3D shape template generation from RGB-D images capturing a moving and deforming object

Hikari Takehara, Yuta Nakashima, Tomokazu Sato, Naokazu Yokoya
Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayamacho, Ikoma, Nara, Japan

Abstract

Automatically reconstructing a 3D shape model of a non-rigid object using a sequence from a single commodity RGB-D sensor is a challenging problem. Some techniques use a 3D shape template of a target object; however, in order to generate the template automatically, the target object required to be stationary. Otherwise, a non-rigid ICP algorithm, which registers a pair of point clouds, can be used for reconstructing 3D geometry of a non-rigid object directly, but it often fails due to the ambiguity in point correspondences. This paper presents a method for generating a 3D shape template from a single RGB-D sequence. In order to reduce the ambiguity in point correspondences, our method leverages point trajectories obtained in the RGB images, which can be used for associating points in different point clouds. We demonstrate the capability of our method using deforming human bodies.

Introduction

Recently, various applications that present a moving and deforming object to users, such as virtual fitting room [1] and virtual pets [2], have become available to ordinary users. These applications render objects based on 3D geometry of their entire shapes (which we refer to as full-body shape models) and their non-rigid motion, both of which are usually handcrafted. Automatic techniques for reconstructing full-body shape models at each frame can drastically reduce the cost for creating a 3D shape model and motion (*e.g.*, [3, 4]). They use multiple sensors (*e.g.*, RGB or RGB-D sensors), whose relative poses are known, to capture the object from different viewpoints simultaneously. It then applies an existing 3D reconstruction technique for rigid objects, such as [5, 6, 7]. However, the use of multiple sensors may be still cumbersome for some applications in which ordinary users need their own shape models and motions.

Reconstructing 3D shape and motion from a single sensor is a challenging problem. Two approaches have been proposed: one uses 3D shape templates of the target object and the other does not. Former approach [8, 9] generates a 3D shape template using a 3D shape reconstruction technique for rigid objects [5, 6, 7], assuming the object is almost stationary. They then fit the 3D shape template to a 3D point cloud at each frame of a single depth map sequence. One major limitation of this approach is that it requires an extra burden to capture the target object while it is stationary, which is practically infeasible, especially for objects like animals.

Latter approach [10, 11] registers 3D point clouds in all frame of a single depth map sequence to any other frames using non-rigid iterative closest point (ICP) [12, 13]; however, it often fails due to the ambiguity in point correspondences and the high degrees of freedom in non-rigid transformation. In addition, non-rigid ICP is formulated as a non-convex optimization problem and thus depends on initial values of transformation parameters.

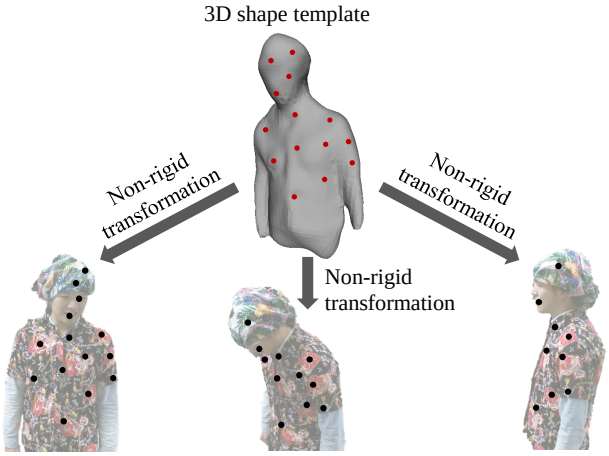


Figure 1: Template points (red) and point trajectories (black).

This paper presents a novel non-rigid point cloud registration method for 3D shape template generation, designed for an single RGB-D sensor. Observing that the RGB image sequence contains rich cues for finding the correspondences between a pair of points on an image and another, our method reduces the ambiguity in point correspondences via optical flow-based point trajectories obtained by, *e.g.*, [14, 15]. Since the point trajectories terminate when they disappear, assuming that the sequence captures the object from all around it, local descriptors (*e.g.*, [16, 17, 18]) loosely associate the point trajectories in different frames to reduce registration errors.

Point cloud registration in our method is based on the non-rigid ICP algorithms by Li *et al.* [12] and Amberg *et al.* [13], which directly register point clouds assuming non-rigid transformation. In order to incorporate the point trajectories obtained from the RGB image sequence into these algorithms, we introduce template points, each of which is a point on the 3D shape template associated with a point trajectory. The correspondences among 3D points in different point clouds are represented by the associated template point as shown in Fig. 1. Our method thus registers the point clouds via the template points. We formulate this non-rigid registration as an optimization problem to find the template points as well as non-rigid transformations from the template points to each point cloud. Due to its non-convexity, our method still depends on initial values of the parameters, and we

alleviate this by adopting two step optimization.

The main contributions of this paper are as follows.

- Our method uses the RGB image sequence for explicitly identifying point correspondences. To the best of our knowledge, no prior work uses the RGB image sequence for the purpose of non-rigid registration. This method well suits for commodity RGB-D sensors, such as Microsoft Kinect.
- Our two step optimization provides stable registration with less dependency on the initial parameter values by gradually increasing the degrees of freedom of the transformation assumed in registration.

Related work

This section reviews the relevant work in full-body 3D shape reconstruction of non-rigid objects.

Multiple sensors. Shape reconstruction using multiple sensors basically is a registration problem for rigid objects if the sensors are synchronized. Starck *et al.* [3] designed a system with eight calibrated RGB sensors, whose relative poses are known. Their system reconstructs the 3D shape of non-rigid objects by combining shape-from-silhouette and multi-view stereo. Dou *et al.* [4] presented a full-body scanning system with eight calibrated RGB-D sensors. It relies on non-rigid registration [19] in order to make the system more robust against unsynchronized sensors and measurement noises. Ye *et al.* [20] developed a motion capturing system with three hand-held RGB-D sensors. Since the poses of these sensors are unknown, it registers point clouds from them using ICP. These systems with multiple sensors can reconstruct the full-body shape of target objects at each frame solely from observations but requires synchronization among the sensors and their relative poses.

Single sensor with prior knowledge. Full-body reconstruction with a single sensor is generally more challenging than the multiple sensor case because it needs to handle unobserved parts of the target object due to, *e.g.*, occlusion.

Some techniques use various types prior knowledge to facilitate shape reconstruction from a single sensor. Prior knowledge includes skeletons of articulated objects [21, 22], statistical shape models [23, 24], have been proposed. Malleson *et al.* [21] proposed a full-body shape reconstruction method designed for human body. The point cloud is divided into several partial point clouds, so that each of them can be handled by rigid registration, which is followed by non-rigid registration. Schmidt *et al.*'s method [22] reconstructs 3D shapes of articulated objects in real-time using GPU accelerated optimization. Anguelov *et al.* [23] fit a partial 3D point cloud to a full-body shape model of human with a trained dataset of various shapes and poses. Chen *et al.* [24] acquire higher quality 3D shapes of non-rigid objects than that by Anguelov *et al.*'s method [23] using a tensor-based deformation model. These approaches are effective for specific objects, such as a human; however, they cannot be applied to other objects.

Single sensor without prior knowledge. Some methods directly register 3D point cloud in every frame to any other frame [10, 11, 12, 13, 25, 26, 27]. Li *et al.* [12] and Amberg *et al.* [13] proposed a non-rigid ICP algorithm, which can handle larger deformation than Brown *et al.*'s [25]. Dou *et al.* [10] acquired high-quality 3D shape of non-rigid objects, using the non-rigid ICP and bundle adjustment algorithms. Newcombe *et al.*[11] presented real-time non-rigid 3D shape reconstruction with GPU-accelerated optimization. This type of approaches can reconstruct full-body shapes from a single RGB-D sequence; however, it often fails due to the ambiguity in point correspondences and the high degrees of freedom of non-rigid transformation. In addition, such algorithms are formulated as a non-convex optimization problem and thus are dependent on initial values of transformation parameters.

Another approach for single-sensor 3D shape reconstruction is to generate a full-body 3D shape beforehand (namely, a 3D shape template) and fit it to a point cloud at each frame [8, 9]. Li *et al.* [8] fit a low resolution 3D shape template to a point cloud and update it for finer geometric details. The 3D shape template is generated by an existing reconstruction method for rigid object (*e.g.*, [5, 6, 7]) assuming the object is almost stationary. Zollhöfer *et al.* [9] presented a real-time method, which is similar to Li *et al.*'s [8] using GPU-accelerated optimization. These method can reconstruct full-body shape models more stably than the methods without shape templates; however, it requires a capturing step dedicated for template generation, which can be practically impossible when the target is, for example, an animal.

Most existing methods with a single RGB-D sensor do not leverage RGB images. Our method provides stable non-rigid registration of point clouds using the RGB image sequence for explicitly identifying point correspondences.

Template generation using point trajectories

Given an RGB-D image sequence capturing a moving and deforming object from entire directions around it, our method first tracks the 2D point trajectories in RGB images using Sundaram *et al.*'s method [14] and then extract 3D point trajectories, each of which is a series of 3D points in each point cloud given by the depth maps. The i-th 3D point trajectory is denoted by $\mathscr{X}_i = \{\mathbf{x}_i^t | t = t_i^{\mathrm{s}}, \ldots, t_i^{\mathrm{e}}\}$, where t_i^{s} and t_i^{e} are the frame indexes at which the point trajectory starts and terminates. The template point associated with $\mathscr{X}_i$ is denoted by $\mathbf{p}_i \in \mathscr{P}$, where $\mathscr{P}$ is the set of template points. We assume that $\mathbf{x}_i^t$ and $\mathbf{p}_i$ is associated by local affine transformation identified by $\mathbf{A}_i^t$ and $\mathbf{b}_i^t$ (*i.e.*, $\mathbf{p}_i = \mathbf{A}_i^t \mathbf{x}_i^t + \mathbf{b}_i^t$). Under this assumption we formulate the non-rigid registration problem as a minimization problem of an energy function with respect to $\mathbf{p}_i$ and $(\mathbf{A}_i^t, \mathbf{b}_i^t)$ for all i and t. The point trajectory is lost when the corresponding point is occluded, potentially resulting in erroneous registration. To alleviate this, we also use local descriptor-based point correspondences in RGB images to loosely associate point trajectories in different frames.

Local descriptor-based point correspondences

Our method uses Sundaram *et al.*'s method [14] to obtain point trajectories, which is based on optical flow in RGB images sequence. This method can track relatively dense points over a longer period; however it cannot re-identify the same point once it is lost. To remedy this problem, we utilize point correspondences by SIFT feature points and local descriptors [16].

To obtain reliable point correspondences, as shown in Fig. 2, we first find point correspondences between a target frame t and the rest, and identify the frames whose number of point correspondences exceed a certain threshold T_C. Since the sensor captures the target object from all around it, the number of the point correspondences usually gives several sets of continuous frames.

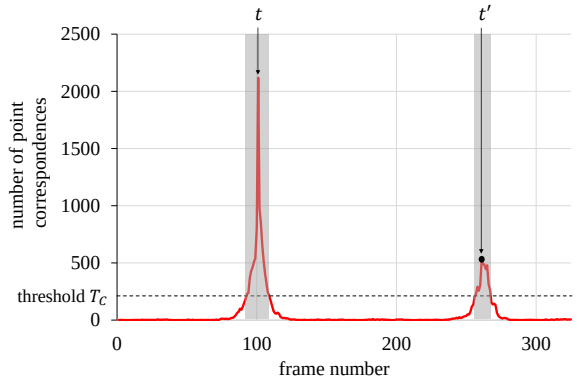


Figure 2: Obtaining a frame pair (t,t') for finding local descriptor-based point correspondences.

For each set, we find the frame t' that gives the largest number of point correspondences. We denote the set of all frame pairs (t,t') obtained in this process by $\mathscr{G}$.

Energy function

Let $\mathscr{A}$ and $\mathscr{B}$ the sets of linear and translation components of the affine transformations. Our energy function is

$$E(\mathscr{P},\mathscr{A},\mathscr{B}) = \alpha_{\mathrm{F}}E_{\mathrm{F}} + \alpha_{\mathrm{C}}E_{\mathrm{C}} + \alpha_{\mathrm{R}}E_{\mathrm{R}} + \alpha_{\mathrm{S}}E_{\mathrm{S}}, \tag{1}$$

where α_{F}, α_{C}, α_{R}, and α_{S} are weights to determine the contributions of terms E_{F}, E_{C}, E_{R}, and E_{S}, respectively.

Registration error term E_{F} involves the registration error defined by the sum of distances between the template points $\mathbf{p}_i$ and corresponding 3D trajectory points transformed by its associated affine transformation $\mathbf{A}_i^t\mathbf{x}_i^t + \mathbf{b}_i^t$, *i.e.*,

$$E_{\mathrm{F}}(\mathscr{P},\mathscr{A},\mathscr{B}) = \sum_t \sum_{i\in\mathscr{V}(t)} ||\mathbf{p}_i - (\mathbf{A}_i^t\mathbf{x}_i^t + \mathbf{b}_i^t)||_2^2, \tag{2}$$

where $\mathscr{V}(t)$ is the index set of point trajectory-based 3D points observed in frame t. The smaller E_{F} is, the closer the template point and the transformed point are to each other.

Local descriptor-based error term E_{C} penalizes large distances between pairs of local descriptor-based 3D points in different frames in the template point space, inspired by Li *et al.*'s method [8]. Instead of defining a template point for each local descriptor-based 3D point $\mathbf{y}_j^t$ as with point trajectory-based one, we choose to loosely associate $\mathbf{y}_j^t$ with several $\mathbf{p}_i$'s so that the point correspondences by local descriptors can directly affects $\mathbf{p}_i$'s. For this, we represent $\mathbf{y}_j^t$'s corresponding point in the template space by a weighted sum of template points associated with 3D points $\mathbf{x}_i^t$ around $\mathbf{y}_j^t$ as shown in Fig. 3, assuming that the spatial relationship among $\mathbf{y}_j^t$ and neighboring $\mathbf{x}_i^t$'s is preserved in the template point space, *i.e.*,

$$\tilde{\mathbf{y}}_j^t = \sum_{i\in\mathscr{M}(t,j)} w_{ji}\mathbf{p}_i, \tag{3}$$

where $\mathscr{M}(t,j)$ is the index set of 3D points $\mathbf{x}_i^t$ in frame t that are m nearest neighbors of $\mathbf{y}_j^t$, and w_{ji} is a weight determined by the distance between $\mathbf{y}_j^t$ and $\mathbf{x}_i^t$. We calculate w_{ji} using Eq. (1) in [8]. This leads our local descriptor-based error term

$$E_{\mathrm{C}}(\mathscr{P}) = \sum_{(t,t')\in\mathscr{G}} \sum_j ||\tilde{\mathbf{y}}_j^t - \tilde{\mathbf{y}}_j^{t'}||_2^2, \tag{4}$$

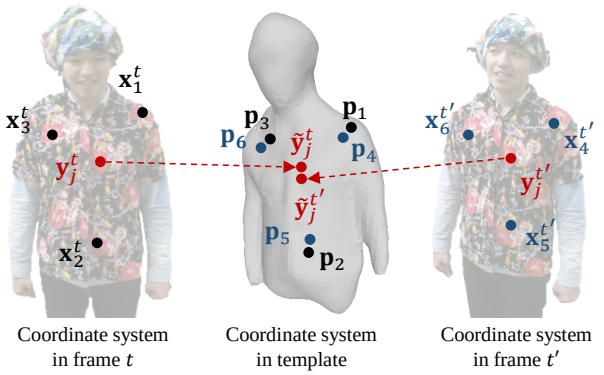


Figure 3: Local descriptor-based 3D point $\mathbf{y}_j^t$ and its representation in the template point space.

where $\mathscr{G}$ is the set of frame pairs described in the previous section. This term implies that the local descriptor-based 3D points in frames t and s should be close enough to each other in the template space as they represent the same point on the object.

Regularization terms E_{R} and E_{S} are introduced to constrain transformation parameters $\mathscr{A}$ and $\mathscr{B}$, assuming that the target object is articulated as in Li *et al.* [8]. Motions of most points on an articulated object can be described by rigid transformations; therefore, E_{R} constrains the linear component $\mathbf{A}_i^t$ to be almost a rotation matrix, which can be represented by

$$E_{\mathrm{R}}(\mathscr{A}) = \sum_t \sum_{i\in\mathscr{V}(t)} ||(\mathbf{A}_i^t)^{\mathrm{T}}\mathbf{A}_i^t - \mathbf{I}||_F^2, \tag{5}$$

where $||\bullet||_F$ is the Frobenius norm.

In addition, because the motions of neighboring points are similar to each other under rigid motions, a 3D point $\mathbf{x}_i^t$ transformed by its own transformation $(\mathbf{A}_i^t,\mathbf{b}_i^t)$ and one of its neighbor's transformation $(\mathbf{A}_j^t,\mathbf{b}_j^t)$ must be close to each other in the template point space. This can be encoded by

$$E_{\mathrm{S}}(\mathscr{A},\mathscr{B}) = \sum_t \sum_i \sum_j ||\mathbf{A}_i^t\mathbf{x}_i^t + \mathbf{b}_i^t - (\mathbf{A}_j^t\mathbf{x}_i^t + \mathbf{b}_j^t)||_2^2, \tag{6}$$

where the second and third summations are calculated for $i\in\mathscr{V}(t)$ and $j\in\mathscr{N}(t,i)$, respectively, and $\mathscr{N}(t,i)$ is the index set of n nearest neighbors of the 3D point $\mathbf{x}_i^t$.

Two-step optimization

Our proposed method minimizes the energy function in Eq. (1), which is a non-convex optimization problem because E_{R} is a quartic function with respect to each element of $\mathbf{A}_i^t$. The solution thus depends on initial values of $\mathscr{P}$, $\mathscr{A}$, and $\mathscr{B}$. To alleviate this dependency, we adopt two-step optimization based on our heuristics.

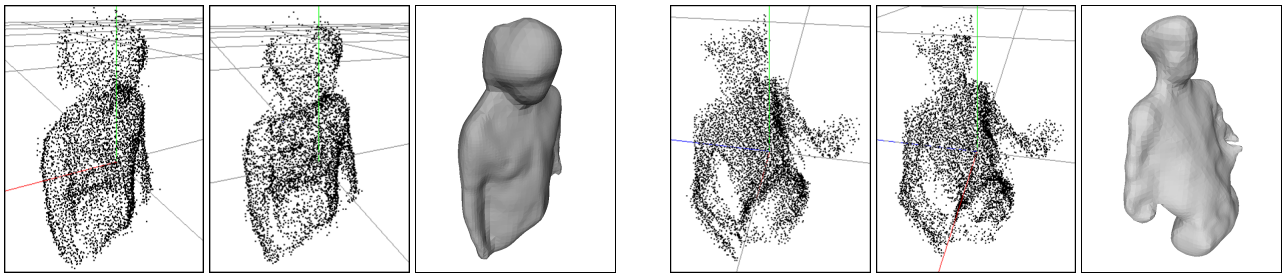
In the first step, we assume that all $\mathbf{x}_i^t$'s share the same transformation, *i.e.*, $\mathbf{A}_i^t = \mathbf{A}^t$ and $\mathbf{b}_i^t = \mathbf{b}^t$ for all i. Under this assumption, the number of free parameters is drastically reduced. Also E_{S} is identically 0 and thus Eq. (1) can be rewritten to

$$E'(\mathscr{P},\mathscr{A},\mathscr{B}) = \alpha'_{\mathrm{F}}E'_{\mathrm{F}} + \alpha'_{\mathrm{R}}E'_{\mathrm{R}} + \alpha'_{\mathrm{C}}E_{\mathrm{C}}, \tag{7}$$

where

$$E'_{\mathrm{F}}(\mathscr{P},\mathscr{A},\mathscr{B}) = \sum_t \sum_{i\in\mathscr{V}(t)} ||\mathbf{p}_i - (\mathbf{A}^t\mathbf{x}_i^t + \mathbf{b}^t)||_2^2, \tag{8}$$

Figure 4: Examples of masked RGB images and depth images in DS1 (left) and DS2 (right).



(a) DS1 (b) DS2

Figure 5: Obtained template points in first (left) and second (center) steps, and 3D shape template (right) in second step for DS1 and DS2.

$$E'_{\mathrm{R}}(\mathscr{A}) \quad = \quad \sum_t ||(\mathbf{A}^t)^{\mathrm{T}}\mathbf{A}^t - \mathbf{I}||_F^2. \tag{9}$$

α'_{F}, α'_{R}, and α'_{C} are weights. The proposed method minimizes E' using the gradient descent method. Since E'_{R} is still a quartic function, this minimization is again a non-convex problem; however, our preliminary study has demonstrated that it gives good initial estimates of parameter values for the later step.

In the second step, Eq. (1) is minimized with initial estimates of the parameter values obtained in the first step, again using the gradient decent method.

Experimental results

Implementation and dataset

We heuristically determined the threshold value $T_C = 80$. The weight values for the first step optimization α'_{F}, α'_{R}, and α'_{C} were set to 1, 100, and 0.1, respectively, and for the second step $\alpha_{\mathrm{F}} = 0.1$, $\alpha_{\mathrm{R}} = 1.0$. Those for the second step optimization were $\alpha_{\mathrm{C}} = 1.0$, $\alpha_{\mathrm{S}} = 1.0$. The numbers m and n of nearest neighbors in $\mathscr{M}$ and $\mathscr{N}$ were both set to 4.

For the first step optimization, we set all the template points $\mathbf{p}_i$ to $\mathbf{0}$ as initial values. For $\mathbf{A}^t$ and $\mathbf{b}^t$, we use the parameters of rigid transformations as their initial values for fast convergence. To obtain the rigid transformation, we registered point clouds from consecutive two RGB-D images sequentially, assuming that the transformation between two point clouds can be described by a rigid transformation. We then calculated the transformation from each frame's point cloud to, *e.g.*, the first frame's one, by recursively multiplying the transformations.

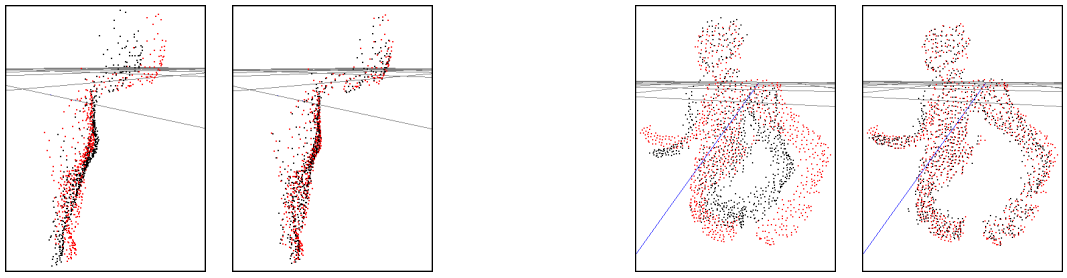
In this experiment, we generated the 3D shape template from two RGB-D sequences, *i.e.*, DS1 capturing a human moving his body and head, and DS2 capturing a human moving his arms, in which the numbers of frames are 350 and 325, respectively. They were captured using fixed Microsoft Kinect v2. The subjects rotated in front of the sensor while capturing. We consider that this is almost equivalent to moving the sensor. Since the captured images included background regions, we extracted the human regions by thresholding the depth values, as shown in Fig. 4.

Results

The template points and shape templates by the first and second optimization steps are shown in Fig. 5, where the mesh models were generated by using an algorithm based on the Poisson formulation [28].

For DS1, the template points around the head converged in the second step optimization compared with those after the first step. This demonstrates that our energy function with local affine transformations works properly. On the other hand, the point around the neck spread after the second step. One of possible reasons is that the local descriptor-based correspondences were erroneous around the neck (*i.e.*, some points around neck were associated with those around the shoulder). For DS2, the template points around the right hand, which was divided into two parts, got closer because of the loose correspondences by local descriptors. However, particularly for forearms, the numbers of point trajectories and local descriptor-based point correspondences were relatively small, and thus the template points did not exhibit good convergence.

To quantitatively evaluate the accuracy of the proposed method, we transformed the template points to the sensor's coordinate system during the first and second optimization steps, and then calculated the distance between each transformed tem-



(a) DS1 (b) DS2

Figure 6: Examples of template points (black) transformed back to the sensor's coordinate system in a certain frame and the point trajectory-based 3D points in that frame (red) in first and second steps for DS1 and DS2.

plate point and its associated point trajectory-based 3D point using the inverse transformation of $\mathbf{A}_i^t$ and $\mathbf{b}_i^t$, as shown in Fig. 6. Fig. 7 shows the average and largest distances as well as the distances' 5th and 95th percentile for DS1 and DS2. These results indicate that our two-step optimization works well, decreasing the distances in both steps, although the largest distance did not decrease in the second step.

For better 3D shape templates, we need removal of wrong correspondences by local descriptors as well as uniformly extracted point trajectories. The geodesic distance-based weights for Eq. (3) may also improve the local descriptor-based point correspondences, especially for the neck and the forearms in DS1 and DS2, respectively.

Conclusion

In this paper, we have proposed a method for 3D shape template generation leveraging RGB image sequence obtained while capturing depth images with an RGB-D sensor. For stability, we have also proposed the two-step optimization strategy, which gradually increases the degree of freedom of transformations assumed in the energy function. Our experimental results have demonstrated that the proposed method can generate the 3D shape template with about 0.01 meter of the averaged distance between a transformed template point and its associated 3D point. Our future work includes the improvement of the quality of shape template by, *e.g.*, removing wrong local descriptor-based point correspondences and using geodesic distances to determine the weights in Eq. (3).

Acknowledgements

This work was supported in part by JSPS KAKENHI No. 25540086.

References

[1] "TriMirror Virtual Fitting Room." http://www.trimirror.com/en/about/.

[2] "FooPets — Real Virtual Pets Online." http://www.foopets.com/.

[3] J. Starck and A. Hilton, "Surface capture for performance-based animation," *IEEE Computer Graphics and Applications*, vol. 27, no. 3, pp. 21–31, 2007.

[4] M. Dou, H. Fuchs, and F. J., "Scanning and tracking dynamic objects with commodity depth cameras," *Proc. IEEE Int'l Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 99–106, 2013.

[5] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," *Proc. IEEE Int'l Conf. on Computer Vision (ICCV)*, pp. 2320–2327, 2011.

[6] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," *Proc. IEEE Int'l Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 127–136, 2011.

[7] V. Pradeep, C. Rhemann, S. Izadi, C. Zach, M. Bleyer, and S. Bathiche, "MonoFusion: Real-time 3D reconstruction of small scenes with a single web camera," *Proc. IEEE Int'l Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 83–88, 2013.

[8] H. Li, B. Adams, L. J. Guibas, and M. Pauly, "Robust single-view geometry and motion reconstruction," *ACM Trans. on Graphics (TOG)*, vol. 28, no. 5, pp. 175:1–175:10, 2009.

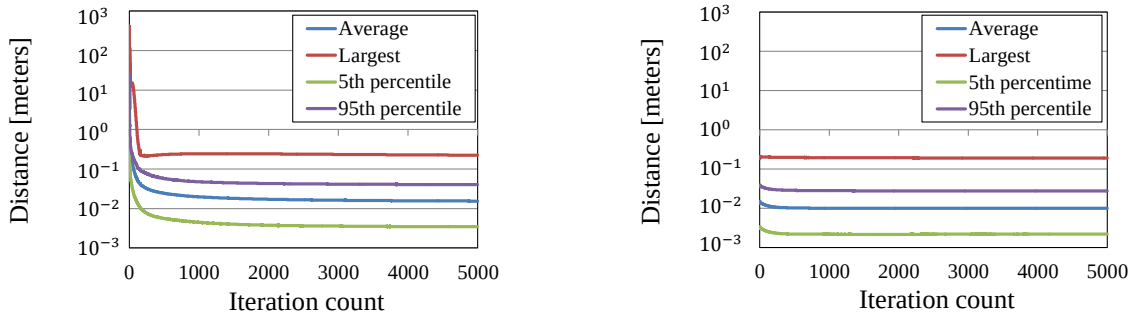
[9] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger, "Real-time non-rigid reconstruction using an RGB-D camera," *ACM Trans. on Graphics (TOG)*, vol. 33, no. 4, pp. 156:1–156:12, 2014.

[10] M. Dou, J. Taylor, H. Fuchs, A. Fitzgibbon, and S. Izadi, "3D scanning deformable objects with a single RGBD sensor," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 493–501, 2015.

[11] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 343–352, 2015.

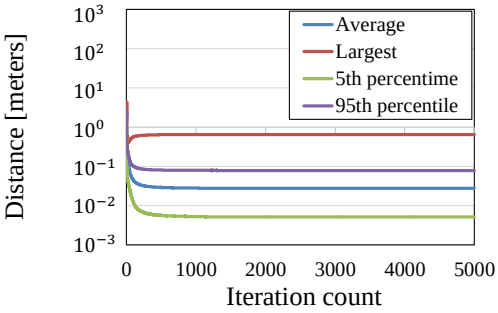
[12] H. Li, R. W. Sumner, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," *Proc. Eurographics Symposium on Geometry Processing (SGP)*, pp. 1421–1430, 2008.

[13] B. Amberg, S. Romdhani, and T. Vetter, "Optimal step nonrigid ICP algorithms for surface registration," *Proc. IEEE Computer So-*

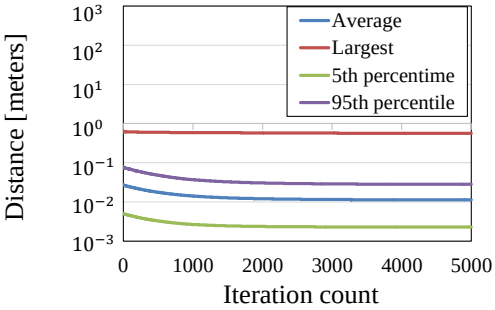


(a) DS1, first step

(b) DS1, second step



(c) DS2, first step



(d) DS2, second step

Figure 7: The evolution of average distance, largest distance, 5th and 95th percentile value of distance in first (a)(c) and second (b)(d) steps for DS1 and DS2.

ciety Conf. on Computer Vision and Pattern Recognition (CVPR), 8 pages, 2007.

[14] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by GPU-accelerated large displacement optical flow," *Proc. European Conf. on Computer Vision (ECCV)*, pp. 438–451, 2010.

[15] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," *Int'l Journal of Computer Vision*, vol. 80, no. 1, pp. 72–91, 2008.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int'l Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[17] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *Proc. European Conf. on Computer Vision (ECCV)*, pp. 404–417, 2006.

[18] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," *Proc. IEEE Int'l Conf. on Computer Vision (ICCV)*, pp. 2564–2571, 2011.

[19] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded deformation for shape manipulation," *ACM Trans. on Graphics (TOG)*, vol. 26, no. 3, pp. 80:1–80:7, 2007.

[20] G. Ye, Y. Liu, N. Hasler, X. Ji, D. Q, and C. Theobalt, "Performance capture of interacting characters with handheld kinects," *Proc. European Conf. on Computer Vision (ECCV)*, pp. 828–841, 2012.

[21] C. Malleson, M. Klaudiny, A. Hilton, and J. Guillemaut, "Single-view rgbd-based reconstruction of dynamic human geometry," *Proc. IEEE Int'l Conf. on Computer Vision Workshops (ICCVW)*, pp. 307–314, 2013.

[22] T. Schmidt, R. A. Newcombe, and D. Fox, "DART: Dense articulated real-time tracking with consumer depth cameras," *Autonomous Robots*, vol. 39, no. 3, pp. 239–258, 2015.

[23] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "SCAPE: shape completion and animation of people," *ACM Trans. on Graphics (TOG)*, vol. 24, no. 3, pp. 408–416, 2005.

[24] Y. Chen, Z. Liu, and Z. Zhang, "Tensor-based human body modeling," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 105–112, 2013.

[25] B. J. Brown and S. Rusinkiewicz, "Global non-rigid alignment of 3-D scans," *ACM Trans. on Graphics (TOG)*, vol. 26, no. 3, pp. 21:1–21:9, 2007.

[26] H. Li, E. Vouga, A. Gudym, L. Luo, J. T. Barron, and G. Gusev, "3D self-portraits," *ACM Trans. on Graphpics (TOG)*, vol. 32, no. 6, pp. 187:1–187:9, 2013.

[27] M. Zeng, J. Zheng, X. Cheng, and X. Liu, "Templateless quasi-rigid shape modeling with implicit loop-closure," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 145–152, 2013.

[28] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," *Proc. Eurographics Symposium on Geometry Processing (SGP)*, pp. 61–70, 2006.

Author Biography

Hikari Takehara *received his B.E. in electrical and electronic engineering from the National Institute of Technology, Kumamoto in 2013 and his M.E. in information science from Nara Institute of Science and Technology (NAIST) in 2015. He is currently pursuing his Ph.D. at NAIST. His main research interests include computer vision and computer graphics.*

Yuta Nakashima *received his B.E. and M.E. degrees in communication engineering from Osaka University, Osaka, Japan in 2006 and 2008, respectively, and the Ph.D. degree in engineering from Osaka University, Osaka, Japan, in 2012. He is currently an assistant professor at Graduate School of Information Science, Nara Institute of Science and Technology (NAIST). He was a research fellow of the Japan Society for the Promo-*

tion of Science (JSPS) from 2010 to 2012, and was a Visiting Scholar at the University of North Carolina at Charlotte in 2012. His research interests include video content analysis using probabilistic and statistical approaches. He is a member of the IEEE, the ACM, and the IEICE.

Tomokazu Sato *received his B.E. degree in computer and system science from Osaka Prefecture University in 1999. He received his M.E. and Ph.D. degrees in information science from Nara Institute of Science and Technology (NAIST) in 2001 and 2003, respectively. He was an assistant professor at NAIST in 2003-2011. He was a visiting researcher at Czech Technical University in Prague in 2010-2011. He has been an associate professor at NAIST since 2011. He is a member of IEEE, IEICE, IPSJ, VRSJ and ITE.*

Naokazu Yokoya *received his B.E., M.E., and Ph.D. degrees in information and computer sciences from Osaka University in 1974, 1976, and 1979, respectively. He joined Electrotechnical Laboratory (ETL) of the Ministry of International Trade and Industry in 1979. He was a visiting professor at McGill University in Montreal in 1986-87 and has been a professor at Nara Institute of Science and Technology (NAIST) since 1992. He has also been a vice president at NAIST since April 2013. He is a fellow of IPSJ, IEICE and VRSJ and a member of IEEE, ACM SIGGRAPH, JSAI, JCSS and ITE.*