3D Free-Form Object Recognition in Range Images Using Local Surface Patches

Hui Chen and Bir Bhanu
Center for Research in Intelligent Systems
University of California, Riverside, California 92521, USA
{hchen, bhanu}@vislab.ucr.edu

Abstract

This paper introduces an integrated local surface descriptor for surface representation and object recognition. A local surface descriptor is defined by a centroid, its surface type and 2D histogram. The 2D histogram consists of shape indexes, calculated from principal curvatures, and angles between the normal of reference point and that of its neighbors. Instead of calculating local surface descriptors for all the 3D surface points, we only calculate them for feature points which are areas with large shape variation. Furthermore, in order to speed up the search process and deal with a large set of objects, model local surface patches are indexed into a hash table. Given a set of test local surface patches, we cast votes for models containing similar surface descriptors. Potential corresponding local surface patches and candidate models are hypothesized. Verification is performed by aligning models with scenes for the most likely models. Experiment results with real range data are presented to demonstrate the effectiveness of our approach.

1. Introduction

This paper addresses recognition of 3D free-form objects from range images. In 3D object recognition, the key problems are how to represent free-form surfaces effectively and how to match the surfaces using the selected representation. In the early years of 3D computer vision, the representation schemes included wire-frame, constructive solid geometry, extended gaussian image, generalized cylinders, planar faces and superquadric. All of these are not highly suitable for representing free-from surfaces.Recently, researchers have been focusing on using local surface properties to represent the shape of objects and recognize free-form objects in range images [3]. They include the splash representation [11], the point signatures [4], the spin image representation [9], the spherical spin image representation [5], the surface point signatures [12], and the harmonic shape images [13].

In this paper, we introduce an integrated local surface descriptor for 3D object representation. We calculate the local surface descriptors only for the feature points which are areas with large shape variation measured by shape index [6]. Our approach starts from extracting feature points from range images, then define the local surface patch as the feature point and its neighbors, next calculate local surface properties which are 2D histogram, surface type and the centroid. The 2D histogram consists of shape indexes and angles between the normal of reference point and that of its neighbors. For every local surface patch, we compute the mean and standard deviation of shape indexes and use them as indexes to a hash table. By comparing histograms and surface types and casting the votes to the hash table, we find the potential corresponding local surface patches and potential models. Finally, we estimate the transformation based on the corresponding surface patches and calculate the match quality between the hypothesized model and test image.

2. Technical approach

• **Feature points extraction:** In our approach, feature points are defined as areas with large shape variation measured by shape index calculated from principal curvatures [6]. In order to estimate curvatures, we fit a quadratic surface $f(x,y) = ax^2 + by^2 + cxy + dx + ey + f$ to a local window and use the least square method to estimate the parameters of the quadratic surface, and then use differential geometry to calculate the surface normal, Gaussian and mean curvatures and principal curvatures [2, 7].

Shape index (S_i), a quantitative measure of the shape of a surface at a point p, is defined by (1) where k_1 and k_2 are maximum and minimum principal curvatures respectively.

$$S_i(p) = \frac{1}{2} - \frac{1}{\pi} tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \tag{1}$$

Within a local window, the center point is marked as a feature point if its shape index S_i satisfies the following condition:

$$\begin{aligned} S_i &= max\ of\ shape\ indexes \text{ and } S_i \geq (1+\alpha) * \mu \\ \text{or } S_i &= min\ of shape\ indexes \text{ and } S_i \leq (1-\beta) * \mu \\ \mu &= \frac{1}{M} \sum_{j=1}^{M} S_i(j) \quad 0 \leq \beta \leq 1 \end{aligned} \tag{2}$$

where α, β parameters controll feature points selection and M is the number of points in the local window. The feature points extraction results are shown in Figure 1 where the feature points are marked by red point sign. From Figure 1, we can clearly see that some feature points corresponding to the same physical area appear in both image.

• **Local surface patches(LSP):** We define a "local surface patch" as the region consisting of a feature point P and

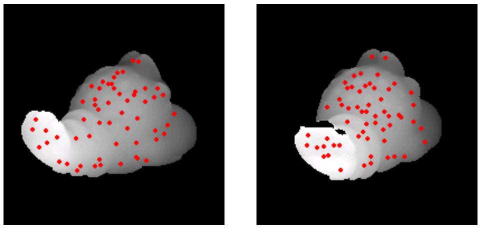


Figure 1. Feature points location in two range images of the same object as gray scale images.

its neighbors N. A local surface patch is shown in Figure 2. The neighbors should satisfy the following two conditions,

$$\begin{aligned} N &= \{pixels \quad N, ||N-P|| \le \epsilon_1\} \\ &\quad \text{and } acos(n_p \bullet n_n) < A, \end{aligned} \qquad (3)$$

where n_p and n_n are the surface normal vectors at point P and N. For every LSP, we compute the shape indexes and normal angles between point P and its neighbors. Then we can form a 2D histogram. One axis of this histogram is the shape index which is in the range [0,1]; the other is the dot product of surface normal vectors at P and N which is in the range [-1,1]. In order to reduce the effect of the noise, we use bilinear interpolation when we calculate the 2D histogram [9].

We also compute the centroid of LSPs. For the feature point, we can get the surface type Tp based on the Gaussian and mean curvatures using (4) [1].

$$Tp = 1 + 3(1 + sgn_{\epsilon_H}(H)) + (1 - sgn_{\epsilon_K}(K)) \qquad (4)$$

where H are mean curvatures and K are Gaussian curvatures. Note that a feature point and the centroid of a patch may not coincide.

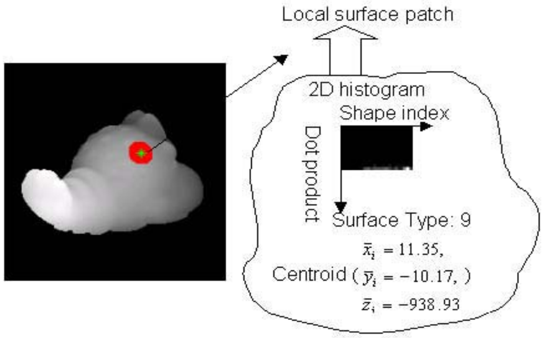


Figure 2. Illustration of Local Surface Patch.

In summary, every LSP is described by a 2D histogram, surface type and the centroid. LSP encodes the geometric information of a local surface.

- **Hash table building:** Considering the uncertainty of location of a feature point, we repeat the above process to calculate descriptor of local surface patches for neighbors of feature point P. To speed up the comparison process, we use the mean and standard deviation of shape index to index a hash table and insert the corresponding hash bin the information (model ID, 2D histogram, surface type, the centroid). Therefore, we save model local surface descriptors into the hash table. For each model object, we repeat the same process to build the model database. The structure of the hash table is explained in Figure 3.

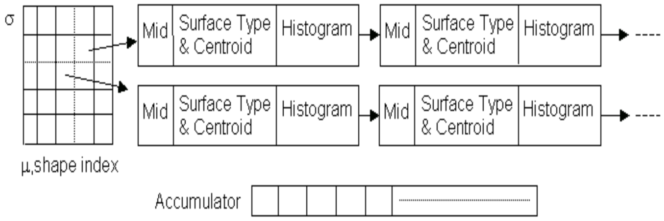


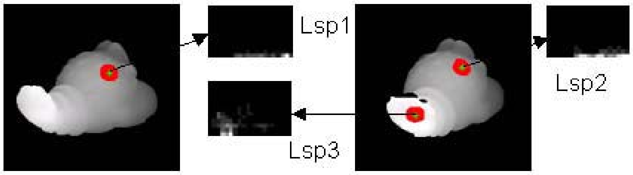
Figure 3. Structure of the hash table.

- **Recognition**
- **(a) Comparing local surface patches:** Given a test range image, we extract feature points and get LSPs. Then we calculate the mean and stand deviation of shape index, and cast votes to the hash table if the histogram dissimilarity falls in a preset threshold ϵ_2 and the surface type is the same. Since histogram can be thought of as an approximation of probability distribution function, it is natural to use the $\chi^2 - divergence$ function (5) to measure the dissimilarity [10].

$$\chi^2(Q,V) = \sum_i \frac{(q_i - v_i)^2}{q_i + v_i} \qquad (5)$$

Where Q and V are normalized histograms.

Figure 4 shows an experimental validation that the local surface patch has the discriminative power to distinguish shapes. First we generate a local surface patch (Lsp1) for the lobster object, then compare it to another local surface patch (Lsp2) corresponding to the same physical area of the same object imaged at different viewpoints. We find that they have a low dissimilarity and the same surface type. However, when Lsp1 is compared to Lsp3 lying in a different area, the dissimilarity is high. The experimental results suggest that the local surface patch can be used for distinguishing objects.



Comparsion of local surface patches

	Mean	Std.	Surface Type	$\chi^2 - divergence$
Lsp1	0.672	0.043	9	χ^2(Lsp1, Lsp2)=0.24
Lsp2	0.669	0.038	9	
Lsp3	0.274	0.019	9	χ^2(Lsp1, Lsp3)=1.91

Figure 4. Experimental validation of discriminatory power of Local Surface Patches.

• **(b): Grouping corresponding pairs of local surface patch:** After voting, we histogram all hash table entries and get models which received the top three highest votes. By casting votes, not only we know which models get higher votes, but also we know the potential corresponding local surface patch pairs. Note that a hash table entry may have multiple items, we choose the local surface patch from the database with minimum dissimilarity and the same surface type as the possible corresponding patch. We filter the possible corresponding pairs based on the geometric constraints given below.

$$d_{C_1,C_2} = |d_{S_1,S_2} - d_{M_1,M_2}| < \epsilon_3, \tag{6}$$

Where d_{S_1,S_2} and d_{M_1,M_2} are Euclidean distance between centroids of two surface patches. For two correspondences $C_1 = \{S_1, M_1\}$ and $C_2 = \{S_2, M_2\}$ where S means test surface patch and M means model surface patch, they should satisfy (6) if they are consistent corresponding pairs. Thus, we use geometric constraints to partition the potential corresponding pairs into different groups. The largest group would more likely to be the true corresponding pair.

Given a list of corresponding pairs $L = \{C_1, C_2, \ldots, C_n\}$, the grouping procedure for every pair in the list is as follows: Initialize each pair of a group. For every group, add other pairs to it if they satisfy (6). Repeat the same procedure for every group. Select the group which has the largest size.

• **(c) Verification:** Our verification procedure has two steps: 1) Estimate rigid transformation; 2) Evaluate the registration results. For the first step, we calculate the rotation matrix and translation vector by using quaternion representation [8]. For the second step, we use match quality defined below to measure how well the test surfaces are aligned with model surfaces.

• **(d) Match quality:** Applying the rigid transformation to the model object, we get a transformed data set. For every point in this dataset, we search the closest point in the test image. If the Euclidean distance between them is less than ϵ_4, they are declared as corresponding points. Thus, we can get the match quality, MQ defined below.

$$MQ = \frac{\#\ of\ corresponding\ points}{\#\ of\ total\ model\ points} \tag{7}$$

It's obvious that MQ is in $[0, 1]$. The larger MQ is, the more likely two objects are. In order to speed up the nearest point search process, we use k-d tree.

3. Experimental Results

• **Data and parameters:** We use real range data taken by Minolta Vivid 700 and we get the data from OSU web site.[1] There are nine objects in our database and they are angel(0), pooh(1),bird(2), buddha(3), dough_boy(4), duck(5), frog(6), lobster(7) and orangedino(8) where the number represents model ID. The average number of 3D points is 12022. The model objects are shown in Figure 5. We apply our approach to the single-object and two-object scenes. The model objects and scene objects are two different views of objects.

The parameters of our approach are $\epsilon_1 = 6.5mm$, $A = \pi/3$, $\epsilon_2 = 0.38$, $\epsilon_3 = 9.4mm$,$\epsilon_4 = 4.8mm$, $\alpha = 0.5$, $\beta = 0.35$, and $\epsilon_H = \epsilon_K = 0.003$. The bin size of the two dimensions of 2D histogram is 0.05 and 0.04 respectively. The average size of local surface patch is 246 pixels.

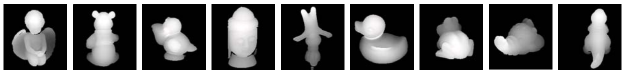


Figure 5. Model objects range images 0-8.

• **Single-object scenes:** In this test case, we show the effectiveness of the voting scheme and discriminating power of LSP in the hypothesis generation. For a given test object, feature points are extracted and the properties of LSPs are calculated. Then LSPs are indexed into the database of model LSPs. For each model indexed, its vote is increased. We show the voting results (shown as a percentage of the number of LSPs in the scene) for the nine objects in Table 1. Note that in some cases the numbers shown are larger than 100 since some models may get more than one vote. We can observe that most of the highest votes go to the correct models. For every test object, we do the verification for the top 3 models which received the highest votes. The verification results are listed in Table 2. We show the relative pose of the test objects and their corresponding models before the registration in Figure 6(a). The pose results after registrations are shown in Figure 6(b).

Test/ Model	0	1	2	3	4	5	6	7	8
0	**92**	0	45	88	11	53	82	61	16
1	18	**75**	13	28	3	13	10	18	28
2	20	0	38	33	13	27	16	33	**40**
3	19	6	16	**132**	18	23	30	18	27
4	0	3	9	0	**31**	9	14	3	0
5	8	6	17	25	0	**80**	37	40	25
6	6	0	25	41	10	67	**87**	65	25
7	11	6	16	20	22	21	40	**86**	28
8	14	15	24	6	8	1	24	13	**104**

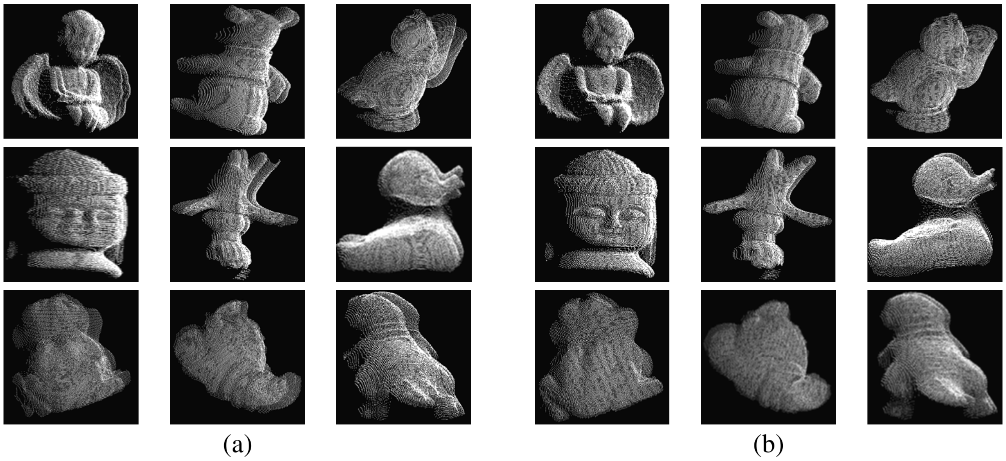
Table 1. Voting results for nine models in the single-object scenes.

• **Two-object scenes:** We created two-object scenes to make one object partially overlap the other object by first properly translating objects and then putting two objects together. Three two-object scenes and recognition results are shown in Table 3. In Table 3, the first scene contains model 2 and 6; the second contains model 3 and 7; the third contains model 1 and 8. It's clearly seen that we recognize objects correctly.

4. Conclusions

We have presented an algorithm for recognition of 3D objects in single-object scenes and two-object scenes. The experimental results show the validity and effectiveness of our algorithm. We show that the local surface patch is a good local surface descriptor, since we can get good corresponding pairs based on comparing local surface patches. Moreover, our approach can deal with a large set of objects since we use hash table to save model information and select candidate models by casting votes to the hash table.

[1] http://sampl.eng.ohio-state.edu/ sampl/data/3DDB/RID/minolta/



(a) (b)

Figure 6. (a)Test objects with the corresponding models before registration. (b)Test objects with the correctly recognized models after registration.

Test objects	Results (Top 3 matches)			
Angel	Model ID	0	3	6
	Match Quality	97.2%		
Pooh	Model ID	1	3	8
	Match Quality	96.7%		19.1%
Bird	Model ID	8	2	7
	Match Quality	8.2%	99.7%	30.8%
Buddha	Model ID	3	6	8
	Match Quality	96.4%	39.8%	25.7%
dough_boy	Model ID	4	6	5
	Match Quality	99.6%		
duck	Model ID	5	7	6
	Match Quality	97.9%	24.1%	43.1%
frog	Model ID	6	7	3
	Match Quality	99.6%	26.4%	7.2%
lobster	Model ID	7	6	8
	Match Quality	91.6%	47.7%	35.6%
orangedino	Model ID	8	7	2
	Match Quality	96.6%	38.5%	48.4%

Table 2. Verification results for single-object scenes.

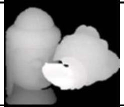
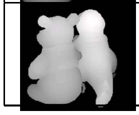
Overlapping objects	Results (Top 3 matches)			
	Model ID	6	7	2
	Percentage votes	62.4	47.7	32.2
	Match Quality	96.8%	33.0%	99.5%
	Model ID	3	7	6
	Percentage votes	83.5	59.0	37.2
	Match Quality	90.1%	91.2%	39.8%
	Model ID	8	1	7
	Percentage votes	115	51.6	46.7
	Match Quality	90.8%	95.2%	60.5%

Table 3. Recognition results for two-object scenes.

References

[1] P. Besl and R. Jain. Segmentation through variable-order surface fitting. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 10(2):167–192, 1988.

[2] B. Bhanu and H. Chen. Human ear recognition in 3D. *Workshop on Multimodal User Authentication*, pages 91–98, 2003.

[3] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Image Understanding*, 81:166–210, 2001.

[4] C. Chua and R. Jarvis. Point signatures: A new representation for 3D object recognition. *International Journal of Computer Vision*, 25(1):63–85, 1994.

[5] S. Correa and L. Shapiro. A new signature-based method for efficient 3-D object recognition. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1:769–776, 2001.

[6] C. Dorai and A. Jain. COSMOS-a representation scheme for free-form surfaces. *Proc. Int. Conf. on Computer Vision*, pages 1024–1029, 1995.

[7] P. Flynn and A. Jain. On reliable curvature estimation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 110–116, 1989.

[8] B. Horn. Close-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America*, 4(4):629–642, 1987.

[9] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999.

[10] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50, 2000.

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

[12] S. M. Yamany and A. Farag. Free-form surface registration using surface signatures. *Proc. Int. Conf. on Computer Vision*, 2:1098–1104, 1999.

[13] D. Zhang and M. Herbert. Harmonic maps and their applications in surface matching. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2:524–530, 1999.