A coarse-to-fine algorithm for matching and registration in 3D cross-source point clouds

Xiaoshui Huang, Jian Zhang, Qiang Wu, Lixin Fan, Chun Yuan

***Abstract*—We propose an efficient method to deal with the matching and registration problem found in cross-source point clouds captured by different types of sensors. This task is especially challenging due to the presence of density variation, scale difference, a large proportion of noise and outliers, missing data and viewpoint variation. The proposed method has two stages: in the coarse matching stage, we use the ESF descriptor to select potential K regions from the candidate point clouds for the target. In the fine stage, we propose a scale embedded generative GMM registration method to refine the results from the coarse matching stage. Following the fine stage, both the best region and accurate camera pose relationships between the candidates and target are found. We conduct experiments in which we apply the method to two applications: one is 3D object detection and localization in street-view ourdoor (LiDAR/VSFM) cross-source point clouds, and the other is 3D scene matching and registration in indoor (KinectFusion/VSFM) cross-source point clouds. The experiment results show that the proposed method performs well when compared with the existing methods. It also shows that the proposed method is robust under various sensing techniques such as LiDAR, Kinect and RGB camera.**

***Index Terms*—Cross-source; point cloud; registration; matching; detection; localization; robotics;smart city**

I. Introduction

Researchers have shown great interest in matching and registration applications, such as view searching in smart cities, location-based services, street-view reconstruction and augmented reality. With the progress of sensing technology, many types of 3D point cloud sensors have been developed. Cross-source point clouds originate from different types of sensors. Compared to the same source, matching and registration on cross-source point clouds show great generalization. In this paper, we propose a method for conducting the matching and registration of a small-scale point cloud (e.g. Visual Structure from Motion (VSFM [1])) on a large-scale point cloud (e.g. street-view).

Figure 1 is a typical example of cross-source point clouds, containing cross-source problems. At least four challenges arise in solving cross-source point cloud matching and registration: (1) Density variation. Different levels of sampling density and sampling theory from various sensors result in large differences in the number of 3D points in contrasting types of point clouds. One point cloud may therefore be much denser than another; (2) Scale variation. Because the point clouds sense within their own local coordinate system, it is difficult to maintain same scale metric in two different types of sensor. Furthermore, the scale information in relation to point clouds reconstructed by VSFM is usually unknown, thus it is necessary to estimate the scale; (3) Noise, outliers and missing data. Different sensing mechanisms create a large amount of noise and a large number of outliers in cross-source point clouds, and some parts of scenes cannot produce points in point clouds. For example, VSFM is unable to generate points in textureless images; (4) Viewpoint variation. Viewpoint divergence is normal in 3D cross-source point clouds because the 3D cross-source point clouds are captured from different kinds of sensors. In a 3D point cloud, viewpoint variation often leads to the partial overlapping of point clouds, which is extremely challenging for many registration methods. Moreover, occlusions caused by viewpoint change may look remarkably different between cross-source point clouds.

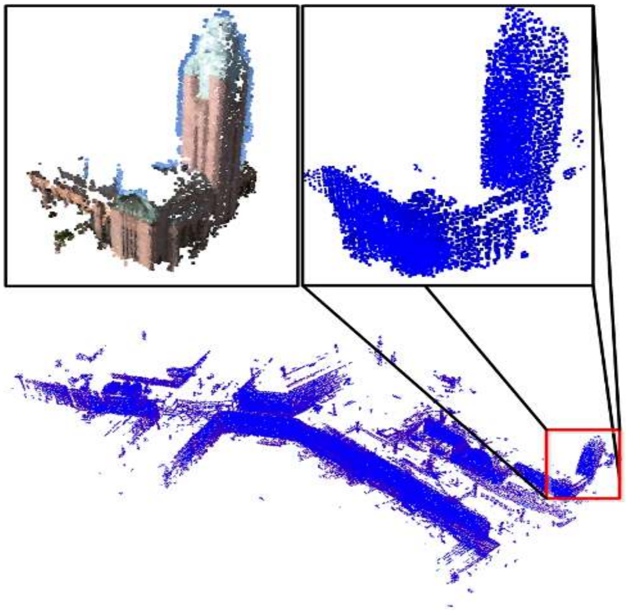

Fig. 1. An example of cross-source point clouds of VSFM and LiDAR highlighted from the street view scene. The top left is the VSFM point cloud and the top right is the detected registration result on LiDAR point cloud.

The goal in point cloud matching is to identify potential regions for the small-scale point cloud in the large-scale point cloud, where the small-scale point cloud is usually describing a small part of the large-scale point cloud. Existing methods can be categorized as feature-based methods [2] or slice window methods [3] .

Xiaoshui Huang, Jian Zhang and Qiang Wu are with Global Big Data Technologies Centre, School of Computing and Communication. University of Technology Sydney, Australia. (Emails: Xiaoshui.Huang@student.uts.edu.au, Jian.Zhang@uts.edu.au and Qiang.Wu@uts.edu.au)

Lixin Fan is with Nokia Technology Company, Finland (Email: lixin.fan@nokia.com).

Chun Yuan is with Graduate school of Shenzhen, Tsinghua University, China (yuanc@sz.tsing.edu.cn).

Existing methods of point cloud registration fall into two categories: direct methods and transformation methods. Direct methods use point cloud coordinates and extract descriptors to assist matching and registration. Typical examples are iterative closest point (ICP) [4] and other feature-based methods [5], [6], [7]. These methods deal with the registration problem in same-source point clouds however they experience many limitations when handling the cross-source point cloud registration problem (See Section V). Their aim is to look for exact matching points which make up only a small proportion in cross-source point clouds. Consequently, matched point searching with direct methods is difficult when there is a large amount of outliers and noise. In addition, these methods rely on initialization. For example, similar to our work, Peng. et al [8] proposed a two-stage algorithm using ICP. Due to the limitations of ICP, the accuracy of the final registration results is low, and the results are not visually registered correctly.

Transformation methods first transform 3D point clouds into another space or another model and then use these transformed data for matching and registration. A typical example is the Gaussian mixture model (GMM) [9], [10], which uses GMM to describe and match point clouds. These methods focus more on global information while ignoring local structure distortion. They have many advantages over direct methods in dealing with cross-source problems. Our algorithm belongs to the category of transformation methods.

In this paper, a novel coarse-to-fine algorithm is proposed to match and register two cross-source point clouds. There are two main stages: 1) in the coarse stage, top K potential regions are detected by coarse matching , which are regarded as the candidates of point cloud significantly overlapping with the 2nd point cloud set; 2) in the fine stage, two cross-source point clouds are assumed to be scaled samples according to the same GMM model. We register them based on this assumption and use the registration error to refine the ranking in the first stage.

The main contributions can be summarized as follows: (1) an effective coarse-to-fine pipeline is proposed which specifically considers the problem caused by scale changing issue seen in cross-source point clouds. The key aspects of our pipeline are: the top K potential regions are coarsely selected by using the efficient Ensemble of Shape Functions (ESF) [2] descriptor; and a scale-embedded generative GMM method is proposed to refine the coarse selected regions, which takes into account the impact of scale variation in two cross-source point clouds. This is not seen in [38]; (2) scale difference between the cross-source point cloud has been considered into a generative GMM cost function.

II. Related work

Same-sourced point clouds are captured from the same kinds of sensors (e.g. all captured from Kinect), while cross sources are captured from different kinds of sensors (e.g one from Kinect, the other from a RGB camera). Because only a few methods directly focus on cross-source point clouds matching and registration problem, in this section, the related methods are reviewed in terms of their ability to deal with the four challenges raised in cross-source point cloud matching and registration. The existing 3D matching methods can be categorized into feature-based and slice window methods. The existing registration methods can be divided into two categories: direct methods and transformed methods.

A. 3D Matching

The 3D matching methods can be divided into two types: feature-based methods and slice window methods. The key element of the feature-based methods is to design a discriminative 3D feature. Regarding 3D features [1], the existing methods can be categorized as global [2] and local categories [11]. The key elements of the slice window methods are the searching strategy and the compare strategy. The typical example is [3], which uses a voting scheme to efficiently detect 3D objects in 3D point clouds. [12] uses CNN to generate 3D object proposals and is applied to autonomous driving. In this paper, as there is a large variation in cross-source point clouds, we choose global feature-based method to overcome the local variations and efficiently detect the potential regions in the large-scale point cloud.

B. Registration: direct methods

Direct point set registration methods usually minimize Euclidean distances between nearby points. The typical example is the iterative closest point (ICP) [4] algorithm [13], [14], [15]. Due to its efficiency, it is widely used in same-source and cross-source registration. [16] also improves ICP to deal with non-rigid point set matching and incorporates a outlier detection strategy. In summary, the above methods are all heuristic-related methods; hence they cannot guarantee global optimality of the solutions. Go-ICP [17] improves ICP to obtain a globally optimal solution by combining ICP with a branch-and-bound (BnB) scheme. Also, [18] proposes a method using L2E estimator and ICP, which is potential to handle 2D and 3D registration. The only difference for 2D and 3D is to select different feature descriptor for its points (shape context for 2D and spin image for 3D). Despite these improvements to the ICP method, these direct registration approaches are intrinsically sensitive to missing data, large variations in point densities and scale differences, thus rendering them useless for cross-source point cloud matching and registration (see the experimental results in Section V for examples).

Different to these ICP-based methods, when scan pairs start in arbitrary initial poses, registration amounts to solving a global problem to find the best aligning rigid transform over the 6DOF space of all possible rigid transforms comprising translations and rotations. Since aligning rigid transforms is uniquely determined by three pairs of (non-degenerate) corresponding points, one popular strategy is to invoke RANSAC [19] to find such aligning triplets of point pairs [20]. However, RANSAC regularly degrades to its worst case $O(n^3)$ complexity in the number n of data samples in the presence

[1] For more information about 3D features. See http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_4:_3D_object_recognition_(descriptors)

of partial matching with low overlap. Various alternatives have been proposed to encounter the cubic complexity: hierarchical representation in the normal space [21]; super-symmetric tensors to represent the constraints between the tuples [22] ; stochastic non-linear optimization to reduce distance between scan pairs [23]; branch-and-bound using pairwise distance invariants [24]; and evolutionary game theoretic matching [25], [26] as an alternative to RANSAC. However, these methods are all sensitive to missing data.

Following the concept of RANSAC, another kind of method is 4PCS [27], which uses the randomized alignment approach and the idea of planar congruent sets to compute optimal global rigid transformation. The 4PCS is widely used and has also been extended to take into account uniform scale variations [28]. However, these methods have a complexity of $O(n * 2 + k)$ where n denotes the size of the point clouds and k the set of candidate congruent 4-points. It has a great limitation when the point number is large. To remove the quadratic complexity of the original 4PCS, [29] extends it to a fast algorithm where only linear computation time is needed. It reports the points or spheres in R^3 and uses a smart index to quickly find the matched plane in all candidate congruent 4-point planes. One cross-source point cloud registration experiment is reported in this paper. However, these methods face challenges when confronted with complete cross-source problems.

Although these direct methods show some degree of ability in addressing parts of the cross-source problem, none of them deal with all the cross-source problems. To robustly deal with all the cross-source problems, we reformulate the registration with the cross-source scales into a generative GMM cost function and propose a method considering two cross-source point clouds as scaled samples from a GMM. More specifically, we treats all the point sets on an equal footing. The GMM parameters and rigid transformation are simultaneously estimated by smoothly transforming two point clouds in a consistent GMM.

C. Registration: transformed methods

The typical example of transformed methods are feature-based methods, extracting features from 3D point clouds and transforming the point cloud registration Euclidean space to a feature space. The typical 3D feature extraction methods are FPFH [30], ESF [2], Spin image [31] and SHOT [32]. Torki [33] uses local features in images to learn manifold symbol. They first learn a feature embedding representation that harbors spatial structure of the features as well as the local appearance similarity. Following learning, they use out-of-sample method to embed features from a new image. Similarly, Yuan [34] transforms every given point in point clouds into a shape representation in order to cast the problem of point set matching as a shape registration problem, which is the Schrodinger distance transform (SDT) representation. These feature-based methods have exciting results on same-source point clouds. However, extracting reliable features is very difficult in cross-source point clouds which always cause feature-based methods to fail on cross-source point clouds.

Related to point cloud registration, GMM-based methods are another kind of method to deal with point set registration. Bing et al. [35] proposed an method to represent point clouds as Gaussian Mixture Models(GMM) and, subsequently, to solve the registration problem by minimizing the statistical discrepancies between corresponding GMMs. The approach can be used for both rigid and non-rigid point cloud registration, and has shown the ability to deal with noise and outliers to some extent. Georgios et al. [36] introduce a motion drift idea into the GMM framework and achieve good results on rigid and non-rigid point set registration. [37] extracts the convex hull and uses GMM to estimate the transformation matrix on these convex sets of a point set. [38] proposes a joint registration of a multiple point cloud (JR-MPC) solution to GMM-based approach by recasting registration as a clustering problem. In the previous GMM-based registration methods [36], [35], [37], they estimate one GMM using one point clouds or two GMMs using two point clouds. This makes the reasonable assumption that points from one set are normally distributed around points belonging to the other set. Hence, the point-to-point assignment problem can be recast into that of estimating the parameters of a mixture distribution or minimizing the GMM distance. However, when there is cross-source scale, missing data and density variation in cross-source point clouds, the estimated GMMs originally show a lot of difference which results in registration inaccuracy or even failure.

Although these aforementioned transformed methods shows ability in dealing with parts of noise and outliers or density variation, none of them successfully addresses the cross-source registration problem, which contains scale, density variation, noise and outliers and missing data. In this paper, we aim to deal with the difficult cross-source problem. Motivated by the sensor capturing process and [38], a scale embedded generative GMM method is proposed to robustly register two cross-source point clouds. Our algorithm assumes there is a virtual GMM describing the real scene and the two cross-source point clouds are two scaled samples to the GMM. This procedure is similar to capturing different point clouds from different sensors for the same scene. The different point clouds within their local coordinate system are scaled samples to the same scene [2] .

The remaining sections of this paper are organized as follows: Section III describes the proposed coarse-to-fine algorithm; Section IV provides a discussion and the implementation details; Section V describes the experiments and applications; and Section VI concludes the paper.

III. Coarse-to-fine Algorithm

The coarse-to-file algorithm is illustrated in Figure 2. It comprises coarse matching and fine registration, where coarse matching aims at finding the top K potential regions in the candidate point cloud that potentially match with the target point cloud. It substantially reduces the number of candidate regions and hence saves computation cost in the next

[2] The reason for the scale samples is that cross-source point clouds usually show different scale metrics when put in a same coordinate system.

stage. We compute ESF descriptors of these potential regions and use them to conduct the first coarse matching. Then, a scaled embedded generative GMM registration is performed to obtain the transformation of two cross-source point clouds and use the transformation error to refine the matching results. Furthermore, the main steps of the second stage are (1) obtaining the transformation matrix of each registration; (2) acquiring the residual error of each registration by applying the transformation matrix to the original two cross-source point clouds (e.g. selected LiDAR region and VSFM); and (3) using the residual error to re-rank the matching results and output the ranked registration results. After registration, an accurate transformation matrix is obtained which can be used for applications such as location-based services.

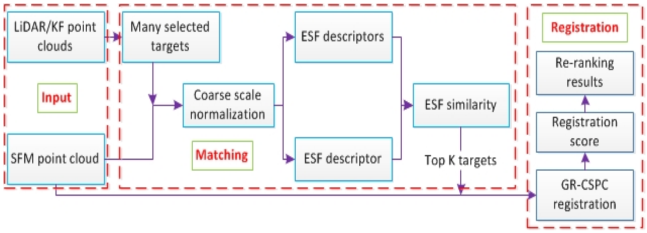

Fig. 2. Overview of the proposed coarse-to-fine algorithm. With Lidar/KinectFusion(KF) and VSFM cross-source point clouds input, the matching stage aims at detecting the most potential registration targets; the registration stage aims at finding the optimal registration relation and refining the previous matching results.

A. Coarse Matching

In the coarse matching stage, top K potential regions are obtained for the VSFM point cloud from LiDAR/KinectFusion (KF) point clouds. In this stage, ESF descriptors are computed for two selected point clouds (VSFM point cloud and LiDAR/KF targets) and used for coarse matching. Using ESF alone is very hard to find the correct result for top potential regions in the cross-source problems, hence, to detect the most reliable result, the fine registration step is indispensable.

B. Fine Registration

In this paper, we propose a Generative gaussian mixture Model for the Cross-Source Point Cloud registration (GM-CSPC). We consider that two cross-source point clouds are scaled samples from a virtual GMM and the GMM is generated from these two point clouds. If they come from the same GMM, the registration error will be very small. We select GMM because it is a robust model to describe a complexity scene [39]. The GMM is robust to density, missing data, noise and outliers. The model is simple whilst being very robust to describe a complexity scene in real world.

To achieve the overall pipeline, a scale embedded generative GMM method is proposed, which has considered the scale issue in the case of cross-source point cloud. It is rarely discussed in the existing paper. Our method is inspired by [38], but different from [38], we assume that there is one virtual GMM to describe the scene or object, and the two cross-source point clouds are two different samples with the different scales and captured by different sensors (Figure 3). Figure 4 shows [38] and the proposed scale embedded generative GMM method for cross-source point cloud registration. However, [38] fails at cross-source point cloud registration because the scale problem has not been considered in the multiple point set registration. In this paper, the challenging factors including scale, rotation and translation are systematically considered in a generative GMM process.

More precisely, we formulate the registration of cross-source point clouds into a model generative problem, where two point clouds are different samples of a GMM and they have scale, rotation and translation transformation. We propose a scaled embedded generative GMM method to solve it. At the optimum, two cross-source point clouds become registered and the rigid transformation of the two cross-source point clouds is obtained by maximizing the GMM posterior probability. Core to our registration method is the generative concept considering two point clouds as different scaled samples for a virtual object/scene (describing as GMM), and optimizing the GMM parameters and transformation matrix simultaneously.

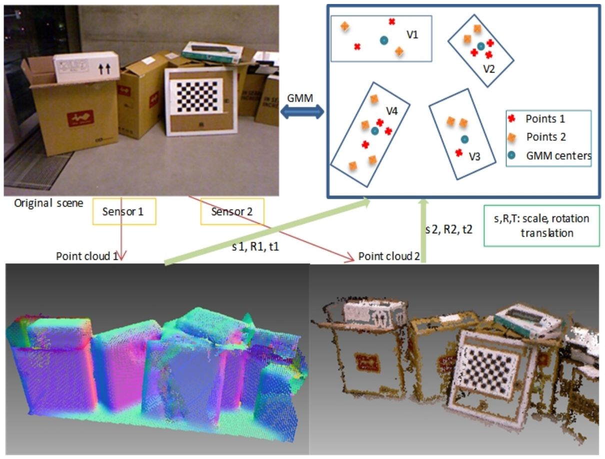

Fig. 3. Problem definition: the proposed Generative Model for Cross-Source Point Cloud registration (GM-CSPC). The proposed algorithm simultaneously estimates both the GMM and transformation parameters (scale, rotation and translation).

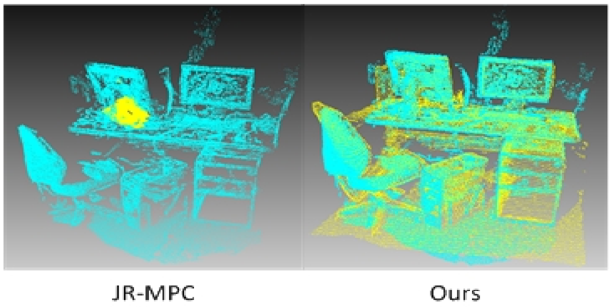

Fig. 4. The registration results of JR-MPC [38] and the proposed scale embedded generative model for cross-source point clouds. The green is VSFM point cloud, the red is KinectFusion point cloud.

The two cross-source point clouds are represented as Y_1 and Y_2, which is $N_1 \times 3$ and $N_2 \times 3$ matrix respectively, N_1 and N_2 are 3D point number of Y_1 and Y_2. According to [36],

[39], considering all GMM components as equal membership, the mixture model considering the noise and outliers can be written as:

$$p(X_{ji}) = (1-w)\sum_{k=1}^{K}\frac{1}{K}p(T_j(X_{ji})|u_k,\sigma_k) + w\frac{1}{h} \tag{1}$$

where T_j is rigid transformation model, $T_j(X_{ji}) = s_j R_j X_{ji} + t_j$, s_j is a scale factor, R_j is a 3×3 rotation matrix, t_j is 3×1 translation matrix; u_k and σ_k are mean and variance parameters of k^{th} Gaussian model; w is the weight of noise and outliers, $0 \leq w \leq 1$; h is the volume of the 3D convex hull encompassing the data [41]; K is the number of Gaussian models.

The parameters can be estimated by using the framework of expectation-maximization. We define a corresponding latent variable $Z = z_{ji}$, where $z_{ji} = k$ means $T_j(X_{ji})$ is assigned to the k-th component of the GMM. The complete data set is $\{X, Z\}$. In order to compute the parameters of GMM, we need to maximize the complete-data log likelihood. However, the complete data is usually not given and only incomplete data X can be utilized. According to [39], the complete-data log likelihood can be computed by E and M steps. In the E step, we estimate the posterior probability of the latent variables given by $P(Z|X, u_k, \sigma_k, T_j)$; in the M step, we use this posterior probability to find the maximization of the expectation of the complete-data likelihood function, which is

$$Q(\theta) = \sum_{Z} p(Z|X,\theta) log(p(X,Z|\theta)) \tag{2}$$

where θ represents the parameters containing u_k, σ_k, $k = 1...K$ and T_j, $j = 1, 2$.

According to [39], ignoring the constants independent of θ, (2) can be rewritten as

$$\begin{aligned} Q(\theta) = -\frac{1}{2}\sum_{j=1}^{2}\sum_{i=1}^{N_j}\sum_{k=1}^{K}\alpha_{jik}(\|T_j(X_{ji}) - u_k\|^2 \\ + log|\sigma_k|),\ s.t.\ R^T R = I, det(R) = 1. \end{aligned} \tag{3}$$

where α_{ijk} is the posterior probability which can be computed by the previous parameter values:

$$\alpha_{jik} = \frac{p_k \sigma_k^{-3} exp(-\frac{1}{2\sigma_k^2} \parallel T_j(X_{ji}) - u_k \parallel^2)}{\sum\limits_{s=1}^{K}[p_s \sigma_s^{-3} exp(-\frac{1}{2\sigma_s^2} \parallel T_j(X_{ji}) - u_s \parallel^2)] + \beta} \tag{4}$$

where $w = w/h(w+1)$ accounts for the outlier term and $\alpha_{ji(K+1)} = 1 - \sum\limits_{k=1}^{K}\alpha_{jik}$ accounts for the posterior could be a outlier.

E-step. Computing the posterior probability is the E-step. In this step, the previous θ value and equation (4) are used to compute the posterior probability α_{jik}. Note that the computation of the posterior probability in the i^{th} step needs the parameters of the $(i-1)^{th}$ step.

M-step. With the posterior probability known, the maximization(M) step aims at estimating the parameter of θ by maximizing the objective function $Q(\theta)$. As the T_j associated with each point cloud are shared with the same GMM parameters, they can be estimated independently. By setting the current GMM parameters, the estimation of T_j can by reformulated as the following constraint problem [36]

$$\begin{cases} \min\limits_{s_j, R_j, t_j} \|(s_j R_j V_j + t_j - X)\Lambda_j\|_F^2 \\ s.t. \quad R_j^T R_j = I,\ |R_j| = 1 \end{cases} \tag{5}$$

where X is the weighted value of whole means of GMM components. $\|\cdot\|_F$ is the Frobenois norm; V_j is the virtual 3D points related to given points which is given by

$$V_{jk} = \frac{\sum_{i=1}^{N_j}\alpha_{jik}X_{ji}}{\sum_{i=1}^{N_j}\alpha_{jik}} \tag{6}$$

In order to solve formulation 5, we introduce the following Theorem 1. It has a close-form solution.

Theorem 1. *Let A and B be two $m \times n$ point clouds, m is the points' dimension, and UDV is the singular value decomposition of $\bar{A}\Lambda\Lambda^T\bar{B}^T$ ($\bar{A} = A - \frac{A\Lambda^2}{tr(\Lambda^2)}, \bar{B} = A - \frac{B\Lambda^2}{tr(\Lambda^2)}$),$\Lambda$ is a weight matrix. The minimum value of ξ of the* ***weighted mean squared error***

$$\xi(s, R, t) = \|(sRA + t - B)\Lambda\|_F^2 \tag{7}$$

of two point clouds with respect to their transformation matrices (s: scale factor, R: rotation and t:translation matrix) are given as

$$R = USV^T \tag{8}$$

$$t = -\frac{1}{tr(\Lambda)^2}(sRA - B)\Lambda^2 \tag{9}$$

$$s = \frac{tr((\bar{A}\Lambda\Lambda^T\bar{B}^T)^T R)}{tr\{(R\bar{A})^T(R\bar{A})\}} \tag{10}$$

where $S = diag(1, 1, det(UV^T))$.

Proof: Equation (7) can be rewritten as

$$\begin{aligned} \xi(s, R, t) &= ((sRA + t - B)\Lambda)^T((sRA + t - B)\Lambda) \\ &= \{((sRA - B)\Lambda)^T((sRA - B)\Lambda)\} \\ &\quad + 2\{((sRA - B)\Lambda)^T t\Lambda\} + \{(t\Lambda)^T(t\Lambda)\} \end{aligned} \tag{11}$$

Taking the partial derivative of $\xi(s, R, t)$ with respect to t, we obtain:

$$\frac{\partial\xi(s, R, t)}{\partial t} = ((sRA - B)\Lambda)^T\Lambda + \Lambda^T\Lambda t$$

Setting $\frac{\partial\xi(s,R,t)}{\partial t} = 0$, we obtain:

$$t = -\frac{1}{tr(\Lambda^2)}(sRA - B)\Lambda^2) \tag{12}$$

Substituting (12) back into (7) and represent $\bar{A} = A - \frac{A\Lambda^2}{tr(\Lambda^2)}$, $\bar{B} = B - \frac{B\Lambda^2}{tr(\Lambda^2)}$, $R^T R = I$, we obtain:

$$\xi(s, R, t) = -2s * tr((\bar{A}\Lambda\Lambda^T\bar{B}^T)^T R) \tag{13}$$

In order to prove rotation matrix R in equation (8), we need to introduce a Lemma [42].

Lemma 1. *Let $R_{D\times D}$ be an unknown rotation matrix and $A_{D\times D}$ be a known real square matrix. Let USV^T be a Singular Value Decomposition of A, where $UU^T = VV^T = I$*

and $S = d(S_i)$, *with* $s_1 \geq s_2 \geq, ..., \geq s_D \geq 0$. *Then, the optimal rotation matrix* R *that maximizes* $tr(A^T R)$ *is* $R = UCV^T$, *where* $C = d(1, 1, ..., det(UV^T))$.

Using this Lemma and equation (13), we can conclude that

$$R = USV^T \tag{14}$$

where, U and V are matrices from the singular value decomposition of matrix $\bar{A}\Lambda\Lambda^T\bar{B}^T$, and $S = diag(1, 1, det(UV^T))$.

To proof scale equation in (10), we need to rewrite (10) and take partial derivative of $\xi(s, R, t)$ with respect to s,

$$\begin{aligned}\frac{\partial\xi(s,R,t)}{\partial s} =& 2s * tr(R\bar{A}^T(R\bar{A})^T) \\ &- 2tr((\bar{A}\Lambda\Lambda^T\bar{B}^T)^T R)\end{aligned}$$

setting $\frac{\partial\xi(s,R,t)}{\partial s} = 0$, we obtain

$$s = \frac{tr((\bar{A}\Lambda\Lambda^T\bar{B}^T)^T R)}{tr\{(R\bar{A})^T(R\bar{A})\}} \tag{15}$$

We have proofed the theorem.

Using $Theorem1$, the optimal parameters in equation (5) are obtained as a close-form solution given as

$$R_j^{new} = USV^T \tag{16}$$

$$t_j^{new} = -\frac{1}{tr(\Lambda)^2}(sRV_j - X)\Lambda_j^2 \tag{17}$$

$$s_j^{new} = \frac{tr((\bar{V_j}\Lambda\Lambda^T\bar{X}^T)^T R)}{tr\{(R\bar{V_j})^T(R\bar{V_j})\}} \tag{18}$$

where $UDV^T = svd(\bar{V_j}\Lambda_j\Lambda_j^T\bar{X}^T)$, $S = diag(1, 1..., det(UV^T))$.

After the transformation parameters θ are obtained, we use the new θ and the posterior probability to compute the GMM parameters. For the means x_k of GMM, it can be easily obtained by taking the partial derivative of (3) with respect to x_k and setting to 0. $\partial Q(\theta)/\partial x_k = 0$. Then, we substitute the new x_k to equation (3) and set $\partial Q(\theta)/\partial\theta_k = 0$ to obtain optimal variances. These formulas of these parameters are given as

$$u_k^{new} = \frac{\sum\limits_{j=1}^{2}\sum\limits_{i=1}^{N_j}\alpha_{jik}(s_j^{new}R_j^{new}X_{ji} + t_j^{new})}{\sum\limits_{j=1}^{2}\sum\limits_{i=1}^{N_j}\alpha_{jik}} \tag{19}$$

$$\begin{aligned}(\sigma_k^{new})^2 =& \frac{\sum\limits_{j=1}^{2}\sum\limits_{i=1}^{N_j}\|\alpha_{jik}(s_j^{new}R_j^{new}X_{ji} + t_j^{new} - u_k^{new}\|}{3\sum\limits_{j=1}^{2}\sum\limits_{i=1}^{N_j}\alpha_{jik}} \\ &+ \varepsilon^2\end{aligned} \tag{20}$$

where ε^2 is a very small positive value to avoid singularities [41].

```
Algorithm 1 Generative registration algorithm
Input: Two cross-source  point clouds $P_1, P_2$
Output: $\theta$. $(u_k, \sigma_k, s_j, R_j, t_j)$
  Initialization : $\theta \leftarrow 1$
  EM optimization,repeat until convergence :
    • E-step: compute $\alpha_{jik}$ by using Eq.(4)
    • M-step: compute optimal $s_j, R_j, t_j, u_k, \sigma_k$.
   ·  Solve $R_j, t_j, s_j$ by using Eq.(16), (17), (18).
   ·  Solve $u_k, \sigma_k$ by using Eq.(19), (20).
   Return T: $s = s_1/s_2, R = R_1/R_2, t = R_1(t_2 - t_1)$.
   Return aligned points: $P_2' = sRP_1 + t$.
```

IV. Implementation details and discussion

All GMM-based methods need to estimate the posterior probability of each point belonging to every Gaussian model in the expectation (E) step. The computation and memory complexity are very large which is $O(M * K + N * K)$, where M and N are the number of two point clouds and K is the Gaussian model. In the generative GMM model like the proposed GM-CSPC, it is even worse, as the complexity is $O(M * N * K)$ which is prohibitive for large scale cross-source point clouds. We will describe how to effectively deal with these problems.

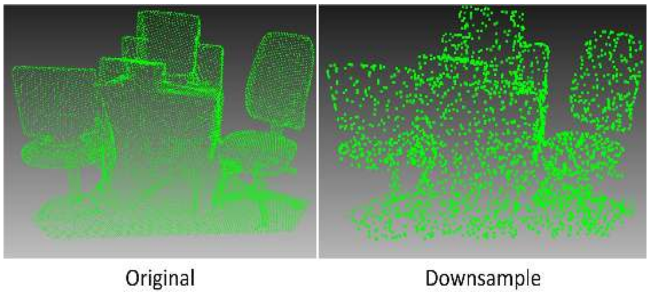

Fig. 5. Visual results of original and down-sample point clouds.

In this paper, the cross-source point cloud registration problem only involves rigid transformation. Hence, if we uniformly down-sample the point cloud, the global shape or structure and rigid transformation will remain the same as the original point cloud (see Figure 5). As the shape of the two point clouds are all retained, the region of the GM-CSPC which is depicted is the same. So, the transformation matrix computed by the down-sample point cloud is the same as the original point cloud. In this way, these two point cloud registrations can be successfully converted from a large complexity problem to a feasibility problem. If a rigid transformation is computed by using a down-sampled point cloud, we can directly apply the rigid transformation to the original point clouds and obtain the final registration results.

In the coarse matching stage, we do a rough scale normalization by assuming the 3D containing sphere of the two cross-source point clouds is the same. The ratio between two 3D containing boxes is used to conduct scale normalization as a pre-processing step before ESF selection. We do this rough scale normalization because ESF is not scale invariant. In the fine registration stage, when the GM-CSPC is completed, a residual error is computed to re-rank the matching results in the previous stage. To compute the residual error, the computed transformation matrix from the revised JR-MPC is applied to perform transformation to the original point clouds. Next, the nearest neighbour of each point in one point cloud (e.g. transformed VSFM point cloud) is computed in another point cloud (e.g. transformed Lidar point cloud) and the mean of the residual error between points and their neighbors is computed as follows, where the residual error is computed by:

$$E(T) = \frac{1}{N}\sum_{i}^{N} \|m_i - T(d_i)\|_2 \tag{21}$$

where m_i is the i^{th} point in point cloud A; and d_i is the nearest neighbor of m_i in the matched point cloud B. A lower $E(T)$ means the two point clouds are more similar. However, based on our observation, the $E(T)$ always has a lower value in small-scale point clouds. To eliminate this scale bias, a penalty is defined related to the scale value:

$$E'(T) = exp(-\frac{s^2}{\alpha}) * E(T) \tag{22}$$

where, $exp(-\frac{s^2}{a})$ is the penalty for scale variation. α is the parameter to control the penalty, and scale is estimated by the proposed generative GMM registration method. The final ranking regions are sorted by the $E'(T)$ value and the top ranked one represents the best match to the VSFM point cloud. The whole coarse-to-fine algorithm is shown in Algorithm 2.

Algorithm 2 Pseudocode of coarse-to-fine algorithm

```
Input: cross-source point clouds
Output: Top 5 Registered regions
  Matching :
    1. Select multi-scale regions from LiDAR
    2. Scale normalization
    3. Compute ESF for these regions
    4. Select Top K regions
  Registration :
    5. Down - sample point cloud
    6. Compute Transformation T by Algorithm 1
    7. Compute $E'(T)$ by Eq. (22)
    8. Re-ranking using $E'(T)$
    9. Cut off at Top 5
```

V. Experimental results

In this section, we conduct four experiments on three kinds of datasets. The first experiment is to demonstrate our scale embedded registration algorithm on cross-source scale problem. We test and compare with other scale estimation methods on scale estimation in section V-A. The following three experiments are to evaluate the performance of our pipeline and the registration algorithm in Section V-B, V-C and V-D. We build three datasets that are Lidar+VSFM,KF+VSFM and synthetic datasets. The LiDAR+VSFM and KF+VSFM datasets represent cross-source outdoor and indoor scene respectively, and the synthetic datasets contain the registration transformation ground truth to test both the pipeline and the registration. We evaluate the accuracy and the efficiency of the pipeline and the registration. The accuracy of the pipeline, we use top-5 measurement on Lidar+VSFM and correspondent relations on KF+VSFM and synthetic datasets. The accuracy of registration algorithm, we use F-norm between transformation matrix and ground truth.

A. Scale estimation comparison

Scale change in the cross-source point cloud has been considered in this paper. [43], [44] are two related methods which tried to estimate the scale value in this case. Given that the software code of [44] is not available, in this paper, only the method of [43] is compared. The comparison results show in Table I, which shows our method achieve better performance. Since the datasets in [43] is captured from same sources, it does not fully reflect cross-source problem. In order to test the ability in cross-source cases, we manually build a dataset of different point clouds using KinectFusion (KF) and Visual Structure-from-motion (VSFM) for a comprehensive comparison. Figure 6 shows our method can estimate a reasonable scale for cross-source point clouds which can transform the cross-source into a unified coordinate system, whereas it is difficult for the ratioICP [43] to achieve the same performance.

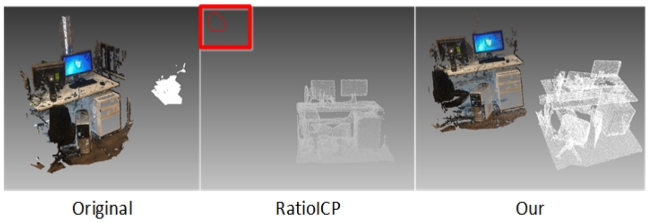

Fig. 6. Cropped point clouds from two sensing techniques. ratioICP [43] estimate the scale is 0.0032 and the proposed method estimates the scale is 0.62. The right two images are results of two scaled point clouds displaying in the same coordinate system.

B. LiDAR and VSFM cross-source point cloud datasets

The experiments are conducted on real cross-source point clouds that are built by LiDAR and VSFM point cloud respectively. The LiDAR point clouds are captured from three different scenes in Helsinki (Helsinki Cathedral, Helsinki station and Library of University of Helsinki), with hundreds of millions of points on each original LiDAR point cloud. To efficiently match and register the large volume data, the LiDAR point clouds are down-sampled into 10% of the original points. To build the VSFM point clouds, these three typical buildings were taken by a digital camera and their 2D images selected. Helsinki Station is divided into two objects: station south and station east.

We build a software-reconstructed point clouds by using the 2D images and through VSFM [1]. The four objects of LiDAR and VSFM point clouds are illustrated in Figure 7. Before applying our algorithm, standard pre-processing, such as removal of sparse outliers, is conducted on both point clouds. Considering computation complexity, the performance

TABLE I
COMPARISON OF RELATIVE SCALE ESTIMATION FOR SEVERAL METHODS, WITH ESTIMATED SCALE AND PERCENTAGE ERROR.

Dataset	Ground Truth	Standard Deviation	Mesh Resolution [45]	Keyscale [46]	Standard ICP [4]	RatioICP [43]	GLS [44]	GLS+ICP [44]	Ours
Bunny	5.000	5.000	5.000	5.000	5.000	5.000	5.000	5.000	5.000
Small Blocks (no change)	2.364	4.855 (105.37%)	1.162 (50.85%)	1.400 (40.78%)	3.029 (28.13%)	2.502 (5.84%)	2.430 (2.81%)	2.382 (1.01%)	**2.351** **(1.21%)**
Small Blocks (with change)	2.424	3.833 (58.13%)	1.684 (30.53%)	2.250 (7.18%)	2.561 (5.65%)	2.543 (4.91%)	2.525 (4.16%)	2.505 (3.34%)	**2.400** **(1.81%)**
Real Blocks	2.364	4.855 (105.37%)	1.162 (50.85%)	1.400 (40.78%)	3.029 (28.13%)	2.502 (5.84%)	2.430 (2.81%)	**2.382** **(1.01%)**	2.462 (3.20%)

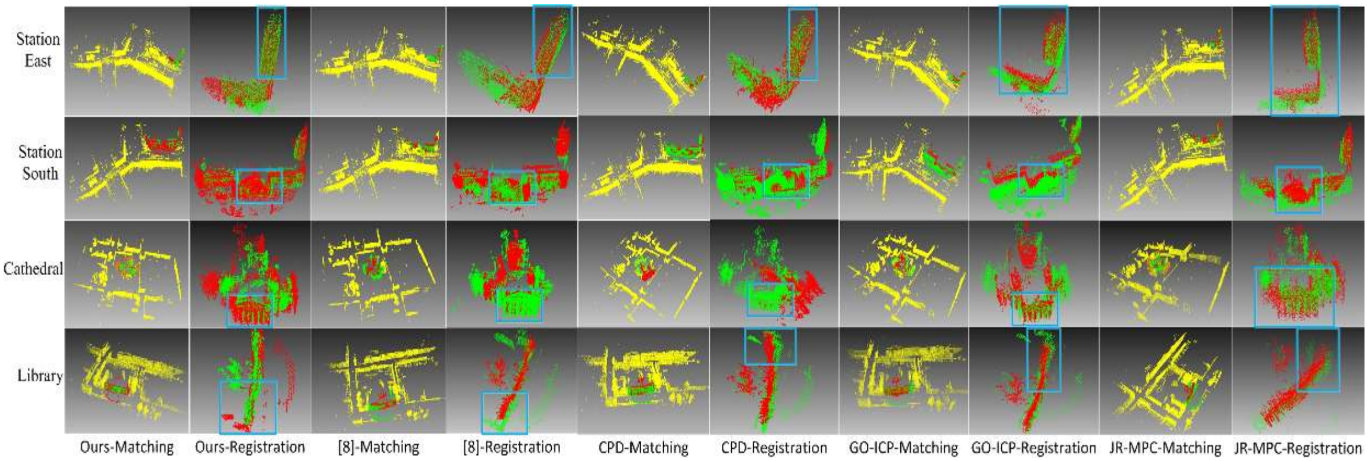

Fig. 8. The top 1 matching and registration results of the 4 objects with the proposed method and [8], CPD [36], GO-ICP [17]. Each row represents the results for one object. The odd columns show matching results and the even columns show the detailed registration results with LiDAR in green and VSFM in red. For each row, the blue regions are selected regions showing better registration accuracy of the proposed method.

TABLE II
THE PERFORMANCE OF THE PROPOSED METHOD AND THE COMPARED METHODS

	cathedral		library		station south		station east	
	accuracy	time(s)	accuracy	time(s)	accuracy	time(s)	accuracy	time(s)
Baseline: single stage ESF	4	24	0	25	2	23	0	25
Baseline: single stage ICP	5	305	5	241	0	167	5	522
ESF-64 + ICP without adjusting final residual [8]	5	85	3	73	0	56	4	139
ESF-64 + ICP with adjusting final residual [8]	5	85	3	73	4	56	5	129
ESF-64 + CPD[36]	2	623	4	656	4	443	5	115
ESF-64 + Go-ICP[17]	5	4345	3	586	3	506	5	503
ESF-64 + JR-MPC[17]	3	302	4	224	2	325	5	256
The proposed method (2000)	5	300	**5**	223	**5**	320	5	256
The proposed method (150)	5	54	**5**	65	**5**	57	5	67

of the proposed algorithm is evaluated on a subset data. The subset data is generated by 7 different scale spheres which scan all the LiDAR point clouds. The radius of the spheres ranges from 30 to 60 with an interval of 5. A hundred regions are selected in each scale. The subset data are used as candidate regions for matching and registration. In our experiments, the candidate regions cover more than 50% areas of LiDAR point clouds. In this paper, we run a retrieval-like process to evaluate the performance of registration over cross source point clouds. To generate ground-truth (GT) of the dataset, we manually select the candidate regions which cover more than 90% area of the target object and less than 10% points associated with the background from LiDAR point cloud. There are more than 5 GT regions for each VSFM point cloud. To run the evaluation, the target point cloud (VSFM) is retrieved from the 100 candidate LiDAR regions for each scale and 7 scales are applied which reaches 700 candidates in total. Since the evaluation is over the whole data sets with multi-scale regions, it includes a comprehensive set of negative regions for tests. This should well demonstrate the challenging situations.

In our study, in the first matching stage, 64-ESF sampling levels are selected, which is the best performance that can be achieved in our experiments. In the second fine registration stage, top 20 (K=20) regions are selected for the fine granularity registration. In this paper, we run a process to re-rank the top 5 candidates from those top 20 regions. The re-

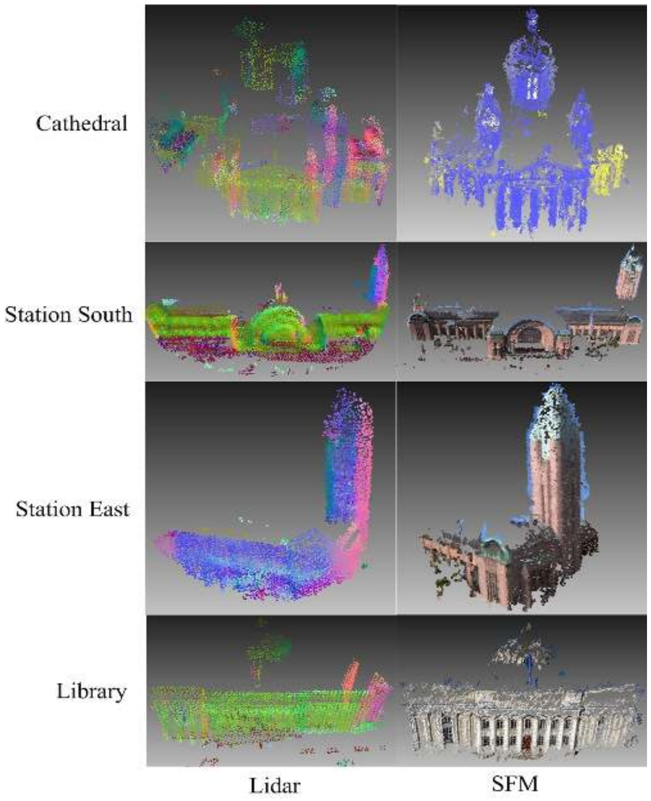

Fig. 7. Eight point clouds of four objects named Cathedral, Station south, Station east and Library. Each row represents one object with two cross-source point clouds. The left column is LiDAR and the right is VSFM.

ranking is conducted by measuring their residual error or their ESF similarity. We then find the number that top 5 candidates lies in GT candidate regions. Five baseline methods, ESF, ICP, ESF+ICP, ESF+CPD, ESF+GO-ICP are evaluated against ours. All experiments are conducted on a PC with 4-core 3.2GHz CPU and 8GB memory. The results are illustrated in Table II.

As shown in Table II, compared with the baseline of ESF+ICP, ESF+CPD, ESF+GO-ICP, the results show that our method achieves the highest accuracy in all datasets. Our method can detect all top 5 ground-truth regions from cross-source point clouds, even in a challenging case of high residual errors due to the significant differences between these two point clouds. In terms of efficiency, our method is faster than the others. In relation to the baseline ESF and the baseline ICP, they show low accuracy and low efficiency, respectively. Figure 8 shows a visual comparison of the top 1 matching and registration results. The results indicate that our algorithm achieves much better performance than any other methods in the blue box regions, especially for the Station South and Cathedral datasets.

In this paper, we conduct a comprehensively experiments to test the accuracy and time efficiency under different kernel number such as from 50 Gaussian models to 6000 Gaussian models. The details of the accuracy and time on different Gaussian models are shown in Figure 9, showing that the accuracy turning point is around 150 Gaussian models.

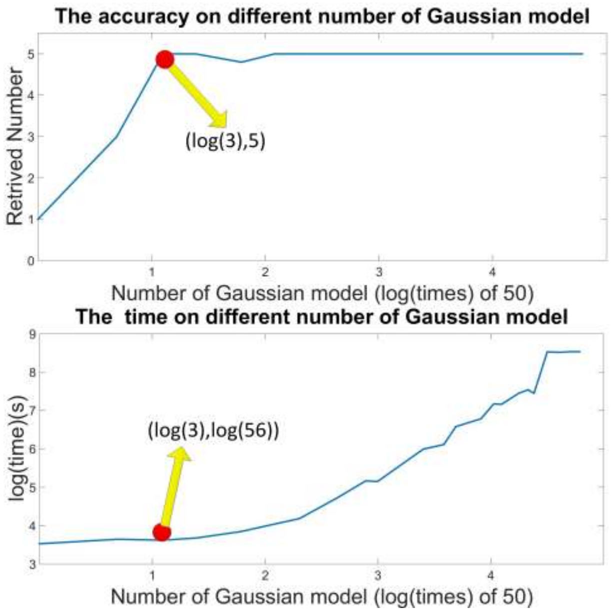

Fig. 9. The accuracy and efficiency on different Gaussian models. The turning points of accuracy and efficiency are located in (3*50,5) and (3*50,56) respectively.

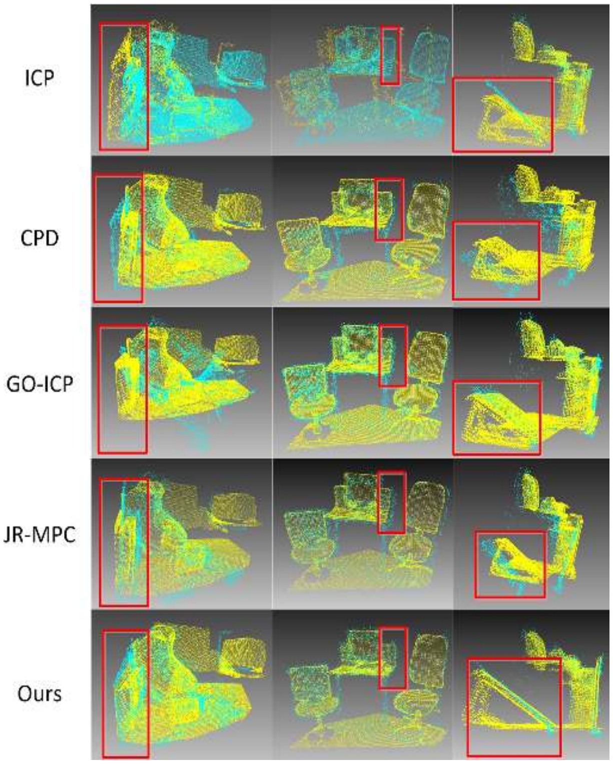

Fig. 10. The final registration results of comparison methods and our method that obtain correct result from the pipeline.

C. KF and VSFM cross-source point cloud datasets

In this section, the experiments are conducted on real cross-source point clouds that are built by KinectFusion (KF) and VSFM respectively. We use KF and video camera to capture 22 indoor scenes. Each scene includes 22 sets of KinectFusion point clouds. For the corresponding 22 VSFM datasets, each data set includes 200 2D images. Hence, 22 sets of VSFM point clouds are constructed. The groundtruth relation between these two point clounds has been annotated. For example, if the first set of KF point cloud corresponds to the same scene with the third set of VSFM point cloud, they are the pair of ground truth point clouds.

TABLE III
THE QUANTITATIVE EVALUATION RESULTS ON KF+VSFM DATASETS

	KF+VSFM	
	accuracy	time(s)
ESF-64 + ICP[8]	0.364	109
ESF-64 + CPD[36]	0.136	434
ESF-64 + Go-ICP[17]	0.455	544
ESF-64 + JR-MPC[38]	0.591	389
The proposed method (2000)	0.864	389
The proposed method (150)	0.818	69

In this experiment, starting from the VSFM point clouds, we conduct coarse matching to select 10 potential point clouds; then, we undertake fine registration and compare this with other methods. We use method in [47] to downsample the point cloud to 10% of the original points and run our GM-CSPC method with [8], ESF+CPD, ESF+GO-ICP, ESF+JR-MPC. The quantitative evaluation results are shown in Table III, indicating that our results achieve the highest accuracy in matching and registration. In terms of efficiency, our method runs faster than other methods.

Figure 10 shows the overall registration results after coarse matching and fine registration by using various existing methods and the proposed methods. It is shown that our method achieves better results. The key parts of the objects to be compared have been highlighted in red box.

D. Synthetic cross-source point cloud datasets

In this section, the experiments are conducted on cross-source point clouds based on synthetic datasets. We evaluate the coarse-to-fine pipeline and compare the performance of proposed fine registration algorithm against the well known registration methods. For coarse matching, we use the ESF method; for the fine registration, we compare our method with registration of [8], CPD [36], JR-MPC [38] and GO-ICP [17].

Firstly, to generate synthetic datasets, the datasets are simulated by three steps according to the cross-source properties discussed in Section I. Step 1: Simulation on different densities and different viewpoints. For different densities, the original point cloud is up-sampled by adding one new point to the gravity center of each small triangle on the original surface. Around 300% points will be added. For different viewpoints, these 3D points are removed if their z coordinates are less than 0. The current view and its point cloud are known as view-1 and S1 respectively. To generate another view as view-2 and its point cloud S2, the coordinate system is rotated 60° relative to the y axis and down-samples by 30% to its original point cloud. Step 2: Construction of missing point. Starting from view 2, we aim to remove 10% of whole point cloud. By defining a plane region with its radius of 5% of the diameter of a ball that contains the whole point cloud, we then randomly delete several plane regions to simulate a defect point cloud. Step 3: Rigid transformation. A random scaling up of 3 to 5 times of the point cloud, a random rotation matrix in the x, y, z axis between 30° and 60°, and a random translation in the z axis between 0 and 50% of the largest point-point distance are added to the view-2. Step 4: Construction of noise and outliers. A white Gaussian noise with predefined signal-to-noise ratio (SNR)[3] SNR = 40dB is added to the point cloud of view-2. The outliers are constructed by down-sampling 30% of point cloud in the view-2 and adding random offsets to the coordinates for these down-sampled points. The simulated noises and outliers are combined to form a final point cloud S2. The simulated S1 and S2 point clouds perceive the cross-source problems. Seven cross-source datasets are synthesized using Stanford 3D objects [4].

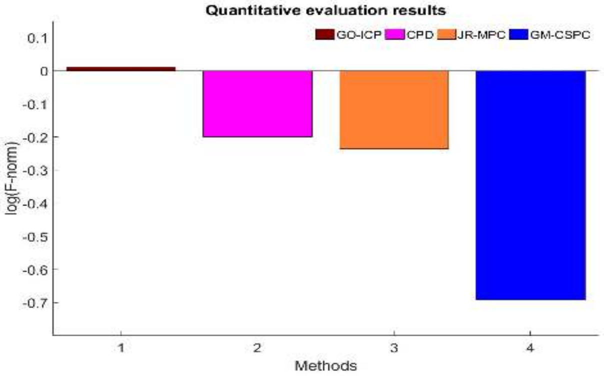

Fig. 11. Quantitative evaluation results of F-norm metric. Our method achieves highest accuracy among these comparison methods.

In our experiments, scale normalization are conducted for JR-MPC and GO-ICP by referring to the method in [8]. In our method, the scale is automatically estimated by the reformulated generative GMM model. To conduct a quantitative evaluation, by following JR-MPC [9], the F-norm value between estimated transformation matrix [5] and groundtruth transformation matrix is used to evaluate the performance of algorithms. The lower the F-norm value, the higher the accuracy of the algorithm.

TABLE IV
THE QUANTITATIVE EVALUATION RESULTS ON SYNTHETIC DATASETS

	Synthetic datasets	
	accuracy	time(s)
ESF-64 + ICP[8]	0.571	615
ESF-64 + CPD[36]	0.571	891
ESF-64 + Go-ICP[17]	0.285	1391
ESF-64 + JR-MPC[38]	0.571	441
The proposed method (2000)	0.857	441
The proposed method (150)	0.857	35.5

To evaluate the coarse-to-fine pipeline, one data P1 in source 1 is used to retrieve P2 in source 2. If the nearest point clouds

[3] Note that SNR is inversely proportional to the variance of added Gaussian noise. In this work we set SNR at a fixed value, thus, the variance of added Gaussian noise is actually adaptive to the variance of input point clouds.

[4] http://graphics.stanford.edu/data/3Dscanrep/

[5] the F-norm of transformation matrix is computed by $e = \| T_i - T_g \|_F$, where T_i is the estimated transformation matrix and T_g is ground truth.

P2 matches the ground-truth correspondent point clouds, the registration is successful. We measure the successful ratio over the ten datasets. The quantitative result are shown in Table IV. The successful ratio of our method is 85.7%, while the ESF+CPD, ESF+GO-ICP, ESF+JRMPC have lower accuracy in terms of their performance.

To evaluate our scale embedded generative GMM algorithm, we compare it to CPD, GO-ICP, JRMPC separately, in terms of its registration accuracy. The quantitatively evaluation results are shown in Figure 11, indicating that our method achieve higher accuracy than the other methods. Figure 12 shows the visual registration results on the synthetic datasets indicating that our method is robust enough for do the registration on cross-source point clouds including Armadillo, Dragon, and horse. However, other existing methods have difficulty in handling cross-source registrations (e.g. Happy).

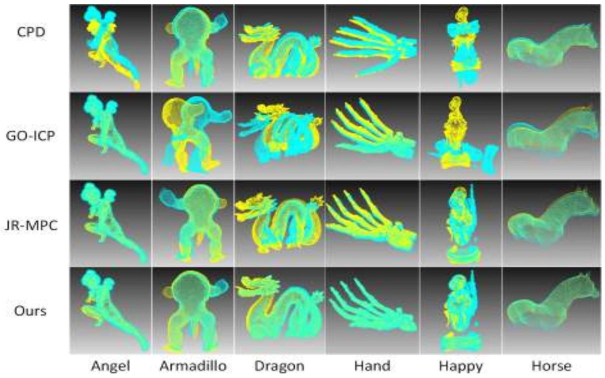

Fig. 12. The visual registration results of our method and comparison methods on Synthetic datasets.

We also test our algorithm on the PISA[6] dataset, which is a real cross-source large building. Figure 13 shows the visual registration results of our method which can successfully handle a large building point cloud (e.g. PISA). The detailed regions in the bottom row show that our method can accurately align with the cross-source point clouds.

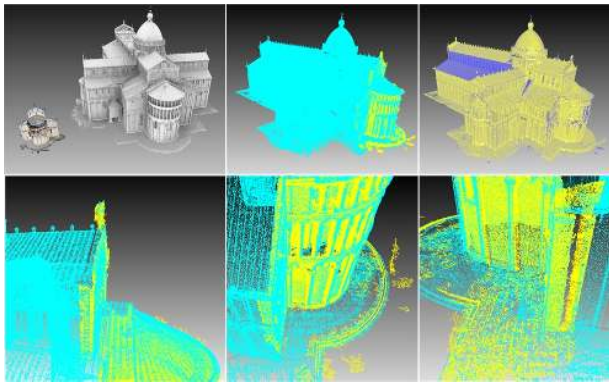

Fig. 13. The two rows show visual results of original point clouds, the registration result in different colors, the registration in different shading techniques. The bottom row shows three sample regions of our registration result in top middle picture.

VI. Conclusion

In this paper, a novel coarse-to-fine algorithm is proposed to address the problem of cross-source point cloud matching and registration. In the first stage, coarse matching is performed to quickly detect a few potential matched regions. In the second stage, a scale estimation embedded GMM-based method is proposed to deal with the cross-source point clouds registration problem. The key points are as follows: 1) the generative GMM framework consider two cross-source point clouds as two samples from this virtual GMM and estimates the parameters and rigid transformation. By smoothly transforming two cross-source point clouds and the maximum posterior probability of GMM, the transformation matrices and the GMM parameters are obtained. This framework obtains better performance in dealing with cross-source noise, outlier, density variation and small missing data; 2) we reformulate the cross-source scale variation into a generative GMM framework and consider the scale in maximizing the posterior probability of GMM. It handles the cross-source scale better than the other methods, such as JR-MPC. Our method not only detects the regions where the reference point cloud is located in the big scene but also obtains an accurate pose related to the big scene. Future work will develop many applications with this method in areas such as location-based services in smart cities and robotics.

Acknowledgment

The authors would like to thank the Nokia Corporation for their help and acknowledge the useful discussions with colleagues in GBDTC. This work is partially supported by Nokia research funding (MM12030846235).

References

[1] C. Wu, “Visualsfm: A visual structure from motion system,” 2011.

[2] W. Wohlkinger and M. Vincze, “Ensemble of shape functions for 3d object classification,” in *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2987–2992.

[3] D. Z. Wang and I. Posner, “Voting for voting in online point cloud object detection,” *Proceedings of the Robotics: Science and Systems, Rome, Italy*, vol. 1317, 2015.

[4] P. J. Besl and N. D. McKay, “Method for registration of 3-d shapes,” in *Robotics-DL tentative*. International Society for Optics and Photonics, 1992, pp. 586–606.

[5] H. Hontani and W. Watanabe, “Point-based non-rigid surface registration with accuracy estimation,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 446–452.

[6] G. C. Sharp, S. W. Lee, and D. K. Wehe, “Icp registration using invariant features,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 1, pp. 90–102, 2002.

[7] X. Huang, J. Zhang, Q. Wu, C. Yuan, and L. Fan, “Dense correspondence using non-local daisy forest,” in *Digital Image Computing: Techniques and Applications (DICTA), 2015 International Conference on*. IEEE, 2015, pp. 1–8.

[8] F. Peng, Q. Wu, L. Fan, J. Zhang, Y. You, J. Lu, and J. Y. Yang, “Street view cross-sourced point cloud matching and registration,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014, pp. 2026–2030.

[9] G. D. Evangelidis, D. Kounades-Bastian, R. Horaud, and E. Z. Psarakis, “A generative model for the joint registration of multiple point sets,” in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 109–122.

[10] B. Jian and B. C. Vemuri, “Robust point set registration using gaussian mixture models,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1633–1645, 2011.

[6] https://www.irit.fr/recherches/VORTEX/MelladoNicolas/category/datasets/

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

[12] X. Chen, K. Kundu, Y. Zhu, A. G. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *Advances in Neural Information Processing Systems*, 2015, pp. 424–432.

[13] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.

[14] D. F. Huber and M. Hebert, "Fully automatic registration of multiple 3d data sets," *Image and Vision Computing*, vol. 21, no. 7, pp. 637–650, 2003.

[15] A. Torsello, E. Rodola, and A. Albarelli, "Multiview registration via graph diffusion of dual quaternions," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2441–2448.

[16] H. Chui and A. Rangarajan, "A new algorithm for non-rigid point matching," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 44–51.

[17] J. Yang, H. Li, D. Campbell, and Y. Jia, "Go-icp: a globally optimal solution to 3d icp point-set registration," in *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 11. IEEE, 2016, pp. 2241–2254.

[18] J. Ma, J. Zhao, J. Tian, Z. Tu, and A. L. Yuille, "Robust estimation of nonrigid transformation for point set registration," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2147–2154.

[19] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[20] C.-S. Chen, Y.-P. Hung, and J.-B. Cheng, "Ransac-based darces: A new approach to fast automatic registration of partially overlapping range images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1229–1234, 1999.

[21] Y. Diez, J. Martí, and J. Salvi, "Hierarchical normal space sampling to speed up point cloud coarse matching," *Pattern Recognition Letters*, vol. 33, no. 16, pp. 2127–2133, 2012.

[22] Z.-Q. Cheng, Y. Chen, R. R. Martin, Y.-K. Lai, and A. Wang, "Supermatching: Feature matching using supersymmetric geometric constraints," *IEEE Transactions on Visualization and Computer graphics*, vol. 19, no. 11, pp. 1885–1894, 2013.

[23] C. Papazov and D. Burschka, "Stochastic global optimization for robust point set registration," *Computer Vision and Image Understanding*, vol. 115, no. 12, pp. 1598–1609, 2011.

[24] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust global registration," in *Symposium on geometry processing*, vol. 2, no. 3, 2005, p. 5.

[25] A. Albarelli, E. Rodola, and A. Torsello, "Loosely distinctive features for robust surface alignment," in *European Conference on Computer Vision*. Springer, 2010, pp. 519–532.

[26] E. Rodolà, A. Albarelli, F. Bergamasco, and A. Torsello, "A scale independent selection process for 3d object recognition in cluttered scenes," *International journal of computer vision*, vol. 102, no. 1-3, pp. 129–145, 2013.

[27] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, p. 85, 2008.

[28] M. Corsini, M. Dellepiane, F. Ganovelli, R. Gherardi, A. Fusiello, and R. Scopigno, "Fully automatic registration of image sets on approximate geometry," *International journal of computer vision*, vol. 102, no. 1-3, pp. 91–111, 2013.

[29] N. Mellado, D. Aiger, and N. J. Mitra, "Super 4pcs fast global pointcloud registration via smart indexing," in *Computer Graphics Forum*, vol. 33, no. 5. Wiley Online Library, 2014, pp. 205–215.

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

[31] A. E. Johnson, "Spin-images: a representation for 3-d surface matching," Ph.D. dissertation, Citeseer, 1997.

[32] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *European conference on computer vision*. Springer, 2010, pp. 356–369.

[33] M. Torki and A. M. Elgammal, "Putting local features on a manifold." in *CVPR*, vol. 2, 2010, p. 4.

[34] Y. Deng, A. Rangarajan, S. Eisenschenk, and B. C. Vemuri, "A riemannian framework for matching point clouds represented by the schrodinger distance transform," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3756–3761.

[35] B. Jian and B. C. Vemuri, "Robust point set registration using gaussian mixture models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1633–1645, 2011.

[36] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.

[37] J. Fan, J. Yang, D. Ai, L. Xia, Y. Zhao, X. Gao, and Y. Wang, "Convex hull indexed gaussian mixture model (ch-gmm) for 3d point set registration," *Pattern Recognition*, vol. 59, pp. 126–141, 2016.

[38] G. D. Evangelidis, D. Kounades-Bastian, R. Horaud, and E. Z. Psarakis, "A generative model for the joint registration of multiple point sets," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 109–122.

[39] C. M. Bishop, "Pattern recognition," *Machine Learning*, 2006.

[40] J. Ma, J. Zhao, and A. L. Yuille, "Non-rigid point set registration by preserving global and local structures," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 53–64, 2016.

[41] R. Horaud, F. Forbes, M. Yguel, G. Dewaele, and J. Zhang, "Rigid and articulated point registration with expectation conditional maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 587–602, 2011.

[42] A. Myronenko and X. Song, "On the closed-form solution of the rotation matrix arising in computer vision problems," *arXiv preprint arXiv:0904.1613*, 2009.

[43] B. Lin, T. Tamaki, F. Zhao, B. Raytchev, K. Kaneda, and K. Ichii, "Scale alignment of 3d point clouds with different scales," *Machine Vision and Applications*, vol. 25, no. 8, pp. 1989–2002, 2014.

[44] N. Mellado, M. Dellepiane, and R. Scopigno, "Relative scale estimation and 3d registration of multi-modal geometry using growing least squares," 2015.

[45] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 21, no. 5, pp. 433–449, 1999.

[46] T. Tamaki, S. Tanigawa, Y. Ueno, B. Raytchev, and K. Kaneda, "Scale matching of 3d point clouds by finding keyscales with spin images," in *Pattern Recognition (ICPR), 2010 20th International Conference on*. IEEE, 2010, pp. 3480–3483.

[47] M. Corsini, P. Cignoni, and R. Scopigno, "Efficient and flexible sampling with blue noise properties of triangular meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 6, pp. 914–924, 2012.