040    |
| NN point-to-point        | 0.401 ± 0.516     |
| Hybrid ICP              | 0.542 ± 0.436     |
| NN point-to-plane        | 0.786 ± 3.535     |
| NN cascading plane-point | 0.879 ± 3.713     |
| NN cascading plane-point | 1.090 ± 3.869     |

C. Sequential ICP

We now consider the setting where ICP is applied sequentially with a camera moving into a static object along a trajectory and refer to this setting as a **Sequential ICP**. Here, we study different ICP variants for their trade-off of accuracy and convergence speed, since faster variants allow for more pose estimations in a given time.

1) Trajectory definition: We define a trajectory as a linear path between some initial camera pose and a point in the proximity of an object. The camera pose is sampled in line with the description in section VI-A.3, and the camera moves in the direction of its optical axis at a velocity of 0.1 ms\(^{-1}\). The magnitude of the camera movement between any two subsequent time steps is determined by the time it takes a method to obtain ICP initialisation and calculate the uncertainty proxy when required.

2) Experiment: To study the trade-off between the accuracy of each ICP estimate and the number of available estimates, we use projective Cascading (point-plane) and Hybrid ICP as the underlying ICP algorithms. Hybrid ICP was shown to consistently perform best in single-image experiments while having a relatively high inference time. On the other hand, projective Cascading ICP was shown to perform well in the low pre-ICP VSD regime and poorly in the high pre-ICP VSD regime while having a low inference time. By using these two variants, we investigate if it is better to have many low-quality estimates or fewer high-quality estimates. In line with the single image experiment, to initialise each trajectory, we sample translation perturbations from the uniform distribution \(\delta_t \sim U[0, 0.15] m\) and calculate the corresponding rotation perturbations \(\delta_\theta\) according to the ratio described in section VI-A.5. We use rejection sampling to ensure a uniform distribution of pre-trajectory VSD errors across 10 equally spaced bins. To simulate a realistic setting, we assume the depth noise model proposed by Ahn et al. [21] for the Intel RealSense D435 depth camera and access to a noisy reconstruction. To this end, we reconstruct each of the objects using 100 random images assuming the RealSense D435 noise model and by perturbing ground-truth camera poses by a translation and rotation perturbation \(\delta_t = 5\) mm and \(\delta_\theta = 1.1^\circ\). These magnitudes were obtained by calculating the mean error for multiple extrinsic calibrations of a RealSense D435 camera mounted to a real-world Sawyer robot’s wrist. In total, we sample 5 trajectories per object per pre-trajectory VSD bin.

3) Results: The results for this experiment are illustrated in Figure 5. In this figure, the graphs on the left and right show the performance of all methods for fusing multiple estimates when using projective Cascading ICP and Hybrid ICP as the underlying ICP algorithms respectively. As these two graphs illustrate, the choice of the underlying ICP algorithm depends on the VSD distribution of the global pose estimator used to initialise sequential ICP. For example, Labbe et al. [2] report that 63.8\% of all VSD errors of the CosyPose pose estimator are below 0.3 for a misalignment tolerance of \(\tau = 2\) cm. Hence, when used for sequential ICP initialisation, it would be best combined with projective Cascading ICP because in the low pre-trajectory VSD error regime, this performs better than Hybrid ICP. This is because for low pre-ICP VSD errors, projective Cascading ICP has a similar performance to Hybrid ICP while being significantly faster.

Whilst there are some trends in terms of which methods are worst, there is no obvious trend in terms of which method...
is best. Therefore, we conclude that the Average method is a suitable method for using sequential ICP estimates as it is one of the simplest methods and is at least as good as the others. Also, as methods that used per-image uncertainties did not perform better than other methods, this suggests that either the uncertainty estimates were poor, or these uncertainties were not appropriate for the method proposed by Barfoot et al. [18] for fusing SE(3) estimates. Johns [6] has also found that methods that used per-image uncertainties were no better than other baselines for fusing sequential pose estimates.

VII. CONCLUSIONS

We have proposed Hybrid ICP, a flexible ICP variant that uses the current image observation to dynamically choose the optimal data association method and the optimal error metric. As a component of this algorithm, we also introduced Cascading ICP. We showed that Hybrid ICP is more robust to initialisation and noise than other commonly used ICP variants when estimating object poses. We also studied sequential ICP and the trade-off between accuracy and convergence speed and found that for initialisation errors based on state-of-the-art pose estimators, projective Cascading ICP is optimal.

REFERENCES

[1] T. Hodan et al., “BOP challenge 2020 on 6D object localization,” in Computer Vision – ECCV 2020 Workshops, A. Bartoli and A. Fusiello, Eds., Springer International Publishing, 2020, pp. 577–594.
[2] Y. Lambé et al., “Cosypose: Consistent multi-view multi-object 6D pose estimation,” in European Conference on Computer Vision. Springer, 2020, pp. 574–591.
[3] K. Park et al., “Pix2Pose: Pixel-Wise coordinate regression of objects for 6D pose estimation,” in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 7667–7676.
[4] S. Rusinkiewicz et al., “Efficient variants of the ICP algorithm,” in Proceedings Third International Conference on 3-D Digital Imaging and Modeling, 2001, pp. 145–152.
[5] T. Hodañ et al., “On evaluation of 6D object pose estimation,” in Computer Vision – ECCV 2016 Workshops. Cham: Springer International Publishing, 2016, pp. 606–619.
[6] E. Johns, “Coarse-to-fine imitation learning: Robot manipulation from a single demonstration,” in IEEE International Conference on Robotics and Automation (ICRA), 2021.
[7] E. Valassakis, N. Di Palo, and E. Johns, “Coarse-to-fine for sim-to-real: Sub-millimetre precision across wide task spaces,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021.
[8] G. Turk et al., “Zippered polygon meshes from range images,” Proceedings of the 21st annual conference on Computer graphics and interactive techniques, 1994.
[9] K. Pulli, “Multiview registration for large data sets,” in Second International Conference on 3-D Digital Imaging and Modeling (Cat. No. PR00062), 1999, pp. 160–168.
[10] P. Besl et al., “A method for registration of 3-D shapes,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp. 239–256, 1992.
[11] Y. Chen et al., “Object modeling by registration of multiple range images,” in Proceedings. 1991 IEEE International Conference on Robotics and Automation, 1991, pp. 2724–2729 vol.3.
[12] G. Blais et al., “Registering multiview range data to create 3D computer objects,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 8, pp. 820–824, 1995.
[13] A. Segal et al., “Generalized-ICP,” Proc. of Robotics: Science and Systems, 06 2009.
[14] J. Minguet et al., “Metric-based iterative closest point scan matching for sensor displacement estimation,” IEEE Transactions on Robotics, vol. 22, no. 5, pp. 1047–1054, 2006.
[15] O. Ronneberger et al., “U-net: Convolutional networks for biomedical image segmentation,” in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Cham: Springer International Publishing, 2015, pp. 234–241.
[16] R. Newcombe and othersw, “Kinectfusion: Real-time dense surface mapping and tracking,” in 2011 10th IEEE International Symposium on Mixed and Augmented Reality, 2011, pp. 127–136.
[17] D. Forsyth et al., Computer vision : a modern approach, 2nd ed. Pearson, 2012.
[18] T. Barfoot et al., “Associating uncertainty with three-dimensional poses for use in estimation problems,” IEEE Transactions on Robotics, vol. 30, no. 3, pp. 679–693, 2014.
[19] A. Chang et al., “ShapeNet: An Information-Rich 3D Model Repository,” arXiv e-prints, p. arXiv:1512.03012, Dec. 2015.
[20] B. Curless et al., “A volumetric method for building complex models from range images,” in Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, ser. SIGGRAPH ’96. New York, NY, USA: Association for Computing Machinery, 1996, p. 303–312.
[21] M. Ahn et al., “Analysis and noise modeling of the intel realsense D435 for mobile robots,” in 2019 16th International Conference on Ubiquitous Robots (UR), 2019, pp. 707–711.