3D Local Features for Direct Pairwise Registration

Haowen Deng [†, *, ✣] Tolga Birdal [†, *] Slobodan Ilic [†, *]

[†] Technische Universität München, Germany, [*] Siemens AG, München, Germany
[✣] National University of Defense Technology, China

Abstract

We present a novel, data driven approach for solving the problem of registration of two point cloud scans. Our approach is direct in the sense that a single pair of corresponding local patches already provides the necessary transformation cue for the global registration. To achieve that, we first endow the state of the art PPF-FoldNet [18] auto-encoder (AE) with a pose-variant sibling, where the discrepancy between the two leads to pose-specific descriptors. Based upon this, we introduce RelativeNet, a relative pose estimation network to assign correspondence-specific orientations to the keypoints, eliminating any local reference frame computations. Finally, we devise a simple yet effective hypothesize-and-verify algorithm to quickly use the predictions and align two point sets. Our extensive quantitative and qualitative experiments suggests that our approach outperforms the state of the art in challenging real datasets of pairwise registration and that augmenting the keypoints with local pose information leads to better generalization and a dramatic speed-up.

1. Introduction

Learning and matching local features have fueled computer vision for many years. Scholars have first handcrafted their descriptors [36] and with the advances in deep learning, devised data driven methods that are more reliable, robust and practical [34, 52]. These developments in the image domain have quickly escalated to 3D where 3D descriptors [44, 53, 19] have been developed.

Having 3D local features at hand is usually seen as an intermediate step towards solving more challenging 3D vision problems. One of the most prominent of such problems is 3D pose estimation, where the six degree-of-freedom (6DoF) rigid transformations relating 3D data pairs are sought. This problem is also known as *pairwise 3D registration*. While the quality of the intermediary descriptors is undoubtedly an important aspect towards good registration performance [25], directly solving the final problem at hand is certainly more critical. Unfortunately, contrary to 2D descriptors, the current deeply learned 3D descriptors [53, 19, 18] are still not tailored for the task we consider, i.e. they lack any kind of local orientation assignment and hence, any subsequent pose estimator is coerced to settle for nearest neighbor queries and exhaustive RANSAC iterations to robustly compute the aligning transformation. This is neither reliable nor computationally efficient.

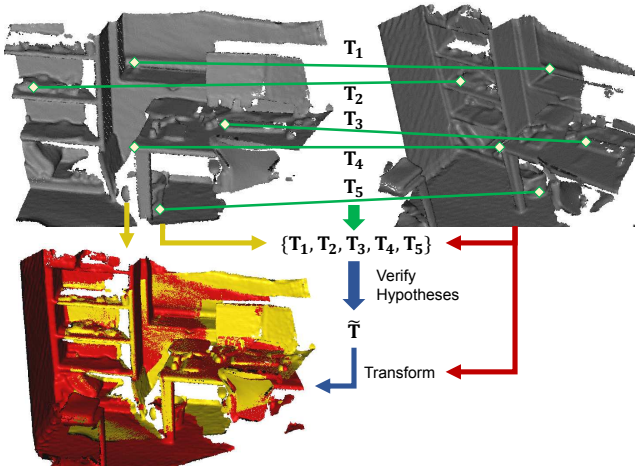

Figure 1. Our method provides not only powerful features for establishing correspondences, but also directly predicts a rigid transformation attached to each correspondence. Final estimation of the rigid pose between fragment pairs can then be made efficiently by operating on the pool of pose predictions.

In this paper, we argue that descriptors that are good for pairwise registration should also provide cues for direct computation of local rotations and propose a novel, robust and end-to-end algorithm for local feature based 3D registration of two point clouds (See Fig. 1). We begin by augmenting the state-of-the-art unsupervised, 6DoF-invariant local descriptor PPF-FoldNet [18] with a deeply learned orientation. Via our pose-variant orientation learning, we can decouple the *3D structure* from *6DoF motion*. This can result in features solely explaining the pose variability up to

a reasonable approximation. Our network architecture is shown in Fig. 2. We then make the observation that locally good registration leads to good global alignment and vice versa. Based on that, we propose a simple yet effective *hypothesize-and-verify* scheme to find the optimal alignment conditioned on an initial correspondence pool that is simply retrieved from the (mutually) closest nearest neighbors in the latent space.

For the aforementioned idea to work well, the local orientations assigned to our keypoints (sampled with spatial uniformity) should be extremely reliable. Unfortunately, finding such repeatable orientations of local patches immediately calls for local reference frames (LRF), which are by themselves a large source of ambiguity and error [41]. Therefore, we instead choose to learn to estimate *relative* transformations instead of aiming to find a canonical frame. We find the relative motion to be way more robust and easier-to-train for than an LRF. To this end, we introduce *RelativeNet*, a specialized architecture for relative pose estimation.

We train all of our networks end-to-end by combining three loss functions: 1) Chamfer reconstruction loss for the unsupervised PPF-FoldNet [18], 2) Weakly-supervised relative pose cues for the transformation-variant local features, 3) A feature-consistency loss which enforces the nearby points to give rise to nearby features in the embedding space. We evaluate our method extensively against multiple widely accepted benchmark datasets of 3DMatch-benchmark [53] and Redwood [13], on the important tasks of feature matching and geometric registration. On our assessments, we improve the state of the art by 6.83% in pairwise registration while reducing the runtime by 20 folds. This dramatic improvement in both aspects stems from the weak supervision making the local features capable of spilling rotation estimates and thereby easing the job of the final transformation estimator. The interaction of three multi-task losses in return enhances all predictions. Overall, our contributions are:

1. Invariant + pose-variant network for local feature learning designed to generate pose-related descriptors that are insensitive to geometrical variations.
2. A multi-task training scheme which could assign orientations to matching pairs and simultaneously strengthen the learned descriptors for finding better correspondences.
3. Improvement of geometric registration performance on given correspondence set using direct network predictions both interms of speed and accuracy.

2. Related Work

Local descriptors There has been a long history of handcrafted features, designed by studying the geometric properties of local structures. FPFH [43], SHOT [44], USC [46] and Spin Images [29] all use different ideas to capture these properties. Unfortunately, the challenges of real data, such as the presence of noise, missing structures, occlusions or clutter significantly harm such descriptors [25]. Recent trends in data driven approaches have encouraged the researchers to harness deep learning to surmount these nuisances. Representative works include 3DMatch [53], PPFNet [19], CGF [32], 3D-FeatNet [28], PPF-FoldNet [18] and 3D point-capsule networks [54], all outperforming the handcrafted alternatives by large margin. While the descriptors in 2D are typically complemented by the useful information of local orientation, derived from the local image appearance [36], the nature of 3D data renders the task of finding a unique and consistent local coordinate frame way more challenging [23, 41]. Hence, none of the aforementioned works were able attach local orientation information to 3D patches. This motivates us to jointly consider descriptor extraction and the direct 3D alignment.

Pairwise registration The approaches to pairwise registration fork into two main research directions.

The first school tries to find an alignment of two point sets globally. Iterative closest point (ICP) [2] and its transcendents [45, 50, 2, 35] alternatively hypothesize a correspondence set and minimize the 3D registration error optimizing for the rigid pose. Despite its success, making ICP outlier-robust is considered, even today, to be an open problem [30, 21, 48, 11]. Practical applications of ICP also incorporate geometric, photometric or temporal consistency cues [39] or odometry constraints [55], whenever available. ICP is prone to the initialization and is known to tolerate only up to a $15 - 30^\circ$ misalignment [5, 3].

Another family branches off from Random Sample Consensus (RANSAC) [22]. These works hypothesize a set of putative matches of keypoints and attempt to disable the erroneous ones via a subsequent rejection. The discovered inliers can then be used in a Kabsch-like [31] algorithm to estimate the optimal transformation. A notable drawback of RANSAC is the huge amount of trials required, especially when the inlier ratio is low and the expected confidence of finding a correct subset of inliers is high [12]. This encouraged the researchers to propose accelerations to the original framework, and at this time, the literature is filled with an abundance of RANSAC-derived methods [15, 16, 33, 14], unified under the USAC framework [42].

Even though RANSAC is now a well developed tool, heuristics associated to it facilitated the scholars to look for more *direct* detection and pose estimation approaches, hopefully alleviating the flaws of feature extraction and randomized inlier maximization. Recently, the geometric hashing of point pair features (PPF) [4, 20, 6, 26, 47] is found to be the most reliable solution [27]. Another alternative includes 4-point congruent set (4-PCS) [1, 9] further made ef-

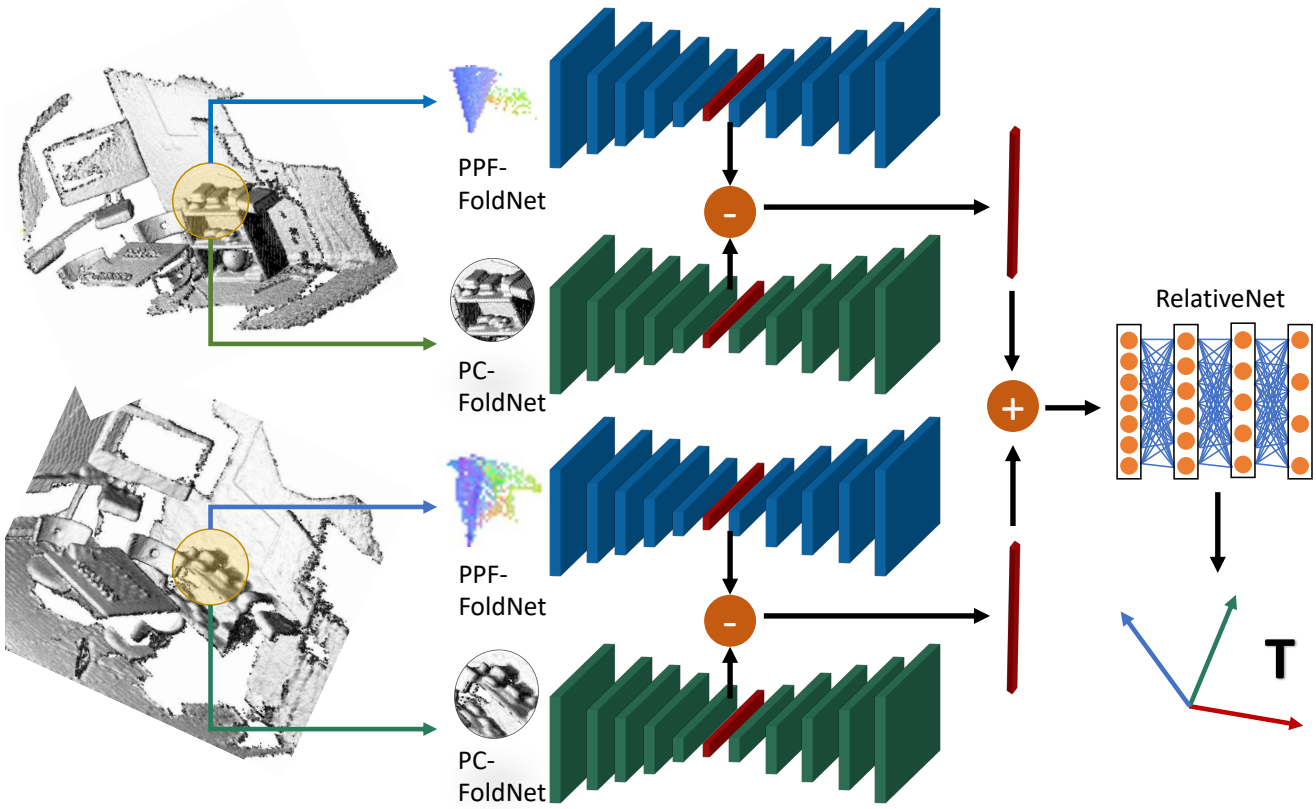

Figure 2. Overview of proposed pipeline. Given two point clouds, we first feed all the patches into PPF-FoldNet and PC-FoldNet auto-encoders to extract invariant and pose-variant local descriptors, respectively. Patch pairs are then matched by their intermediate invariant features. The pairs that are found to match are further processed to compute the discrepancy between invariant PPF-based features and PC-based features. These ratio features belonging to pairs of matching keypoints are concatenated and sent into RelativeNet, generating relative pose predictions. Multiple signals are imposed on reconstruction, pose prediction and feature consistency during the training stage.

ficient by the Super4PCS [37] and generalized by [38]. As we will elaborate in the upcoming sections, our approach lies at the intersection of local feature learning and direct pairwise registration inheriting the good traits of both.

3. Method

Purely geometric local patches typically carry two pieces of information: (1) 3D *structure*, summarized by the sample points themselves $\mathbf{P} = \{\mathbf{p}_i \,|\, \mathbf{p}_i \in \mathbb{R}^{N\times 3}\}$ where $\mathbf{p} = [x, y, z]^\top$ and (2) *motion*, which in our context corresponds to the 3D transformation or the *pose* $\mathbf{T}_i \in SE(3)$ holistically orienting and spatially positioning the point set $\mathbf{P}$:

$$SE(3) = \left\{ \mathbf{T} \in \mathbb{R}^{4\times 4} : \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \right\}. \qquad (1)$$

where $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$. A point set $\mathbf{P}_i$, representing a local patch is generally viewed as a transformed replica of its canonical version $\mathbf{P}_i^c$: $\mathbf{P}_i = \mathbf{T}_i \otimes \mathbf{P}_i^c$. Oftentimes, finding such a canonical absolute pose $\mathbf{T}_i$ from a single local patch involves computing *local reference frames* [44], that are known to be unreliable [41]. We instead base our idea on the premise that a good local (patch-wise) pose estimation leads to a good global rigid alignment of two fragments. First, by decoupling the pose component from the structure information, we devise a data driven predictor network capable of regressing the pose for arbitrary patches and showing good generalization properties. Fig. 2 depicts our architectural design. In a following part, we tackle the problem of relative pose labeling without the need for a canonical frame computation.

Generalized pose prediction A naive way to achieve tolerance to 3D-structure is to train the network for pose prediction conditioned on a database of input patches and leave the invariance up to the network [53, 19]. Unfortunately, networks trained in this manner either demand a very large collection of unique local patches or simply lack generalization. To alleviate this drawback, we opt to eliminate the structural components by training an invariant-equivariant network pair and using the intermediary latent space arithmetic. We characterize an equivariant function Ψ as [49]:

$$\Psi(\mathbf{P}) = \Psi(\mathbf{T} \otimes \mathbf{P}^c) = g(\mathbf{T})\Psi(\mathbf{P}^c) \qquad (2)$$

where $g(\cdot)$ is a function dependent only upon the pose. When $g(\mathbf{T}) = \mathbf{I}$, Ψ is said to be $\mathbf{T}$-*invariant* and for the scope of our application, for any input $\mathbf{P}$ leads to the outcome of the canonical one $\Psi(\mathbf{P}) \leftarrow \Psi(\mathbf{P}^c)$. Note that eq. (2) is more general than Cohen's definition [17] as the group element $\mathbf{T}$ is not restricted to act linearly. Within the body of this paper the term *equivariant* will loosely refer to such *quasi-equivariance* or *co-variance*. When $g(\mathbf{T}) \neq \mathbf{I}$, we further assume that the action of $\mathbf{T}$ can be approximated by some additive linear operation:

$$g(\mathbf{T})\Psi(\mathbf{P}^c) \approx h(\mathbf{T}) + \Psi(\mathbf{P}^c). \tag{3}$$

$h(\mathbf{T})$ being a probably highly non-linear function of $\mathbf{T}$. By plugging eq. (3) into eq. (2), we arrive at:

$$\Psi(\mathbf{P}) - \Psi(\mathbf{P}^c) \approx h(\mathbf{T}) \tag{4}$$

that is, the difference in the latent space can approximate the pose up to a non-linearity, h. We approximate the inverse of h by a four-layer MLP network $h^{-1}(\cdot) \triangleq \rho(\cdot)$ and propose to regress the motion (rotational) terms:

$$\rho(\mathbf{f}) \approx \mathbf{R} \,|\, \mathbf{t} \tag{5}$$

where $\mathbf{f} = \Psi(\mathbf{P}) - \Psi(\mathbf{P}^c)$. Note that $\mathbf{f}$ solely explains the motion and hence, can generalize to any local patch structure, leading to a powerful pose predictor under our mild assumptions.

The manifolds formed by deep networks are found sufficiently close to a Euclidean flatness. This rather flat nature has already motivated prominent works such as GANs [24] to use simple latent space arithmetic to modify faces, objects etc. Our assumption in eq. (3) follows a similar premise. Semantically speaking, by *subtracting out* the structure specific information from point cloud features, we end up with descriptors that are pose/motion-focused.

Relative pose estimation Note that $\rho(\cdot)$ can be directly used to regress the absolute pose to a canonical frame. Yet, due to the aforementioned difficulties of defining a unique local reference frame, it is not advised [41]. Since our scenario considers a pair of scenes, we can safely estimate a *relative pose* rather than the absolute, ousting the prerequisite for a nicely estimated LRF. This also helps us to easily forge the labels needed for training. Thus, we model $\rho(\cdot)$ by a relative pose predictor, *RelativeNet*, as shown in Fig. 2.

We further make the observation that, correspondent local structures of two scenes (i, j) that are well-registered under a rigid transformation $\mathbf{T}_{ij}$ also align well with $\mathbf{T}_{ij}$. As a result, the relative pose between local patches could be easily obtained by calculating the relative pose between the fragments and vice versa. We will use these ideas in the following section § 3.1 to design our networks, and in § 3.2 explain how to train them.

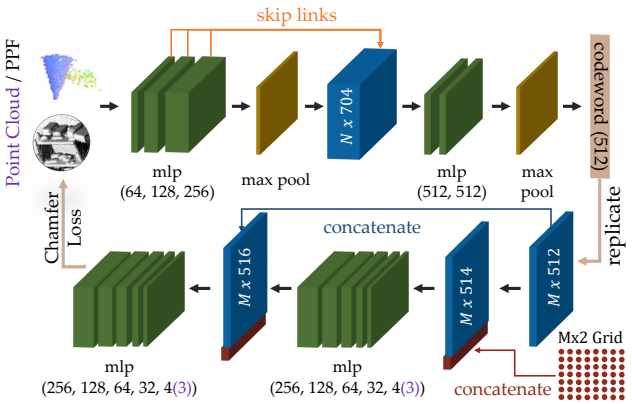

Figure 3. The architecture of PC/PPF-FoldNet. Depending on the input source, the number of last layers of unfolding module is 3 for point clouds and 4 for point pair features, respectively.

3.1. Network Design

To realize our generalized relative pose prediction, we need to implement three key components: the invariant network $\Psi(\mathbf{P}^c)$ where $g(\mathbf{T}) = \mathbf{I}$, the network $\Psi(\mathbf{P})$ that varies as a function of the input and the MLP $\rho(\cdot)$. The recent PPF-FoldNet [18] auto-encoder is luckily very suitable to model $\Psi(\mathbf{P}^c)$, as it is unsupervised, works on point patches and achieves true invariance thanks to the point pair features (PPF) fully marginalizing the motion terms. Interestingly, keeping the network architecture identical as PPF-FoldNet, if we were to substitute the PPF part with the 3D points themselves ($\mathbf{P}$), the intermediate feature would be dependent upon both structure and pose information. We coin this version as *PC-FoldNet* and use it as our equivariant network $\Psi(\mathbf{P}) = g(\mathbf{T})\Psi(\mathbf{P}^c)$. We rely on using PPF-FoldNet and PC-FoldNet to learn rotation-invariant and -variant features respectively. They share the same architecture while take in a different encoding of local patches, as shown in Fig. 3. Taking the difference of the encoder outputs of the two networks, i.e. the latent features of PPF- and PC-FoldNet respectively, results in new features which specialize almost exclusively on the pose (motion) information. Those features are subsequently fed into the generalized pose predictor RelativeNet to recover the rigid relative transformation. The overall architecture of our complete relative pose prediction is illustrated in Fig. 2.

3.2. Multi-Task Training Scheme

We train our networks with multiple cues, supervised and unsupervised. In particular, our loss function L is composed of three parts:

$$L = L_{rec} + \lambda_1 L_{pose} + \lambda_2 L_{feat} \tag{6}$$

L_{rec}, L_{pose} and L_{feat} are the reconstruction, pose prediction and feature consistency losses, respectively. For the sake of clarity, we omit the function arguments.

Reconstruction loss L_{rec} reflects the reconstruction fidelity of PC/PPF-FoldNet. To enable the encoders of PPF/PC-FoldNet to generate good features for pose regression, as well as for finding robust local correspondences, similar to the steps in PPF-FoldNet[18], use the *Chamfer Distance* as the metric to train the both of the auto-encoders in an unsupervised manner:

$$L_{rec} = \frac{1}{2}\Big(d_{cham}(\mathbf{P}, \hat{\mathbf{P}}) + d_{cham}(\mathbf{F}_{ppf}, \hat{\mathbf{F}}_{ppf})\Big) \tag{7}$$

$$d_{cham}(\mathbf{X}, \hat{\mathbf{X}}) = \tag{8}$$
$$\max\left\{\frac{1}{|\mathbf{X}|}\sum_{\mathbf{x}\in\mathbf{X}}\min_{\hat{\mathbf{x}}\in\hat{\mathbf{x}}}\|\mathbf{x}-\hat{\mathbf{x}}\|_2,\ \frac{1}{|\hat{\mathbf{X}}|}\sum_{\hat{\mathbf{x}}\in\hat{\mathbf{X}}}\min_{\mathbf{x}\in\mathbf{X}}\|\mathbf{x}-\hat{\mathbf{x}}\|_2\right\}.$$

$\hat{}$ operator denotes the reconstructed (estimated) set and $\mathbf{F}_{ppf}$ the PPFs of the points computed identically as [18].

Pose prediction loss A correspondence of two local patches are centralized and normalized before being sent into PC/PPF-FoldNets. This cancels the translational part $\mathbf{t} \in \mathbb{R}^3$. The main task of our pose prediction loss is then to enable our RelativeNet to predict the relative rotation $\mathbf{R}_{12} \in SO(3)$ between given patches $(1, 2)$. Hence, a natural choice for L_{pose} describes the discrepancy between the predicted and the ground truth rotations:

$$L_{pose} = \|\mathbf{q} - \mathbf{q}^*\|_2 \tag{9}$$

Note that we choose to parameterize the spatial rotations by quaternions $\mathbf{q} \in \mathbb{H}_1$, the Hamiltonian 4-tuples [10, 8] due to: 1) decreased the number of parameters to regress, 2) lightweight projection operator - vector-normalization.

Translation $\mathbf{t}^*$, conditioned on the hypothesized pair $(\mathbf{p}_1, \mathbf{p}_2)$ and the predicted rotation $\mathbf{q}^*$ can be computed by:

$$\mathbf{t}^* = \mathbf{p}_1 - \mathbf{R}^*\mathbf{p}_2 \tag{10}$$

where $\mathbf{R}^*$ corresponds to the matrix representation of $\mathbf{q}^*$. Such an L2 error is easier to train with negligible loss compared to the geodesic metric.

Feature consistency loss Unlike [18], our RelativeNet requires pairs of local patches for training. Thus, we can additionally make use of pair information as an extra *weak supervision* signal to further facilitate the training of our PPF-FoldNet. We hypothesize that such guidance would improve the quality of intermediate latent features that were previously trained in a fully unsupervised fashion. In specific, correspondent features subject to noise, missing data or clutter would generate a high reconstruction loss causing the local features to be different even for the same local patches. This new information helps us to guarantee that the features extracted from identical patches live as close as possible in the embedded space, which is extremely beneficial since we establish local correspondences by searching their nearest neighbor in the feature space. The *feature consistency loss* L_{feat} reads:

$$L_{feat} = \sum_{(\mathbf{p}_i, \mathbf{q}_i)\in\mathbf{\Gamma}} \|\mathbf{f}_{\mathbf{p}_i} - \mathbf{f}_{\mathbf{q}_i}\|_2 \tag{11}$$

$\mathbf{\Gamma}$ represents the set of correspondent local patches and $\mathbf{f}_\mathbf{p}$ is the feature extracted at $\mathbf{p}$ by the PPF-FoldNet, $\mathbf{f}_\mathbf{p} \in \mathbf{F}_{ppf}$.

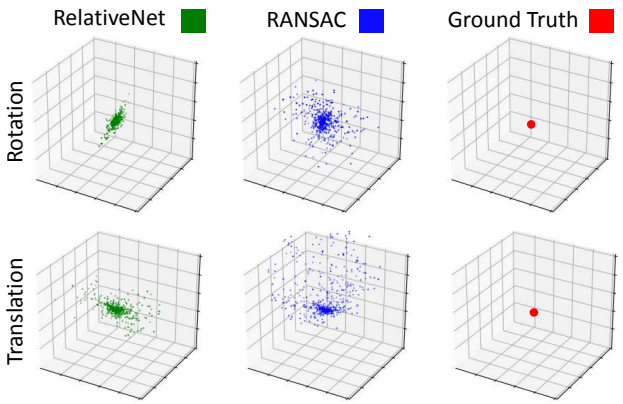

Figure 4. Comparison between the hypotheses generated by our Direct Prediction and RANSAC pipeline. The first row shows the rotational component as 3D Rodrigues vectors, and the second row shows the translational component. Hypotheses generated by our RelativeNet are more centralized around the ground truth.

3.3. Hypotheses Selection

The final stage of our algorithm involves selecting the best hypotheses among many, produced per each sample point. The full 6DoF pose is parameterized by the predicted 3DoF orientation (eq. (9)) and the translation (eq. (10)) conditioned on matching points (3DoF). For our approach, having a set of correspondences is equivalent to having a pre-generated set of transformation hypotheses since each keypoint is associated an LRF. Note that this is contrary to the standard RANSAC approaches where $m = 3$-correspondences parameterize the pose, and establishing N correspondences can lead to $\binom{N}{m}$ hypotheses to be verified. Our small number of hypotheses, already linear in the number of correspondences, makes it possible to exhaustively evaluate the putative matching pairs for verification. We further refine the estimate by recomputing the transformation using all the surviving inliers. The hypothesis with the highest score would be kept as the final decision.

Fig. 4 shows that both translational and rotational components of our hypothesis set are tighter and have smaller deviation from the true pose as opposed to the standard RANSAC hypotheses.

Table 1. Results on 3DMatch benchmark for fragment matching recall [53, 18].

	Kitchen	Home 1	Home 2	Hotel 1	Hotel 2	Hotel 3	Study	MIT Lab	Average
3DMatch [53]	0.5751	0.7372	0.7067	0.5708	0.4423	0.6296	0.5616	0.5455	0.5961
CGF [32]	0.4605	0.6154	0.5625	0.4469	0.3846	0.5926	0.4075	0.3506	0.4776
PPFNet [19]	**0.8972**	0.5577	0.5913	0.5796	0.5769	0.6111	0.5342	0.6364	0.6231
FoldingNet [51]	0.5949	0.7179	0.6058	0.6549	0.4231	0.6111	0.7123	0.5844	0.613
PPF-FoldNet [18]	0.7352	0.7564	0.625	0.6593	0.6058	0.8889	0.5753	0.5974	0.6804
Ours	0.7964	**0.8077**	**0.6971**	**0.7257**	**0.6731**	**0.9444**	**0.6986**	**0.6234**	**0.7458**

4. Experiments

We train our method using the training split of the defacto 3DMatch benchmark dataset [53], containing lots of real local patch pairs with different structure and pose, captured by Kinect cameras. We then conduct evaluations on its own test set and on the challenging synthetic Redwood Benchmark [13]. We assess our performance against the state of the art data-driven algorithms as well as the prosperous handcrafted methods of the RANSAC-family on the tasks of feature matching and geometric registration.

Implementation details We represent a local patch by randomly collecting 2K points around a reference one within 30cm vicinity. To provide relative pose supervision, we associate each patch a pose fetched from the ground truth relative transformations. Local correspondences are established by finding the mutually closest neighbors in the feature space. Our implementation is based on PyTorch [40], a widely used deep learning framework.

4.1. Evaluations on 3D Match Benchmark [53]

How good are our local descriptors? We begin by putting our local features at test for fragment matching task, which reflects how many good correspondence sets could be found by the specific features. A fragment pair is said to match if a true correspondence ratio of 5% and above is achieved. See [18, 19] for details. In Tab. 1 we report the recall of various data driven descriptors, 3DMatch [53], CGF [32], PPFNet [19], FoldingNet [51], PPF-FoldNet [18], as well as ours. It is remarkable to see that our network outperforms the supervised PPFNet [19] by $\sim$ 12% and the unsupervised PPF-FoldNet [18] by $\sim$ 6%. Note that, we are architecturally identical to PPF-FoldNet and hence the improvement is enabled primarily by the multi-task training signals, interacting towards a better minimum and decoupling of the shape and pose within the architecture. Thanks to the double-siamese structure of our network, we can provide both rotation-invariant features like [18], or upright ones, similar to [19].

How useful are our features in geometric registration? To further demonstrate the superiority of our learned local features, we evaluate them for the task of local geometric registration (L-GM). In a typical L-GM pipeline, local features are first extracted and then a set of local correspondences are established by some form of a search in the latent space. Out of these putative matches, a subsequent RANSAC iteratively selects a subset of minimally 3 correspondences in order to estimate a rigid pose. The best relative rigid transformation between the fragment pair is then the one with the highest inlier score. For the sake of fairness among all the methods and to have a controlled setting where the result depends only on the differences in descriptors, we use the simple RANSAC framework [42] across all methods to find the best matches.

The first part of Tab. 2 shows how well different local features could aid RANSAC to register fragments on the 3DMatch Benchmark. Recall and precision are computed the same way as in 3DMatch [53]. For this evaluation, recall is a more important measure, because the precision can be improved by employing better hypothesis pruning schemes filtering out the bad matches without harming recall [33, 32]. The registration result shows that our method is on par with or better than the best performer PPFNet [19] on average recall, while using a much more light-weighted training pipeline. Interestingly, our recall on this task drops when compared to the one of the fragment matching. This means that for certain fragment pairs, even though the inlier ratio is above 5%, RANSAC fails to do the work. Thus, one is motivated to seek better ways to recover the rigid transformation from 3D correspondences.

How accurate is our direct 6D prediction? We now evaluate the contributions of RelativeNet in fixing the aforementioned breaking cases of RANSAC. Thanks to our architecture, we are able to endow each correspondence with a pose information. Normally, each of these correspondences are expected to be good. However, in practice this is not the case. Hence, we devise a linear search to find the best of those, as explained in § 3.3. In Tab. 2 (bottom), we report our L-GM results as an outcome of this verification, on the same 3DMatch Benchmark. As we can see, with the same set of correspondences, our method could yield a much higher recall, reaching up to 77.68%, around 8%

Table 2. Geometric registration performance comparison. The first part lists the performances of some state-of-the-art deeply learned local features combined with RANSAC. The second part shows the performances of our features combined with RANSAC and its variants. The third part shows the results of our features combined with our pose prediction module directly. Not only our learned features are more powerful, but also our pose prediction module demonstrates superiority over RANSAC family.

			Kitchen	Home 1	Home 2	Hotel 1	Hotel 2	Hotel 3	Study	MIT Lab	Average
Different Feautures + RANSAC	3DMatch [53]	Rec.	0.8530	0.7830	0.6101	0.7857	0.5897	0.5769	0.6325	0.5111	0.6678
		Prec.	0.7213	0.3517	0.2861	0.7186	0.4144	0.2459	0.2691	0.2000	0.4009
	CGF [32]	Rec.	0.7171	0.6887	0.4591	0.5495	0.4872	0.6538	0.4786	0.4222	0.5570
		Prec.	0.5430	0.1830	0.1241	0.3759	0.1538	0.1574	0.1605	0.1033	0.2251
	PPFNet [19]	Rec.	**0.9020**	0.5849	0.5723	0.7473	0.6795	0.8846	0.6752	0.6222	0.7085
		Prec.	0.6553	0.1546	0.1572	0.4159	0.2181	0.2018	0.1627	0.1267	0.2615
Our Features + RANSAC variants	USAC [42]	Rec.	0.8820	0.7642	0.6101	0.7527	0.6538	0.8077	0.6709	0.5778	0.7149
		Prec.	0.5083	0.1397	0.1362	0.2972	0.1536	0.1329	0.1530	0.1053	0.2033
	SPRT [15]	Rec.	0.8797	0.7453	0.6101	0.7253	0.6538	0.8462	0.6624	0.4444	0.6959
		Prec.	0.5170	0.1341	0.1374	0.3158	0.1599	0.1384	0.1593	0.0881	0.2062
	LR [33]	Rec.	0.8753	0.7925	0.6038	0.7198	**0.7051**	0.7692	0.6667	0.5556	0.7110
		Prec.	0.5019	0.1348	0.1294	0.2854	0.1549	0.1190	0.1465	0.1012	0.1967
	RAN SAC	Rec.	0.8530	0.7642	0.6038	0.7033	0.6667	0.7692	0.6496	0.5111	0.6901
		Prec.	0.5527	0.1614	0.1479	0.3647	0.1825	0.1587	0.1658	0.1139	0.2309
Our Features + Pose Prediction		Rec.	0.8998	**0.8302**	**0.6352**	**0.8242**	0.6923	**0.9231**	**0.7650**	**0.6444**	**0.7768**
		Prec.	0.5437	0.1778	0.1807	0.4011	0.2061	0.2087	0.1843	0.1465	0.2561

higher than what is achievable by RANSAC. This is 7% higher than PPFNet. Also, this number is around 3% higher than the recall in fragment matching, which means that not only pairs with good correspondences are registered, but also some challenging pairs with even less than 5% inlier ratio are successfully registered, pushing the potential of matched correspondences to the limit.

It is noteworthy to point out that the iterative scheme of RANSAC requires finding at least 3 correct correspondences to estimate $\mathbf{T}$, whereas it is sufficient for us to rely on a single correct match. Moreover due to downsampling [7], poses computed directly from 3-points are crude, whereas patch-wise pose predictions of our network are less prone to the accuracy of exact keypoint location.

Comparisons against the RANSAC-family To further demonstrate the power of RelativeNet, we compare it with some of the state-of-the-art variants of RANSAC, namely USAC [42], SPRT [15] and Latent RANSAC (LR) [33]. Those methods are proved to be both faster and more powerful than the vanilla version [42, 33].

All the methods are given the same set of putative matching points found by our rotation-invariant features. The results depicted in Tab. 2 shows that even a simple hypothesis prunning combined with our data driven RelativeNet can surpass an entire set of hand-crafted methods, achieving approximately 6.19% higher reacall than the best obtained by USAC and 2.61% better than the highest precision obtained by standard RANSAC. In this regard, our method takes a dominant advantage on 3D pairwise geometric registration.

Running times Speed is another important factor regarding any pairwise registration algorithm and it is of interest to see how our work compares to the state of the art in this aspect. We implement our hypotheses verification part based on USAC to make the comparison fair with other USAC-based implementations.

The average time needed for registering a fragment pair is recorded in Tab. 3, feature extraction time excluded. All timings are done on a Intel(R) Core(TM) i7-4820K CPU @ 3.70GHz with a single thread. Note that, our method is much faster than the fastest RANSAC-variant *Latent-RANSAC* [33]. The average time for generating all hypotheses for a fragment pair by RelativeNet is about 0.013s, and the subsequent verification costs 0.016s, making up around 0.03s in total. An important reason why we can terminate so quickly is that the number of hypotheses generated and verified is much smaller compared to the RANSAC methods. While LR is capable of reducing this amount significantly, the number of surviving hypotheses to be verified is still much more than ours.

Table 3. The average runtime for registering one fragment pair and the number of hypotheses generated and verified.

	USAC [42]	SPRT [15]	LR [33]	Ours
Time(s)	0.886	2.661	0.591	0.013 + 0.016
# Hypos	30220	672223	2568 (46198)	335

Effect of correspondence estimation on the registration We put 5 different ways to constructing putative matching pair sets under an ablation study. Strategies include: (1) keeping different number of mutual closest neighboring

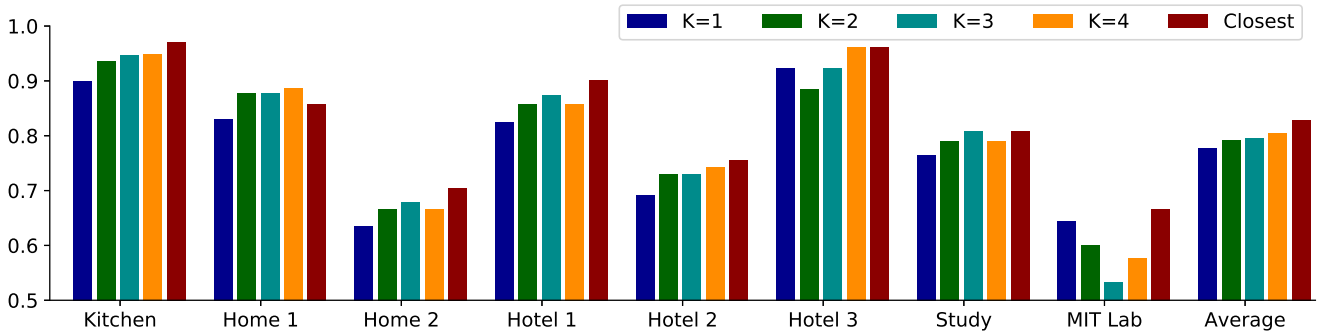

Figure 5. The impact of using different methods to find correspondences. As the number of mutual correspondences kept, K, increases, more hypotheses are verified leading to a trade-off between recall and computation time.

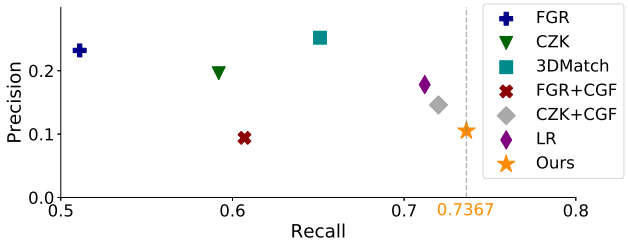

Figure 6. Geometric registration performance of various methods on Redwood Benchmark [13].

patches $k = 1 \dots 4$, each dubbed as $K = k$ and (2) keeping a nearest neighbor for all the local patches from both fragments as a match pair, dubbed *Closest*. These strategies are applied on the same set of local features to estimate initial correspondences for further registration. The results of each method on different scenes and their average are plotted in Fig. 5. As k increases and the criteria for accepting a neighbor to be a pair relaxes, we observe an overall trend of increasing registration recall on different sequences. Not surprisingly, this trend is most obvious in the *Average* column. This is of course not sufficient to conclude that relaxation helps correspondences. The second important observation is that the number of established correspondences also increases as this condition relaxes. The average amount of putative matches found by *Closest* is around 3664, much larger than $K = 1$'s 334, approximately 10 times more, meaning that a subsequent verification would need more time to process them. Hence, we arrive at the conclusion that if recall/accuracy is the main concern, more putative matches should be kept. If, conversely, speed is an issue, *Mutual-1* could achieve a rather satisfying result quicker.

Generalization to unseen domains To show that our algorithm could generalize well to other datasets, we evaluate its performance on the well-known and challenging global registration benchmark provided by Choi *et al*., the Redwood Benchmark [13]. This dataset contains four different synthetic scenes with sequence of fragments. Our network is not fine-tuned with any synthetic data, instead, the weights trained with real data from 3DMatch dataset is used directly. We follow the evaluation settings as Choi *et al*. for an easy and fair comparison, and report the registration results in Fig. 6. This precision and recall plot also depcits results achieved by some recent methods including FGR [55], CZK [13], 3DMatch [53], CGF+FGR [32], CGF+CZK [32], and Latent-Ransac [33]. Among them, 3DMatch and CGF are data-driven. 3DMatch was trained with real data on the same data source as ours, while CGF trained with synthetic data. Note that our method shows $\sim 8.5\%$ higher recall against 3DMatch. Although we are not using any synthetic data for finetuning, we still achieve a better recall of 2.4% w.r.t. CGF and its combination with CZK. In general, our method outperforms all the other state-of-the-art methods on Redwood Benchmark [13], which validates the generalizability and good performance of our method simultaneously. Note that while in general, the maximal precision is low across all the methods, it is not hard to improve it when the recall is high. To show that recall is the primary measure, we ran a global optimization [13] on our initial results, bringing precision up to 91% without big loss of recall - still at 73%.

5. Conclusion

We proposed a unified end-to-end framework for both local feature extraction and pose prediction. Comprehensive experiments on 3DMatch benchmark demonstrate that a multi-task training scheme could inject more power into the learned features, hence improve the quality of the correspondence set for further registration. Geometric registration using the pose predictions by our RelativeNet given the putative matched pairs is also shown to be both more robust and much faster than various state-of-the-art RANSAC methods. We also studied how different methods of establishing local correspondences would affect the registration performance. The outstanding performance on the challenging synthetic Redwood benchmark strongly validates that our method is not only robust, but also generalizes well to unseen datasets. In the future, we also plan to introduce a data driven hypotheses verification approach.

References

[1] D. Aiger, N. J. Mitra, and D. Cohen-Or. 4-points congruent sets for robust pairwise surface registration. In *ACM Transactions on Graphics (TOG)*, volume 27, page 85. ACM, 2008.

[2] P. J. Besl and N. D. McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[3] T. Birdal, E. Bala, T. Eren, and S. Ilic. Online inspection of 3d parts via a locally overlapping camera network. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10. IEEE, 2016.

[4] T. Birdal and S. Ilic. Point pair features based object detection and pose estimation revisited. In *3D Vision*, pages 527–535. IEEE, 2015.

[5] T. Birdal and S. Ilic. Cad priors for accurate and flexible instance reconstruction. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 133–142, Oct 2017.

[6] T. Birdal and S. Ilic. A point sampling algorithm for 3d matching of irregular geometries. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6871–6878. IEEE, 2017.

[7] T. Birdal and S. Ilic. A point sampling algorithm for 3d matching of irregular geometries. In *International Conference on Intelligent Robots and Systems (IROS 2017)*. IEEE, 2017.

[8] T. Birdal, U. Simsekli, M. O. Eken, and S. Ilic. Bayesian pose graph optimization via bingham distributions and tempered geodesic mcmc. In *Advances in Neural Information Processing Systems*, pages 306–317, 2018.

[9] M. Bueno, F. Bosché, H. González-Jorge, J. Martínez-Sánchez, and P. Arias. 4-plane congruent sets for automatic registration of as-is 3d point clouds with 3d bim models. *Automation in Construction*, 89:120–134, 2018.

[10] B. Busam, T. Birdal, and N. Navab. Camera pose filtering with local regression geodesics on the riemannian manifold of dual quaternions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2436–2445, 2017.

[11] Á. P. Bustos and T.-J. Chin. Guaranteed outlier removal for point cloud registration with correspondences. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2868–2882, 2018.

[12] S. Choi, T. Kim, and W. Yu. Performance evaluation of ransac family. *Journal of Computer Vision*, 24(3):271–300, 1997.

[13] S. Choi, Q.-Y. Zhou, and V. Koltun. Robust reconstruction of indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[14] O. Chum and J. Matas. Matching with prosac-progressive sample consensus. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 220–226. IEEE, 2005.

[15] O. Chum and J. Matas. Optimal randomized ransac. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1472–1482, 2008.

[16] O. Chum, J. Matas, and J. Kittler. Locally optimized ransac. In *Joint Pattern Recognition Symposium*, pages 236–243. Springer, 2003.

[17] T. Cohen and M. Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999, 2016.

[18] H. Deng, T. Birdal, and S. Ilic. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[19] H. Deng, T. Birdal, and S. Ilic. Ppfnet: Global context aware local features for robust 3d point matching. *Computer Vision and Pattern Recognition (CVPR). IEEE*, 1, 2018.

[20] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 998–1005. Ieee, 2010.

[21] B. Eckart, K. Kim, and J. Kautz. Hgmr: Hierarchical gaussian mixtures for adaptive 3d registration. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[22] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 1981.

[23] Z. Gojcic, C. Zhou, J. D. Wegner, and W. J. D. The perfect match: 3d point cloud matching with smoothed densities. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[25] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and J. Zhang. Performance evaluation of 3d local feature descriptors. In *Asian Conference on Computer Vision*, pages 178–194. Springer, 2014.

[26] S. Hinterstoisser, V. Lepetit, N. Rajkumar, and K. Konolige. Going further with point pair features. In *European Conference on Computer Vision*, pages 834–848. Springer, 2016.

[27] T. Hodaň, F. Michel, E. Brachmann, W. Kehl, A. G. Buch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, C. Sahin, F. Manhardt, F. Tombari, T.-K. Kim, J. Matas, and C. Rother. Bop: Benchmark for 6d object pose estimation. In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision – ECCV 2018*, pages 19–35, 2018.

[28] Z. Jian Yew and G. Hee Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 607–623, 2018.

[29] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on pattern analysis and machine intelligence*, 21(5):433–449, 1999.

[30] F. Järemo Lawin, M. Danelljan, F. Shahbaz Khan, P.-E. Forssén, and M. Felsberg. Density adaptive point set registration. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[31] W. Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.

[32] M. Khoury, Q.-Y. Zhou, and V. Koltun. Learning compact geometric features. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[33] S. Korman and R. Litman. Latent ransac. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6693–6702, 2018.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[35] H. Li and R. Hartley. The 3d-3d registration problem revisited. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[36] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[37] N. Mellado, D. Aiger, and N. J. Mitra. Super 4pcs fast global pointcloud registration via smart indexing. In *Computer Graphics Forum*, volume 33, pages 205–215. Wiley Online Library, 2014.

[38] M. Mohamad, M. T. Ahmed, D. Rappaport, and M. Greenspan. Super generalized 4pcs for 3d registration. In *3D Vision (3DV), 2015 International Conference on*, pages 598–606. IEEE, 2015.

[39] J. Park, Q.-Y. Zhou, and V. Koltun. Colored point cloud registration revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 143–152, 2017.

[40] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NIPS-Workshops*, 2017.

[41] A. Petrelli and L. Di Stefano. On the repeatability of the local reference frame for partial shape matching. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2244–2251. IEEE, 2011.

[42] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. Usac: a universal framework for random sample consensus. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):2022–2038, 2013.

[43] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.

[44] S. Salti, F. Tombari, and L. Di Stefano. Shot: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125:251–264, 2014.

[45] R. Toldo, A. Beinat, and F. Crosilla. Global registration of multiple point clouds embedding the generalized procrustes analysis into an icp framework. In *3DPVT 2010 Conference*, 2010.

[46] F. Tombari, S. Salti, and L. Di Stefano. Unique shape context for 3d data description. In *Proceedings of the ACM workshop on 3D object retrieval*, pages 57–62. ACM, 2010.

[47] J. Vidal, C.-Y. Lin, and R. Martí. 6d pose estimation using an improved method based on point pair features. In *2018 4th International Conference on Control, Automation and Robotics (ICCAR)*, pages 405–409. IEEE, 2018.

[48] J. Vongkulbhisal, B. I. Ugalde, F. De la Torre, and J. P. Costeira. Inverse composition discriminative optimization for point cloud registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2993–3001, 2018.

[49] D. Worrall and G. Brostow. Cubenet: Equivariance to 3d rotation and translation. In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision – ECCV 2018*, 2018.

[50] J. Yang, H. Li, and Y. Jia. Go-icp: Solving 3d registration efficiently and globally optimally. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1457–1464, 2013.

[51] Y. Yang, C. Feng, Y. Shen, and D. Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[52] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483. Springer, 2016.

[53] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017.

[54] Y. Zhao, T. Birdal, H. Deng, and F. Tombari. 3d point-capsule networks. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[55] Q.-Y. Zhou, J. Park, and V. Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016.