SCALE RATIO ICP FOR 3D POINT CLOUDS WITH DIFFERENT SCALES

Baowei Lin, Toru Tamaki, Bisser Raytchev, Kazufumi Kaneda, Koji Ichii

Hiroshima University, Japan

ABSTRACT

In this paper we propose a method for matching the scales of 3D point clouds. 3D point sets of the same scene obtained by 3D reconstruction techniques usually differ in scale. To match scales, we estimate the ratio of scales of two given 3D point clouds. By performing PCA of spin images over different scales of two point clouds, two sets of cumulative contribution rate curves are generated. Such sets of curves can be considered to characterize the scale of the given 3D point clouds. To find the scale ratio of two point clouds, we register the two sets of curves by using a variant of ICP that estimates the ratio of scales. Simulations with the Stanford bunny and experimental results with 3D reconstructions of artificial and real scenes demonstrate that the ratio of any 3D point clouds can be effectively used for scale matching.

***Index Terms*—** scale ratio, ICP, 3D point cloud, spin images, registration

1. INTRODUCTION

In this paper, we propose a method for estimating the scale ratio of two sets of 3D points, or 3D point clouds, to align the scales of point clouds. This is important because once point clouds are transformed into the same scale, feature descriptors which are not scale invariant such as spin image [10, 9] and NARF [16] can be used for robust 3D registration. To estimate the scale ratio, first, Principal Component Analysis (PCA) is used to generate two sets of cumulative contribution rate curves. Then the *scale ratio* is defined as the scale translation from one set of curves to the other, which can be found by registration of two sets of curves.

The absolute scale of a point cloud is usually unknown when a 3D point cloud is generated by using recently widespread 3D reconstruction techniques [15, 7]. It could be very useful to capture at least the relative scales of point clouds for many applications because point clouds can be aligned in scale and registered with features which need not be scale invariant. However, not much work has been done on scale ratio estimation or scale matching despite of recent researches in 3D scene analysis [2, 8, 12, 6]. Besides most of the works have been proposed for range data or depth maps (that is, a depth value z is given at each (x,y) coordinates) usually obtained by a range finder which provide the absolute scales.

For 3D point clouds, there are some scale invariant features. Some extensions of 2D features have been proposed such as 3D SIFT [14] and nD SIFT [5], but they just describe features of volumes or n-dimensional data, not a sparse set of 3D points. Instead of directly extending those 2D features, some researchers have proposed to combine 2D features with 3D point clouds [18, 3]. Those are effective when 2D images are available, but not applicable to characterize the scale of 3D point clouds themselves.

To align the scales of two arbitrary 3D point clouds with no associated 2D images, one could use mesh-resolution [9, 10], which compute the median of distances between all 3D points. Obviously, it will fail when two point clouds have different density of points. To overcome this problem, a *keyscale* was proposed by Tamaki et al [17]. A keyscale is a representative scale and is defined for a given 3D point cloud as the *minimum* of the cumulative contribution rates of PCA of spin images over different scales (or widths of spin images) at a fixed eigenspace dimension. The 3D point clouds ratio is then estimated by the ratio of their keyscales. The scale ratio is used to align the scale of one point cloud to the other, which enables us to use any feature descriptors which are not scale invariant such as spin images.

This method however has the following two problems. First, the minimum is not stable and may change against small amount of noise. Second, the minimum is found at discrete steps which is neither accurate nor efficient. There is a tradeoff between these problems because we may find a stable minimum with a larger and sparser discrete step size, but the minimum is less accurate. A more accurate estimate may be obtained with a dense step size, while it is expensive and less stable.

Instead of determining a keyscale for each point cloud, in this paper we propose a method to find the scale ratio directly, which is more efficient and robust than obtaining the ratio of independently estimated keyscales. Our method uses mesh-resolution [9, 10] for initial search range of scale. Our method also uses PCA of spin images over different scales of two point clouds to generate two sets of cumulative contribution rates *curves* for *all* dimensions of eigenspace. Assuming that the two sets of curves differ only in scale, we propose scale ratio ICP, a variant of ICP (Iterative Closest Point [19]), to estimate the scale ratio.

After the concept of spin images has been reviewed in section 2, we will describe the details of the proposed method in section 3 and then show experimental results in section 4.

2. FEATURES DESCRIPTORS

To calculate the scale ratio, any feature descriptors which are not scale invariant such as spin images [9, 10] and NARF [16] can be used. In this paper, we use spin images to find the scale ratio and register point clouds, however, NARF or Shape context [4], etc. can also be used.

2.1. Spin images

A spin image [9, 10] is a local feature at a 3D point with an associated normal vector. It describes local geometry by a two-dimensional histogram of distances to neighbor points in cylindrical coordinates. Let $\boldsymbol{p}_i$ be a point in a 3D point cloud P and $\boldsymbol{n}_i$ its associated normal vector. A spin map is defined for any other points $\boldsymbol{p}_j \in P$ by distances α, β along the normal vector and tangent plane at $\boldsymbol{p}_i$:

$$\alpha_{ij} = \sqrt{\| \boldsymbol{p}_i - \boldsymbol{p}_j \|^2 - (\boldsymbol{n}_i^T(\boldsymbol{p}_i - \boldsymbol{p}_j))^2}, \qquad (1)$$

$$\beta_{ij} = \boldsymbol{n}_i^T(\boldsymbol{p}_i - \boldsymbol{p}_j). \qquad (2)$$

Then, only points close to $\boldsymbol{p}_i$ are used to make a spin image. First, points with $\alpha_{ij} \leq w$ and $|\beta_{ij}| \leq w$ are selected, where $w > 0$ is a pre-defined threshold called *image width*. Next, distances $(\alpha_{ij}, \beta_{ij})$ are discretized into an $m \times m$ grid and then voted to a spin image, a two dimensional distance histogram of $m \times m$ bins, denoted by $Spin_i(\alpha, \beta, w)$. The spin images of corresponding points in different point clouds are usually very similar and can be used for matching.

Because spin images are not scale invariant a certain range of local area (neighborhood) should be specified. Hence, to find the appropriate image width, or scale, w becomes very important.

3. FINDING THE RATIO

We describe our method to estimate two 3D point clouds ratio in this section.

3.1. PCA and Contribution Rate

First, we describe briefly the cumulative contribution rate which is used for finding the scale ratio. We denote spin images $Spin_i(\alpha, \beta, w)$ of $m \times m$ bins as vectors $\boldsymbol{s}_i^w$ of m^2 dimensions. By performing PCA for a set of spin images $\{\boldsymbol{s}_i^w\}$, we have m^2 eigenvectors $e_1^w, \ldots, e_{m^2}^w$ of m^2 dimensions and corresponding real eigenvalues $\lambda_1^w \geq \cdots \geq \lambda_m^w \geq 0$.

The cumulative contribution rate at dimension $d = 1, 2, \ldots, m^2$ is defined as $c_d^w = \frac{\sum_{i=1}^{d} \lambda_i^w}{\sum_{i=1}^{m^2} \lambda_i^w}$.

3.2. Finding scale ratio of two point clouds

The idea of keyscale [17] was to determine the appropriate size of neighborhood. If w is too small, all spin images would represent just a local surface plane. If w is too large, spin images do not represent the scene geometry correctly. In both extreme cases, spin images become similar and the cumulative contribution rates of PCA of the spin images approach to 1. In other words, there is a minimum value of cumulative contribution rate make the spin images most different from each other. Therefore, they used the *minimum* of cumulative contribution rate curve at a specific dimension as a keyscale to characterize the behavior of w over different scales. As shown in Fig. 1(a), a curve representing cumulative contribution rate at a specific dimension over different w clearly has a minimum.

This idea is however not limited to the minimum of a single curve. We extend this idea to use all curves of all dimensions. Two sets of curves are shown in Fig. 1 (a) and Fig. 1 (b). We assume that two sets of curves differ only in scales except non-overlapped parts. This is reasonable as two sets of curves shown in Fig. 1 (a) and Fig. 1 (b) have a similar overlapped part. Those parts can be used to register two sets of curves.

In other words, we estimate a scale difference (i.e., *scale ratio*) between two sets of curves. This idea simplifies registration problem of matching scales of 3D point clouds into a 1D registration problem along the scale (w) axis in Fig. 1. In the following subsection, we propose scale ratio ICP to estimate the scale ratio.

3.2.1. Scale ratio ICP

As shown in Fig. 1(a)(b), suppose there are two sets of curves $c1_d^w = \{x_{w_i}^d, w_i\}$ and $c2_d^w = \{y_{w_i'}^d, w_i'\}$ where $1 \leq d \leq m^2$. The horizontal axis is the width w of spin images, and the vertical axis is cumulative contribution rate.

The objective function we want to minimize is

$$E(t) = \sum_d \sum_i \left\| \begin{pmatrix} y_{w_i'}^d \\ w_i' \end{pmatrix} - \begin{pmatrix} x_{w_i}^d \\ tw_i \end{pmatrix} \right\|^2 \qquad (3)$$

where t is the unknown scale ratio. Currently we assume that points on a curve have corresponding points on the other curve, however, in fact we do not know the correspondence.

Therefore, we use the strategy of ICP to estimate t as follows:

1. initialization

   An exhaustive search is used to find an initial rough estimate of t. First, mesh-resolutions w_{mesh1} and

w_{mesh2} of two sets of point clouds are used to set the minimums $t_{m1} = w_{mesh1}, t_{m2} = w_{mesh2}$ and maximums $1000t_{m1}, 1000t_{m2}$ of spin image width w to find overlapped part. The overlapped parts are $c1_d^w = \{x_{w_i}^d, w_i\}$ where $t_{m1} \leq w_i \leq 1000t_{m1}$ and $c2_d^w = \{y_{w_i'}^d, w_i'\}$ where $t_{m2} \leq w_i' \leq 1000t_{m2}$. Then we find the minimum in the range at discrete steps as the initial estimate t_{init}:

$$t_{init} = \underset{0<t\leq 100\frac{t_{m2}}{t_{m1}}}{\operatorname{argmin}} E(t). \tag{4}$$

2. find putative correspondences

   For each point on one set of a curve $c1_d^w$, find the closest point on the curve $c2_d^w$ with the current estimate t. Note that this process is performed for different d independently.

3. estimate t

   The estimate of t based on the correspondences can be obtained in a closed-form. By taking the derivative of $E(t)$ with respect to t and setting it to 0, we have the following equation (we omit the derivation due to the space limitation):

$$t = \frac{\sum_i w_i w_i'}{\sum_i w_i^2}. \tag{5}$$

4. iteration

   Step 2 and 3 are iterated as t is updated until the estimate converges.

4. EXPERIMENTAL RESULTS

We demonstrate that the proposed method effectively works on any 3D point clouds by simulations and real data experiments.

4.1. Simulations

For simulations demonstrating the concept, we generated two synthetic 3D point clouds from the Stanford bunny [1]. One point cloud with 69,451 points was created from the bunny. Then, the point cloud was scaled by the factor of 5 to create the other point cloud. Two sets of cumulative contribution rate curves are shown in Fig. 1 (a), the original scale, and 1 (b), 5 times larger scale. To plot these curves, we did not use the initialization described in section 3.2.1. Therefore, two sets of curves are difficult to register without appropriately finding the overlapped part.

The first simulation uses these two point clouds. Fig. 1 (c)and Fig.1 (d) shows curves only in the range determined by the initialization step for finding the initial estimate of t.

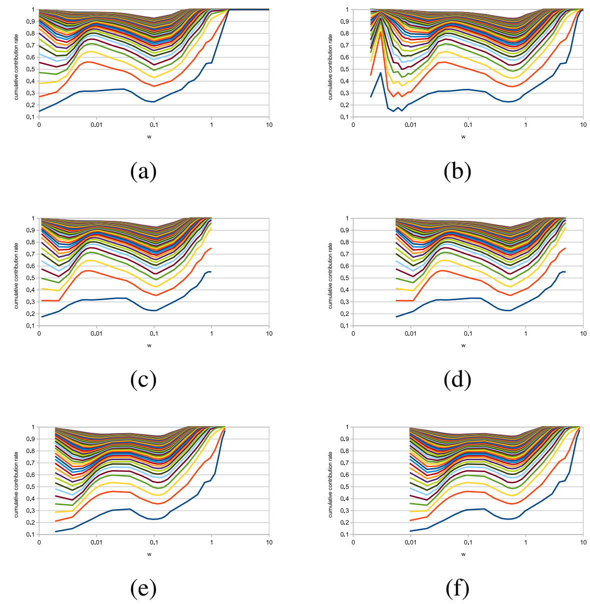

(a) (b) (c) (d) (e) (f)

Fig. 1. c_d^w curves of the bunny simulation dataset in (a) original scale and (b) 5 times larger scale. (c) and (d) Curves above are now shown only in the initial search range found by the initialization step. (e) and (f) Numbers of points in two point clouds are now randomly reduced to approximately 30%.

The ratio found by the proposed scale ratio ICP is shown in the 1st row of Table 1. As the ground truth ratio is 5, our method estimated the ratio accurately.

The second simulation demonstrates the robustness of the proposed method by randomly reducing the number of points to approximately 30% (exactly, 21,664 and 21,473 points). Despite of the random sampling, the two curves are still similar to each other as shown in Fig. 1 (e) and Fig. 1 (f). The estimated result is 5.052 as shown in the 2nd row of Table 1 and the relative error is 1.052% which is still acceptable result.

The third simulation demonstrates robust is the proposed method to different density of points in the point clouds as compared to mesh-resolution. In this case we used the original bunny data and its counterpart randomly sampled and scaled by factor of 5. In other words, we used the two datasets of Fig. 1 (c) and Fig. 1 (f). As shown in the 3rd row of Table 1, the proposed method can estimate the ratio as 4.733 and the relative error is 5.333%. In contrast, mesh-resolution result is 8.727 which is much worse than our result.

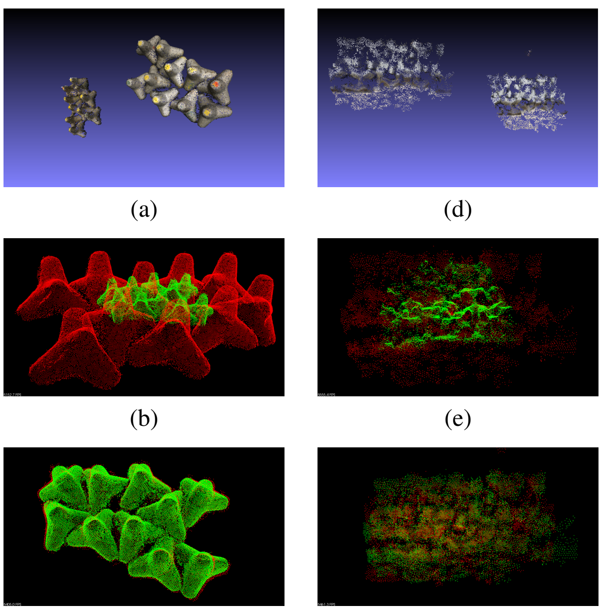
The fourth simulation adds noise to the coordinates of the 3D points for demonstrating the noise tolerance of the proposed method. Uniform noise was added to each x, y, z coordinate of every point in the point clouds. In this case, the

Table 1. Simulation results of different methods.

simulations	ground truth	scale ratio ICP	keyscale [17]	mesh-resolution [9, 10]
noise-free, no random sampling	5	5.000	5.000	5.000
noise-free, both random sampling	5	5.052	5.053	5.053
noise-free, random sampling of one dataset	5	4.733	5.818	8.727
noise (0.1), no random sampling	10	10.000	10.000	10.000
noise (0.5), no random sampling	10	10.086	9.898	12.727
noise (1.0), no random sampling	10	10.842	10.909	16.363

two point clouds are different in scales by a factor of 10. The magnitude of the noise was set to 0.1%, 0.5% and 1.0% of the maximum of the three sides of the bounding box of the bunny. The results in Table 1 (row 4, 5 and 6) show the relative errors are 0.003%, 0.863% and 8.422% which mean that the proposed method is not affected significantly by small levels of noise.

4.2. Small and real blocks of 3D point clouds



(a) (d) (b) (e) (c) (f)

Fig. 2. ICP results for small and real blocks. (a) Point clouds for small blocks. (b) Registration result for the small blocks when using the standard ICP without our method. (c) Registration result for the small blocks when using the standard ICP with our method. (d) Point clouds for real blocks. (e) Registration result for the real blocks when using the standard ICP without our method. (f) Registration result for the real blocks when using the standard ICP with our method.

Now we show experimental results on real datasets. The first dataset consists of the small blocks shown in Fig. 2 (a). Two point clouds with normal vectors of 73,224 and 101,859 points were computed by 3D reconstruction with Bundler [15] followed by Patch-based Multi-view Stereo (PMVS2) [7] from 26 and 29 images respectively. The proposed method provides a ratio of 2.50. Fig. 2 (b) and Fig. 2 (c) shows the alignment results without and with estimated ratio by using the "standard" ICP for 3D point cloud registration. Clearly, scale aligned point clouds are well registered. This registration process is available as a video on YouTube (`http://youtu.be/Sp6ekOV0zMA`).

The second dataset consists of real blocks. Two point clouds of 10,325 and 9,343 points were computed from 24 images per set. Fig. 2 (d) shows the point clouds and (e) and Fig. 2 (f) the alignment results without and with estimated ratio. This registration process is also available as a video on Youtube, which demonstrates that the proposed method works very efficiently even on real datasets.

5. CONCLUSIONS

We have proposed a method for matching scales of 3D point clouds. Experimental results demonstrated that the proposed method works well for easy and difficult point cloud datasets. In future works, we will try to reduce the computational cost. Typical computation time ranges from 10 to 30 minutes for several millions of points with unoptimized codes, which is still not so small due to the repetition of PCA.

6. REFERENCES

[1] The Stanford 3D Scanning Repository. [Online]. Available: http://www.graphics.stanford.edu/data/3Dscanrep/

[2] E. Akagunduz and I. Ulusoy, "Extraction of 3d transform and scale invariant patches from range scans," in *CVPR*, 2007, pp. 1–8.

[3] G. Baatz, K. Koser, D. Chen, R. Grzeszczuk and M. Pollefeys, "Leveraging 3d city models for rotation invariant place-of-interest recognition," in *IJCV*, vol. 96(3), 2012, pp. 315–334.

[4] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," in *PAMI*, vol. 24(4), 2002, pp. 509–522.

[5] W. Cheung and G. Hamarneh, "n-sift: n-dimensional scale invariant feature transform," in *Trans. IP*, vol. 18(9), 2009, pp. 2012–2021.

[6] C. Conde and A. Serrano, "3d facial normalization with spin images and influence of range data calculation over face verification," in *CVPRW*, 2005.

[7] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," in *CVPR*, 2007, pp. 1362–1376.

[8] M. Haker, M. Bohme, T. Martinetz, and E. Barth, "Scale invariant range features for time-of-flight camera applications," in *CVPRW*, 2008.

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

[10] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scene," in *PAMI*, vol. 21, 1999, pp. 433–449.

[11] F. D. la Torre and M. J. Black, "Robust principal component analysis for computer vision," in *ICCV*, 2001, pp. 362–369.

[12] A. S. Mian, M. Bennamoun, and R. Owens, "Keypoint detection and local feature matching for textured 3d face recognition," in *IJCV*, vol. 79(1), 2008, pp. 1–12.

[13] K. Saitwal, A. A. Maciejewski, R. G. Roberts, and B. A. Draper, "Using the low-resolution properites of correlated images to improve the computational efficiency of eigenspace decomposition," in *Trans. IP*, vol. 15(8), 2006, pp. 2376–2387.

[14] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *ACM Multimedia*, 2007, pp. 357–360.

[15] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from internet photo collections," in *IJCV*, vol. 80, 2008, pp. 189–210.

[16] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Narf: 3d range image features for object recognition," in *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at IROS*, 2010.

[17] T. Tamaki, S. Tanigawa, Y. Ueno, B. Raytchev, and K. Kaneda, "Scale matching of 3d point clouds by finding keyscales with spin images," in *ICPR*, 2010, pp. 3480–3483.

[18] C. Wu, B. Clipp, X. Li, J.-M. Frahm, and M. Pollefeys, "3d model matching with viewpoint invariant patches (vips)," in *CVPR*, 2008, pp. 1–8.

[19] P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," in *PAMI*, Vol. 14(2), 1991, pp. 239–256.