Deformable Registration of Textured Range Images by Using Texture and Shape Features

Ryusuke Sagawa, Nanaho Osawa, Yasushi Yagi
Institute of Scientific and Industrial Research, Osaka University,
8-1 Mihogaoka, Ibaraki-shi, Osaka, 567-0047, JAPAN
{sagawa,n-oosawa,yagi}@am.sanken.osaka-u.ac.jp

Abstract

This paper describes a method to align textured range images of deforming objects. The proposed procedure aligns deformable 3D models by matching both texture and shape features. First, the characteristics of each vertex of a 3D mesh model is defined by computing a color histogram for the texture feature and the average signed distance for the shape feature. Next, the key points, which are the distinctive vertices of a model, are extracted with respect to the texture and shape features. Subsequently, the corresponding points are located by matching the key points of the models before and after deformation. The deforming parameters are computed by minimizing the distance between the corresponding points. The proposed method iterates the correspondence search and deformation to align range images. Finally, the deformation for all vertices is computed by interpolating the parameters of the key points. In the experiments, we obtained textured range images by using a real-time range finder and a camera, and evaluated deformable registration for the range images.

1 Introduction

The majority of research in range sensing and 3D modeling involves stationary objects. However, there is an increasing number of studies measuring and modeling moving and non-rigid objects due to the recent progress of range sensing techniques.

Examples of non-rigid objects being observed are articulated objects such as a human body and deformable objects such as cloth. If the motion and pose of a human body is analyzed, this knowledge can then be applied to CG animation, medical research, robotics, and preservation of intangible cultural assets. Since the motion of the human body, which has many joints, is complicated there are many requirements for research such as a large-scale motion capture (mocap) system, equipping markers, and construction of an a priori model from the object's knowledge.

In this paper, we propose a method to model a deforming object by matching range images before and after deformation. These images are captured by a range sensor that obtains range images at a high frame rate. Since the shape of an object is not necessarily an invariant in the modeling of a deformable object, the proposed method utilizes texture images for matching models. The texture and range images are simultaneously obtained with a camera. Many markers are necessary for matching in a mocap system, our procedure is advantageous as it is a marker-less method. Moreover, since our method does not use a priori information of a specific object, it can be applied to deformable objects such as clothes. Though ICP-based (Iterative Closest Point) methods [5, 20] are often used for registration of rigid objects, correct correspondences are not expected to be obtained by searching the closest points. Therefore, the proposed method aligns range images using feature points that are extracted by computing the characteristics of texture and shape.

In related work, many studies have proposed feature-based methods that compute either texture or shape features. For texture feature computation, the Harris operator [14], the Scale Invariant Feature Transform (SIFT) feature [18], and histogram-based methods [24, 4] are often used. When matching two-dimensional features, it is necessary to consider the apparent scale of a feature in an image. However, consideration is not necessary when matching textures are mapped on a shape model since the scale is known from the shape. In contrast, the methods based on shape features use invariants with respect to rigid transformation [16, 19, 12, 26, 7]. Since rigid transformation is assumed, it is necessary to extend the methods to apply to deformable registration. If a method uses a normal vector to compute an invariant the method is inappropriate for range images with high-frequency noise since the normal vector will become unstable. In [6, 13, 28], methods that use both texture and shape are proposed. Because rigid transformation is assumed in these methods, it is necessary to extend these procedures to apply to deformable registration.

As studies of deformable registration, Chui and Rangarajan [8] proposed a framework of deformable registration, which minimized the distance of softassigned corresponding points by deforming models based on thin-plate spline. This method only used shape information for matching. Variational approaches are used for deformable registration of 2D images and 3D volumes [15, 22]. These approaches only used shape information for matching, too. In [9, 2, 1, 17, 3], the reference model of the shape of a human body is fitted to measured data. In these studies, a mesh or bone model is required as reference. Moreover, markers are necessary for matching in [2, 3].

In addition, methods to compute 3D motion of a human body have been proposed. Vedula et al. [27] suggested a method based on optical flow with multiple cameras. A procedure based on matching images must consider changes of apparent scale. This matching is more difficult than the matching of range images as the scale is known if the shape is given. Okada et al. [21] proposed a method to estimate the pose of a human body by matching the input image with the posture dictionary. This technique requires the creation of a database of various postures from the knowledge of human motion. Demirdjian et al. [10] proposed a method to constrain the pose of a human body. This method assumes the link of the articulated human body.

This paper is organized as follows. The overview of the proposed method is described in Section 2. The computation of texture and shape features is explained in Section 3, and a method to extract the key points, which have strong characteristics, is described in Section 4. A technique of deformable registration by matching key points is proposed in Section 5. We experimented with deformable registration using textured range images in Section 6. Finally, the proposed method is summarized in Section 7.

2 Overview

In this paper, it is assumed that textured range images are obtained in high frame rate. A textured range image is captured by using a range sensor and a camera. A range sensor is used to capture the 3D points of an object. If the sensor uses a two-dimensional sensor such as a CCD, the data obtained is called a range image and contains dense information on the distance from the sensor to the object surface for each pixel. For mapping a texture, a mesh model is created by linking the neighboring points of a range image. A camera captures an image synchronously with a range image. By calibrating the camera with the range sensor, the image can be mapped on the 3D model as a texture.

We obtain textured range images of a deforming object at a high frame rate and accomplish deformable registration as shown in Figure 1. The range image A is aligned to the range image B with deformation in this example. Hereinafter, the models of range image A and B are called the deforming model and the reference model, respectively. Each step of the proposed method is as follows:

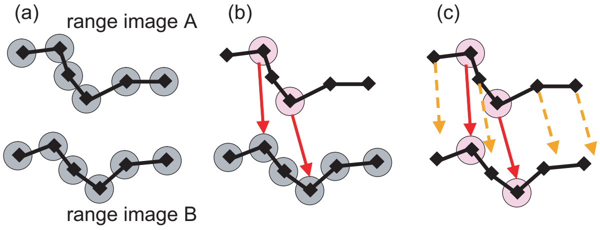


Figure 1. Overview of the deformable registration of range image A to range image B: (a) Texture and shape features are computed for the vertices of each range image. (b) Key points are extracted from range image A and the corresponding points of the key points are located. (c) A deforming vector for each key point is computed and deforming vectors are interpolated for all vertices.

1. Compute texture and shape features for each vertex of the two mesh models.
2. Extract key points, which have strong characteristics, from the deforming model.
3. Search for the corresponding points of the key points in the reference model.
4. Compute a deforming vector for each key point by using the correspondences.
5. Iterate Steps 3 and 4 until convergence.
6. Interpolate deforming vectors for all vertices from the key points.

Each step is described in detail in the following sections.

3 Computing Texture and Shape Features

The proposed method first computes local features for each vertex in the 3D models with respect to texture and shape. In this paper, it is assumed that the change of a local feature is small even if the global shape of a 3D model is grossly deformed. A local feature should be invariant with respect to such aspects as rigid transformation, data noise, illumination, and apparent scale. However, since the shape of an object is known, it is not necessary to consider the apparent scale to construct a feature.

To compute a local feature for a vertex, we consider concentric spheres of different radii of which the center is a vertex v of a textured range image. If a, b, and c in Figure 2 are concentric spheres from the center migrating out, respectively, the differences of the spheres become layers. The layers are hereinafter referred to as Layers 1, 2, and 3 commencing from the inside and moving outwards. Layer 1 is equivalent to sphere a, and Layer 2 is the difference of

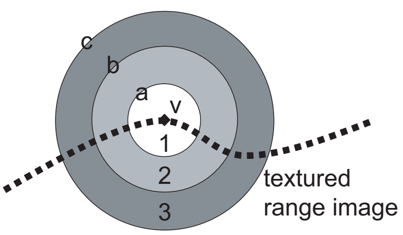


Figure 2. The proposed method computes the local features for each layer around a vertex $\boldsymbol{v}$ of a textured range image. A layer is the difference between concentric spheres with different radii.

sphere a and b. The proposed method computes the texture and shape features for each layer, which then becomes the features of the vertex $\boldsymbol{v}$.

The features computed for each layer is dependent upon the distance from the center $\boldsymbol{v}$. However, they are independent of the rigid transformation of the model. Since the spin-image [16] and local log-polar range image [19] use a normal vector to construct the features, the features become unstable if high-frequency noise is included in a range image. However, because the proposed method does not use a normal vector, it is robust even if high-frequency noise exists.

3.1 Texture Features

In this section, the computation of the texture feature of each layer is described. The proposed method computes a color signature [24] by creating a color histogram of the texture image that mapped to the 3D model with a layer. First, we find a region in the texture image which is mapped to the meshes within the layer. Next, a histogram is created by voting the color of each pixel in the region. Figure 3 shows an example of creating a texture feature for Layer 1.

CIE L*a*b* color space [29] is used for creating a color histogram. The difference of color in L*a*b* color space is computed by a Euclidean distance of a three-dimensional color vector (L, a, b). 125 bins are used in three-dimensional space by dividing the color space to five bins along each axis and a histogram is formed by voting the color of each pixel. The value of each bin is normalized by dividing by the total number of pixels in the region.

Next, the color distribution, represented by a histogram, is converted to a color signature. A color signature is a set of pairs $(\boldsymbol{p}_i, w_i)$, where $\boldsymbol{p}_i$ is a three-dimensional feature vector and w_i is the weight. The central color vector (L_i, a_i, b_i) is used as $\boldsymbol{p}_i$, and the ratio of the votes of the bin with respect to the total votes is used for w_i. If the weight of a bin is 0, the bin is excluded from the color signature. Though a histogram uses a fixed number of bins, a color signature can be used to reduce the number of the elements.

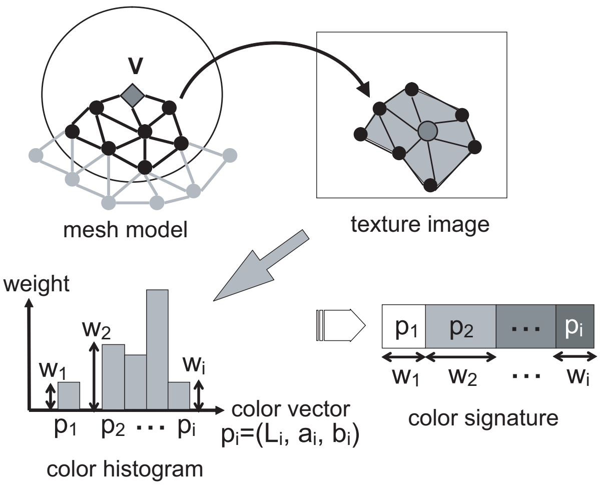
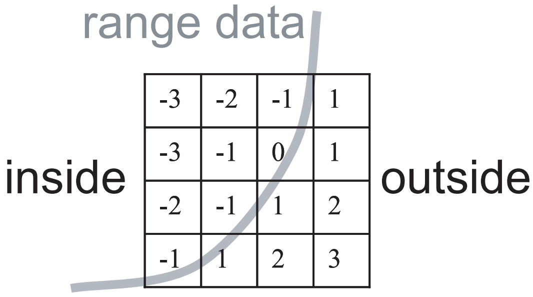


Figure 3. An example of computing a texture feature for Layer 1. First, a region is identified in a texture image which is mapped to the meshes within Layer 1. Next, a color histogram is created by voting the colors of the pixels in the region and converting it to a color signature.



-3	-2	-1	1
-3	-1	0	1
-2	-1	1	2
-1	1	2	3

Figure 4. Example of a SDF: the size of a signed distance is defined as the distance to an object surface. The + or - sign indicates inside or outside of the object surface, respectively.

3.2 Shape Features

This section describes computation of the shape features for each layer. A signed distance field (SDF), which is a volumetric representation of a shape, was used for a shape feature. The 3D space is divided into voxels and a SDF is created by computing the signed distance $f(\boldsymbol{x})$ for the center of each voxel. The magnitude of $f(\boldsymbol{x})$ is the distance from the center of a voxel to the surface of an object and the sign is positive if the voxel is outside of the surface and negative if it is inside. Namely, the surface is represented by the set of points $\boldsymbol{x}$ that satisfy $f(\boldsymbol{x}) = 0$. We used a method proposed by Sagawa et al. [25] to compute a signed distance. Figure 4 shows an example of a SDF.

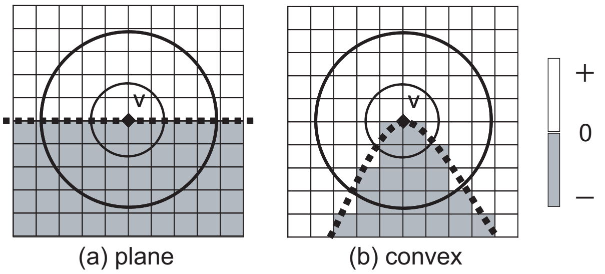


Figure 5. The average of the signed distance in a layer is a shape feature. (a) planar case: the average of the SD is approximately 0. (b) convex case: the average of SD is much greater than 0.

First, the SDFs for both models are calculated. We use the average of the signed distances in a layer to represent the local shape. Gelfand et al. [12] used the ratio of the volume of objects in a unit sphere inside as a shape feature. If an average signed distance is used as a shape feature, the distance from the surface can be represented by this feature. If v is a vertex of a planar surface as shown in Figure 5(a), the same number of positive and negative signed distances is included in a layer. Thus, the average signed distance becomes 0. If a vertex is at the peak of a convex surface as shown in Figure 5(b), the number of positive signed distances increases and the average becomes much larger than 0. In contrast, if concave, the average becomes negative. Therefore, the local shape is represented by the average signed distance in the layers.

However, an actual range image has noise for each vertex. Because Layer 1 is close to the vertex v and the volume is smaller than the other layers, the average signed distance is largely affected by the noise. The average signed distances can be computed for Layers 2 and 3 that have larger volumes. Thus, we used the averages of Layers 2 and 3 as a shape feature.

4 Extracting Key Points

In this section, vertices that have strong characteristics are extracted by using the features computed in the previous section. It can be assumed that a vertex that exhibits strong characteristics is easy to match even if the model is transformed and deformed. We define a vertex of strong characteristics as a key point. To extract a key point, we estimate the strength of features based on the texture and shape features.

4.1 Strength of Texture Feature

A point that has a strong texture feature is namely one that the color distribution is different from that of the surrounding area. In a 2D case, Rosten and Drummond [23] used a circle around a feature point. If the intensity of the center pixel is different from the intensities of the pixels on the circle, a pixel is extracted as a feature point. Thus, we define a key point as a vertex which the color distributions differ greatly between the inner and outer layers. In Figure 6(a), v has a strong texture feature. The color distributions of Layers 1 and 2 are greatly different each other. If v is off the point as shown in Figure 6(b), the color distributions of Layers 1 and 2 are similar.

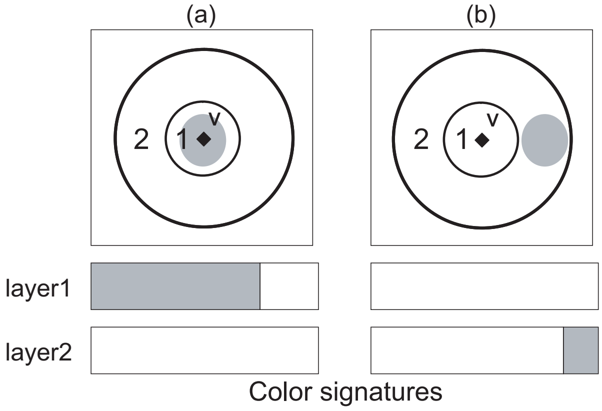


Figure 6. A strong texture feature is extracted by using the difference between the color distributions of Layers 1 and 2. (a) If a vertex v is on a texture feature, the difference of distribution is large. (b) If the vertex v is off a texture feature, the difference of distribution is small.

The proposed method estimates the dissimilarity of color signatures by Earth Mover's Distance (EMD) [24]. The EMD is computed as a solution to the transportation problem of linear optimization. A large transportation cost means that the distance between two color signatures is large. Therefore, we compute the distance between Layers 1 and 2 for each vertex and use the transportation cost as the strength of the texture feature. The dissimilarity measured by EMD has better results in image retrieval since it estimates the distance by transportation cost, though the distance of the bins is not considered in a color histogram [24]. In this paper, we used Euclidean distance of color vectors as the distance of the bins.

4.2 Strength of Shape Features

A point that has a characteristic local shape is one of largely convex or concave surface. Since we use the average signed distance as a shape feature, convex and concave surfaces have a large magnitude of the average signed distance. Thus, we use the magnitude as the strength of shape feature.

4.3 Choosing Key Points

To extract the candidates of the key points, vertices are chosen that have either texture or shape features stronger than a threshold. Since the candidates may gather to each other, we choose the vertex that has the maximum value in a local space. Namely, if a candidate has the greatest strength compared to the other candidates in a local space, it is extracted as a key point.

5 Deformable Registration

In this section, we describe a method of deformable registration using the key points extracted in the previous section. First, the proposed method searches the corresponding point and computes the deforming vector for each key point by using the correspondences. Next, it interpolates the deforming vectors for all vertices of a deforming model. The deformable registration is accomplished by using the deforming vectors to translate the vertices of the deforming model.

5.1 Correspondence Search

To find the corresponding point in a reference model to a key point in a deforming model, it is assumed that textured range images are captured at a high frame rate and the corresponding point is close to the key point. Even if the data does not satisfy this assumption, the proposed method can be applied if the initial registration is done by manual or rigid transformation.

The proposed method initially searches the vertices that are within a certain distance from a key point. If both distances of the texture and shape features are smaller than the thresholds, the points are determined as a corresponding point. If multiple vertices are found, the closest point in the candidates is chosen. To compute the distance of features, we use the EMD for texture feature and the absolute difference of two average signed distances. To find the points within a distance from the key point, we used a k-d tree [11] to reduce the cost.

5.2 Computing the Deforming Vectors for Key Points

Allen et al. [2] proposed a method to fit a template 3D model to a measured range image by deforming the model. The method used markers as feature points and computed deforming parameters by minimizing a cost function. The cost function consisted of the weighted sum of the distances between the closest points, the marker distances, and the differences of the transformation matrices of neighboring points. It is minimized by a quasi-Newtonian solver.

The proposed method minimizes a cost function to compute an optimal deforming vector in a similar manner. Instead of using markers, it minimizes the distance between key points and the corresponding points. The cost function of the proposed method consists of the distances of the corresponding points E_c and the smoothness of deformation E_s. If the i-th key point is $\boldsymbol{k}_i$, the corresponding point of $\boldsymbol{k}_i$ is $\boldsymbol{c}_i$, and the deforming vector of $\boldsymbol{k}_i$ is $\boldsymbol{T}_i$, E_c is the sum of the distance of corresponding points as follows:

$$E_c = \sum_{i}^{n} \parallel \boldsymbol{k}_i + \boldsymbol{T}_i - \boldsymbol{c}_i \parallel^2, \tag{1}$$

where n is the number of the key points.

E_s is a constraint to smoothly deform a mesh model. E_s is defined by the following equation to penalize the difference of the deforming vectors of neighboring key points:

$$E_s = \sum_{i}^{n} \sum_{j}^{n} w_{i,j} \parallel \boldsymbol{T}_i - \boldsymbol{T}_j \parallel^2, \tag{2}$$

where $w_{i,j} = 1/ \parallel \boldsymbol{k}_i - \boldsymbol{k}_j \parallel$. Finally, the cost function E becomes the following function, which is minimized by a quasi-Newtonian solver:

$$E = \alpha E_c + \beta E_s, \tag{3}$$

where α and β are the weights given by the user. The deforming vector is initialized to $\boldsymbol{c}_i - \boldsymbol{k}_i$ for non-linear minimization.

5.3 Iterative Registration

Since the smoothness constraint E_s restricts the deformation, the distance between corresponding points does not become 0. Therefore, we iterate the correspondence search and deformation alternately and the distances are gradually minimized. This idea is similar to ICP and is effective in avoiding the effect of the wrong correspondences of the key points. During iteration, we change the weights α and β to adjust the smoothness constraint. In the experiment, α is fixed to 1 and β is linearly reduced from 2 to 0 during iteration.

5.4 Computing Deforming Vector for All Vertices

Finally, we compute the deforming vectors for all vertices of a deforming model. A deforming vector $\boldsymbol{T}_v$ of a vertex $\boldsymbol{v}$ is interpolated by using the deforming vectors $\boldsymbol{T}_i$ of the key points $k_i(i = 1, \ldots, n$ as follows:

$$\boldsymbol{T}_v = \sum_{i}^{n} W_i \boldsymbol{T}_i, \tag{4}$$

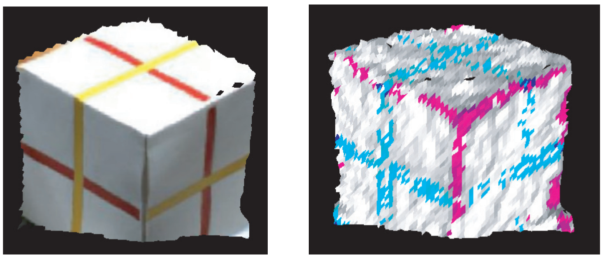
where $W_i = \frac{1}{\|\boldsymbol{v}-\boldsymbol{k}_i\|} / \sum_j^n \frac{1}{\|\boldsymbol{v}-\boldsymbol{k}_j\|}$.

6 Experiments

In the experiments, we used a range sensor, Canesta DP300, and a camera, Point Grey Research Flea2. The DP300 is a fast range sensor, which measures distance by infra-red lights. We captured textured range images by calibrating the range sensor and camera a priori. The range of searching corresponding points was set at 10cm and the maximum iteration was 10.

6.1 Estimating the Accuracy of Deformation

First, we estimate the accuracy of deformation. We obtained a textured range image shown in Figure 7(a) and deformed the model manually, which is used as a reference model. To estimate the accuracy, we compare the deforming vectors computed by the proposed method and that of



(a) a textured range image (b) candidates of key points

Figure 7. Target object: (a) a textured range image and (b) candidates of key points. The light blue points represent texture key-points, and the light pink points are shape key-points.

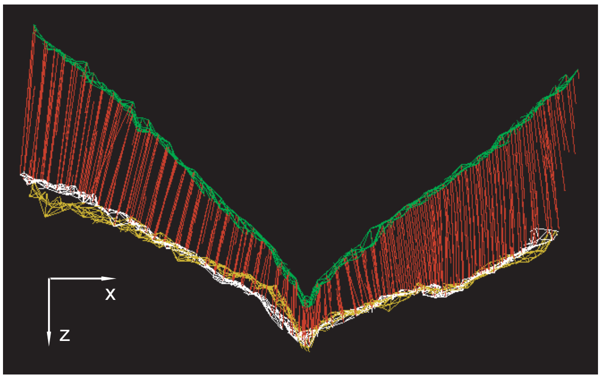


Figure 8. Slices of mesh models: The green mesh model is the model before deformable registration. The yellow one is the model deformed by the transform vectors of ground truth. The white one is the model after deformable registration. The red lines denote deforming vectors.

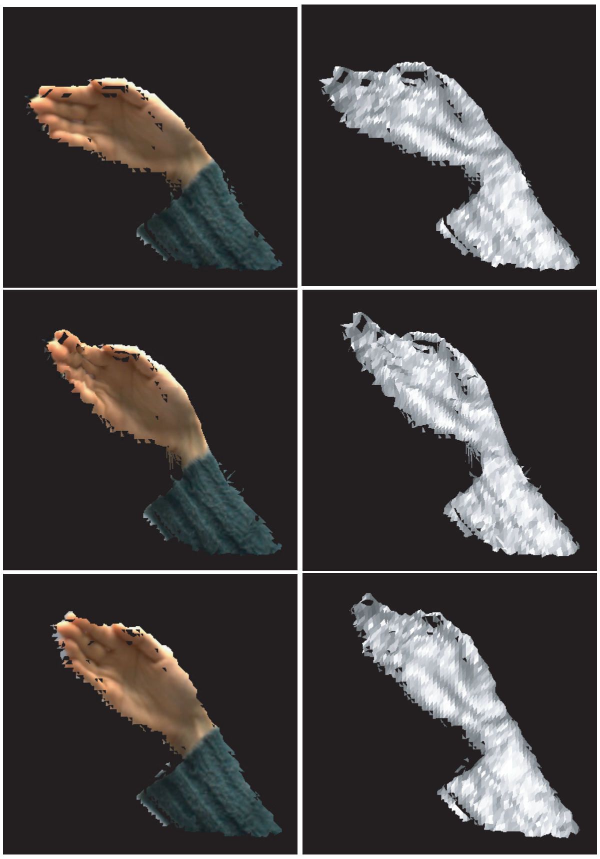


Figure 9. Deformable registration is applied to the range images of a human hand. The range images without texture and the textured range images are the left and right columns, respectively. The range images in the top and bottom rows are used as deforming and reference models, respectively. The deformed model is shown in the middle row.

the ground truth given manually. Figure 7(b) shows the candidates of key points. The light blue points represent texture key-points, and the light pink points are shape key-points. The characteristic points are extracted by both texture and shape features. The deformation by manual input is a translation along the z-axis, which is parallel to the optical axis of the range sensor. The magnitude of translation increases linearly with respect to the distance from the corner of the box. Figure 8 shows slices of the mesh models. The green mesh model is the model before deformable registration. The yellow model is deformed by the transform vectors of ground truth. The white illustrates the model after deformable registration. The red lines show deforming vectors. Since the model after deformation overlaps with the reference model, the deforming vectors are successfully estimated. The RMSs of the deforming vectors are 0.67, 0.83, and 1.6 along x-, y-, and z-axes, respectively, while the mean length of the initial vectors is 7.5cm.

6.2 Deformable Registration of Articulated Object

Next, we tested the deformable registration for a human hand and arm as articulated objects. Figure 9 shows the result of deformable registration when the wrist moves. The wrist is bent while the arm is stationary before and after the motion. The left column shows the range images without texture and the right column designates the textured range images. The range images in the top and bottom rows are used as deforming and reference models, respectively. The deformed model is shown in the middle row.

Next, we apply deformable registration to sequentially captured range images of a hand and an arm Figure 10 and Figure 11 show the results, respectively. The wrist is bending during the motion in Figure 10, and the arm is bending in Figure 11. The hand and arm deformable registration is

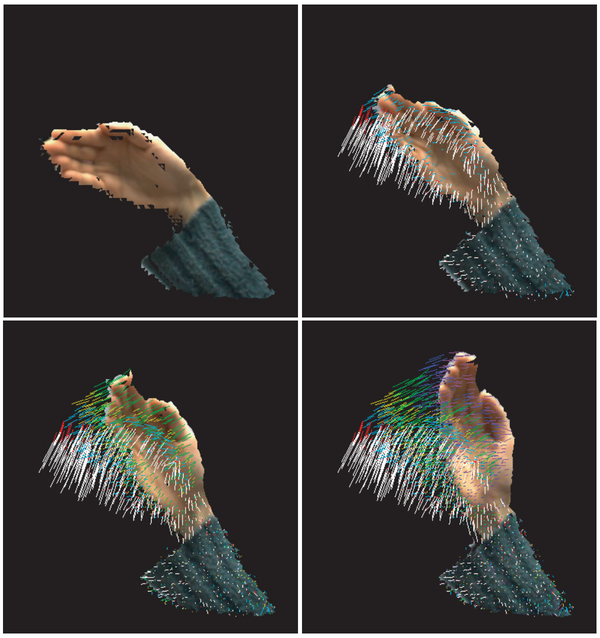


Figure 10. Deformable registration is applied to the motion sequence of a wrist. The deformable registration is applied to 13 sequential frames. The motions of different frames are distinguished by the colors of the lines.

applied to 13 and 17 frames, respectively. The deforming vectors are represented by the lines in the figures. The motions of different frames are distinguished by the colors of the lines. The proposed method successfully estimated the 3D motion of the hand and arm.

7 Conclusion

This paper described a method of deformable registration of textured range images by matching both texture and shape features. First, the method defines the characteristics of each vertex of a 3D mesh model by computing a color histogram for texture feature and the average signed distance for the shape feature. Next, the method extracts key points, which are the distinctive vertices of a model, with respect to the texture and shape features. Next, the method locates the corresponding points by matching the key points between the deforming and reference models. The deforming vectors are computed by minimizing the distance between the corresponding points. The smoothness constraint is used to avoid the effect of wrong correspondence. The proposed method iterates the correspondence search and deformation to gradually align range images. Finally, the deformation for all vertices is computed by interpolating the parameters of the key points.

In the experiments, we obtained textured range images by using a real-time range finder and a camera, and evaluated the deformable registration for range images of a hand

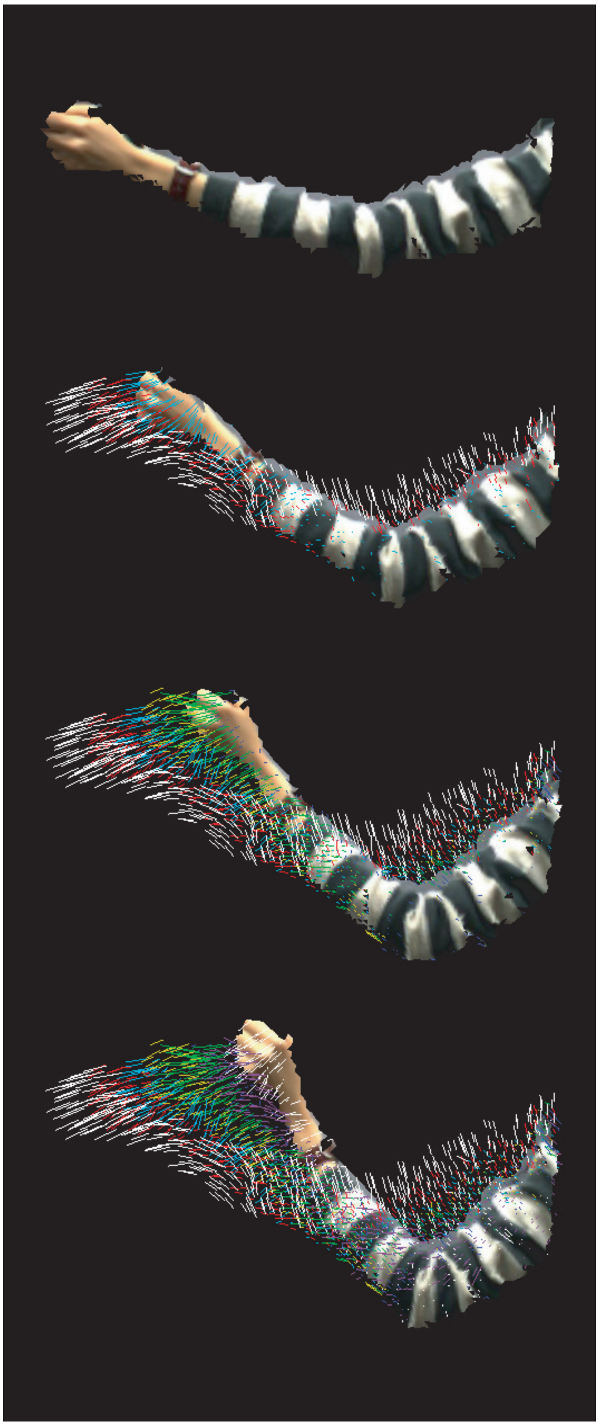


Figure 11. Deformable registration is applied to the motion sequence of an arm. The deformable registration is applied to 17 sequential frames. The motions of different frames are distinguished by the colors of the lines.

and an arm. The proposed method successfully estimated the motion of the articulated objects.

For future work, we plan to use all the vertices of a model to improve the accuracy of the deforming vectors. This additional information will be applied after registration by using the key points. In addition, since the deforming vectors are estimated between two frames in this paper, estimating deforming vectors in a long sequence is an issue that needs to be solved.

References

[1] B. Allen, B. Curless, and Z. Popović. Articulated body deformation from range scan data. *ACM Trans. Graph.*, 21(3):612–619, 2002.

[2] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. on Graphics (TOG)*, 22(3):587–594, 2003.

[3] Z. B. Azouz, C. Shu, and A. Mantel. Automatic locating of anthropometric landmarks on 3d human models. In *Proc. Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[5] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Trans. Patt. Anal. Machine Intell.*, 14(2):239–256, Feb 1992.

[6] N. Brusco, M. Andreetto, A. Giorgi, and G. Cortelazzo. 3d registration by textured spin-images. In *3DIM '05: Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling*, pages 262–269, Washington, DC, USA, 2005. IEEE Computer Society.

[7] C. Chua and R. Jarvis. 3-d free-form surface registration and object recognition. *Int'l Jour. Computer Vision*, 17(1):77–99, 1996.

[8] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.

[9] Q. Delamarre and O. Faugeras. 3d articulated models and multi-view tracking with silhouettes. In *ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2*, page 716, Washington, DC, USA, 1999. IEEE Computer Society.

[10] D. Demirdjian, T. Ko, and T. Darrell. Constraining human body tracking. In *Proc. IEEE International Conference on Computer Vision*, volume 2, pages 1071–1078, 2003.

[11] J. Friedman, J. Bentley, and R. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3):209–226, 1977.

[12] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Symposium on Geometry Processing*, pages 197–206, 2005.

[13] G. Godin, D. Laurendeau, and R. Bergevin. A method for the registration of attributed range images. In *Proc. 3-D Digital Imaging and Modeling (3DIM)*, pages 179–186, 2001.

[14] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[15] G. Hermosillo, C. Chefd'Hotel, and O. Faugeras. A variational approach to multi-modal image matching. *International Journal of Computer Vision*, 50(3):329–343, November 2002.

[16] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(5):433–449, 1999.

[17] S. Knoop and R. Vacek, S.and Dillmann. Modeling joint constraints for an articulated 3d human body model with artificial correspondences in icp. In *Humanoid Robots, 2005 5th IEEE-RAS International Conference on Volume*, pages 74 – 79, Dec. 2005.

[18] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[19] T. Masuda. Multiple range image registration by matching local log-polar range images. In *The 7th Asian Conference on Computer Vision*, pages 948–957, Hyderabad, India, January 2006.

[20] P. Neugebauer. Geometrical cloning of 3d objects via simultaneous registration of multiple range images. In *Proc. Int. Conf. on Shape Modeling and Application*, pages 130–139, Mar 1997.

[21] R. Okada, B. Stenger, T. Ike, and N. Kondoh. Virtual fashion show using real-time markerless motion capture. In *Proc. Asian Conference on Computer Vision*, volume 2, pages 801–810, 2006.

[22] J. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *Proc. the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 822–827, Washington, DC, USA, 2005. IEEE Computer Society.

[23] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511, October 2005.

[24] Y. Rubner, C. Tomasi, and L. Guibas. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision*, 40(2):99–121, 2000.

[25] R. Sagawa, K. Nishino, and K. Ikeuchi. Adaptively merging large-scale range data with reflectance properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):392–405, March 2005.

[26] F. Stein and G. Medioni. Structural indexing: efficient 3-d object recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(2):125–145, 1992.

[27] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):475–480, March 2005.

[28] J. Wyngaerd. Combining texture and shape for automatic crude patch registration. In *3-D Digital Imaging and Modeling, 2003*, pages 179 – 186, October 2003.

[29] G. Wyszecki and W. Styles. Color Science: Concepts and Methods, Quantitative Data and Formulae (2nd ed.). 1982.