

## 3-D FEATURE-BASED MATCHING BY RSTG APPROACH

Jen-Jar Jaw \*, Tzu-Yi Chuang

Department of Civil Engineering, National Taiwan University, 1, Roosevelt Rd., Sec. 4, Taipei 10617, Taiwan  
(jejaw,d95521008)@ntu.edu.tw

Commission III, WG III/2

**KEY WORDS:** LiDAR; Matching; Feature; Automation; Transformation; Laser scanning; Three-dimensional;

### ABSTRACT:

3-D feature matching is the essential kernel in a fully automated feature-based LiDAR point cloud registration. After feasible procedures of feature acquisition, connecting corresponding features in different data frames is imperative to be solved. The objective addressed in this paper is developing an approach coined RSTG to retrieve corresponding counterparts of unsorted multiple 3-D features extracted from sets of LiDAR point clouds. RSTG stands for the four major processes, “Rotation alignment”; “Scale estimation”; “Translation alignment” and “Geometric check,” strategically formulated towards finding out matching solution with high efficiency and leading to accomplishing the 3-D similarity transformation among all sets. The workable types of features to RSTG comprise points, lines, planes and clustered point groups. Each type of features can be employed exclusively or combined with others, if sufficiently supplied, throughout the matching scheme. The paper gives a detailed description of the matching methodology and discusses on the matching effects based on the statistical assessment which revealed that the RSTG approach reached an average matching rate of success up to 93% with around 6.6 % of statistical type 1 error. Notably, statistical type 2 error, the critical indicator of matching reliability, was kept 0% throughout all the experiments.

### 1 INTRODUCTION

Laser scanning, often referred to as LiDAR (Light Detection And Ranging), is nowadays widely employed as an efficient technology for generating 3-D data representing physical surface and object models in application areas as diverse as topographic mapping, pattern recognition, heritage recording and engineering metrology. LiDAR systems generate dense 3-D point clouds, and these individual point clouds are registered to form a comprehensive surface for more information and better inference. Registration constitutes an essential data processing step; it commonly can be carried out with the aid of artificial markers (Akca, 2003; Franaszek et al., 2009), utilizing iterative closest point (ICP) approaches (Besl and McKay, 1992; Chen and Medioni, 1992; Zhang, 1994) or employing geometric features implied within point clouds (Huang and Menq, 2001; Habib et al., 2005; Dold and Brenner, 2006; Rabbani et al., 2007; Jaw and Chuang, 2008, 2010).

In the perspective of feature-based techniques, automatic registration generally confronts three issues: acquisition of registration features which represent the involved data; a similarity measure that mathematically ensures the correspondence of conjugate features, and a transformation function that describes the transformation between the data sets. This paper assumes that geometric features have been acquired, and focuses on solving the issue of retrieving correspondence of 3-D features. 3-D feature matching has attracted research attention in a number of communities. In computer graphics and pattern recognition communities, the shape histograms

(Ankerst et al., 1999), depth buffer descriptor (Heczko et al., 2002) and light field descriptor (Chen et al., 2003) are a few common approaches in the retrieval of 3-D objects domain. Matching is performed by pairing global or local shape descriptors for optimal relative transformation, where the optimal matched mates minimize the differences between corresponding shape features (Chua and Jarvis, 1996; Belongie et al., 2001; Gelfand et al., 2005; Li and Guskov, 2005; Gal and Cohen-Or, 2006). However, retrieval precision is generally poor in particular when overall shapes of objects have obvious discrepancy and the presence of numerous surface self-similarities. Also, some methods implement principal component analysis (PCA) and centers of gravity of two data sets to retrieve the rotation component and translation vector, respectively; it actually only works in very specific conditions and obtains coarse estimation (Kim et al., 2011).

In the fields of photogrammetry and remote sensing, the general strategy of surface matching techniques utilize a least-squares approach that minimizes the distances between corresponding surface elements to solve for the registration problem (Xu and Li, 2000; Gruen and Akca, 2005), and some pre-processing procedures may be needed (e.g., interpolation). Regarding points, lines, planes and clustered point groups are generally employed as the primary primitives in existing feature-based registration methods, and research into integrated multiple feature matching without any aid of initial information (e.g., a prior alignment or approximations) is currently limited. To benefit from the rich sets of features, this paper proposes a

matching approach named RSTG to retrieve corresponding counterparts from unsorted multiple 3-D features, where different types of features can be integrated and employed simultaneously; Moreover, the RSTG can synchronously estimate the transformation parameters relating two data frames while establishing the correspondence. Notably, uncertainty of features be matched is taken into account with a weighting arrangement throughout the process, which is rarely considered in current matching methods. In addition, point-to-point correspondence is needless retrieving the translation parameters as the line-based similarity transformation model (Jaw and Chang, 2008) is employed. The methodology of the RSTG is given as follows.

## 2 RSTG APPROACH

The RSTG approach is strategically formulated towards finding out 3-D matching solution with high efficiency and leading to accomplishing the 3-D similarity transformation with four major processes, namely “Rotation alignment”; “Scale estimation”; “Translation alignment” and “Geometric check.” Starting from the rotation alignment step, RSTG produces a queue of possible associations for a set of features in both query and reference frame data. The decision whether the set of features is matched or not would be made by a judging function taking account into both similarity and relative geometry of features. This operation proceeds iteratively until the desired number of matches is achieved or there is no more matches found. That is to say, only some of qualified matches, called ‘initial matches,’ are required in this phase as long as the matched mates are satisfactory for estimating scale and translation parameters in the subsequent procedures. Instead of exploring whole data, the strategy would significantly reduce the computational time. The remaining correspondence will be retrieved in the final phase.

### 2.1 Rotation alignment

The purpose of rotation alignment is to find out the initial matches, while constructing the rotation matrix between two data frames as illustrated in Fig. 1.

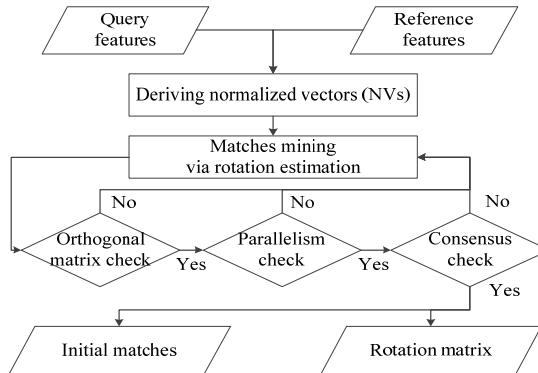


Fig. 1. the workflow of rotation alignment

At the beginning of the procedure, multiple features are first converted into the form of normalized vectors (NVs). This

allows different types of features which may be collected through dissimilar methods with distinct scale can work together. Utilizing the form of NVs makes the RSTG approach a scale invariant algorithm. Additionally, the relevant scale would be reconstructed in the scale estimation step. The NVs of different features can be derived in both query and reference frames by:

$$\begin{cases} v_{(1)i} = \frac{\overrightarrow{\Delta P t_{(1)ij}}}{\|\overrightarrow{\Delta P t_{(1)ij}}\|} \text{ or } \frac{\overrightarrow{Dv_{(1)i}}}{\|\overrightarrow{Dv_{(1)i}}\|} \text{ or } \frac{\overrightarrow{Nv_{(1)i}}}{\|\overrightarrow{Nv_{(1)i}}\|} \text{ or } \frac{\overrightarrow{\lambda v_{(1)i}}}{\|\overrightarrow{\lambda v_{(1)i}}\|}, i \neq j \\ v_{(2)i} = \frac{\overrightarrow{\Delta P t_{(2)ij}}}{\|\overrightarrow{\Delta P t_{(2)ij}}\|} \text{ or } \frac{\overrightarrow{Dv_{(2)i}}}{\|\overrightarrow{Dv_{(2)i}}\|} \text{ or } \frac{\overrightarrow{Nv_{(2)i}}}{\|\overrightarrow{Nv_{(2)i}}\|} \text{ or } \frac{\overrightarrow{\lambda v_{(2)i}}}{\|\overrightarrow{\lambda v_{(2)i}}\|} \end{cases} \quad (1)$$

where  $v_{(1)i}$  and  $v_{(2)i}$  are the  $i^{th}$  normalized vector in the query and reference frames, respectively;  $\overrightarrow{\Delta P t_{(1)ij}}$  is the difference vector derived by the  $i^{th}$  and  $j^{th}$  points in the query;  $\overrightarrow{Dv_{(1)i}}$ ,  $\overrightarrow{Nv_{(1)i}}$  and  $\overrightarrow{\lambda v_{(1)i}}$  individually stand for the  $i^{th}$  direction vector of a line, the  $i^{th}$  normal vector of a plane and the  $i^{th}$  eigenvector vector of a cluster of points in the query frame, whereas  $\overrightarrow{Dv_{(2)i}}$ ,  $\overrightarrow{Nv_{(2)i}}$  and  $\overrightarrow{\lambda v_{(2)i}}$  represent their counterparts in the reference frame, respectively. Therefore, the relative geometric matrix  $R_G$  can be established by Eq. (2), where  $P_{(1)}$  and  $P_{(2)}$  represent the weight matrices of features in query and reference frames, respectively;  $C_{(1)}$  and  $C_{(2)}$  matrices, as shown in Eq. (3), consist of stacking NVs acquired from multiple features with respect to each data frame. The normalized row vectors  $v_{(1)n}^T$  and  $v_{(2)n}^T$  can arbitrarily be acquired by points, lines, planes or clustered points with Eq. (1), and the  $n$  in both  $C_{(1)}$  and  $C_{(2)}$  is defined in advance as long as it embraces at least three independent NVs forming the relative geometric matrix. Consequently, the rotation matrix is derived by using the SVD on the relative geometric matrix:

$$R_G = C_{(2)}^T P_{(2)}^T P_{(1)} C_{(1)} (C_{(1)}^T P_{(1)}^T P_{(1)} C_{(1)})^{-1} = U \Sigma V^T \quad (2)$$

$$C_{(1)} = \begin{bmatrix} v_{(1)1}^T \\ v_{(1)2}^T \\ \vdots \\ v_{(1)n}^T \end{bmatrix}, \quad C_{(2)} = \begin{bmatrix} v_{(2)1}^T \\ v_{(2)2}^T \\ \vdots \\ v_{(2)n}^T \end{bmatrix} \quad (n \geq 3) \quad (3)$$

$$R = U V^T \quad (4)$$

where  $R$  is a rotation matrix;  $U$  and  $V^T$  are two orthogonal matrices, and  $\Sigma$  is a diagonal matrix. However, to estimate a rotation matrix, the correspondences between features in the query and reference frames has to be investigated in advance. The way to find initial matches is described as follows.

**2.1.1 Matches Mining via Rotation Estimation:** Assume that there are  $m_1$  and  $m_2$  NVs in each query and reference frame, and the row number of matrices  $C_{(1)}$  and  $C_{(2)}$  is given as  $n$ . The process starts from generating a short list for all possible connections between the NVs in each frame by using the permutation to pick  $n$  NVs from  $m_1$  and  $m_2$  individually, and then implementing the combination for the set of NVs. During the iterative calculation, the order and the

elements within the  $C_{(1)}$  and  $C_{(2)}$  change based on the short list as forming the relative geometric matrix. And, the acquired rotation matrix would go through three checks (introduced in subsequent sections) to verify whether the relevant corresponding NVs are matched or not. The procedure would keep operating until the assigned number of initial matches is fulfilled, or there is no matches accepted throughout the data. Finally, the rotation matrix and the initial matches are determined.

**2.1.2 Orthogonal Matrix Check:** The first check for the nominative rotation matrices is based on the definition of the orthogonal matrix. Although there are several ways to verify whether a matrix is an orthogonal matrix, this paper examines the rotation matrix by utilizing the ratio of its three singular values. If the singular values have similar ratio with each other within a tolerance, as described in Eq. (5), then the matrix is regarded as an orthogonal matrix passing the first inspection.

$$\lambda_1/\lambda_2 \cong \lambda_2/\lambda_3 \cong \lambda_3/\lambda_1 \quad (5)$$

where  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  are the three singular values. However, this check is not always strict in every situation. For example, this principle would fail when a set of NVs in the query frame conjugates with its mirror projection in the reference frame. Therefore, the rotation matrices are further inspected by the parallelism and the consensus checks.

**2.1.3 Parallelism Check:** The concept of parallelism check is that the NVs in a set should be parallel with each counterpart after removing the variant of rotation. If the angles between corresponding NVs are all close to zero, the rotation matrix is accepted. Additionally, the subsequent consensus check would further ensure that all the approved rotation matrices have the same trend.

**2.1.4 Consensus Check:** Although all approved rotation matrices should be nearly identical with each other as long as every matched sets of NVs are correct, the uncertainty of correlative features may cause a little discrepancy. This check statistically observes the values where the rotation matrices are clustered, and applies all the sets of NVs whose corresponding rotation matrices are within the cluster to recompute a global rotation matrix. Finally, the features corresponding to each matched normalized vector are regarded as the initial matches.

## 2.2 Scale estimation

The scale factor between two data sets can be evaluated by the ratio of two identical distances measured in different frames, and thus the corresponding features are prerequisite to connect identical distances. The initial matches are used to solve the scale factor avoiding manual identification for the correspondents via Eq. (6):

$$S = (\sum_{i=1}^n P_i)^{-1} \sum_{i=1}^n P_i \frac{|dist_{(2)i}|}{|dist_{(1)i}|} \quad (6)$$

where  $S$  is the scale factor, and  $P_i$  stands for the weight of the  $i^{th}$  distance ratio;  $dist_{(1)i}$  and  $dist_{(2)i}$  denote the  $i^{th}$  spatial distance in the query and reference frames, respectively. The measure of distances can be acquired by using the perpendicular distance between two lines or intersected lines of two adjacent planes.

## 2.3 Translation alignment

Provided that the scale factor and rotation matrix between two data frames are known, translation parameters can be calculated by the point-based 3-D similarity transformation. However, reliable corresponding points have to be known in advance due to the point-to-point restriction of the transformation. For a more flexible processing, this paper employs the line-based similarity transformation model (Jaw and Chuang, 2008), which lies on trajectory-based restriction to get a closed-form solution of transformation parameters. The collinear property can be described by:

$$\begin{bmatrix} -u_{(2)1} & t_{(2)1} & 0 \\ 0 & -v_{(2)1} & u_{(2)1} \\ \vdots & \vdots & \vdots \\ -u_{(2)n} & t_{(2)n} & 0 \\ 0 & -v_{(2)n} & u_{(2)n} \end{bmatrix}_{2n \times 3} T - \begin{bmatrix} -u_{(2)1}X_{(2)1} + u_{(2)1}F_{X(1)1} + t_{(2)1}Y_{(2)1} - t_{(2)1}F_{Y(1)1} \\ -v_{(2)1}Y_{(2)1} + v_{(2)1}F_{Y(1)1} + u_{(2)1}Y_{(2)1} - u_{(2)1}F_{Z(1)1} \\ \vdots \\ -u_{(2)i}X_{(2)i} + u_{(2)i}F_{X(1)i} + t_{(2)i}Y_{(2)i} - t_{(2)i}F_{Y(1)i} \\ -v_{(2)i}Y_{(2)i} + v_{(2)i}F_{Y(1)i} + u_{(2)i}Y_{(2)i} - u_{(2)i}F_{Z(1)i} \end{bmatrix}_{2n \times 1} = 0 \quad (7)$$

where  $T = [T_X \ T_Y \ T_Z]^T$  is the translation vector;  $(X_{(1)i}, Y_{(1)i}, Z_{(1)i})$  and  $(X_{(2)i}, Y_{(2)i}, Z_{(2)i})$  represent the  $i^{th}$  point in the query and reference frames, which can be assigned with an end-point of line or a foot-point of normal vector, etc.;  $[t_{(2)i} \ u_{(2)i} \ v_{(2)i}]^T$  is the normalized vector of  $i^{th}$  features in the reference frame;  $[F_{X(1)i} \ F_{Y(1)i} \ F_{Z(1)i}]^T = SR[X_{(1)i} \ Y_{(1)i} \ Z_{(1)i}]^T$ ,  $i = 1 \sim n$ ,  $n \geq 2$ ;  $n$  is the number of transformed points.

## 2.4 Geometric check

After the first three steps of the RSTG approach, initial matches and the similarity transformation are both acquired, and thus the variants of scale, rotation and translation are eliminated by transforming each kind of features into the same data frame with the multi-feature transformation (Jaw and Chuang, 2010). Therefore, the correspondence of remaining features is investigated step by step with three geometric constraints: 1) the minimum distance constraint; 2) the minimum included angle, and 3) the relative connections with the initial matches. The first constraint is utilized to search the nearest correspondent which is the most possible matched entity, whereas the second constraint which is specific for lines and planes is employed to inspect the consistency of orientation between matches. Furthermore, matches are verified by comparing the relative connections with the initial matches. The relative connections mentioned here comprise the distances and angles between a feature and initial matches, if available. Since the correspondence of initial matches has been known,

the relative connections of conjugate features in the query and reference data would be comparatively similar with each other. Finally, these three constraints make certain erroneous matches disappear from the final matching result, and features in the query data are all explored for correspondents. Notably, the thresholds involved are all determined by the error propagation based on the uncertainty of features. In addition, a feature may have multiple corresponding mates as long as they all pass through the constraints.

### 3 EXPERIMENTAL ASSESSMENTS

For a practical matching algorithm, the effectiveness and efficiency are the most significant issues that should be verified. Therefore, the RSTG approach was inspected with both a statistical assessment and a real case to prove the feasibility and the effectiveness. The explicit setting of each examination is introduced as follows.

#### 3.1 Statistical Assessment

All experiments in this assessment were performed by multiple 3-D features. We investigated the performance and efficiency which related to the distribution and uncertainty of features between two data frames. The experiments were carried out repeatedly for 100 times to derive a statistic result. There comprised 40 points, 30 lines and 30 planes, 100 features totally, generated randomly for every calculation. That is, the distribution of features changes every time. Besides, random errors, ranging from 0.01m to 0.1m, were also added to simulate the uncertainty of features. As shown in Table 1, the type 1 error is defined as the wrong decision that rejects a true match, whereas the type 2 error indicates the error that fails to reject a false match. The statistical results reveal that the RSTG reached an average successful matching rate of 98.83% while the feature were added with  $\pm 0.01$ m random errors; an average successful matching rate of 85.90% while the features were added with  $\pm 0.1$ m random errors. It is worth to point out that the percentage of statistical type 2 error, which has critical influences on the matching reliability, was kept 0% throughout all tests. The operations took an average time of 67.87 seconds to accomplish the matching based on Intel i5 CPU 2.53GHz in Matlab. Yet, the efficiency would highly depend on programming skills and language, facilities, and so on. It still has space for further optimization. The statistical illustrations are shown in Fig. 2, where the blue square line indicates the  $i^{th}$  successful matching rate; the red dotted line is the average successful matching rate; the green star line represents the type 1 error, and the cyan star line denotes the type 2 error.

Table 1. the matching performance.

Std. of features (m)	$\pm 0.01$	$\pm 0.03$	$\pm 0.05$	$\pm 0.1$
Avg. matching rate (%)	98.83	96.10	92.5	85.90
Avg. type 1 error (%)	1.16	3.90	7.50	14.10
Avg. type 2 error (%)	0	0	0	0
Avg. operation time (sec)	72.2	70.4	67.5	61.4

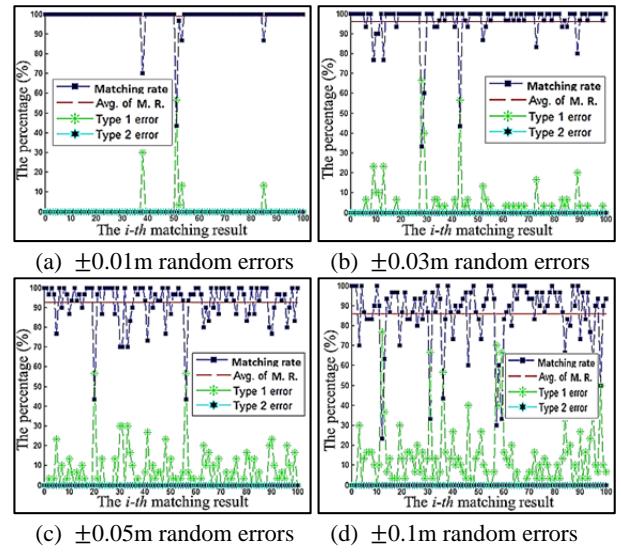


Fig. 2. the statistic results.

#### 3.2 The Real Case

Due to visibility constraints from a single terrestrial laser scanner, scenes are normally performed by combining multiple overlapping scans into a single data set. It depends on the correspondents in different scan to establish the transformation. Assuming that the features in each scan had been acquired already, the RSTG approach assisted in finding out the corresponding features in an automatic fashion. There were five successive scans, marked "S-1", "S-2", "S-3", "S-4" and "S-5", collected by Trimble (Mensi) GS200 describing the front door of National Taiwan University in Taiwan. The nominal positional accuracy of 4 shots as reported by Trimble (2005) was up to 2.5mm at 25m range. The extracted features totally contained 21 points, 46 lines and 22 planes as illustrated in Fig. 8 with white colour, and the relevant uncertainty was derived based on error propagation from the origin accuracy of point clouds.

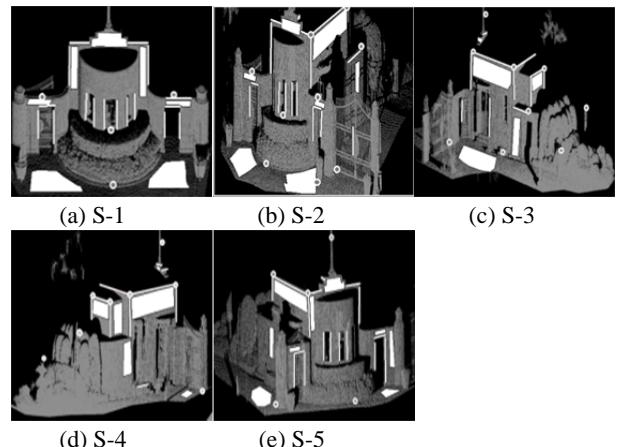


Fig. 3. the point clouds and extracted features.

Although the LiDAR point clouds were scanned sequentially, we assumed that we didn't have the information so feature matching between scans were performed randomly. Once a data set has matched, it would not be involved again in

subsequent matching processes. In this case, we chose the scan 1 to start the matching process. As Table 2 shows, the successful matching rate maintains above 92.3% and the total operation time takes 1.85 seconds to fulfil the whole matching procedures. In addition, Table 2 also reveals that the RSTG approach remains obtaining correct result when two scans have no overlapping features (e.g., S-1 and S-3; S-1 and S-4).

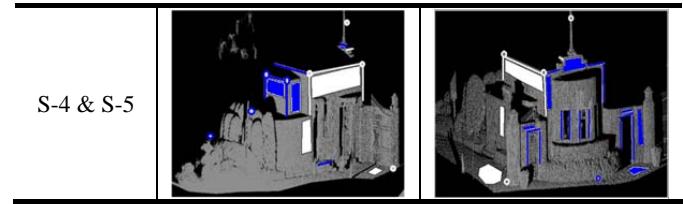
Table 2. the matching performances.

Successful matching rate (%)	S-1	S-2	S-3	S-4	S-5
Operation time (sec)					
S-1		100	N/A	N/A	100
S-2	0.18		92.3	N/A	N/A
S-3	0.09	0.21		92.8	N/A
S-4	0.11	0.11	0.19		100
S-5	0.23	0.10	0.07	0.16	
Total	1.01	0.42	0.26	0.16	

The visual inspections are shown in Table 3 where the white colour denotes the correct matches of features; the blue colour represents those features appearing only in one scan; the red colour indicates false matches with type 1 error. In this case, there was no false match with type 2 error.

Table 3. the visual inspection of matching results.

	Query frame	Reference frame
S-1 & S-2		
S-1 & S-5		
S-2 & S-3		
S-3 & S-4		



Based on the experiments shown above, the RSTG approach was proved as a workable algorithm with qualified performances in terms of effectiveness, efficiency and feasibility.

### 3.3 Evaluation of Algorithmic Contributions

The RSTG approach has some characteristics which can be remarked as follows.

**Scale invariant and multiple feature integration:** Features being matched may be acquired from distinct methods with different types so they might not have an identical scale. By utilizing the form of normalized vector, it allows that multiple kinds of features or features with different scale work together. And the relative scale between data is reconstructed in the scale estimation stage.

**Impact of the amount of feature data:** As other matching algorithms, efficiency would be degraded when dealing with a large number of data sets. However, apart from the stage of geometric check, the RSTG approach only performs a partial matching process and employs initial matches, which can help to reduce the computational cost. Indeed, the step of geometric check would still be affected by the heavy loading of data.

**Uncertainty of features:** As long as the uncertainty of multiple features is given, the tolerances such as distance and angular thresholds can be determined by error propagation. According to the experimental experiences, it would degrade the successful match rate when the tolerance is either too strict or loose. Therefore, the proper and reasonable error propagation based on quality of features plays a significant role in the RSTG approach.

## 4. CONCLUSIONS

This paper proposes the RSTG approach for 3-D feature matching and has proved its effectiveness and efficiency via successful demonstrations. The contributions of the RSTG approach are the capability of matching multiple kinds of features simultaneously while constructing the 3-D similarity transformation. Furthermore, the RSTG approach also has application potential in other research communities. The methodology would be kept improving and investigating into other types of features which can be involved to the RSTG approach.

## 5. REFERENCES

Akca, D., 2003. Fully automatic registration of laser scanner

point clouds, in: Optical 3D Measurement Techniques VI, volume 1, Zurich, Switzerland, pp. 330–337.

Ankerst, M., Kastenmuller, G., Kriegel, H. P., and Seidl, T., 1999. 3D shape histograms for similarity search and classification in spatial databases. In Proc. SSD.

Belongie, S., Malik, J., and Puzicha, J., 2001. Matching shapes, ICCV.

Besl, P.J., and McKay, N.D., 1992. A method for registration of 3D shape, IEEE Trans. Pattern Analysis and Machine Intelligence, 14(2), 239-256.

Chen, Y., and Medioni, G., 1992. Object modelling by registration of multiple range images, Image and Vision Computing, 10(3), 145-155.

Chen, D. Y., Ouhyoung, M., Tian, X. P., and Shen, Y. T., 2003. On visual similarity based 3D model retrieval, Computer Graphics Forum, pp. 223.232.

Chua, C., and Jarvis, R., 1996. Point signatures: A new representation for 3D object recognition, International Journal of Computer Vision 25(1), 63-85.

Dold, C. and Brenner, C., 2006. Registration of terrestrial laser scanning data using planar patches and image data, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, volume 36, pp. 78–83.

Franaszek, M., Cheok, G.S., and Witzgall, C., 2009. Fast automatic registration of range images from 3D imaging systems using sphere targets, Automation in Construction 18, 265–274.

Gal, R., and Cohen-or, D., 2006. Salient geometric features for partial shape matching and similarity, ACM Transaction on Graphics.

Gelfand, N., Mitra, N. J., Guibas, L. J., and Pottmann, H., 2005. Robust global registration, Eurographics Symposium on Geometry Processing , pp. 197–206.

Gruen, A., and Akca, D., 2005. Least squares 3-D surface and curve matching, ISPRS Journal of Photogrammetry and Remote Sensing, 59(3), 151-174.

Habib, A., Mwafag, G., Michel, M., and Al-Ruzouq, R., 2005. Photogrammetric and LiDAR data registration using linear features, Photogrammetric Engineering & Remote Sensing, 71(6), 699–707.

Heczko, M., Keim, D., Saupe, D., and Vrancic, D., 2002. Methods for similarity search on 3D databases, Datenbank-Spektrum, 2, pp. 54-63.

Huang, J., and Menq, C. H., 2001. Automatic data segmentation for geometric feature extraction from unorganized 3-D coordinate points, IEEE Transactions on Robotics and Automation (17): 268–278.

Jaw, J.J., and Chuang, T.Y., 2008. Registration of LIDAR point clouds by means of 3D line features, JCIE Journal of the Chinese Institute of Engineers, 31(6), 1031-1045.

Jaw, J.J., and Chuang, T.Y., 2010. On the effectiveness of feature-based LIDAR point cloud registration, Photogrammetric Computer Vision and Image Analysis, ISPRS Comm. III Symposium, 38(3B), pp. 60-65.

Kim, C., Lee, J., Cho, M., and C. Kim, 2011. Fully automated registration of 3D CAD models with point cloud from construction sites, in: Proceedings of the International Symposium on Automation and Robotics in Construction (ISARC), vol. 2,pp. 311–316.

Li, X., and Guskov, I., 2005. Multi-scale features for approximate alignment of point-based surfaces, Symposium on Geometry Processing.

Rabbani, T., Dijkman, S., Heuvel, F., and Vosselman , G., 2007. An integrated approach for modelling and global registration of point clouds, ISPRS journal of Photogrammetry and Remote Sensing, 61, 355-370.

Trimble, 2005. GS200 Declaration of conformity, <http://www.accupoint.com/uploads/ebay/Untitled.PDF>.

Xu, Z., and Li, Z., 2000. Least Median of Squares Matching for Automated Detection of Surface Deformations, Int'l Archives of Photogrammetry and Remote Sensing, vol. 33, no. B3/2, pp. 1000-1007.

Zhang, Z., 1994. Iterative point matching for registration of freeform curves and surfaces, International Journal of Computer Vision, 13(2), 119-152.