Scale registration based on descriptor analysis and B-spline matching

Wenyu Chen, Wei Xiong, Jia Du, Jierong Cheng
Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore.
e-mail: chenw@i2r.a-star.edu.sg

***Abstract*— By adopting different reconstruction techniques, 3D sensors generate different point clouds of various scales for the same object. This paper presents a new algorithm to align point clouds of different scales. First, shape descriptors based on different neighboring sizes are calculated on each point. Secondly, for each neighboring size, principle components analysis is applied on all descriptors to generate the cumulative contribution rates, which are fitted by B-spline curves. Then, finding scale different between two 3D point clouds in 3D space, is reduced to finding scale difference between two sets of 2D B-spline curves in 2D space. Finally, a numerical solution is presented to find the unknown scale, which is experimentally proved to have higher accuracy than existing methods.**

***Keywords*— point cloud; scale registration; B-spline**

I. Introduction

Many applications are relied on fusion of data from different sensors [1]. However, different sensors based on different reconstruction techniques may sense the same object into point clouds of different scales [2]. Without a baseline, a single camera with structure-from-motion techniques cannot image an object with the desired scale [3]. As such, the scale difference between two point clouds need to be identified before point cloud fusion [4-6].

Point cloud library (PCL) [7] implements many scale invariant shape descriptors for aligning and fusing point clouds of the same scale, including Spin Image [8], Fast Point Feature Histograms (FPFH) [9], Rotational Projection Statistics (RoPs) [10], and Signature of Histograms of Orientations (SHOT) [11]. Each descriptor is based on a neighboring size defining a neighboring region for each point, thus does not encode the scale information of the whole point cloud. They cannot be directly deployed for fusion of point clouds with different scales.

Based on point cloud resolution as the median of distances between points, the scale difference was estimated as the resolution difference between two point clouds [12]. Such method is not stable for point clouds with different point densities. Instead of the resolution, a key scale [4] was derived as a scale descriptor for each point cloud. Accordingly, the scale difference between two point clouds became their key scale difference. To obtain the key scale, the cumulative contributions rates (CCR) was derived from principle components analysis (PCA) on spin images [8] based the same neighboring size. Merging all CCRs with different neighboring sizes together, the point cloud's key scale is a minimum of CCRs. Similar to the solution, key scale may not be captured correctly due to noises. As a result, estimated scale based on such single-value descriptors may not be accurate. Scale ratio method [5] identified the scale difference by comparing the two CCRs instead of the two key scales. The algorithm formed a point-to-point correspondence between the two CCRs, and formulated the scale difference as a value give the best scale matching. For higher accuracy, both key scale method [4] and scale ratio method [5] rely on a dense set of neighboring sizes, which is computationally expensive. To work with a sparse set of neighboring sizes, polynomial curves are fit to CCRs, which turn the scale matching into polynomial curves matching instead of CCRs matching [13]. However, polynomial curves may not have very good fitting to the discrete CCRs. Compared with polynomial curves, B-spline curves [14] can have better fitting to discrete data.

This paper proposes a B-spline scale registration method by further improving the scale ratio method [5] to avoid the discrete CCRs matching, and avoiding the ill-fitting problem using polynomial curves [13]. Analytical B-spline curves are used to replace the CCRs and polynomial curves for a robust matching. By reformulating the best matching problem into finding the best match between two sets of B-spline curves, a numerical solution based on B-spline tessellation is presented and analyzed. The matching performance of the proposed algorithm is illustrated in different experiments.

II. Scale Registration

A. Descriptor Analysis

Similar to the key scale method [4], scale ratio method [5] and polynomial method [13], the proposed B-spline scale registration method also relies on descriptor analysis to derive CCR. In this section, we extend the calculation of CCR [4, 5] to different descriptors.

Generally, given a point cloud P, the shape descriptor on the point $p \in P$ is defined based on a neighboring size e and denoted as p_e, which varies for different neighboring sizes

$$e \in \{e_1, e_2, \cdots, e_m\}, \tag{1}$$

where m is the number of neighboring sizes.

For different points, all descriptors are of the same dimension d. PCA on all descriptors of the same neighboring size, i.e.

$$P_e = \{p_e | p \in P\}, \tag{2}$$

gives d eigenvalues

$$\lambda_{e1} \ge \lambda_{e2} \ge \cdots \ge \lambda_{ed} \ge 0. \tag{3}$$

The CCR for the point cloud P are

$$\eta_{ie} = \frac{\sum_{j=1}^{i} \lambda_{ej}}{\sum_{j=1}^{d} \lambda_{ej}}, \quad \begin{array}{l} e \in \{e_1, e_2, \cdots, e_m\}, \\ i \in I = \{1, 2, \cdots, d\}. \end{array} \tag{4}$$

In this paper, we refer CCR as the discrete points

$$\eta_i = \left\{ \left(r_j, \eta_{i,e_j} \right) \right\}_{j=1}^{m}, i \in I = \{1, 2, \cdots, d\}, \tag{5}$$

and denote the point cloud as

$$\left\{ P, \{e_1, e_2, \cdots, e_m\}, \{\eta_i\}_{i \in I} \right\}. \tag{6}$$

PCL [7] implementations of Spin Image [8], FPFH [9], RoPs [10], and SHOT [11] are used as shape descriptors for scale matching (TABLE I). Fig. 1 plots different CCRs using different descriptors for the Standford bunny [15], where the black dots are the discrete CCRs.

TABLE I. PCL IMPLEMENTATION OF SHAPE DESCRIPTORS

	Spin Image[8]	FPFH[9]	RoPS[10]	SHOT[11]
Descriptor size d	225	33	135	352
CCR in Fig. 1	top left	top right	bottom left	bottom right

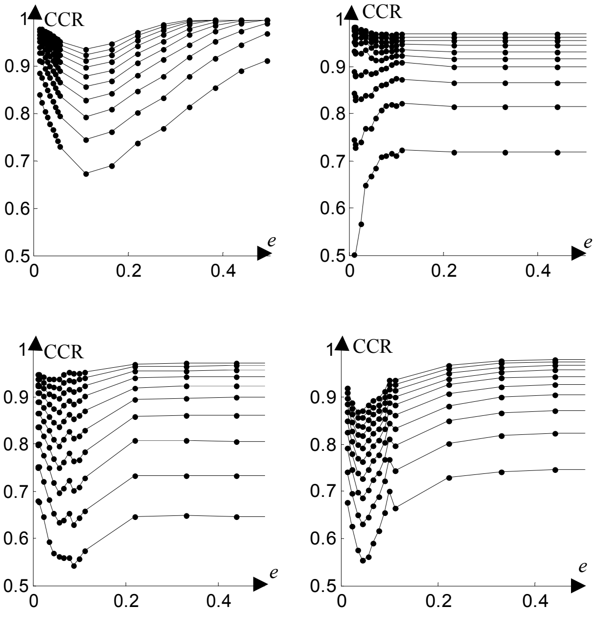

Fig. 1. CCRs for different shape descriptors.

B. B-spline Fitting

Given two different point clouds with corresponding CCRs as $\left\{ P, \{e_1, e_2, \cdots, e_m\}, \{\eta_i\}_{i \in I} \right\}$ and $\left\{ P', \{e'_1, e'_2, \cdots, e'_m\}, \{\eta'_i\}_{i \in I} \right\}$, Keyscale method [4], which identifies the key scale as the minimal CCR, may fail for descriptors providing no minimal in CCR (FPFH [9] produce CCR in top right image in Fig. 1. Scale ratio [5] formed a point-to-point correspondence between $\left(e_j, \eta_{i,e_j}\right)$ and $\left(e_j, \eta'_{i,e'_j}\right)$, and derived the unknown scale as

$$\underset{\alpha}{\operatorname{argmin}} \sum_i \sum_j \left\| \begin{pmatrix} \eta_{i,e_j} \\ \alpha \cdot e_j \end{pmatrix} - \begin{pmatrix} \eta'_{i,e'_j} \\ e'_j \end{pmatrix} \right\|^2. \tag{7}$$

Without a dense set of neighboring sizes, the incorrect point-to-point correspondence will highly reduce the accuracy (see example in Section III.B). Polynomial method [13] can improve the accuracy via polynomial fitting, which, however, may not fit well to the discrete points (see comparison in Section III.E).

Given a tolerance, B-spline can fit the discrete points with a given tolerance, thus provide higher accuracy [14]. An algorithm based on matching B-spline curves instead of matching the discrete points and polynomial curves will be developed which can

- avoid searching minimum from discrete points [4],
- avoid incorrect correspondence [5],
- avoid ill-fitting [13], and
- provide higher accuracy.

For the point cloud in Eq.(6), each CCR η_i can be least square fitted [14] by a cubic B-spline curve with n segments as

$$f_i(t) = \sum_{j=0}^{n+2} A_{ij} N_{ij}(t), t \in [0,1], \tag{8}$$

where A_{ij} are control points, $N_{ij}(t)$ are cubic B-spline base functions defined over a knot sequence T_i, and n is the number of segments during the fitting. Generally, increasing n makes the B-spline curve closer to the CCR.

The most important step for the B-spline fitting is to decide an initial parameterization of the sampling points, i.e. finding parameter t_j for each CCR point as

$$f_i(t_j) = \left(e_j, \eta_{i,e_j}\right). \tag{9}$$

These parameters can defined as

$$t_j = \frac{\sum_{k=2}^{j} \left\| \left(e_k, \eta_{i,e_k}\right) - \left(e_{k-1}, \eta_{i,e_{k-1}}\right) \right\|^{\beta}}{\sum_{k=2}^{m} \left\| \left(e_k, \eta_{i,e_k}\right) - \left(e_{k-1}, \eta_{i,e_{k-1}}\right) \right\|^{\beta}} \in [0,1]. \tag{10}$$

Fig. 2 presents different the fitted B-spline curves for different β values. In this paper, $\beta = 1$ is assumed to have a better fitting result.

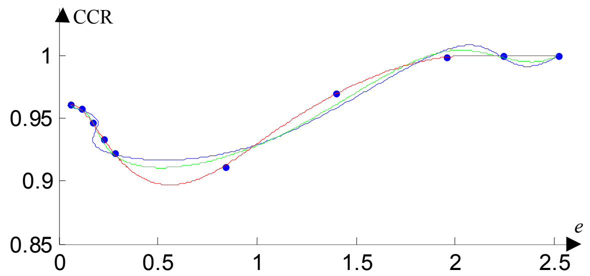

Fig. 2. B-spline fitting of CCR with different parameters: blue dots for CCR; red curve for $\beta = 1$; green curve for $\beta = 0.5$; blue curve for $\beta = 0$.

C. B-spline Tessellation

As will discussed in the following sections, the proposed algorithm tessellates a B-spline curve to a polygon during the calculation for the unknown scale. This section will prove that such tessellation only need to be done once on curves of different scales.

The distance from a point p to a B-spline curve $f_i(t)$ is

$$D(p, f_i(t)) = \min_t \|p - f_i(t)\|. \tag{11}$$

Given a tolerance ε, the B-spline curve $f_i(t) = \left(f_{ix}(t), f_{iy}(t)\right)$ can be tessellated into a polygon

$$F_i = \left\{f_{ij} = \left(f_{ijx}, f_{ijy}\right)\right\}_{j \in J_i} \tag{12}$$

such that the distance between the polygon and the curve is not bigger than ε, i.e

$$D\left(f_i(t), F_i\right) = \min_{j \in J_i} D\left(f_i(t), f_{ij} f_{i,j+1}\right) \le \varepsilon, \quad \forall t \in [0,1]. \tag{13}$$

Where $D\left(f_i(t), f_{ij} f_{i,j+1}\right)$ is the distance from the point $f_i(t)$ to the edge $f_{ij} f_{i,j+1}$.

Scaling α on the first coordinate of $f_i(t)$ gives

$$f_i(\alpha, t) = \left(\alpha f_{ix}(t), f_{iy}(t)\right) \tag{14}$$

And the scaled polygon becomes

$$F_i(\alpha) = \left\{\left(\alpha f_{ijx}, f_{ijy}\right)\right\}_{j \in J_i} \tag{15}$$

According to the scale invariance property of B-spline, for $\alpha \le 1$, we have

$$D\left(f_i(\alpha, t), F_i(\alpha)\right) \le \varepsilon, \quad \forall t \in [0,1]. \tag{16}$$

As such, the tessellation needs to be done once for the original B-spline curve $f_i(t)$. Eq.(15) is a tessellated polygon of Eq.(14) with the tolerance ε.

D. B-spline Based Scale Matching

For $i \in I$, denote B-spline curves for two different point cloud as

$$\begin{aligned} f_i(t) &= \left(f_{ix}(t), f_{iy}(t)\right), t \in [0,1], \\ g_i(s) &= \left(g_{ix}(s), g_{iy}(s)\right), s \in [0,1]. \end{aligned} \tag{17}$$

For each α, the matching between $f_i(\alpha, t)$ in Eq.(14) and $g_i(s)$ is evaluated as

$$\int_0^1 D\left(f_i(\alpha, t) - g_i(s)\right) dt. \tag{18}$$

Scale matching is to find the unknown scale α such that $\left\{f_i(\alpha, t)\right\}_{i \in I}$ best matches $\left\{g_i(s)\right\}_{i \in I}$, i.e.

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \sum_{i \in I} \int_0^1 \min_s \left\| f_i(\alpha, t) - g_i(s) \right\| dt. \tag{19}$$

Since Eq.(19) is a nonlinear minimization problem with non-differentiable objective function, we proposed a numerical method to find a solution. Substituting $f_i(\alpha, t)$ with its tessellated polygon $F_i(\alpha)$ in Eq.(15) into Eq.(19) gives

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \sum_{i \in I} \sum_{j \in J_i} D\left(F_i(\alpha), g_i(s)\right). \tag{20}$$

Project $F_i(\alpha)$ to the curve $g_i(s)$ to derived the projected points

$$G_i(\alpha) = \left\{\left(g_{ijx}, g_{ijy}\right)\right\}_{j \in J_i}. \tag{21}$$

Eq.(20) becomes

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \sum_{i \in I} \sum_{j \in J_i} \left\| F_i(\alpha) - G_i(\alpha) \right\|. \tag{22}$$

The solution for the above equation can be derived iteratively as

$$\alpha_{k+1} = \underset{\alpha_k}{\operatorname{argmin}} \sum_{i \in I} \sum_{j \in J_i} \left\| F_i(\alpha_k) - G_i(\alpha_k) \right\|. \tag{23}$$

Coupling with Eq.(15) and Eq.(21), we have

$$H = \sum_{i \in I} \sum_{j \in J_i} \left\| F_i(\alpha_k) - G_i(\alpha_k) \right\| = \sum_{i \in I} \sum_{j \in J_i} \left\| \begin{pmatrix} \alpha_k f_{jx} \\ f_{jy} \end{pmatrix} - \begin{pmatrix} g_{jx} \\ g_{jy} \end{pmatrix} \right\|$$

$$= \sum_{i \in I} \sum_{j \in J_i} \left(\left(\alpha_k f_{ix} - g_{ix}\right)^2 + \left(f_{iy} - g_{iy}\right)^2 \right)$$

Coupling

$$\frac{\partial H}{\partial \alpha_k} = 2 \sum_{i \in I} \sum_{j \in J_i} f_{ix} \left(\alpha_k f_{ix} - g_{ix}\right) = 2 \sum_{i \in I} \sum_{j \in J_i} \left(\alpha_k f_{ix}^2 - f_{ix} g_{ix}\right)$$

with $\frac{\partial H}{\partial \alpha_k} = 0$ leads to

$$\alpha_{k+1} = \frac{\sum_{i \in I} \sum_{j \in J_i} f_{ijx} g_{ijx}}{\sum_{i \in I} \sum_{j \in J_i} f_{ijx}^{\ 2}}. \tag{24}$$

E. Scale Registration Algorithm

The proposed scale registration algorithm is summarized as follow:

Input: two different point clouds P_1 and P_2

Output: the scale α

Step 1: Initialization

Choose a descriptor

```
    Specify the number of neighboring size $m$
    Initialize the size $d$ and set $I = \{1, 2, \cdots, d\}$
Step 2: B-spline fitting
    For $P \in \{P_1, P_2\}$
        Estimate the resolution $\mu$ of $P$
        Initialize $\{e_1, e_2, \cdots, e_m\}$ according to $\mu$
        For $e \in \{e_1, e_2, \cdots, e_m\}$
            Derive the descriptors $P_e$ in Eq.(2)
            Perform PCA on $P_e$ to get $\{\lambda_{ei}\}_{i=1}^{d}$ in Eq.(3)
            Calculate CCR in $\{\eta_{ie}\}_{i=1}^{d}$ Eq.(4)
        For $i \in I$
            Form discrete points $\eta_i$ in Eq.(5)
            Fit a B-spline $f_i(t)$ to $\eta_i$
Step 3: Scale registration
    Step 3.1 Initialize $\alpha_0, k = 0$
        Initialize $\{f_i(t), g_i(t)\}_{i \in I}$ in Eq.(17)
        Tessellate $\{f_i(t), g_i(t)\}_{i \in I}$ into $F_i$ in Eq.(12) and $G_i$
        Derive $\alpha_0$ using scale ratio [5] based on $F_i$ and $G_i$
    Step 3.2 Calculate $\alpha_{k+1}$ and update $k \leftarrow k+1$
        Derive $F_i(\alpha_k)$ in Eq.(15)
        Derive $G_i(\alpha_k)$ in Eq.(21)
        Update $\alpha_{k+1}$ in Eq.(24)
    Step 3.3 Repeat Step 3.2 until $\alpha_k$ converge. Set $\alpha = \alpha_k$.
```

III. Experiments

The proposed B-spline scale registration algorithm can achieve higher accuracy than existing methods [5, 12, 13]. The first point cloud is taken as the Standford bunny [15] with 69451 points, and the second point cloud is a scaling of the first one by the factor of 0.2. The accuracy is

$$\delta = \delta(m,n) = \frac{|\alpha - 0.2|}{0.2}.$$

The accuracy δ is a function of the B-spline segments n and the number of neighboring size m. First experiment will be conducted to show how accuracy various according to n and m. Second experiment applies different levels of noises onto both point clouds to analyze the accuracy. The third experiment will study how different descriptors and CCRs affect the accuracy.

A. Selection of B-spline Segments n

Spin Images are selected as shape descriptors. CCRs are fitted by different B-spline curves with different number of segments. Different α values are derived for $m \in \{11, 12, \cdots, 17\}$ and $n \in \{2, 3, \cdots, 7\}$. For each *m,* the accuracy adopting different *n* values is plotted as one curve in Fig. 3. Meanwhile, for each n, the accuracy variation

$$\delta' = \delta'(n) = \max_m \delta(m,n) - \min_m \delta(m,n),$$

is plotted in Fig. 4. Combining the data from Fig. 3 and Fig. 4, all *n* values can provide a high accuracy $(\delta \geq 98.5\%)$ with a low variation $(\delta' < 0.013)$. In the following we assume *n*=2.

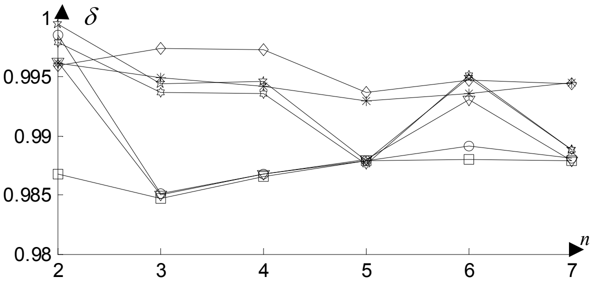

Fig. 3. Accuracy δ over *n*: different curves for different *m*.

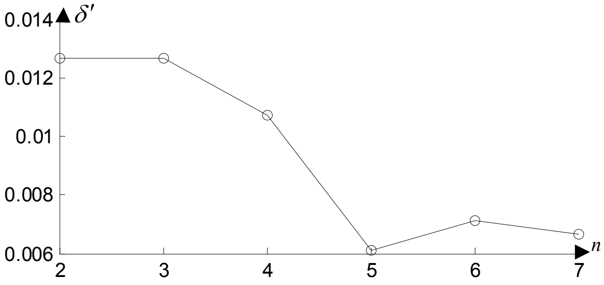

Fig. 4. Accuracy variatoin δ' over *n*.

B. Performance over different m

For $m \in \{5, 6, \cdots, 17\}$, the accuracies δ for the proposed method and scale ratio method [5] are collected separated and presented in different curves in Fig. 5: the red curve indicates the accuracy of the proposed method; the blue curve presents the accuracy of scale ratio method [5]. Based on the same CCRs, the proposed method can achieve higher accuracy and outperform the scale ratio method [5].

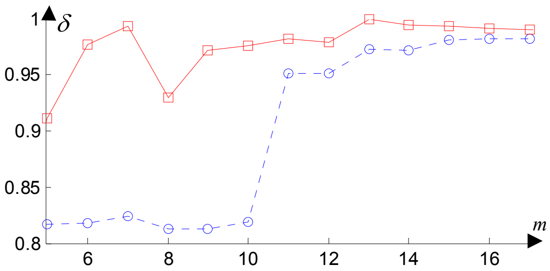

Fig. 5. Accuracy for scale ratio method [5] (blue) and the proposed method (red).

For selected *m* values, Fig. 6 and Fig. 7 show the different results from both methods. Compared with the proposed method, the scale ratio method [5] can better match CCRs, i.e. matching the red circles to the blue circles in Fig. 6 and Fig. 7. However, as illustrated in the left bottom image in Fig. 6, the two B-spline curves may not match very well. The proposed method provides a better matching between B-spline curves (the right bottom image in Fig. 6) and achieve a higher accuracy (Fig. 5).

Since the shape descriptor on each point need to be calculated for m different neighboring sizes, smaller m is desired for a faster computation (the computation time for $m = 10$ is only 56% of that for $m = 18$). However, for a smaller m, the error in point-to-point correspondence in [5] may highly reduce the accuracy (Fig. 7). On the other hand, the proposed method still can achieve a higher accuracy. In the following, $m = 10$ is assumed.

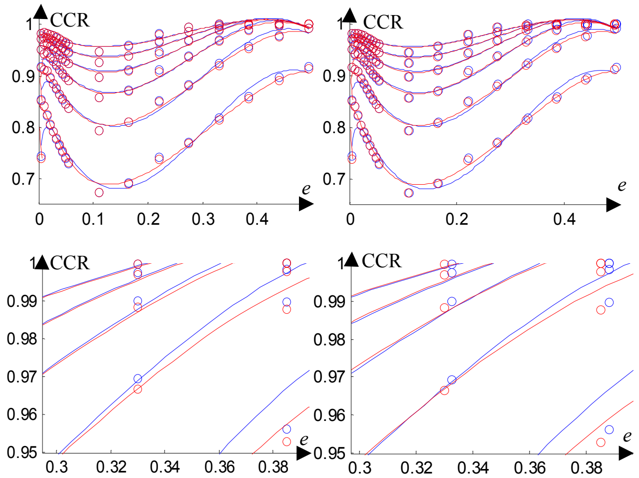

Fig. 6. Point based matching VS B-spline based matching: circles for CCR; left column for scale ratio [5]; right column for our results; second row for zoom view of the first row.

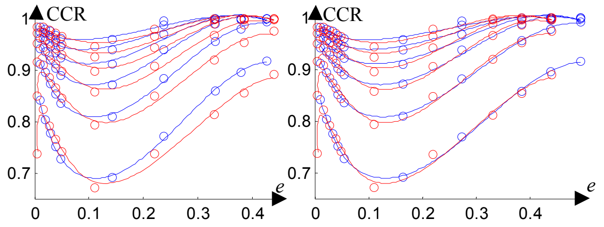

Fig. 7. Point based matching VS curve based matching: circles for CCR; left for scale ratio [5]; right for our results.

C. *Performance over noises*

Define noise at level L as Gaussian noise with mean zero and standard deviation $\mu \cdot L$, where μ is the point cloud resolution. The same level of noise are added to the both point clouds, and the accuracy for different methods under different levels of noises is presented in Fig. 8. The proposed method does not significantly affected by Gaussian noise.

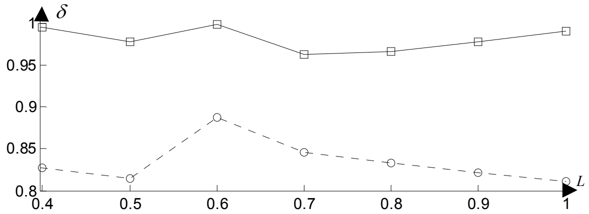

Fig. 8. Accuracy in different noises: curve with circles for [5]; curve with squares for ours.

D. *Performance of descriptors*

Adopting different descriptors in TABLE I will generate different CCRs. Accordingly, the accuracy varies as listed in TABLE II. Our method can provide a higher accuracy.

TABLE II. SCALE ACCURACY USING DIFFERENT DESCRIPTORS

Descriptors	***Scale ratio*[5]**	***Our method***
Spine Image [8]	81.93%	97.72%
FPFH [9]	80.14%	83.90%
RoPS [10]	80.19%	90.89%
SHOT [11]	79.25%	93.90%

E. *Compared with polynomial method*

Based on a sparse set of radii, the accuracy of the scale ratio method [5] is below 85% (Fig.5 first 4 blue circles). Polynomial based method can improve the registration accuracy to 85%~90% [13]. The proposed B-spline based method can further increase the accuracy to above 90% (Fig.5 first 4 red squares).

IV. Conclusions

A new scale registration algorithm based on B-spline curves is proposed in this paper. For each point cloud, PCA on selected descriptors generates CCR. Cubic B-spline curves are fitted to the CCR and used for scale registration. The unknown scale is formulated as the one giving the best matching between two sets of B-spline curves. A numerical algorithm has been implemented to derive the desired scale difference. In different experiments, the proposed method outperforms existing methods with higher accuracy.

References

[1] J. Zhang and X. Lin, "Advances in fusion of optical imagery and LiDAR point cloud applied to photogrammetry and remote sensing," *International Journal of Image and Data Fusion,* vol. 8, pp. 1-31, 2017.

[2] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE transactions on pattern analysis and machine intelligence,* vol. 32, pp. 1362-1376, 2010.

[3] D. Scaramuzza, F. Fraundorfer, M. Pollefeys, and R. Siegwart, "Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints," in *IEEE 12th International Conference on Computer Vision*, 2009, pp. 1413-1419.

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

[5] B. Lin, T. Tamaki, B. Raytchev, K. Kaneda, and K. Ichii, "Scale ratio ICP for 3D point clouds with different scales," in *20th IEEE International Conference on Image Processing (ICIP)*, 2013, pp. 2217-2221.

[6] N. Mellado, M. Dellepiane, and R. Scopigno, "Relative scale estimation and 3D registration of multi-modal geometry using Growing Least Squares," *IEEE transactions on visualization and computer graphics,* vol. 22, pp. 2160-2173, 2016.

[7] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 1-4.

[8] 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, pp. 433-449, 1999.

[9] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3212-3217.

[10] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *International journal of computer vision,* vol. 105, pp. 63-86, 2013.

[11] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *European Conference on Computer Vision*, 2010, pp. 356-369.

[12] A. E. Johnson and M. Hebert, "Surface matching for object recognition in complex three-dimensional scenes," *Image and Vision Computing,* vol. 16, pp. 635-651, 1998.

[13] W. Chen, W. Xiong, J. Du, and J. Cheng, "Polynomial Curve Registration for Matching Point Clouds of Different Scales," in *IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*, 2017, pp. 122-126.

[14] L. Piegl and W. Tiller, *The NURBS book*: Springer Science & Business Media, 2012.

[15] The Stanford 3D scanning repository [Online]. Available: http://www-graphics.stanford.edu/data/3dscanrep