remote sensing

MDPI

Review

Review on Deep Learning Algorithms and Benchmark Datasets for Pairwise Global Point Cloud Registration

Yang Zhao and Lei Fan *

Department of Civil Engineering, Design School, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China
* Correspondence: lei.fan@xjtlu.edu.cn

Abstract: Point cloud registration is the process of aligning point clouds collected at different locations of the same scene, which transforms the data into a common coordinate system and forms an integrated dataset. It is a fundamental task before the application of point cloud data. Recent years have witnessed the rapid development of various deep-learning-based global registration methods to improve performance. Therefore, it is appropriate to carry out a comprehensive review of the more recent developments in this area. As the developments require access to large benchmark point cloud datasets, the most widely used public datasets are also reviewed. The performance of deep-learning-based registration methods on the benchmark datasets are summarized using the reported performance metrics in the literature. This forms part of a critical discussion of the strengths and weaknesses of the various methods considered in this article, which supports presentation of the main challenges currently faced in typical global point cloud registration tasks that use deep learning methods. Recommendations for potential future studies on this topic are provided.

Keywords: registration; point cloud; deep learning; benchmark; dataset; feature descriptor; performance

Citation: Zhao, Y.; Fan, L. Review on Deep Learning Algorithms and Benchmark Datasets for Pairwise Global Point Cloud Registration. *Remote Sens.* **2023**, *15*, 2060. https://doi.org/10.3390/rs15082060

Academic Editors: Sara Gonizzi Barsanti and Sajjad Roshandel

Received: 8 March 2023
Revised: 7 April 2023
Accepted: 11 April 2023
Published: 13 April 2023

1. Introduction

Registration aligns point clouds from one source, multiple sources, or multiple epochs in a common coordinate system. It plays an indispensable role in most point cloud applications, such as documentation, change detection, deformation measurement and navigation, and its accuracy directly determines the quality of these applications. Typical applications of point clouds in documentation include the reconstruction of cultural heritage [1], 'as-built' and 'as-is' conditions of building information modelling [2–4], and forest inventories [5]. Applications involving change detection or quantitative deformation estimation can be realized by comparing point clouds obtained at multiple epochs; they include excavation volume estimation [6–8], deformation monitoring of buildings [9,10] and civil structures [11–13], and quantification of landform variations of terrain surfaces [14–16]. Registered point clouds are also widely used to assist with the navigation of robots [17] and autonomous driving vehicles [18], for which not only accuracy, but also real-time efficiency of registration, are considered.

Among various forms of registration, the pairwise global registration of point clouds from the same source is fundamental for the following reasons: (a) multi-view registration can be built on multiple pairwise registrations of every two-point cloud; (b) the accuracy of a local registration often depends on the quality of its proceeding global registration; (c) multi-modal registration is an extension from same-source registration. Unless otherwise stated, registration hereafter in this article generally refers to the pairwise global registration of same-source point clouds.

The mathematical representation of point cloud registration is given as follows: Suppose there are two partially overlapping point clouds $P = \{p_i \in \mathrm{R}^3\}_{i=1}^{M}$ and $Q = \{q_i \in \mathrm{R}^3\}_{i=1}^{N}$. P and Q are termed the source point cloud and the template point cloud, respectively. M and N represent the number of points in P and Q, respectively.

Suppose p_x and q_y are a matched point pair established by the mutual nearest neighbour search of point positions, and $\boldsymbol{C}_{gt}$ is a set containing all the matching pairs found in the overlapping area of $\boldsymbol{P}$ and $\boldsymbol{Q}$ when the ground truth registration is applied. The goal of registration is to find a rotation matrix $\boldsymbol{R} \in \mathcal{SO}(3)$ and a translation vector $\boldsymbol{t} \in \mathbb{R}^3$, by minimising the sum of the distances between all matched point pairs, as defined in

$$\arg \min_{R \in \mathcal{SO}(3),\, t \in \mathbb{R}^3} \frac{1}{\left|C_{gt}\right|} \sum_{(p_x, q_y) \in C_{gt}} \| \, Rp_x + t - q_y \, \|^2, \tag{1}$$

where $\left|C_{gt}\right|$ is the number of matched point pairs in C_{gt}.

In practice, the ground-truth matched point pair set C_{gt} is unknown. Therefore, many efforts towards a registration pipeline have been made for the estimation of C_{gt}. The accuracy of the estimated set of matched point pairs, C_{est}, is crucial for the performance of registration methods. C_{est} is usually obtained by matching the points of similar feature values from source and template point clouds. Therefore, the point feature extraction is a fundamental step in point cloud registration.

A point feature is often represented by a vector incorporating the information on the characteristics of data points in a neighbourhood. An algorithm that extracts point features from the neighbourhood geometric information is called a feature descriptor. 'Traditional' point cloud registration methods adopt manually defined feature descriptors [19–23], which are referred to as 'handcrafted' feature descriptors. The distinguishability of the handcrafted feature descriptors is limited, which could lead to spurious matching of points and, hence, inaccurate registration.

To overcome this limitation, in recent years, deep neural networks [24–28] have been proposed with the growth of various benchmark registration datasets [29–32] for more competent feature descriptors. Using deep neural networks as backbones, various 'deep' feature descriptors (e.g., [32–40]) have been trained and evaluated on various benchmark registration datasets. As shown in Figure 1, deep feature descriptors [32,34–36,38,40] outperformed traditional handcrafted feature descriptors [19–22] by a large margin in terms of feature match recall (FMR) on the 3DMatch dataset [32]. It should be noted that the FMR of the handcrafted feature descriptors [19–22] were cited from [33], where the handcrafted feature descriptors [19–22] were evaluated on the 3DMatch dataset as baselines for comparison. As reported in [41], some recently proposed handcrafted descriptors, such as SGC [42] and LoVS [43], performed better than their earlier counterparts, such as USC [21] and SHOT [22]. However, their performances have not been compared with those of deep-learning-based descriptors.

The trend in research interests in deep-learning-based point cloud registration may be represented by the number of relevant publications. Therefore, screening of relevant publications from 2017 to 2022 was conducted. Journal and conference publications were searched for using Google Scholar. The key words used for the search included point cloud, registration, and deep learning. However, this choice also resulted in the identification of publications focusing on other topics, such as point cloud segmentation. Therefore, a further manual screening was conducted after checking the abstracts of all search results. During the manual screening, it was found that many relevant studies were published in proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), the International Conference on Computer Vision (ICCV), and the European Conference on Computer Vision (ECCV) since 2017. Therefore, the abstracts of the articles in the proceedings of these three conferences published since 2017 were exhaustively browsed for any relevant publications missed by the search using the key words.

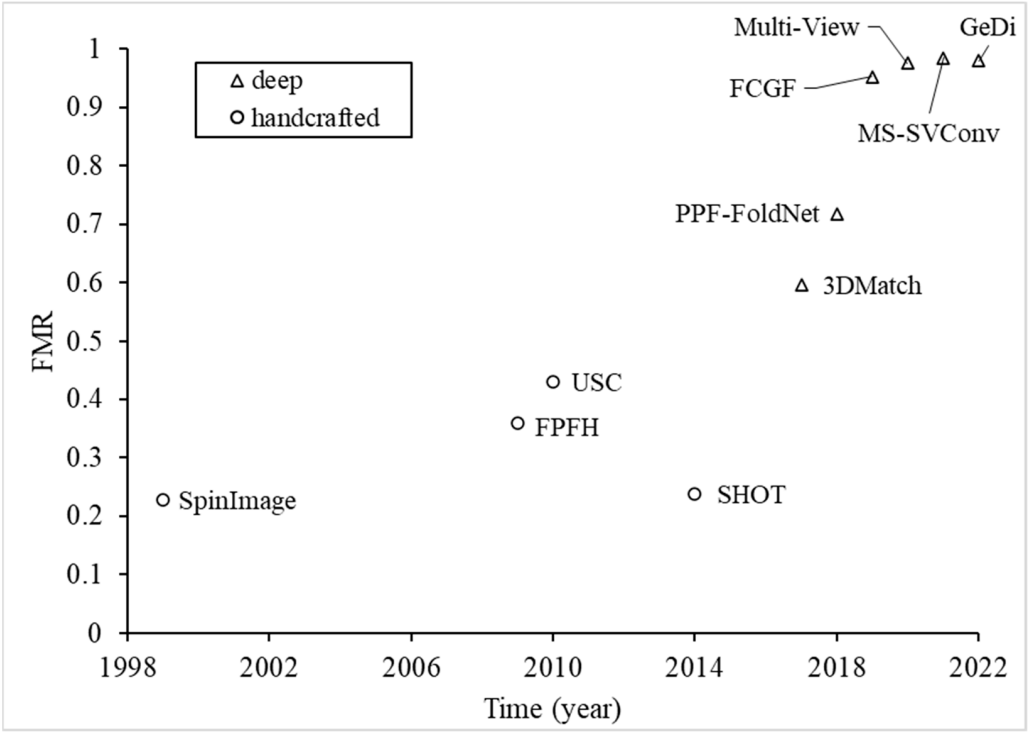

Figure 1. FMR for various representative feature descriptors of different years, based on the 3Dmatch dataset. In an individual year, when multiple deep feature descriptors were proposed, only that with the highest FMR is shown. The presented feature descriptors are SpinImage [19], FPFH [20], USC [21], SHOT [22], 3DMatch [32], PPF-FoldNet [34], FCGG [35], Multi-view [36], MS-SVConv [38], and GeDi [40].

As shown in Figure 2, the publication number increased gradually for the period from 2017 to 2021, during which 2021 witnessed a big increase in the number of publications. However, by the end of 2022, when this review was conducted, the number of publications seemingly suggested a decreasing interest in this topic. This was probably due to further improvements in the registration performance on the existing datasets being limited, as indicated in Figure 1. Therefore, it is a suitable time to review the current status quo and the progress of deep-learning-based point cloud registration algorithms.

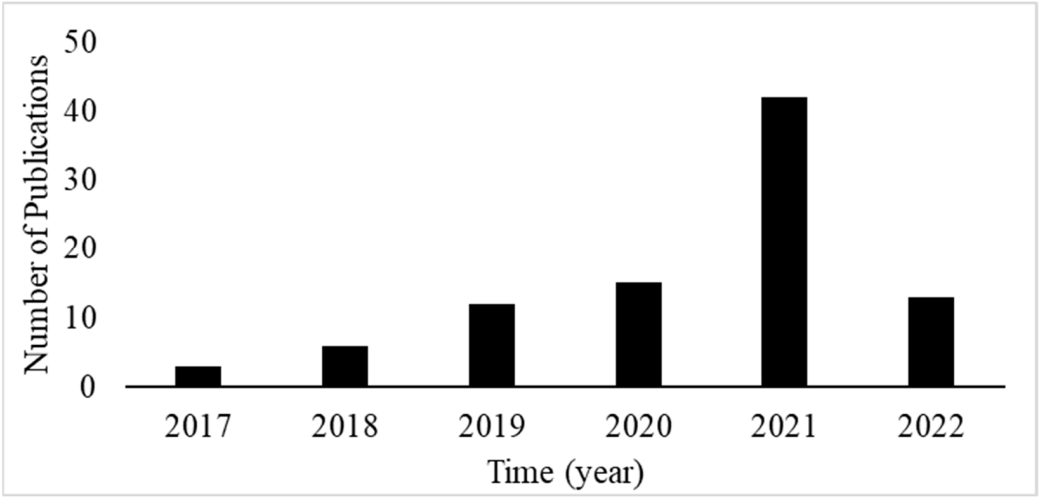

Figure 2. The number of annual publications on the deep-learning-based point cloud registration from years 2017–2022 based on our screening statistics.

Figure 3 visualizes a network of the keywords from the surveyed literature using VOSviewer software [44]. Challenging factors such as 'rotation', 'noise', 'point density', 'point order' and 'partial overlap' are shown. The figure also indicates that most of the pairwise global point cloud registration algorithms rely on the 'correspondence' of a point 'feature' extracted by a 'descriptor'. For developing deep-learning-based algorithms, a 'benchmark' 'dataset' is essential for training and testing. Recent research has focused on self-supervised and 'end-to-end' deep learning. Figure 3 shows a high frequency of occurrence of the phrase 'state-of-the-art', suggesting that 'state-of-the-art' performances have continuously been updated with new algorithms for improved registration 'accuracy' and 'robustness'. However, the rate of update is slowing down, as indicated by the number of publications in 2022.

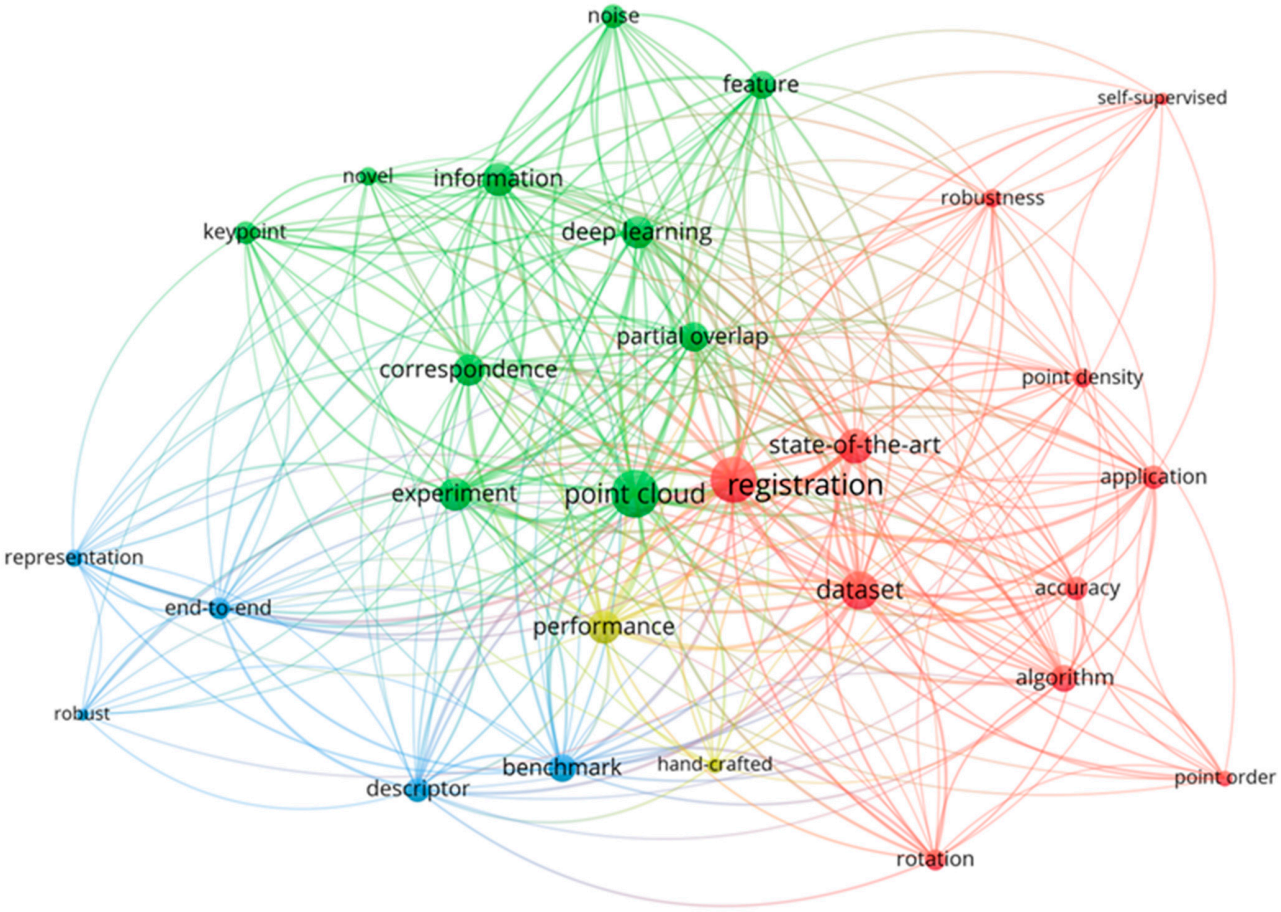

Figure 3. Keyword network of the surveyed literature.

There are several existing reviews [17,45–53] on point cloud registration algorithms from various perspectives. Some [17,45–48] have focused mainly on traditional methods. There are also a few reviews [49–53] on learning-based registration. However, due to their different focuses, they did not cover all algorithms or benchmark datasets for pairwise global point cloud registration. More importantly, the interpretations in these reviews were often centred on individual algorithms and the associated contributions. This may be less useful for audiences who wish to understand the advancements made to individual key steps involved in the whole pipeline of deep-learning-based pairwise global point cloud registration. The present review article comprehensively reviews the key steps of deep-learning-based methods for pairwise global point cloud registration, presents the advancements in those key steps, discusses the performance of different algorithms, and provides insights for future developments. In addition, the main benchmark registration datasets are also reviewed and discussed because the development and evaluation of

registration algorithms rely on those datasets. The review also presents the performance of various deep-learning-based registration methods on the benchmark point cloud datasets in the literatures considered.

The remainder of this article is structured as follows: Section 2 introduces two strategies of deep-learning-based point cloud registration, one of which is based on pose-invariant point features and the other on pose-variant point features. Section 3 introduces key steps of deep-learning-based registration relying on pose-invariant features. Section 4 covers the main benchmark datasets that are of use for the training and testing of deep-learning-based algorithms for point cloud registration. Section 5 summarizes the reported performance in the literature of these algorithms on the benchmark datasets. Section 6 discusses current challenges and suggests future research directions. In Section 7, the conclusions are drawn.

2. Deep-Learning-Based Point Cloud Registration Strategy

2.1. 'Hybrid' Methods Utilising 'Pose-Invariant' Features

Typically, a registration method, where some but not all steps are realized using deep neural networks, is termed a 'hybrid' registration method [54]. Many 'hybrid' registration algorithms share a similar pipeline of processing modules, typically comprising point feature descriptors, key point detection, feature matching, outlier rejection, and rigid motion estimation, as shown in Figure 4. Among these modules, deep-learning-based implementation is often considered for feature extraction, key point detection, and outlier rejection modules in the 'hybrid' methods.

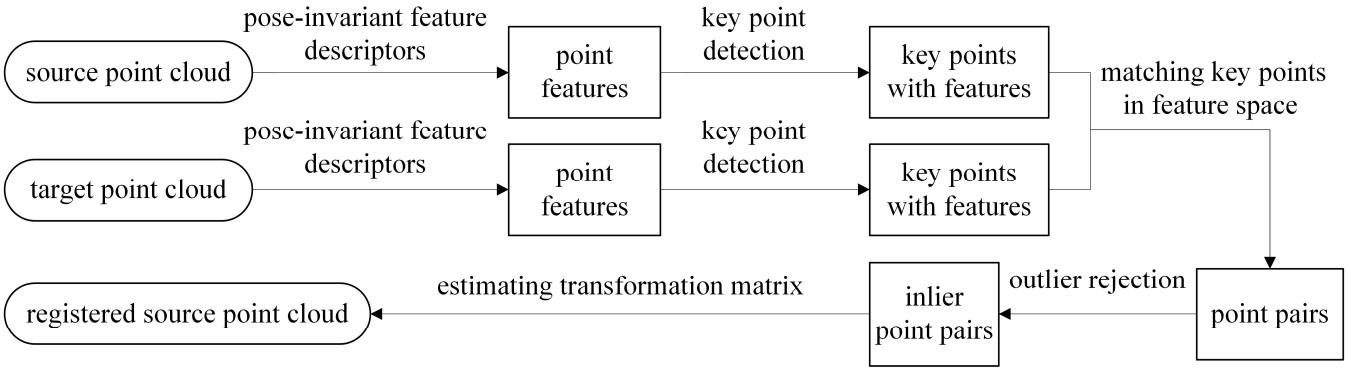

Figure 4. The pipeline of the 'hybrid' and 'end-to-end' algorithms relying on pose-invariant features.

A feature descriptor extracts local features of an input point cloud. The extracted features are designed to be pose-invariant, and, therefore, can be matched between point cloud pairs regardless of their differences in initial poses. The key point detection module estimates score for each point based on which key points are detected. It should be noted that, more often than not, the key point detection is based on the extracted feature values and, hence, is implemented after the feature descriptor, as indicated in Figure 4. However, some methods (e.g., [55]) detect key points directly from input point cloud data, in which the key point detection may be conducted parallel to, or even before, the feature descriptor. Feature matching yields initial matches based on the features of the detected key points. An outlier rejection algorithm rejects spurious matches out of the initial matches. Finally, the rigid motion parameters, which are the goal of a registration, can be estimated from the constraints provided by the coordinates of the matched point pairs. Among those modules, the feature descriptors, feature matching and motion estimation modules are mandatory to produce a registration. Key point detection and outlier rejection are optional and, hence, their corresponding arrow lines in Figure 4 are dashed.

Taking GeDi [40] (one of the best-performing 'hybrid' methods), for example, the following steps are involved. First, it extracts all points within a spherical neighbourhood of each (key) point in both source and template point clouds. These points form a point

cloud patch. Then, each point cloud patch is canonicalized, and the feature of the point cloud patch is extracted using PointNet++ [25]. Afterwards, the (key) points in a source point cloud are matched to the corresponding (key) points in a target point cloud (i.e., point pairs), based on the features of the point cloud patches that they are centering. Finally, the outliers in the corresponding (key) point pairs are rejected in a RANdom SAmple Consensus (RANSAC) [56] process and the rigid motion is estimated based on the spatial constraints provided by the correspondences of (key) point pairs. Note that GeDi does not incorporate a key point detection module and every point is treated as a key point.

2.2. 'End-to-End' Methods

Registration methods (e.g., [57–63]) where all essential registration steps are executed using deep neural networks are referred to as 'end-to-end' methods. Depending on the type of features used, 'end-to-end' methods can be categorized into the two categories described below.

2.2.1. 'End-to-End' Methods Utilising Pose-Invariant Features

The majority of 'end-to-end' methods conform to a pipeline similar to the 'hybrid' methods (see Figure 4), utilising the pose-invariant features. In addition, some 'end-to-end' (e.g., [57–60]) methods iterate the entire pipeline to refine the registration with the updated poses of the source point cloud.

Except for nearest neighbour search (NNS) used in the feature-matching module, and RANSAC [56] used in the outlier rejection module, the implementation of the other individual modules in the 'hybrid' methods is also compatible with 'end-to-end' methods. Singular value decomposition (SVD) [64] is the protocol motion estimation algorithm of both 'hybrid' and 'end-to-end' methods.

For example, REGTR [60] is one of the best-performing 'end-to-end' methods based on pose-invariant features. Firstly, the REGTR method utilizes the KPConv [28] convolutional backbone to extract features of key points in both source and template point clouds. These features are subsequently fed into multiple Transformer cross-encoder layers to condition the features of key points with context-awareness and positional encoding. The conditioned features of key points are passed into the output decoder to predict the overlap scores and the corresponding point pairs of the key points between source and template point clouds. The overlap scores serve to reject outliers, which are probably outside of the overlapping area between the source point cloud and the target point cloud. Finally, the rigid motion is estimated based on the spatial constraints provided by the corresponding point pairs.

2.2.2. 'End-to-End' Methods Utilising Pose-Variant Features

Some 'end-to-end' methods (e.g., PointNetLK [61], PCRNet [62], and RelativeNet [63]) seek to embed pose information in pose-variant local or global features of point clouds and to infer the rigid motion directly from the pose-variant features via regression algorithms [61–63,65], as shown in Figure 5.

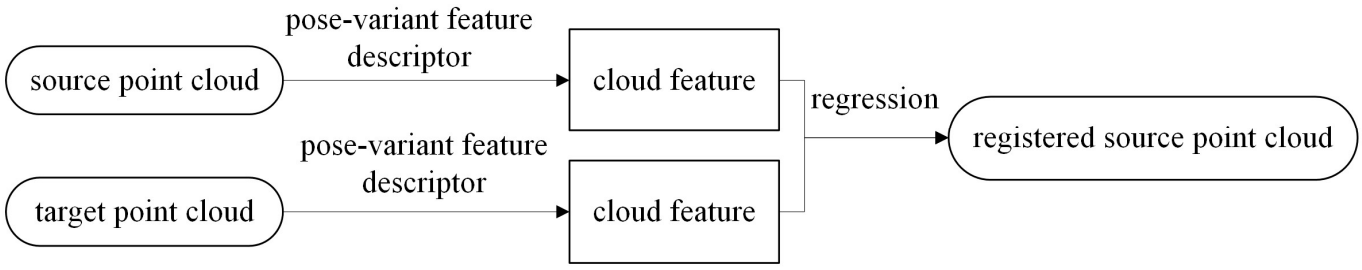

Figure 5. The pipeline of the registration that is based on pose-variant features.

The global features in PointNetLK [61] and PCRNet [62] are extracted by similar PointNet-like MLPs. It should be noted that the T-Net with a canonical orientation ability,

which is a part of the original PointNet, is excluded from the PointNetLK and PCRNet to ensure that the global features are pose-variant. However, a global feature learned in this way embeds both structure and pose information. Therefore, PointNetLK and PCRNet are not robust to point cloud pairs with low overlapping because the rigid motion estimation is confused by the differences in structure information.

In order to eliminate this confusion, a pose-specific feature, which ideally embeds only pose information, was designed in RelativeNet [63]. In RelativeNet, PPF-FoldNet [34] features and PC-FoldNet [66] features are extracted from each point cloud patch, respectively. The PPF-FoldNet feature is pose-invariant and only contains structure information. In contrast, PC-FoldNet is pose-variant and contains both structure and pose information. The discrepancy between PPF-FoldNet and PC-FoldNet features is deemed pose-specific (i.e., it contains only pose information) and is fed into a regression network for the rigid motion estimation.

2.2.3. Pose-Invariant Features Versus Pose-Variant Features

In most cases, pose-invariant features were used because they typically outperform pose-variant features. There are a very limited number of methods utilising pose-variant features, mainly including PointNetLK [61], PCRNet [62] RelativeNet [63], and OMNet [67]. It was observed that the performances of PointNetLK, PCRNet, and RelativeNet were significantly worse than those of pose-invariant methods, as reported in [18,40,60].

2.3. Performance Comparisons between 'Hybrid' and 'End-to-End' Methods

As the 3DMatch dataset has been the most widely used dataset for testing the performance of deep-learning-based methods, more comprehensive test results are available to facilitate direct comparisons between performances of different methods. The reported performances, in terms of registration recall (RR), of representative deep-learning-based methods on the 3DMatch dataset is illustrated in Figure 6. It should be noted that the 'hybrid' [32,33,35,68–70] and 'end-to-end' [59,60] methods shown in Figure 6 are those with the best performance(s) of each year. For more detailed information on the performances of all methods considered on various datasets, refer to Section 5.2.

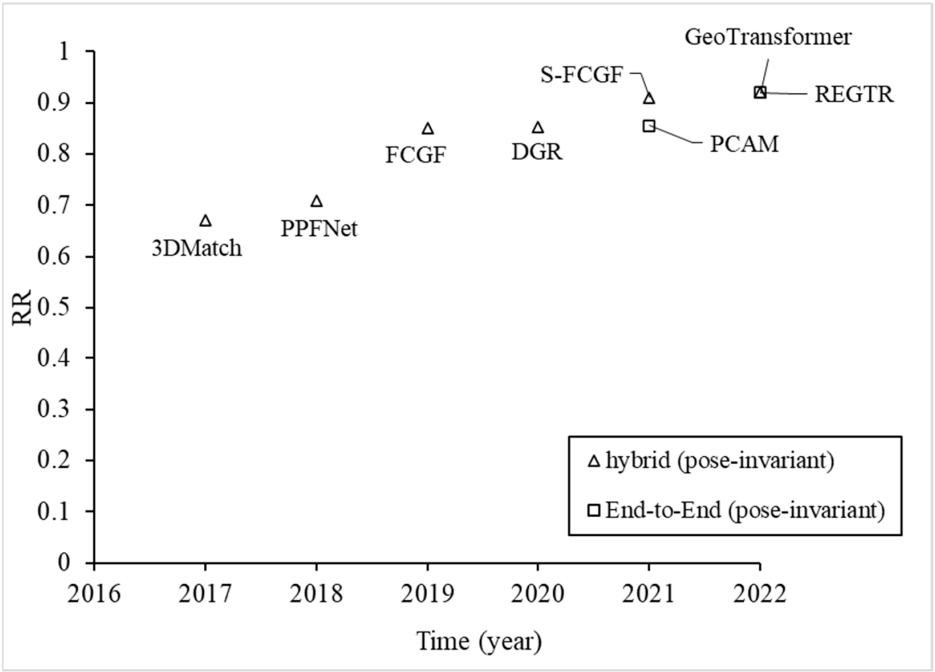

Figure 6. RR for 'hybrid' and 'end-to-end' registration methods of the best performance(s) in each of recent years, based on the 3DMatch dataset. The Presented methods are 3DMatch [32], PPFNet [33], FCGF [35], DGR [68], S-FCGF [69], PCAM [59], REGTR [60], and GeoTransformer [70].

By 2022, the state-of-the-art performances of 'end-to-end' methods and 'hybrid' methods (i.e., REGTR [60] and GeoTransformer [70], respectively.) were very similar. It was also observed that the test results of 'end-to-end' methods on 3DMatch have only become available in the last two years, likely because such methods are often very computationally demanding.

3. Key Steps of Deep-Learning-Based Point Cloud Registration Utilising Pose-Invariant Features

3.1. Feature Descriptors

The most commonly used backbones in the feature extraction networks for point cloud registration are the sparse 3D convolutional neural network (sparse 3D CNN) [26], the shared multi-layer perceptron (MLP) derived from PointNet [24] (referred to as PointNet-like MLP hereafter), the dynamic graph convolutional neural network (DGCNN) [27], and kernel points convolution (KPConv) [28]. These networks were originally proposed for other tasks, such as semantic segmentations and classifications. However, their utility in similar architectures was later extended to point cloud registration. The parameters of these networks were trained on the benchmark datasets for registration, using the loss functions relevant to the point cloud registration. A trained feature extraction network, combined with any necessary pre-processing or post-processing, is essentially a 'feature descriptor' (see Figure 4). A desirable feature descriptor should possess two properties, including description ability (represented by performances on seen datasets) and generalization ability (represented by performances on unseen datasets). These two properties are typically related to the pose-invariance, the size of the receptive field, and the 'context-awareness' of point features, which are elaborated in Sections 3.1.1–3.1.3, respectively.

3.1.1. Pose Invariance

The source and the template point clouds are of arbitrarily initial poses. If a point cloud registration is based on point feature matching, the matched points, regardless of their initial poses, should ideally have the same feature values quantified by the feature descriptors. A feature descriptor of such a capability is termed pose-invariant. As the effect of the initial poses is eliminated, the point features produced by the pose-invariant feature descriptor contain only structure information. This can be realized by canonical orientation or data augmentation, which are elaborated, respectively, below.

Before introducing different realizations of pose-invariance, it is necessary to distinguish 'patch-based' and 'frame-based' feature descriptors, following the definition in Ao et al. [71]. The former take point cloud patches as an input. A point cloud patch is formed by a centre point and its neighbourhood that can be defined using K nearest neighbour (KNN) search or radius neighbourhood search. The output of a 'patch-based' feature descriptor is the point feature of the centre point, embedding the information of its neighbourhood. A 'frame-based' feature descriptor takes an entire point cloud as an input and outputs features of each point, embedding the neighbourhood information via a fully convolutional network.

Canonical orientation is applied to patch-based feature descriptors [33,34,39,57,58,72–74], and it aligns point cloud patches of arbitrary initial poses into a canonical pose. Ideally, the canonical pose only depends on the distribution of points within the point cloud patches. Therefore, matched point cloud patches of different initial poses are expected to have similar poses after the canonical orientation, and, consequently, the effect of the initial poses is eliminated. The canonical orientation can be realized by either a handcrafted local reference frame (LRF) or a trained deep neural network [39,55].

Canonical orientations of corresponding point clouds or point cloud patches can be inconsistent due to variations in data densities, noises, and clutters. For 'patched-based' feature descriptors, the risk of such an inconsistency can be tolerated because individual patches of inconsistent canonical orientations can be rejected in subsequent feature-matching and/or outlier-rejection modules. However, in the case of 'frame-based' feature descriptors, if inconsistent canonical orientations are applied to the whole point

clouds, the pose invariance of all point features may not be ensured. Therefore, the canonical orientation is often not applicable to 'frame-based' feature descriptors [35,37,38,57,60,75,76]. Pose-invariance can be broken down into rotational invariance and translational invariance. Luckily, as the translational invariance is an inherent property of feature extraction networks (e.g., sparse 3D CNN, DGCNN, and KPConv) of 'frame-based' feature descriptors, only the rotational invariance is of concern. The rotational invariances of those feature descriptors are learned during training implicitly through random rotations applied to the input training data, which is termed 'data augmentation'.

3.1.2. Size of Receptive Field

Except for a few feature descriptors [33,34,72] that incorporate the parameterization of point clouds in pre-processing, many feature descriptor descriptions and generalization abilities are often solely offered by feature extraction networks. The receptive field of a feature extraction network should be large enough for capturing the objects of large sizes [77]. However, increasing the size of the receptive field is usually restricted by computational costs. As a mitigating measure, such an increase often requires more computationally efficient deep neural networks.

In early studies, standard 3D convolutional neural networks (CNNs) were adopted as feature extraction networks [32,74]. In these CNNs, their computational costs increased with the number of cells of the voxel-grid in the receptive field because the convolution was conducted on all cells, regardless of whether a cell was occupied or empty. In many point clouds, large portions of their 3D space are empty. Convolutions on empty cells contribute little to the description ability but consumes much computational resource. To counter this, 'point-based' deep neural networks were proposed for feature extraction, including PointNet-MLP, sparse 3D CNN, DGCNN and KPConv. In these networks, the computational cost increases with increase in the number of points within the receptive field, while empty spaces are disregarded. As such, 'point-based' networks are more computationally efficient [24,27] than standard 3D CNNs, and, hence, have been commonly adopted in more recent investigations.

Reducing the density of input point clouds is another means of facilitating the increase in the receptive field while the overall computational cost is under control. However, the side-effect is that it also reduces geometrical details. To keep the benefits of both detailed geometries and large receptive fields, 'multi-scale' feature extraction networks [38,40,70,78,79] were proposed. In these networks, an input point cloud is subsampled to multiple levels of data densities and fed into the feature extraction networks under different sizes of receptive fields. Low-density inputs correspond to large receptive fields, while high-density inputs correspond to small receptive fields. The outputs under the multiple sizes were fused into a final feature.

3.1.3. Context Awareness

Although the receptive field of point features can be increased, a point feature descriptor without 'context awareness' cannot incorporate information outside its receptive fields. As such, it is believed that adding 'context awareness' can enhance the description ability and the generalization ability of a feature descriptor [33,57,58]. To this end, 'global context awareness' and 'mutual context awareness' are often considered. 'Global context awareness' is the awareness of the global geometry of the current point cloud. 'Mutual context awareness' is the awareness of the geometry of the other point cloud. Usually, 'global context awareness' and 'mutual context awareness' are realized by applying self-attention and cross-attention layers of Transformer [80], respectively, on point features. DCP [57] and PRNet [58] incorporate mutual context awareness between source and template point clouds by applying cross-attention layers to the features initially extracted. Similarly, REGTR [60] and Lepard [81] use both self-attention layers and cross-attention layers to incorporate both 'global context awareness' and 'mutual context awareness'. Alternatively, in PPFNet [33], the global geometry is summarized in a global feature, which is obtained

by the channel-wise pooling of initial point features. Then, the initial point features and the global feature are fused into the final point features with 'global context awareness'.

As described in Section 3.1.1, feature descriptors are designed to embed only structure information but not to pose information. However, without pose information, it is difficult to distinguish repetitive objects (e.g., multiple streetlamps in a street scene) and to correctly match the points representing the repetitive objects based on point features. To this end, Lepard [81] also enhanced point features with 'positional context awareness' using rotary positional encoding [82], while the source template cloud is repositioned with rigid motion parameters from a soft Procrustes layer to mitigate the effect of initial pose.

3.2. Key Point Detection

Key point detection is a common module in the point cloud registration pipeline. Since a 3D rigid motion contains only six degrees of freedom, matching of a small number of key points would often be sufficient for registration. By filtering out a large number of 'unimportant' points, the key point detection module helps to reduce the computational cost and, at the same time, increases the accuracy of point feature matching. The key point detection module can be executed before, after, or in parallel to the local feature extraction module [55,83].

There are generally two steps in the key point detection module. The first is to estimate the scores of individual points, which represent the saliency of geometry and the likelihood of overlapping. Point scores can be handcrafted or learned. Handcrafted scores are usually derived from feature values, such as the norm [39,58] or channel-max [37] of point features. The estimation of these handcrafted point scores incurs little additional computational cost. Alternatively, deep-learning-based key point detectors [55,75,83–85] are incorporated to estimate point scores. Learned point scores are usually based on an attention module [80]. The second step is to detect the key points by filtering in the top K points or the points above a certain score threshold, which is straightforward to implement.

Most of the early key point detectors [37,39,55,83,84] estimate the point scores of sources and template point clouds independently, without referring to each other, and, hence, cannot specifically predict the likelihood of overlapping. When source and template point clouds have a low overlap ratio, the inclusion of non-overlapping points in feature matching would cause significant spurious matches. Therefore, the prediction of overlapping points is also considered. MaskNet [85] uses a PointNet-like network to predict whether points in template point clouds overlap with those in source point clouds. However, MaskNet detects the overlapping points only in the template point cloud. To counter this, PREDATOR [75] uses a deep-learning-based cross-attention module to predict the overlapping points from both the source and the template point clouds. Alternatively, NgeNet [79] uses a specially designed voting scheme to detect overlapping points. PR-Net [58] embeds mutual context awareness in the feature descriptor, and, hence, its key point detection (based on L_2 norm of the feature values) also implicitly contains the ability to predict the overlapping likelihood.

It was also observed that some methods [38–40] did not adopt a learned key point detector but still achieved the state-of-the-art performance on benchmark datasets. This suggests that a learned key point detector is not always essential in a point cloud registration pipeline. A common limitation of the current key point detectors is that the criterion of filtering is usually manually set, which may need to be altered if the key point detectors are applied to unseen datasets.

3.3. Feature Matching

Key points detected from source and template point clouds need to be matched into pairs to enable rigid motion estimation. In most 'hybrid' methods, matched point pairs are obtained using mutual nearest neighbour search (MNNS) of point features.

Due to its robustness and simple implementation, MNNS is still widely used and has been shown to cause no harm to the state-of-the-art performances of the methods adopting it (e.g., [38–40]).

However, MNNS is not desirable in two respects. Firstly, it produces a 'one-to-one' match of points, which may contain errors resulting from low data densities and data noises of point clouds. Secondly, it produces a 'hard' match, i.e., either 0 (not matched) or 1 (matched). The 'hard match' forbids the backpropagation of a deep neural network and, therefore, cannot be incorporated in an 'end-to-end' method.

'End-to-end' methods (e.g., [57–59,76]) adopt differentiable soft matching between the features of source and template point clouds. Soft matching enables one-to-multiple matching, in which the likelihood of matching is represented by a value within an interval, usually from 0 to 1 (increase with increasing likelihood). However, such a soft matching lacks the ability to reject the points outside the overlapping area(s). Therefore, soft matching is not ideal when source and template point clouds have a low overlap ratio. This limitation can be addressed in the key point detection module or the outlier rejection module.

Some recent methods [70,86,87] also implemented feature matching in a 'coarse-to-fine' manner. For the coarse match step, the features of point cloud patches are aggregated and matched. For the fine match step, within coarsely matched point cloud patches, individual points are matched based on their features.

3.4. Outlier Rejection

In most cases, many of the initial matched point pairs are inaccurate. If all matched point pairs are used to estimate the rigid motion, the rigid motion parameters estimated are most likely to be inaccurate. Therefore, inaccurate matchings of point pairs (referred to as 'outliers') need to be rejected, leading to a set where accurately matched point pairs (i.e., 'inliers') are kept. This process is termed 'outlier rejection'.

RANSAC is the most widely used outlier rejection algorithm in 'hybrid' methods due to its robustness and simple implementation. However, RANSAC is time-consuming, especially when the inlier ratio is low [68]. Therefore, it is worthwhile incorporating additional means of outlier rejection before applying RANSAC. For 'hybrid methods', the matched point pairs resulting from MNNS can be reused for outlier rejection. Both Unsupervied R&R [88], Lepard [81] and GeoTransfomer [70] assumed that matchings should be more accurate when the point pairs are closer in the feature space. These methods estimate the weights of matched point pairs based on their distance in feature space, with higher weights corresponding to smaller distances. The Unsupervised R&R and Lepard reject matched point pairs of low weights according to a certain threshold, while GeoTransfomer rejects matched point pairs outsides of the top k rankings by weights. Moreover, deep learning algorithms dedicated for outlier rejection [68,89,90] were also developed for 'hybrid' methods. These dedicated outlier rejection networks are trained separately from the feature extraction networks and classify matched point pairs into inliers and outliers.

However, all the aforementioned outlier rejection methods are based on 'hard match' of point pairs and, hence, are not applicable to 'end-to-end' methods. This is because the 'hard match' of point pairs cannot be differentiated, and, as such, do not support the backpropagation of deep neural networks, as mentioned in Section 3.3. For 'end-to-end' methods, differentiable outlier rejection methods based on 'soft match' of point pairs have been devised. For example, the soft feature matching in PRNet was refined by applying a gumbel-sampler [91] on the matching matrix.

3.5. Rigid Motion Estimation

It is common practice to apply SVD in deep-learning-based registration methods for estimating the rigid motion, and, as such, this step is not elaborated here.

4. Benchmark Datasets

Point cloud datasets are fundamental to deep-learning-based methods for the registration of point clouds, which are used to train and/or to test those methods. Some widely used benchmark datasets are elaborated in Section 4, and their key information is presented in Table 1. Overall, these benchmark datasets evolved in the following respects: from small to large amounts of data, from small to large spatial scales, and from indoor scenes to diverse scenes.

Table 1. Summary of some widely used benchmark datasets for point cloud registration.

Dataset	# Scenes	# Frames or Scans	Format	Scenario
Stanford [92]	9	-	TIN mesh	individual objects
ModelNet [31]	151,128	-	CAD	synthetic individual objects
ModelNet40 [31]	12,311	-	CAD	synthetic individual objects
3Dmatch [32]	62	200,000	RGB-D	indoor
ScanNet [93]	1513	2,500,000	RGB-D	indoor
TUM [94]	2	39	RGB-D	indoor
ETH [30]	8	276	point cloud	indoor and outdoor
KITTI [29]	39.2 km	-	point cloud	outdoor
RobotCar [95]	1000 km	-	point cloud	outdoor
WHU [47]	11	115	point cloud	indoor and outdoor
Composite [96]	14	400	RGB-D and point cloud	indoor and outdoor

means 'number of'.

A single point cloud is often not sufficient to train or to test deep-learning-based registration algorithms. Attempts were made to combine point clouds from multiple publications to establish more useful datasets (e.g., Stanford 3D Scanning Repository [92] and Bologna Retrieval (BR) [97]). However, these datasets are still comparatively small, and, thus, few researchers [72] have used them for the training of deep learning models. Moreover, these datasets usually contain data representing individual objects rather than complex scenes, which are incompetent for producing deep learning models for real-life application scenarios.

3DMatch [32] is one of the most popular real-life datasets used for developing deep-learning-based point cloud registration methods. It has widely been adopted in many previous studies [32–35,37–39,63,70,71,73,88,90,98,99]. It includes over 8 million correspondences out of 200,000 registered RGB-D images from 62 RGB-D real-life indoor scene reconstructions conducted in previous studies [100–104]. A total of 54 and 8 scenes can be used for training and testing, respectively. Such a partition was also conveniently followed by later researchers. To limit the difficulties of training and testing, 3DMatch utilized only the image pairs with over 30% overlap. The dataset is valuable because it includes a large number of correspondences and covers real-life scenes. Deng, Birdal and Ilic [34] found that the frames in 3DMatch were already oriented to the correct orientations to a certain extent, and, as such, it would not challenge the rotation invariance of point feature descriptors. Therefore, the Rotated 3DMatch [34] datasets were derived by applying random rotations to the original 3DMatch. Moreover, Huang, Gojcic, Usvyatsov, Wieser and Schindler [75] gathered the 3DLoMatch dataset by considering only the image pairs with overlaps of less than 30% in 3DMatch, to test the performance of their method under the low overlap condition. ScanNet [93] is also a large-size real-life indoor dataset comprising 2.5 million RGB-D images in 1513 scenes. It was used for both the training and testing of deep learning registration models [88]. TUM [94] is another real-life indoor dataset containing 39 RGB-D images in two scenes. Due to its limited number of images, it has usually been used as a test dataset [20] for verifying the generalization ability of a deep learning method. The 3Dmatch, ScanNet and TUM datasets consist of only indoor scenes.

ETH [30] contains both indoor and outdoor scenes, consisting of 276 LIDAR scans in eight scenes. The ETH dataset provides realistic outdoor scenes, which have often been missing in previous registration benchmark datasets. However, due to its limited number

of scans, it was often used as a test dataset [37–39,55,71]. For both training and testing of deep learning models aiming at outdoors scenes, the KITTI [29] dataset is the most widely used [35,37,55,70,71,78,90,99,105]. KITTI includes a range of point clouds, covering a total length of 39.2 km, which were acquired by a Velodyne laser scanner mounted on a car travelling in rural areas and along highways. The ground-truth registration was given by the GPS (global positioning system) and IMU (inertial measurement unit) measurements synchronized with the scanner. The point cloud sequence can be cropped to make overlapping pairs of point clouds [55]. Oxford RobotCar [95] is a multi-platform dataset of outdoor scenes with large spatial and temporal spans. It consists of 100 traversals of an outdoor route of over 10 km in Oxford, taken over a one-year period under various weather conditions. 2D scans were obtained by 2D LIDAR sensors mounted on a vehicle. Each 2D scan contained points with the coordinates in two axes, and the coordinates in the third axis were provided by the forward motion of the car in a 'push-broom' manner. 3D point cloud data were generated by registering numerous 2D scans using the 'ground-truth' measurements of the forward motion of the vehicle, which were provided by a combination of GPS and INS. However, the authors of Oxford RobotCar did not suggest use of the 'ground truth' for benchmarking localization and mapping algorithms because the qualities of the GPS reception and the fused GPS and INS solution varied significantly during data collection. Therefore, in 3DFeat-Net [55], the initial ground-truth registration of Oxford RobotCar was refined using the iterative closest point (ICP) [106] before it was used as the test benchmark. Recently, researchers [47,96] have proposed outdoor datasets of larger scales; however, tests (e.g., [107,108]) on these datasets are rare

Compared to the establishment of a benchmark dataset using a large amount of real-life point clouds, it is more convenient to use synthetically generated point clouds from 3D CAD models. ModelNet [31] is a collection of 3D CAD models, including 151,128 models from 660 categories. ModelNet40, a subset of ModelNet, was often used as a point cloud registration benchmark dataset. ModelNet40 consists of 12,311 3D CAD models from 40 categories. Points are sampled on exterior surfaces of the CAD models of ModelNet40 to generate point clouds. Random rotations and translations can be applied to the synthetic point clouds to produce template and source point clouds with the known registration ground-truth. ModelNet is popular in the training and testing of many deep-learning registration models [38,57,58,76]. However, synthetic point clouds often have less complexity compared to those acquired from real-life scans. As such, it is perhaps less convincing to use synthetic data for the evaluation of registration methods.

5. Evaluation Metrics

5.1. Definition

In many 'hybrid' registration methods, the processes after deep feature descriptors are similar. Therefore, the performance of deep feature descriptors in terms of feature matching can often represent the overall performances of 'hybrid' methods.

The basic index for assessing the performance of feature matching is the inlier ratio *In* (sometimes referred to as the hit ratio) [38]. Suppose that $p_{x'}$ and $q_{y'}$ are a matched point pair estimated by the MNNS of point feature values, and that C_{est} is a set containing all the matched pairs found between P and Q. *In* is defined as the ratio of the correctly matched pairs to all matched pairs, as in

$$In(C_{est}) = \frac{1}{|C_{est}|} \sum_{(p_{x'}, q_{y'}) \in C_{est}} 1(\| R_{gt} p_{x'} + t_{gt} - q_{y'} \|_2 \leq \tau_{dist}), \quad (2)$$

where R_{gt} and t_{gt} are the ground-truth rotation and translation, respectively, and $|C_{est}|$ is the total number of matched point pairs that are estimated. A matched pair is deemed correct if the distance between the paired points is under a threshold, τ_{dist}, when the ground-truth rigid motion is applied.

For the 'hybrid' methods, to evaluate the performance of feature matchings over a dataset consisting of n pairs of source and template point clouds, FMR is often used [33].

FMR is the ratio of the number of the pairs with an inlier ratio over τ_{In} to the total number (i.e., n) of pairs.

However, the FMR metric is not applicable to 'end-to-end' methods. As such, the evaluation metrics that are based on the estimated rotations and translations are also proposed [76]. The performance of a registration method on a benchmark dataset can also be represented by the mean values of v, as defined in

$$\mathrm{RE} = arccos\left(\frac{tr\left(R_{gt}^{-1}R_{est}\right) - 1}{2}\right) \tag{3}$$

$$\mathrm{TE} = \| t_{gt} - t_{est} \|_2, \tag{4}$$

where the R_{gt} and t_{gt} are the ground-truth rotation and translation, respectively; R_{est} and t_{est} are the estimated rotation and translation, respectively; RE and TE are the rotation and translation errors, respectively.

Instead of using RE and TE, it is more convenient to compare the performances of registration methods using a single evaluation metric, such as FMR for the performance of feature matching. To this end, either success rate (SR) or registration recall (RR) is used, depending on the dataset under consideration.

Both SR and RR represent the ratio of the number of accurately registered pairs of source and template point clouds to the total number of the pairs in a dataset. However, the definitions of 'accurately registered' are different between SR and RR. In the case of SR, a registration is deemed accurate if RE and TE are smaller than the thresholds τ_{RE} and τ_{TE}, respectively. For RR, a registration is deemed accurate when the root mean square error (RMSE) of the ground-truth matched point pairs underestimates the rotation R_{est} and the translation t_{est} (i.e., RMSE (R_{est}, t_{est}) is lower than a threshold τ_{RMSE}. The RMSE (R_{est}, t_{est}) is defined as follows:

$$\mathrm{RMSE}(R_{est}, t_{est}) = \sqrt{\frac{1}{|C_{gt}|}\sum_{(p_x,q_y)\in C_{gt}} \| R_{est}p_x + t_{est} - q_y \|^2}, \tag{5}$$

5.2. Summary of Evaluation Metric Results from the Literature

This section summarises the published results of the evaluation metrics defined in Section 5.1 on the benchmark datasets considered. Based on this summary, the performances of different deep-learning-based registration methods can be compared. It should be noted that some pioneering methods were not evaluated for some benchmark datasets in their original publications but were considered in later studies as baselines for comparison. In such cases, the source(s) where the results were published is cited in the same cell. Table 2 summarises FMR where τ_{dist} 0.1 m and $\tau_{In} = 0.05$ were set for its estimation. Table 3 summarises RR, SR, RE, and TE ($\tau_{RMSE} = 0.2$ m was used for the estimation of RR on 3DMatch, Rotated 3DMatch and 3DLoMatch; $\tau_{RE} = 5^\circ$ and $\tau_{TE} = 2$ m were used for the estimation of SR on KITTI). For evaluating the RR of hybrid methods, the number of sample points was 5000 and the maximum iterations of RANSAC were 55,000.

Overall, it was observed that the deep-learning-based feature descriptors and registration methods outperformed the traditional methods by notable margins in terms of FMR and RR. It is difficult to name an individual method as the top performer, in general, not only because none of the methods were exhaustively tested on every benchmark data configuration, but also because the differences in the performances of some leading methods were small. For example, the following methods GeDi, SpinNet, REGTR, GeoTransformer, Lepard, and NgeNet, showed top performances in the corresponding benchmark datasets where each method was tested. Specifically, GeDi and SpinNet showed good pose-invariance (see Rotated 3DMatch in Table 2) and a generalization ability on the unseen datasets (see ETH (3Dmatch) in Table 2 and KITTI (3DMatch in) Table 3), while Geo-

Transformer and NgeNet demonstrated strong performances on low-overlapping datasets (see 3DLoMatch in Table 3).

Table 2. FMR on the four benchmark datasets.

Method		3DMatch	Rotated 3DMatch	KITTI	ETH (3DMatch)
Title	C				
PPFNet [33]	D	0.623	0.003	-	-
PPF-FoldNet [34]	D	0.718	0.731	-	-
3DMatch [32]	D	0.573	0.011	-	0.169
FCGF [35]	D	0.952	0.953	0.966	0.161
Multi-view [36]	D	0.975	**0.969**	-	0.799
D3Feat [37]	D	0.958	0.955	**0.998**	0.616
MS-SVConv [38]	D	**0.984**	-	-	0.768
DIP [39]	D	0.948	0.946	0.973	**0.928**
GeDi [40]	D	0.979	**0.976**	**0.998**	**0.982**
PREDATOR [75]	D	0.967	-	-	-
3Dfeat-Net [55]	D	-	-	0.960	-
Equivariant3D [73]	D	0.942	0.939	-	-
RelativeNet [63]	D	0.760	-	-	-
CGF [72]	D	0.478	0.499	-	0.202
3DsmoothNet [74]	D	0.947	0.949	-	0.790
FoldingNet [66]	D	0.613	0.023	-	-
CapsuleNet [98]	D	0.807	0.807	-	-
SpinNet [71]	D	0.978	**0.978**	**0.991**	**0.928**
YOHO [107]	D	0.982	-	-	0.920
GeoTransformer [70]	D	0.979	-	-	-
Lepard [81]	D	**0.983**	-	-	-
Equivariant [109]	D	0.976	-	-	-
NgeNet [79]	D	0.982	-	-	-
WSDesc [110]	D	0.967	0.785	-	-
CoFiNet [86]	D	0.981	-	-	-
OCFNet [87]	D	**0.984**	-	-	-
SpinImage [19]	H	0.227, [33]	0.227, [34]	-	-
FPFH [20]	H	0.359, [33]	0.364, [34]	-	0.221, [74]
USC [21]	H	0.400, [33]	-	-	-
SHOT [22]	H	0.238, [33]	0.234, [34]	-	0.611, [74]

The methods of top three in performance are marked in bold. In the column 'C', which represents classification of the feature descriptors, D and H represent deep feature descriptors and handcrafted feature descriptors, respectively. In the column ETH (3DMatch), the methods were trained on 3DMatch but tested on ETH. Zero before the decimal point is left out to save space. The symbol '-' means that the results are unavailable.

Comparisons between the 'hybrid' and the 'end-to-end' methods can be made based on the relatively limited number of results in the literature. Most of the pioneering 'end-to-end' methods (e.g., [57,58,61,62]) are suitable only for simple objects [75], probably because they can only tolerate several thousands of points as the input data. Moreover, some studies of 'end-to-end' methods (e.g., [57,58,61,62,111–119]) did not report the evaluation results in terms of RE and TE metrics, which were adopted in tests of both the 'hybrid' and the 'end-to-end' methods. Therefore, comparisons between the test results of those 'end-to-end' methods were found to be rare in the literature. In terms of the 'hybrid' methods, they were often tested on more challenging datasets, such as 3DMatch, but rarely on synthetic datasets, such as ModelNet40.

Table 3. SR, RR, RE and TE on the benchmark datasets.

Method		3DMatch	3DLoMatch	KITTI	KITTI (3DMatch)	ModelNet40	
Title	C	RR	RR	SR	SR	RE (°)	TE(m)
DCP [57]	E	-	-	-	-	11.975 [75]	0.171 [75]
RelativeNet [63]	E	0.777	-	-	-	-	-
PCAM [59]	E	0.855 [60]	-	-	-	-	-
REGTR [60]	E	**0.920**	-	-	-	**1.473**	**0.014**
PointNetLK [61]	E	-	-	-	-	29.725 [60]	0.29, [60]
RPM-Net [76]	E	-	-	-	-	**1.712** [75]	**0.018** [75]
OMNet [67]	E	0.359 [60]	-	-	-	2.947 [75]	0.032 [75]
RCP [120]	E	-	-	-	-	**1.665**	**0.016**
HRegNet [78]	E	-	-	0.997	-	-	-
WSDesc [110]	E	0.814	-	-	-	-	-
PPFNet [33]	H	0.71	-	-	-	-	-
FCGF [35]	H	0.851 [75]	0.401 [75]	0.966	0.350 [40]	-	-
3DMatch [32]	H	0.670 [33]	-	-	-	-	-
D3Feat [37]	H	0.816 [75]	0.372 [75]	**0.998**	0.387 [40]	-	-
DIP [39]	H	0.890	-	0.973 [40]	**0.750** [40]	-	-
GeDi [40]	H	-	-	**0.998**	**0.831** [40]	-	-
PREDATOR [75]	H	0.890	0.598 [75]	**0.998**	-	1.739	0.019
DGR [68]	H	0.853 [60]	-	-	-	-	-
CGF [72]	H	0.56 [33]	-	-	-	-	-
3DSmoothNet [74]	H	0.784 [75]	0.330 [75]	0.960 [75]	-	-	-
SpinNet [71]	H	-	-	0.991	**0.654** [40]	-	-
GeoTransformer [70]	H	**0.920**	**0.750**	**0.998**	-	-	-
S-FCGF [69]	H	0.914	-	-	-	-	-
NgeNet [79]	H	**0.929**	**0.845**	**0.998**	-	-	-
Lepard [81]	H	**0.935**	**0.690**	-	-	-	-
P2-Net [121]	H	0.880	-	-	-	-	-
CoFiNet [86]	H	0.893	0.675	**0.998**	-	-	-
OCFNet [87]	H	0.897	0.681	**0.998**	-	-	-
SpinImage [19]	T	0.34 [33]	-	-	-	-	-
FPFH [20]	T	0.40 [33]	-	-	-	-	-
USC [21]	T	0.43 [33]	-	-	-	-	-
SHOT [22]	T	0.27 [33]	-	-	-	-	-

The methods of top three in performance are marked in bold. In the column 'C', which represents classification of the methods, T, H and E represent traditional, hybrid and 'end-to-end' methods, respectively. In the column KITTI (3DMatch), the methods were trained on 3DMatch but tested on KITTI. Zero before the decimal point is left out to save space. The symbol '-' means that the results are unavailable.

Based on the reported results of RE and TE on ModelNet40 in Table 3, the best 'end-to-end' method is REGTR. The decent performance of REGTR may be attributed to the 'global context awareness' and 'mutual context awareness' components incorporated in its point features. Based on RR on 3DMatch in Table 3, except for REGTR, the other 'end-to-end' methods are inferior to at least five 'hybrid' methods. As shown by the test results on the seen datasets in Tables 2 and 3, strong description abilities were achieved by the PointNet-like MLP feature descriptors (e.g., DIP, GeDi, SpinNet), the sparse CNN feature descriptors (e.g., MS-SVConv, FCGF), and the KPConv feature descriptors (e.g., D3Feat, CoFiNet, Lepard, and NgeNet). However, from the tests results on the unseen datasets, i.e., FMR on ETH (3DMatch) in Table 2 and SR on KITTI (3Dmatch) in Table 3, the PointNet-like MLP feature descriptors showed better generalization abilities, compared to the sparse CNN and the KPConv feature descriptors. The likely reason is that the latter two feature descriptors are sensitive to scale and density variations.

In Table 3, the RE and TE results of PointNetLK are shown to be inferior to those of the other methods. This shows that the methods that are based on pose-invariant features generally perform better than PointNetLK, which is based on pose-variant features. The pose-invariance of the feature descriptors is inferred by comparing the FMR metric on 3DMatch and Rotated 3DMatch in Table 2. Some of the early-year deep feature descriptors

(e.g., 3DMatch and PPF-FoldNet) show significant drops in performance from 3DMatch to Rotated 3DMatch, suggesting poor rotation invariances. The RR on 3DLoMatch in Table 3 demonstrate that better performance on low-overlapping data can be achieved by specialized efforts in either the overlap detection module (e.g., PREDATOR and NgeNet), or the outlier rejection module (e.g., GeoTransformer). For the size of the receptive field, the advantage of multi-scale feature descriptors can be highlighted by the superior performances exhibited by GeDi, MS-SVConv, GeoTransformer, and NgeNet.

6. Challenges and Future Research

The deep-learning-based methods were tested on benchmark datasets, typically in the studies dedicated to point cloud registration. Although in such studies some methods exhibited excellent performances, reports of their direct application to real-life situations are still rare in the literature [122], which also highlights some challenges and directions for future research.

6.1. Computational Feasibility

The challenge in computational feasibility often results from the fact that real-life point cloud data may contain a large number of points. To this end, patch-based methods may be considered as they can process a point cloud patch-by-patch with each patch containing much less points. However, the feature descriptors of patch-based methods generally lack context awareness, which hinders registration performance in complex scenes. Therefore, how to better balance the trade-off between computational feasibility and registration performance in complex scenes should be further explored.

6.2. Generalization Ability

Although the feature descriptors of KPConv, sparse CNN and PointNet-like MLP achieved decent performances on the seen datasets, only PointNet-like MLP feature descriptors demonstrated strong generalization ability on the unseen datasets. This suggests a challenge in their application, i.e., the generalization of the established methods to other datasets. The challenge in generalization ability is typically caused by the differences in characteristics between training benchmark datasets and data encountered in real-life applications. In real-life applications, as the ground-truth registration is unavailable for training deep learning models, the model parameters of a deep-learning-based method must be pre-trained on a benchmark dataset(s) (ideally similar to the real-life point cloud data to be registered).

To facilitate the applications of pre-trained models to real-life applications, more dedicated benchmark datasets of diverse application scenes should be established. An interesting question to be considered is how to use diverse datasets to pre-train a deep learning model. On one side, a large number of datasets in various scenes may be used to train a 'large, generic' model, aiming for greater generalization ability of a single model for applications in diverse scenes. On the other side, datasets in specific scenes are used to pre-train a model for multiple sets of model parameters that are more suitable for a specific scene. In the second case, the pre-trained model using a benchmark dataset of a more similar scene to a real-life application can be found. If real-life data acquisition is difficult in certain scenarios, point cloud data generation using simulators (e.g., [123–126]) may be considered. Further development of such tools to facilitate the generation of simulated sensor data based on realistic environments is a promising way forward.

6.3. Utilization of Texture Information

Apart from structure information, real-life scenes contain rich texture information, such as colours, reflectance, and spectra, which can be captured by additional sensors. The fused data may be used to aid in feature descriptions (e.g., [127,128]) and improve the overall registration performances. However, the utilization of the texture information for point cloud registration, which is still rare, as such information is often missing in

most existing benchmarks, may be considered in further studies. In addition, data noises, clutters, and density variations are often not considered in the current registration methods and the associated datasets, which should be considered in further developments.

6.4. Evaluation Metrics

In the existing studies, it is a common practice to use a lump-sum single metric(s) to represent the overall registration quality of all experiments on the benchmark dataset(s), but the registration quality of individual experiments is not reported. As such, the spatial variation of the registration errors is unclear, especially in the local areas of relatively large errors. This information is beneficial for investigating the causes of these errors, which can be used to improve the registration methods.

Moreover, various evaluation metrics were used in previous research, including FMR, RR, SR, RE and TE. However, it is rare in the existing studies to consider all these metrics. In most cases, researchers often choose one or some of these evaluation metric(s). The choices of performance metrics and benchmark datasets are probably favourable to their methods. As such, the comparisons between different methods may not be consistent or fair. As such, it would be useful to use consistent evaluation metrics in future studies.

6.5. Multi-Temporal Data Registration

Registration of multi-temporal point clouds is essential for applications where changes in a scene over time need to be monitored. Based on the existing studies, the registration methods were typically tested on the point cloud data obtained in a survey campaign where the variations in objects in the scenes scanned were very small. However, they were rarely tested on multi-temporal point clouds that represent dynamic and changing scenes (e.g., a construction site) over time. Although, feature matching and outlier rejection algorithms can implicitly mitigate the effects of dynamic scenes on point cloud registration, it is suggested to explicitly consider this in future studies for developing new benchmark datasets and registration methods.

In the literature, there are mainly two ways to cope specifically with multi-temporal registration. One is to establish control point networks external to the point cloud data, and these control points are ensured to be stable (e.g., [129]). However, this approach generates additional fieldwork in the data acquisition phase. The other approach (e.g., [130]) is to first roughly register the multi-temporal point clouds and then to filter out parts of the point clouds based on the spatial distribution of the registration error. Then the registration is refined using the filtered point clouds, which can be performed iteratively. However, the filtering involves manual judgements. It is suggested that future research should address the problem of multi-temporal registration with data-driven approaches, which would mitigate the requirement for extensive fieldwork and manual judgements.

6.6. Multi-Modal Data Registration

Multi-modal registration is becoming a new trend in point cloud registration because fused point clouds from different sensors can be more informative for some tasks, such as scene reconstruction [131], environment perception [132], and change detection [133]. However, compared to single-source point cloud registration, multi-modal point cloud registration is more challenging in terms of noise, outliers, partial overlap, density difference and scale variations since different types of sensors have different imaging mechanisms. Without considering these challenges, registration algorithms originally designed for single-source point cloud registration are likely to perform poorly in multi-modal point cloud registration [52]. Although some researchers [133,134] have recently proposed algorithms and benchmark datasets specifically for multi-modal point cloud registration, much more can be done to improve the performance of algorithms and the diversity of benchmark multi-modal datasets in the future.

7. Conclusions

This article reviews deep-learning-based methods and the relevant benchmark datasets (by the end of 2022) for pairwise global registration of same-source point clouds. The performances of the methods are discussed, in general, from their design perspective, and, specifically, based on a set of performance metrics on the benchmark datasets considered.

Based on our statistics, the number of publications in deep-learning-based point cloud registration increased from 2017 to 2021, during which a sharp increase was witnessed in 2021. However, the relevant publications in 2022 seemed to slow down, likely due to the extensive work undertaken in previous years (especially 2021) and limited progress in registration performances since then.

Two types of deep-learning-based methods (i.e., 'hybrid' and 'end-to-end') were considered, which share many key steps in the registration pipeline. It was observed that the deep-learning-based registration methods outperformed the traditional methods by notable margins in terms of FMR and RR. Comparisons between 'hybrid' and 'end-to-end' methods are more challenging because the performances of many of these methods on the considered benchmark datasets were not reported in the literature. For the dataset 3DMatch, on which the performance metric RR of many of the considered methods was tested, the state-of-the-art performance of the 'end-to-end' methods was slightly lower than that of the 'hybrid' methods, as shown in Table 3.

In the 'hybrid' methods, deep learning implementations are often considered for feature extraction, key point detection and outlier rejection. The 'hybrid' methods that are based on pose-invariant features (i.e., GeDi [40], GeoTransformer [70], NgeNet [79], Lepard [81]) show leading performances on the benchmark datasets. All these methods adopted deep learning in the feature extraction step, while GeDi adopted deep learning in only the feature extraction step, which suggests that deep feature descriptors are a critical factor for determining the overall performance of point cloud registration. There are also methods that did not use learned key point detections, but achieved strong performance on the benchmark datasets, which suggests that a learned key point detector is not always essential. However, the criteria of filtering key points are usually set manually and may vary when the key point detectors are applied to unseen datasets.

Author Contributions: Y.Z. drafted and revised the article; L.F. conceptualized and revised the article. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Xi'an Jiaotong-Liverpool University Research Enhancement Fund, grant number REF-21-01-003.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gümüş, K.; Gümüş, M.G.; Erkaya, H. A statistical evaluation of registration methods used in terrestrial laser scanning in cultural heritage applications. *Mediterr. Archaeol. Archaeom.* **2017**, *17*, 53–64.
2. Xiong, X.; Adan, A.; Akinci, B.; Huber, D. Automatic creation of semantically rich 3D building models from laser scanner data. *Autom. Constr.* **2013**, *31*, 325–337. [CrossRef]
3. McGuire, M.P.; Yust, M.B.S.; Shippee, B.J. Application of Terrestrial Lidar and Photogrammetry to the As-Built Verification and Displacement Monitoring of a Segmental Retaining Wall. In Proceedings of the Geotechnical Frontiers 2017, Orlando, FL, USA, 12–15 March 2017; pp. 461–471.
4. Cai, Y.; Fan, L. An Efficient Approach to Automatic Construction of 3D Watertight Geometry of Buildings Using Point Clouds. *Remote Sens.* **2021**, *13*, 1947. [CrossRef]
5. Crespo-Peremarch, P.; Tompalski, P.; Coops, N.C.; Ruiz, L.Á. Characterizing understory vegetation in Mediterranean forests using full-waveform airborne laser scanning data. *Remote Sens. Environ.* **2018**, *217*, 400–413. [CrossRef]
6. Hashash, Y.M.A.; Filho, J.N.O.; Su, Y.Y.; Liu, L.Y. 3D Laser Scanning for Tracking Supported Excavation Construction. In Proceedings of the Geo-Frontiers, Austin, TX, USA, 24–26 January 2005; pp. 1–10.

7. Su, Y.Y.; Hashash, Y.M.A.; Liu, L.Y. Integration of Construction As-Built Data Via Laser Scanning with Geotechnical Monitoring of Urban Excavation. *J. Constr. Eng. Manag.* **2006**, *132*, 1234–1241. [CrossRef]
8. Yakar, M.; Yılmaz, H.M.; Mutluoğlu, Ö. Comparative evaluation of excavation volume by TLS and total topographic station based methods. *Lasers Eng.* **2010**, *19*, 331–345.
9. Pesci, A.; Casula, G.; Boschi, E. Laser scanning the Garisenda and Asinelli towers in Bologna (Italy): Detailed deformation patterns of two ancient leaning buildings. *J. Cult. Herit.* **2011**, *12*, 117–127. [CrossRef]
10. Lee, M.; Tsai, Y.; Wang, R.; Lin, M. Finding the displacement of wood structure in heritage building by 3D laser scanner. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *2*, 165–169. [CrossRef]
11. Chen, D.L.; Lu, Y.Y.; Chen, Y.M.; Ma, L.; Jia, D.Z.; Cheng, L.; Li, M.C.; Hu, D.; He, X.F. Automated and Efficient Extraction of Highway Tunnel Lining Cross-sections Using Terrestrial Laser Scanning (TLS). *Lasers Eng.* **2018**, *39*, 341–353.
12. Batur, M.; Yilmaz, O.; Ozener, H. A Case Study of Deformation Measurements of Istanbul Land Walls via Terrestrial Laser Scanning. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 6362–6371. [CrossRef]
13. Zhao, Y.; Seo, H.; Chen, C. Displacement analysis of point cloud removed ground collapse effect in SMW by CANUPO machine learning algorithm. *J. Civ. Struct. Health Monit.* **2022**, *12*, 447–463. [CrossRef]
14. O'Neal, M.A.; Pizzuto, J.E. The rates and spatial patterns of annual riverbank erosion revealed through terrestrial laser-scanner surveys of the South River, Virginia. *Earth Surf. Process. Landf.* **2011**, *36*, 695–701. [CrossRef]
15. Bremer, M.; Sass, O. Combining airborne and terrestrial laser scanning for quantifying erosion and deposition by a debris flow event. *Geomorphology* **2012**, *138*, 49–60. [CrossRef]
16. Lague, D.; Brodu, N.; Leroux, J. Accurate 3D comparison of complex topography with terrestrial laser scanner: Application to the Rangitikei canyon (N-Z). *ISPRS J. Photogramm. Remote Sens.* **2013**, *82*, 10–26. [CrossRef]
17. Pomerleau, F.; Colas, F.; Siegwart, R. A Review of Point Cloud Registration Algorithms for Mobile Robotics. *Found. Trends Robot.* **2015**, *4*, 1–104. [CrossRef]
18. Zheng, Y.; Li, Y.; Yang, S.; Lu, H. Global-PBNet: A Novel Point Cloud Registration for Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22312–22319. [CrossRef]
19. Johnson, A.E.; Hebert, M. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 433–449. [CrossRef]
20. Rusu, R.B.; Blodow, N.; Beetz, M. Fast Point Feature Histograms (FPFH) for 3D registration. In Proceedings of the ICRA, Kobe, Japan, 12–17 May 2009; pp. 3212–3217.
21. Tombari, F.; Salti, S.; Stefano, L.D. Unique shape context for 3d data description. In Proceedings of the ACM Workshop on 3D Object Retrieval, Firenze, Italy, 25 October 2010; pp. 57–62.
22. Salti, S.; Tombari, F.; Di Stefano, L. SHOT: Unique signatures of histograms for surface and texture description. *Comput. Vis. Imag. Underst.* **2014**, *125*, 251–264. [CrossRef]
23. Zhou, Q.-Y.; Park, J.; Koltun, V. Fast Global Registration. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 766–782.
24. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the CVPR, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
25. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
26. Choy, C.; Gwak, J.; Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In Proceedings of the CVPR, Long Beach, CA, USA, 15–20 June 2019; pp. 3075–3084.
27. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.* **2019**, *38*, 1–12. [CrossRef]
28. Thomas, H.; Qi, C.R.; Deschaud, J.; Marcotegui, B.; Goulette, F.; Guibas, L. KPConv: Flexible and Deformable Convolution for Point Clouds. In Proceedings of the ICCV, Seoul, Korea, 27 October–2 November 2019; pp. 6410–6419.
29. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In Proceedings of the CVPR, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
30. Pomerleau, F.; Liu, M.; Colas, F.; Siegwart, R. Challenging data sets for point cloud registration algorithms. *Int. J. Robot. Res.* **2012**, *31*, 1705–1711. [CrossRef]
31. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the CVPR, Boston, MA, USA, 7–12 June 2015; pp. 1912–1920.
32. Zeng, A.; Song, S.; Nießner, M.; Fisher, M.; Xiao, J.; Funkhouser, T. 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In Proceedings of the CVPR, Honolulu, HI, USA, 21–26 July 2017; pp. 199–208.
33. Deng, H.; Birdal, T.; Ilic, S. Ppfnet: Global context aware local features for robust 3d point matching. In Proceedings of the CVPR, Salt Lake City, UT, USA, 18–23 June 2018; pp. 195–205.
34. Deng, H.; Birdal, T.; Ilic, S. PPF-FoldNet: Unsupervised Learning of Rotation Invariant 3D Local Descriptors. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 620–638.
35. Choy, C.; Park, J.; Koltun, V. Fully convolutional geometric features. In Proceedings of the ICCV, Seoul, Korea, 27 October–2 November 2019; pp. 8958–8966.

36. Li, L.; Zhu, S.; Fu, H.; Tan, P.; Tai, C.L. End-to-End Learning Local Multi-View Descriptors for 3D Point Clouds. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 1916–1925.
37. Bai, X.; Luo, Z.; Zhou, L.; Fu, H.; Quan, L.; Tai, C.-L. D3feat: Joint learning of dense detection and description of 3d local features. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 6359–6367.
38. Horache, S.; Deschaud, J.-E.; Goulette, F. 3D Point Cloud Registration with Multi-Scale Architecture and Self-supervised Fine-tuning. *arXiv* **2021**. *preprint*. [CrossRef]
39. Poiesi, F.; Boscaini, D. Distinctive 3D local deep descriptors. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 5720–5727.
40. Poiesi, F.; Boscaini, D. Learning general and distinctive 3D local deep descriptors for point cloud registration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 3979–3985. [CrossRef] [PubMed]
41. Yang, J.; Quan, S.; Wang, P.; Zhang, Y. Evaluating Local Geometric Feature Representations for 3D Rigid Data Matching. *IEEE Trans. Image Process.* **2020**, *29*, 2522–2535. [CrossRef] [PubMed]
42. Tang, K.; Song, P.; Chen, X. Signature of Geometric Centroids for 3D Local Shape Description and Partial Shape Matching. In Proceedings of the Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2017; pp. 311–326.
43. Quan, S.; Ma, J.; Hu, F.; Fang, B.; Ma, T. Local voxelized structure for 3D binary feature representation and robust registration of point clouds from low-cost sensors. *Inf. Sci.* **2018**, *444*, 153–171. [CrossRef]
44. Van Eck, N.; Waltman, L. Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics* **2010**, *84*, 523–538. [CrossRef]
45. Cheng, L.; Chen, S.; Liu, X.; Xu, H.; Wu, Y.; Li, M.; Chen, Y. Registration of laser scanning point clouds: A review. *Sensors* **2018**, *18*, 1641. [CrossRef]
46. Pan, Y. Target-less registration of point clouds: A review. *arXiv* **2019**. *preprint*. [CrossRef]
47. Dong, Z.; Liang, F.; Yang, B.; Xu, Y.; Zang, Y.; Li, J.; Wang, Y.; Dai, W.; Fan, H.; Hyyppä, J.; et al. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 327–342. [CrossRef]
48. Gu, X.; Wang, X.; Guo, Y. A review of research on point cloud registration methods. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Ho Chi Minh City, Vietnam, 6–8 January 2020; p. 022070.
49. Villena-Martinez, V.; Oprea, S.; Saval-Calvo, M.; Azorin-Lopez, J.; Fuster-Guillo, A.; Fisher, R.B. When Deep Learning Meets Data Alignment: A Review on Deep Registration Networks (DRNs). *Appl. Sci.* **2020**, *10*, 7524. [CrossRef]
50. Zhang, Z.; Dai, Y.; Sun, J. Deep learning based point cloud registration: An overview. *Virtual Real. Intell. Hardw.* **2020**, *2*, 222–246. [CrossRef]
51. Tang, W.; Zou, D.; Li, P. Learning-based Point Cloud Registration: A Short Review and Evaluation. In Proceedings of the International Conference on Artificial Intelligence in Electronics Engineering, Phuket, Thailand, 7–15 January 2021; pp. 27–34.
52. Huang, X.; Mei, G.; Zhang, J.; Abbas, R. A comprehensive survey on point cloud registration. *arXiv* **2021**. *preprint*. [CrossRef]
53. Brightman, N.; Fan, L.; Zhao, Y. Point cloud registration: A mini-review of current state, challenging issues and future directions. *AIMS Geosci.* **2023**, *9*, 68–85. [CrossRef]
54. Li, X.; Pontes, J.K.; Lucey, S. Pointnetlk revisited. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 12763–12772.
55. Yew, Z.J.; Lee, G.H. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 607–623.
56. Fischler, M.A.; Bolles, R.C. Random sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Readings in Computer Vision*; Fischler, M.A., Firschein, O., Eds.; Morgan Kaufmann: San Francisco, CA, USA, 1987; pp. 726–740.
57. Wang, Y.; Solomon, J.M. Deep closest point: Learning representations for point cloud registration. In Proceedings of the ICCV, Seoul, Korea, 27 October–2 November 2019; pp. 3523–3532.
58. Wang, Y.; Solomon, J.M. Prnet: Self-supervised learning for partial-to-partial registration. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 10–12 December 2019.
59. Cao, A.Q.; Puy, G.; Boulch, A.; Marlet, R. PCAM: Product of Cross-Attention Matrices for Rigid Registration of Point Clouds. In Proceedings of the ICCV, Montreal, QC, Canada, 10–17 October 2021; pp. 13209–13218.
60. Yew, Z.J.; Lee, G.H. REGTR: End-to-end Point Cloud Correspondences with Transformers. In Proceedings of the CVPR, New Orleans, LA, USA, 18–24 June 2022; pp. 6677–6686.
61. Aoki, Y.; Goforth, H.; Srivatsan, R.A.; Lucey, S. Pointnetlk: Robust & efficient point cloud registration using pointnet. In Proceedings of the CVPR, Long Beach, CA, USA, 15–20 June 2019; pp. 7163–7172.
62. Sarode, V.; Li, X.; Goforth, H.; Aoki, Y.; Srivatsan, R.A.; Lucey, S.; Choset, H. Pcrnet: Point cloud registration network using pointnet encoding. *arXiv* **2019**. *preprint*. [CrossRef]
63. Deng, H.; Birdal, T.; Ilic, S. 3d local features for direct pairwise registration. In Proceedings of the CVPR, Kalifornija, CA, USA, 1 April–16 May 2019; pp. 3244–3253.
64. Horn, B.K. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A* **1987**, *4*, 629–642. [CrossRef]
65. Baker, S.; Matthews, I. Lucas-kanade 20 years on: A unifying framework. *Int. J. Comput. Vis.* **2004**, *56*, 221–255. [CrossRef]
66. Yang, Y.; Feng, C.; Shen, Y.; Tian, D. Foldingnet: Point cloud auto-encoder via deep grid deformation. In Proceedings of the CVPR, Salt Lake City, UT, USA, 18–23 June 2018; pp. 206–215.

67. Xu, H.; Liu, S.; Wang, G.; Liu, G.; Zeng, B. OMNet: Learning Overlapping Mask for Partial-to-Partial Point Cloud Registration. In Proceedings of the ICCV, Montreal, QC, Canada, 10–17 October 2021; pp. 3112–3121.
68. Choy, C.; Dong, W.; Koltun, V. Deep Global Registration. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 2511–2520.
69. Yang, H.; Dong, W.; Carlone, L.; Koltun, V. Self-supervised geometric perception. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 14350–14361.
70. Qin, Z.; Yu, H.; Wang, C.; Guo, Y.; Peng, Y.; Xu, K. Geometric transformer for fast and robust point cloud registration. In Proceedings of the CVPR, New Orleans, LA, USA, 18–24 June 2022; pp. 11143–11152.
71. Ao, S.; Hu, Q.; Yang, B.; Markham, A.; Guo, Y. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 11753–11762.
72. Khoury, M.; Zhou, Q.-Y.; Koltun, V. Learning compact geometric features. In Proceedings of the ICCV, Venice, Italy, 22–29 October 2017; pp. 153–161.
73. Spezialetti, R.; Salti, S.; Stefano, L.D. Learning an effective equivariant 3d descriptor without supervision. In Proceedings of the ICCV, Seoul, Korea, 27 October–2 November 2019; pp. 6401–6410.
74. Gojcic, Z.; Zhou, C.; Wegner, J.D.; Wieser, A. The perfect match: 3d point cloud matching with smoothed densities. In Proceedings of the CVPR, Long Beach, CA, USA, 15–20 June 2019; pp. 5545–5554.
75. Huang, S.; Gojcic, Z.; Usvyatsov, M.; Wieser, A.; Schindler, K. PREDATOR: Registration of 3D Point Clouds with Low Overlap. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 4265–4274.
76. Yew, Z.J.; Lee, G.H. Rpm-net: Robust point matching using learned features. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 11824–11833.
77. Luo, W.; Li, Y.; Urtasun, R.; Zemel, R. Understanding the effective receptive field in deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016.
78. Lu, F.; Chen, G.; Liu, Y.; Zhang, L.; Qu, S.; Liu, S.; Gu, R. HRegNet: A Hierarchical Network for Large-scale Outdoor LiDAR Point Cloud Registration. In Proceedings of the ICCV, Montreal, QC, Canada, 10–17 October 2021; pp. 16014–16023.
79. Zhu, L.; Guan, H.; Lin, C.; Han, R. Neighborhood-aware Geometric Encoding Network for Point Cloud Registration. *arXiv* **2022**. *preprint*. [CrossRef]
80. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
81. Li, Y.; Harada, T. Lepard: Learning partial point cloud matching in rigid and deformable scenes. In Proceedings of the CVPR, Orleans, LA, USA, 18–24 June 2022; pp. 5554–5564.
82. Su, J.; Lu, Y.; Pan, S.; Wen, B.; Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *arXiv* **2021**. *preprint*. [CrossRef]
83. Georgakis, G.; Karanam, S.; Wu, Z.; Ernst, J.; Košecká, J. End-to-End Learning of Keypoint Detector and Descriptor for Pose Invariant 3D Matching. In Proceedings of the CVPR, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1965–1973.
84. Tinchev, G.; Penate-Sanchez, A.; Fallon, M. SKD: Keypoint Detection for Point Clouds Using Saliency Estimation. *IEEE Robot Autom. Let.* **2021**, *6*, 3785–3792. [CrossRef]
85. Sarode, V.; Dhagat, A.; Srivatsan, R.A.; Zevallos, N.; Lucey, S.; Choset, H. Masknet: A fully-convolutional network to estimate inlier points. In Proceedings of the International Conference on 3D Vision, Fukuoka, Japan, 25–28 November 2020; pp. 1029–1038.
86. Yu, H.; Li, F.; Saleh, M.; Busam, B.; Ilic, S. CoFiNet: Reliable coarse-to-fine correspondences for robust pointcloud registration. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 7–10 December 2021; pp. 23872–23884.
87. Mei, G.; Huang, X.; Zhang, J.; Wu, Q. Overlap-Guided Coarse-to-Fine Correspondence Prediction for Point Cloud Registration. In Proceedings of the 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 18–22 July 2022; pp. 1–6.
88. El Banani, M.; Gao, L.; Johnson, J. Unsupervisedr&r: Unsupervised point cloud registration via differentiable rendering. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 7129–7139.
89. Pais, G.D.; Ramalingam, S.; Govindu, V.M.; Nascimento, J.C.; Chellappa, R.; Miraldo, P. 3dregnet: A deep neural network for 3d point registration. In Proceedings of the CVPR, Seattle, WA, USA, 13–19 June 2020; pp. 7193–7203.
90. Bai, X.; Luo, Z.; Zhou, L.; Chen, H.; Li, L.; Hu, Z.; Fu, H.; Tai, C.-L. Pointdsc: Robust point cloud registration using deep spatial consistency. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 15859–15869.
91. Jang, E.; Gu, S.; Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv* **2016**. *preprint*. [CrossRef]
92. Turk, G.; Levoy, M. The Stanford 3d Scanning Repository. Available online: http://graphics.stanford.edu/data/3Dscanrep (accessed on 13 January 2023).
93. Dai, A.; Chang, A.X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In Proceedings of the CVPR, Honolulu, HI, USA, 21–26 July 2017; pp. 5828–5839.
94. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 573–580.
95. Maddern, W.; Pascoe, G.; Linegar, C.; Newman, P. 1 year, 1000 km: The Oxford RobotCar dataset. *Int. J. Robot. Res.* **2017**, *36*, 3–15. [CrossRef]

96. Fontana, S.; Cattaneo, D.; Ballardini, A.L.; Vaghi, M.; Sorrenti, D.G. A benchmark for point clouds registration algorithms. *Robot. Auton. Syst.* **2021**, *140*, 103734. [CrossRef]
97. Tombari, F.; Salti, S.; Di Stefano, L. Performance evaluation of 3D keypoint detectors. *Int. J. Comput. Vis.* **2013**, *102*, 198–220. [CrossRef]
98. Zhao, Y.; Birdal, T.; Deng, H.; Tombari, F. 3D point capsule networks. In Proceedings of the CVPR, Long Beach, CA, USA, 15–20 June 2019; pp. 1009–1018.
99. Liu, X.; Killeen, B.D.; Sinha, A.; Ishii, M.; Hager, G.D.; Taylor, R.H.; Unberath, M. Neighborhood normalization for robust geometric feature learning. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 13044–13053.
100. Shotton, J.; Glocker, B.; Zach, C.; Izadi, S.; Criminisi, A.; Fitzgibbon, A. Scene coordinate regression forests for camera relocalization in RGB-D images. In Proceedings of the CVPR, Portland, OR, USA, 23–28 June 2013; pp. 2930–2937.
101. Xiao, J.; Owens, A.; Torralba, A. Sun3d: A database of big spaces reconstructed using sfm and object labels. In Proceedings of the ICCV, Sydney, NSW, Australia, 1–8 December 2013; pp. 1625–1632.
102. Lai, K.; Bo, L.; Fox, D. Unsupervised feature learning for 3d scene labeling. In Proceedings of the ICRA, Hong Kong, China, 31 May–7 June 2014; pp. 3050–3057.
103. Valentin, J.; Dai, A.; Nießner, M.; Kohli, P.; Torr, P.; Izadi, S.; Keskin, C. Learning to navigate the energy landscape. In Proceedings of the Fourth International Conference on 3D Vision, Stanford, CA, USA, 25–28 October 2016; pp. 323–332.
104. Halber, M.; Funkhouser, T.A. Structured Global Registration of RGB-D Scans in Indoor Environments. *arXiv* **2016**. *preprint*. [CrossRef]
105. Arnold, E.; Mozaffari, S.; Dianati, M. Fast and Robust Registration of Partially Overlapping Point Clouds. *IEEE Robot. Autom. Lett.* **2021**, *7*, 1502–1509. [CrossRef]
106. Besl, P.J.; McKay, N.D. A method for registration of 3-D shapes. *IEEE T Pattern Anal.* **1992**, *14*, 239–256. [CrossRef]
107. Wang, H.; Liu, Y.; Dong, Z.; Wang, W.; Yang, B. You Only Hypothesize Once: Point Cloud Registration with Rotation-equivariant Descriptors. *arXiv* **2021**. *preprint*. [CrossRef]
108. Huang, R.; Yao, W.; Xu, Y.; Ye, Z.; Stilla, U. Pairwise Point Cloud Registration Using Graph Matching and Rotation-Invariant Features. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
109. Chen, H.; Liu, S.; Chen, W.; Li, H.; Hill, R. Equivariant Point Network for 3D Point Cloud Analysis. In Proceedings of the CVPR, Nashville, TN, USA, 20–25 June 2021; pp. 14509–14518.
110. Li, L.; Fu, H.; Ovsjanikov, M. WSDesc: Weakly Supervised 3D Local Descriptor Learning for Point Cloud Registration. *IEEE Trans. Vis. Comput. Graph.* **2022**, 1. [CrossRef]
111. Zhang, Z.; Chen, G.; Wang, X.; Shu, M. DDRNet: Fast point cloud registration network for large-scale scenes. *ISPRS J. Photogramm. Remote Sens.* **2021**, *175*, 184–198. [CrossRef]
112. Lee, D.; Hamsici, O.C.; Feng, S.; Sharma, P.; Gernoth, T. DeepPRO: Deep partial point cloud registration of objects. In Proceedings of the ICCV, Montreal, QC, Canada, 10–17 October 2021; pp. 5683–5692.
113. Min, T.; Kim, E.; Shim, I. Geometry Guided Network for Point Cloud Registration. *IEEE Robot. Autom. Lett.* **2021**, *6*, 7270–7277. [CrossRef]
114. Wu, B.; Ma, J.; Chen, G.; An, P. Feature interactive representation for point cloud registration. In Proceedings of the ICCV, Montreal, QC, Canada, 10–17 October 2021; pp. 5530–5539.
115. Song, Y.; Shen, W.; Peng, K. A novel partial point cloud registration method based on graph attention network. *Vis. Comput.* **2023**, *39*, 1109–1120. [CrossRef]
116. Kadam, P.; Zhang, M.; Liu, S.; Kuo, C.C.J. R-PointHop: A Green, Accurate, and Unsupervised Point Cloud Registration Method. *IEEE Trans. Image Process.* **2022**, *31*, 2710–2725. [CrossRef] [PubMed]
117. Zhang, Z.; Sun, J.; Dai, Y.; Fan, B.; He, M. VRNet: Learning the Rectified Virtual Corresponding Points for 3D Point Cloud Registration. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 4997–5010. [CrossRef]
118. Wang, Y.; Yan, C.; Feng, Y.; Du, S.; Dai, Q.; Gao, Y. STORM: Structure-Based Overlap Matching for Partial Point Cloud Registration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 1135–1149. [CrossRef]
119. Wang, H.; Liu, X.; Kang, W.; Yan, Z.; Wang, B.; Ning, Q. Multi-features guidance network for partial-to-partial point cloud registration. *Neural Comput. Appl.* **2022**, *34*, 1623–1634. [CrossRef]
120. Gu, X.; Tang, C.; Yuan, W.; Dai, Z.; Zhu, S.; Tan, P. RCP: Recurrent Closest Point for Point Cloud. In Proceedings of the CVPR, Orleans, LA, USA, 18–24 June 2022; pp. 8216–8226.
121. Wang, B.; Chen, C.; Cui, Z.; Qin, J.; Lu, C.X.; Yu, Z.; Zhao, P.; Dong, Z.; Zhu, F.; Trigoni, N.; et al. P2-Net: Joint Description and Detection of Local Features for Pixel and Point Matching. *arXiv* **2021**, arXiv:2103.01055.
122. Dang, Z.; Wang, L.; Qiu, J.; Lu, M.; Salzmann, M. What Stops Learning-based 3D Registration from Working in the Real World? *arXiv* **2021**. *preprint*. [CrossRef]
123. Griffiths, D.; Boehm, J. SynthCity: A large scale synthetic point cloud. *arXiv* **2019**. *preprint*. [CrossRef]
124. Xiao, A.; Huang, J.; Guan, D.; Zhan, F.; Lu, S. Synlidar: Learning from synthetic lidar sequential point cloud for semantic segmentation. *arXiv* **2021**. *preprint*. [CrossRef]
125. Fang, J.; Yan, F.; Zhao, T.; Zhang, F.; Zhou, D.; Yang, R.; Ma, Y.; Wang, L. Simulating LIDAR point cloud for autonomous driving using real-world scenes and traffic flows. *arXiv* **2018**, *1*, preprint. [CrossRef]

126. Wang, F.; Zhuang, Y.; Gu, H.; Hu, H. Automatic Generation of Synthetic LiDAR Point Clouds for 3-D Data Analysis. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 2671–2673. [CrossRef]
127. Huang, X.; Qu, W.; Zuo, Y.; Fang, Y.; Zhao, X. IMFNet: Interpretable Multimodal Fusion for Point Cloud Registration. *arXiv* **2021**. *preprint*. [CrossRef]
128. Sun, C.; Jia, Y.; Guo, Y.; Wu, Y. Global-Aware Registration of Less-Overlap RGB-D Scans. In Proceedings of the CVPR, Orleans, LA, USA, 18–24 June 2022; pp. 6357–6366.
129. Hou, M.; Li, S.; Jiang, L.; Wu, Y.; Hu, Y.; Yang, S.; Zhang, X. A New Method of Gold Foil Damage Detection in Stone Carving Relics Based on Multi-Temporal 3D LiDAR Point Clouds. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 60. [CrossRef]
130. Matwij, W.; Gruszczyński, W.; Puniach, E.; Ćwiąkała, P. Determination of underground mining-induced displacement field using multi-temporal TLS point cloud registration. *Measurement* **2021**, *180*, 109482. [CrossRef]
131. Pang, L.; Liu, D.; Li, C.; Zhang, F. Automatic Registration of Homogeneous and Cross-Source TomoSAR Point Clouds in Urban Areas. *Sensors* **2023**, *23*, 852. [CrossRef] [PubMed]
132. Qin, N.; Hu, X.; Dai, H. Deep fusion of multi-view and multimodal representation of ALS point cloud for 3D terrain scene recognition. *ISPRS J. Photogramm. Remote Sens.* **2018**, *143*, 205–212. [CrossRef]
133. Zováthi, Ö.; Nagy, B.; Benedek, C. Point cloud registration and change detection in urban environment using an onboard Lidar sensor and MLS reference data. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *110*, 102767. [CrossRef]
134. Huang, X.; Wang, Y.; Li, S.; Mei, G.; Xu, Z.; Wang, Y.; Zhang, J.; Bennamoun, M. Robust real-world point cloud registration by inlier detection. *Comput. Vis. Image Underst.* **2022**, *224*, 103556. [CrossRef]