The 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2014)
Nov. 12 – 15, 2014 at Double Tree Hotel by Hilton, Kuala Lumpur, Malaysia

# 3D Registration of Multi-view Depth Data for Hand-Arm Pose Estimation

Yeongmin Ha, Seho Shin, and Jaeheung Park

Department of Transdisciplinary Studies, Seoul National University, Seoul 443-270, Korea
(Tel : +82-31-888-9146; E-mail: shinsh|park73@snu.ac.kr)

***Abstract -*** Human motion analysis has been applied in a wide range of applications. Specifically, hand-arm motion plays an important role in the tele-operation of a robot and in pattern analyses of human motions. However, estimations of the full pose of a human hand with the arm are challenging due to the limited recognition range and the computational cost for real-time processing. In this paper, we propose a fast and efficient solution to articulate hand-arm pose estimations for different modal depth sensors. A marker-less motion capture system is built using multiple depth sensors with different recognition areas. In the 3D registration process, a new and fast outlier rejection method is proposed. It uses the concept of skeletal consistency, which the computation time of the 3D registration process without a loss of robustness. The performance is demonstrated to assess the accuracy, the robustness and the computational cost by of the proposed system in comparison with other methods.

***Keywords -*** 3D registration, point cloud, depth camera, pose estimation, Outlier Rejection.

## 1. Introduction

Human motion analysis has been applied in a wide array of applications. Specifically, the motion of the upper body is very important owing to its multiple uses. When manipulating objects or interacting with other people, articulated hand-arm pose estimations are widely used. Hand-arm motion also plays important roles in the tele-operation of a robot and in pattern analyses of human motion. To acquire human motion data, a variety of motion capture systems can be used. Existing motion capture systems require a spacious studio, a suit or outfit with sensors attached, and a complicated setup and calibration process for the camera. Such systems can obtain comparatively accurate data pertaining to a marker set; however, these systems do not operate in real time, are expensive and require post-processing. With the advent of new-generation depth sensors, the use of 3D depth data is becoming increasingly popular in motion capture applications [1], [9]. These sensors are very simple and easy to use, use regular hardware, and can be bought at a low cost without any setup process, special outfits, or markers. Consequently, people, even non-professionals, can acquire 3D data inexpensively and in real time. However, the limitation of the recognition area and the low resolution of the depth data remain as problems. In order to overcome these problems, we deploy multiple depth sensors with different recognition areas and resolutions. Figure 1 presents the setup of these sensors. As shown in Figure 2, Kinect provides a skeleton model of the arm [17] and Senz3D provides a skeleton model of the hand [18]. These skeleton models have skeletal consistency. Using this consistency, the depth data of each sensor can be effectively aligned. 3D registration is an important process in our simple motion capture system, because point cloud data from different view point is represented in a consistent coordinate frame. 3D registration is the process of finding a solution to the problem of aligning various overlapping 3D point-cloud data views into a complete model [2], [5], [6]. This can be done by formulating an optimization problem for the best rotation and translation between the datasets such that the distance between the overlapping areas of the datasets is minimal [8]. Global searches for this optimization problem use genetic algorithms [10] or evolutionary techniques [11]. Local searches involve a considerable amount of work.

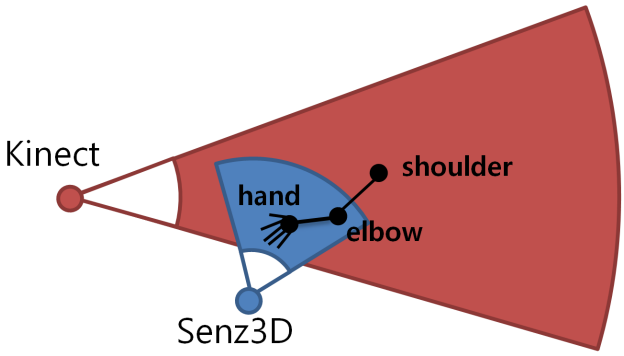



Fig. 1 Illustration of sensors setup. The recognition area of Kinect is wide while that of Senz3D is narrow.

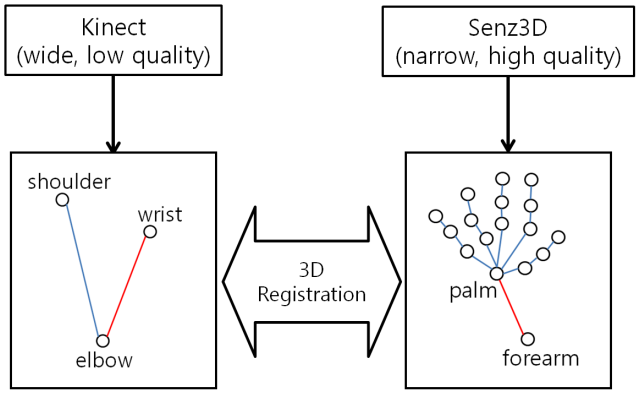



Fig. 2 Skeleton model provided by the middleware of each sensor. Red lines in each skeleton model illustrate instances of skeletal consistency.

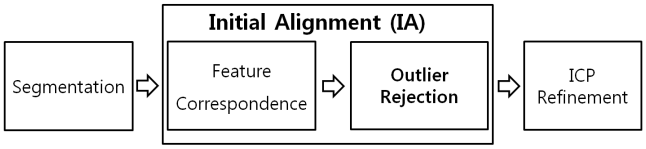



Fig. 3 Pipeline of the 3D registration process.

The most popular method is to use the iterative closest point (ICP) algorithm [7], [12]. Without any information on the initial pose of the datasets, 3D registration is even more difficult and has local optimum solutions. Using the feature of a point cloud with the selection of point correspondences, the process of initial alignment (IA) roughly aligns a source and target dataset [3],[4]. Figure 3 shows the pipeline of the 3D registration process. It consists of three steps. First, skeleton models of each sensor are used for the segmentation step. Then, feature descriptions and searches for the point-to-point correspondences are performed, and false correspondences are removed. In this process, we present a fast and effective outlier rejection method which uses the degree of skeletal consistency. Finally, the third step of the 3D registration process refines the initial alignments above using the ICP algorithm. The key contribution of this research is that the proposed outlier rejection method which uses skeletal consistency reduces the computation time of the 3D registration process without a loss of robustness. The simple motion capture system devised here is very easy and effective because it is simple to handle and has no complicated calibration process involving multiple sensors.

## 2. Segmentation

The first step of the 3D registration pipeline is the segmentation step. Skeleton models provided by the middleware of each sensor are used in this process. Line model segmentation is performed based on the skeletal line from elbow to hand. Let $p_i$ be the $i$-th point in the point cloud data $P = \{p_1, p_2, p_3, \cdots, p_n\}$, with $p_{start}$ and $p_{end}$ both end points of the skeletal line segment. In order to filter the point near the skeletal line, minimum distances between the line and the point cloud data are computed. The line parameter $t_i$ is also calculated to decide whether or not the foot of the perpendicular intersects the skeletal line. The line parameter $t_i$ and the minimum distance $d_i$ can be represented as follows:

$$d_i = \frac{|(p_{start} - p_{end}) \cdot (p_{end} - p_i)|}{\|p_{start} - p_{end}\|} \quad (1)$$

$$t_i = \frac{(p_{start} - p_i) \cdot (p_{end} - p_{start})}{|p_{start} - p_{end}|^2} \quad (2)$$

Using Equations (1) and (2), we can filter the point near the skeletal line. $d_m$ is a predefined threshold value and $t_i$ is the line parameter acquired by (2) above, which must be in the line segment. The point set of the segmentation, $S$, is determined as follows:

$$S = \{p_i : d_i \leq d_m \wedge 0 \leq t_i \leq 1\} \quad (3)$$

(a) (b) (c)

Fig. 4 Segmentation: The red rectangle is a region of interest (ROI). (a) Scene from Kinect. (b) Scene from Senz3D. (c) Point cloud data after segmentation from each viewpoint.

This process can reduce the computation time required for 3D registration through its use of a filtered point set and not the entire point cloud. The line parameter $t_i$ is cached for the proposed outlier rejection method, which is introduced in Section 3.2. It can be used effectively for filtering false correspondences.

## 3. Initial Alignment (IA)

### 3.1 Features and Correspondence

A. Features

Point feature representation can be described as a vector function. It describes local geometric information captured by a neighborhood point set around a query point. That is, each point is associated with a feature describing the local geometry of a point. In this research, two point feature description methods are used. Both utilize geometry-based features. The first is the Signature of Histograms of OrienTation (SHOT) descriptor [13]. It encodes a signature of histograms representing topological traits, making it invariant to rotation and translation and robust against noise and clutter. The second is the Fast Point Feature Histogram (FPFH) descriptor [8], which represents the relative normal orientation, as well as the distance, between point pairs.

B. Correspondence

Once feature vectors are computed for two point cloud datasets, they need to be matched to create point-to-point correspondences. Descriptors are compared using the Euclidean distance, and correspondences are set between each nearest neighbor (NN). In this process, efficient approximated matching schemes are deployed to speed up the matching stage. Here, the fast approximate NN (FLANN) algorithm was used [14].

### 3.2 Outlier Rejection

As a result of the correspondence matching stage, point-to-point correspondences are determined by associating pairs of descriptors that lie close in the descriptor space. A relatively common approach within 3D registration methods is represented by an additional stage, usually referred to as Outlier Rejection, where false correspondences are discarded. There are two types of outlier rejection methods: the geometric methods and the iterative approaches. The geometric-based outlier rejection methods utilize the geometric consistency (GC) and

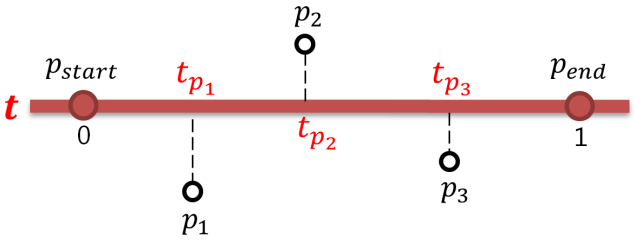
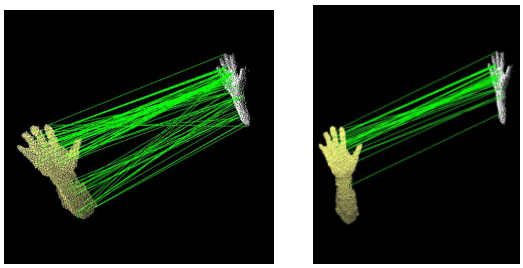



Fig. 5 Points mapped on the skeletal line as line parameter $t$. In this case, $t$ has a range of 0 to 1, as a normalized value.



(a) (b)

Fig. 7 (a) Initial correspondences before the proposed outlier rejection process. (b) Inliers after the process.

Hough 3D voting (Hough). GC [15] uses the Euclidean distance of corresponding point pairs. It is fast and creates a 6D functional vector space which is projected onto 1D. Hough [16] refers to a method based on the local reference frame. It is more robust and uses the 3D functional vector space. The sample consensus (SAC) based method [8] uses an iterative approach to reject outliers. It considers the error metric (1D) for the point cloud that computes the quality of the transformation.

A. Proposed method using skeletal consistency (SC)

As shown in Figure 5, the line parameter is known because is stored during the segmentation process discussed in Section 2. The line parameter refers to the position of the segment on a skeletal line. If two point cloud data are transformed in rotation and translation, the skeletal lines of these data are consistent. Figure 6 shows the process of proposed outlier rejection method. The skeletal data of rejected correspondences are inconsistent. The scene key point is $S_i$ and the model key point $M_i$. The correspondence $C_i$ consists of two point pairs $S_i$ and $M_i$ where $C_i = \{S_i, M_i\}$ . If the distance between line parameters $t_{S_i}$ and $t_{M_i}$ is smaller than a pre-defined threshold

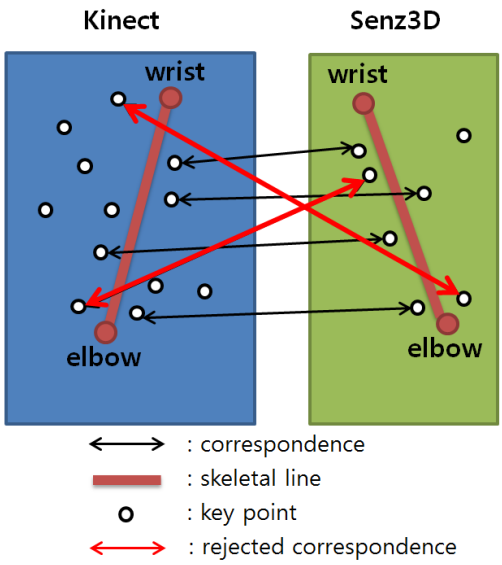



Fig. 6 The proposed outlier rejection method uses skeletal line consistency.

value $\varepsilon$, the correspondence is then rejected. The true correspondence set $C_T$ and the value that represents the skeletal consistency $S_{C_i}$ are determined as follows:

$$S_{c_i} = |t_{S_i} - t_{M_i}| \tag{4}$$

$$C_T = \{C_i : S_{c_i} < \varepsilon\} \tag{5}$$

The computation time of the proposed method is shorter than those of other outlier rejection methods, as the proposed method deploys cashed data $t$ and because its functional vector space is 1D; that is, it only considers the distances between line parameters. The search algorithm is performed at once, i.e.,$O(n)$, where $n$ is the total number correspondences. Figure 7 shows the results of the proposed method.

## 4. ICP Refinement

The last step of the 3D registration process is ICP refinement. It is a post-processing stage which improves the outcome of IA in Section 3. By applying the ICP algorithm [7] to the IA hypotheses, we can refine the estimated 6-DOF transformation. If the outcome of IA does not converge or if it has a lower fitness score, the computation time of ICP refinement becomes slower. The fitness score is the error metric between the corresponding point pairs after ICP. In Section 5.4, this score is used to measure the robustness of the outlier rejection method.

## 5. Experimental Results

### 5.1 System Overview

To validate the proposed simple motion capture system and outlier rejection method, we used a multiple-depth sensor system. Figure 8 presents the system overview used in this experiment. There are three main processes in the system. These are the middleware of body tracking from Kinect, the middleware of hand tracking from Senz3D, and a point cloud library (PCL) based 3D registration process. They run on a multi-process system using shared memory of the type used for storing point cloud data. All experiments were performed on a PC equipped with a 3.4 GHz Intel i5 quad-core processor (i5-3570) with 8GB of memory and a NVIDIA GeForce GTX 650 GPU.

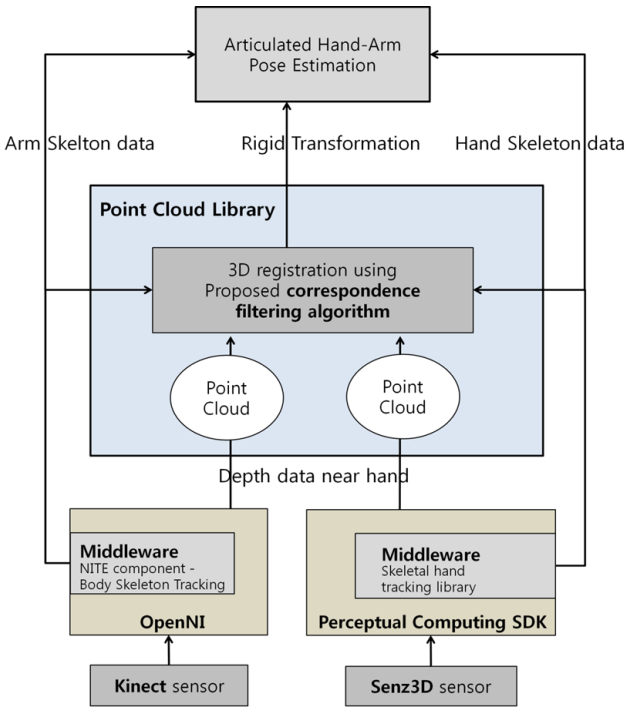
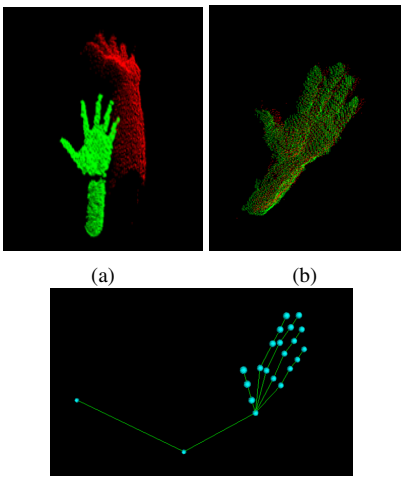



Fig. 8 System overview used in the experiment.

### 5.2 Accuracy

To verify the accuracy of the simple motion capture system, we compared the motion capture data from the Vicon motion capture system at the Advanced Institute of Convergence Technology (AICT, Republic of Korea) with data from our simple motion capture system.

Table 1 shows the average error compared with the data from the Vicon motion capture system. Applying each outlier rejection method to the depth data from our system, the results show few differences. Considering that the average error of Kinect is 0.02m, these are satisfactory results. In Figure 10, average errors of marker



(a) (b)

(c)

Fig. 9 (a) Point cloud data from each camera view point before registration. (b) Point cloud data after registration. (c) Articulated hand-arm skeleton model.

Table 1 Average Error of Each Outlier Rejection Method Compared to the Motion Capture Data.

| Method | Average error (m) |
|---|---|
| SC (Proposed) | 0.0221 |
| GC | 0.0201 |
| Hough | 0.0195 |
| SAC | 0.0203 |

index 0 and 1 are large, as indexes represent a shoulder and an elbow, and because these markers are far from the sensors.

### 5.3 Robustness

We constructed test data set that consists of 162 instances of point cloud data (6 hand gestures $\times$ 27 transformations). They were used to measure the robustness of each method. The fitness score represents the average error of points having a correspondence in the ICP refinement process. Table 2 verifies that the proposed method is more stable than the GC but less stable than the Hough and SAC methods.

Table 2 Fitness Score of Each Outlier Rejection Method.

| Method | Average Fitness Score (m) |
|---|---|
| SC (Proposed) | 88.3106 |
| GC | 189.218 |
| Hough | 24.9871 |
| SAC | 33.8802 |

Table 3 Computation Time of Each Outlier Rejection Method.

| Method | Compuation Time (s) |
|---|---|
| SC (Proposed) | 0.00374 |
| GC | 0.03719 |
| Hough | 0.09592 |
| SAC | 0.09184 |

### 5.4 Computation

We divided the 3D registration process into three stages to verify the computation time of each method. In Figure 11, SC denotes the fastest outlier rejection

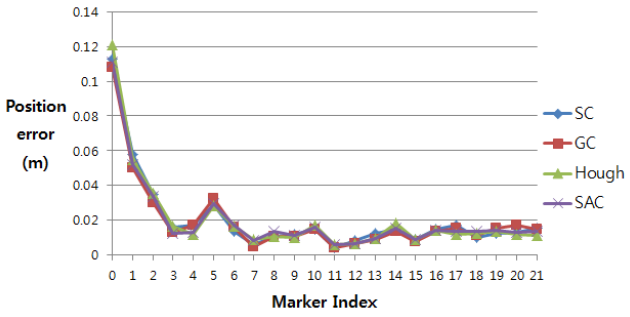



Fig. 10 Average error of each marker.

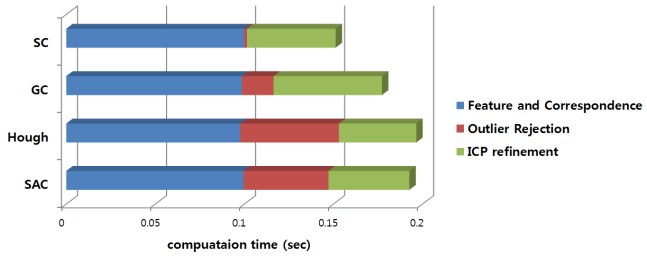



Fig. 11 Computation time to the 3D registration stage of each outlier rejection method.

method, which is approximately 16 times faster than GC. The computation times for the feature descriptions and correspondence matches of each method are similar, as 3D registration stages are independent of each other. In the GC case, the computation time for ICP refinement is longer than the times in other methods due to the fact that GC is weak when it comes to rejecting outliers.

## 6. Conclusion

In this research, a new outlier rejection method for 3D registration is proposed. It takes full advantage of skeleton model constraints. Using the proposed method, we improved the computation speed without a loss of robustness. The workspace of the proposed system is limited to the recognition range of the depth sensors used. In future works, a multiple sensor system with more than two sensors can be developed in order to overcome the limitations of the recognition area.

## References

[1] Y. Mao, Z. Quing, W. Liang, Z. Jiejie, Y. Ruigang, and G. Juergen, Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, Springer, New York, 2013.

[2] R. Madhavan, T. Hong, and E. Messina, "Temporal range registration for unmanned ground and aerial vehicles," *Journal of Intelligent and Robotic Systems*, vol. 44, no. 1, pp. 47-69, 2007.

[3] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, "Robust Global Registration," *in Proc. Symp. Geom. Processing*, 2005.

[4] A. Makadia, A. I. Patterson, and K. Daniilidis, "Fully Automatic Registration of 3D Point Clouds," *in CVPR 06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 12971304, 2006.

[5] F. Tungadi and L. Kleeman, "Multiple laser polar scan matching with application to SLAM," *Australasian Conference on Robotics and Automation*, 2007.

[6] T. Bailey and J. Nieto, "Scan-slam: Recursive mapping and localization with arbitrary-shaped landmarks," *In Robotics: Science and Systems Conference (RSS)*, 2008.

[7] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol.14, No.2, pp.239256, 1992.

[8] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3-D registration," *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pp. 32123217, 2009.

[9] Aldoma A, Marton Z, Tombari F, Wohlkinger W, Potthast C, Zeisl B, et al. "Point Cloud Library: three-dimensional object recognition and 6 dof pose estimation," *IEEE Robotics Autom Mag (RAM)*, vol.19, no.3, pp. 8091, 2012.

[10] L. Silva, O. R. P. Bellon, and K. L. Boyer, "Precision Range Image Registration Using a Robust Surface Interpenetration Measure and Enhanced Genetic Algorithms," *IEEE Trans. Pattern Anal. Mach.Intell.*, vol.27, no.5, pp. 762776, 2005.

[11] O. Cordon, S. Damas, and J. Santamara, "A fast and accurate approach for 3D image registration using the scatter search evolutionary algorithm," *Pattern Recogn. Lett.*, vol.27, no.11, 2006.

[12] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International journal of computer vision*, vol.13, no. 2, pp.119-152, 1994.

[13] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," *in Proc. 11th European Conf. Computer Vision (ECCV 10)*, pp.356-369, 2010.

[14] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," *in Proc. Int. Conf. Computer Vision Theory and Application*, pp.331-340, 2009.

[15] H. Chen and B. Bhanu, "3D Free-Form Object Recognition in Range Images Using Local Surface Patches," *Proc. 17th Intl Conf. Pattern Recognition*, vol. 3, pp.136-139, 2004.

[16] F. Tombari and L. Di Stefano, "Hough voting for 3-D object recognition under occlusion and clutter," *IPSJ Trans. Comput. Vis. Appl. (CVA)*, vol. 4, pp.20-29, 2012.

[17] PrimeSense Ltd. NITE Primesense Middleware, http://www.primesense.com/en/nite, 2011.

[18] The Intel Skeletal Hand Tracking Library, https://software.intel.com/en-us/articles/the-intel-skelskel-hand-tracking-library-experimental-release ,2013