International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:5, 2008

Clustered Signatures for Modeling and Recognizing 3D Rigid Objects

H. B. Darbandi, M. R. Ito, and J. Little

***Abstract*—** This paper describes a probabilistic method for three-dimensional object recognition using a shared pool of surface signatures. This technique uses flatness, orientation, and convexity signatures that encode the surface of a free-form object into three discriminative vectors, and then creates a shared pool of data by clustering the signatures using a distance function. This method applies the Bayes's rule for recognition process, and it is extensible to a large collection of three-dimensional objects.

***Keywords*—**Object recognition, modeling, classification, computer vision.

I. Introduction

THREE-DIMENSIONAL model-based object representation [1] uses the geometry of an object for modeling. In this method, an object's geometric properties and relations are extracted and stored as a model of that particular object. During the matching process, the same procedure is applied to the test object, and its geometric properties and relations are compared against the models for identification purposes. The main goal of all modeling techniques is to extract sufficient object features to enable reliable object recognition during the matching process.

Recently, due to the decreasing cost of 3D scanners, more complex methods that use patches created from dense images have been introduced in the literature. These methods include variety of different techniques such as spin images [2], surface point signatures [3], surface splash [4], harmonic shape contexts [5], flatness and orientation signatures [6], and the tensor method, which models and recognizes 3D objects in cluttered environments [7]. A good survey of different techniques can be found in [8] and [9].

The size of the model created by all the above mentioned modeling techniques increases linearly as more objects are added to the list of the model library. As a result, the efficiency still remains as one of the main problems with these modeling techniques. This problem increases the size of the library model, and makes the recognition process too costly.

In this paper it will be shown that the flatness, orientation, and convexity signatures (FOC signatures) [10] can be shared by different objects in a pool of discriminative signatures. A probabilistic method then can be used to accurately recognize and match the objects on the scene with the library of the models. The size of the model created by clustered signatures is substantially smaller than the original models used in [10], and the recognition results are better.

The rest of this paper is organized as follows: In Section II, the FOC signatures are explained, and set of models used in this paper are described. Section III describes details of the proposed modeling technique. In Section IV, the results and analysis of the experiments are presented and compared with the original method and spin images, and in section V the conclusion and suggestion for further investigation are discussed.

H. B. Darbandi and M. R. Ito are with Electrical and Computer Engineering, University of British Columbia (e-mail: hosseinb@ece.ubc.ca, mito@ece.ubc.ca).

J. Little is with Computer Science, University of British Columbia (e-mail: little@cs.ubc.ca).

II. Background

This paper uses FOC signatures for modeling three-dimensional rigid objects by encoding the fluctuation of the surface and the variation of its normal around an oriented surface point, as the surface expands. In this method, the surface of an object is encoded into three discriminating vectors on each oriented point on the surface of the object termed the flatness, orientation, and convexity signatures. The collection of these three signatures is then used to model and recognize the object.

A. Review

The basic element used to model and recognize an object in [10] is referred to as oriented point. An oriented point is a point (P) on the surface of an object along with its normal (N) at point P.

Consider an oriented point, P, on the surface of an object (please refer to the left-hand column of Fig. 1). Now, assume a sphere with radius R, centered on P. If S is the total area of the object circumscribed inside the sphere, and A is the projection of S on a plane Π, normal to N, then

$$F = \frac{A}{S} \qquad 0 \le F \le 1 \tag{1}$$

F specifies the flatness of the area around an oriented point, P, on the surface. For a flat surface, F is equal to 1; for a curved surface, F is less than one. The more curved the surface is, the lower the value of F would be.

As R increases, the ratio of A/S changes, creating a graph that is the flatness signature of the surface around point P. The flatness signature of a flat area is a horizontal line. Similarly, the angle of the normal, θ, which is equal to the average of the normal of the patches enclosed in the sphere, also

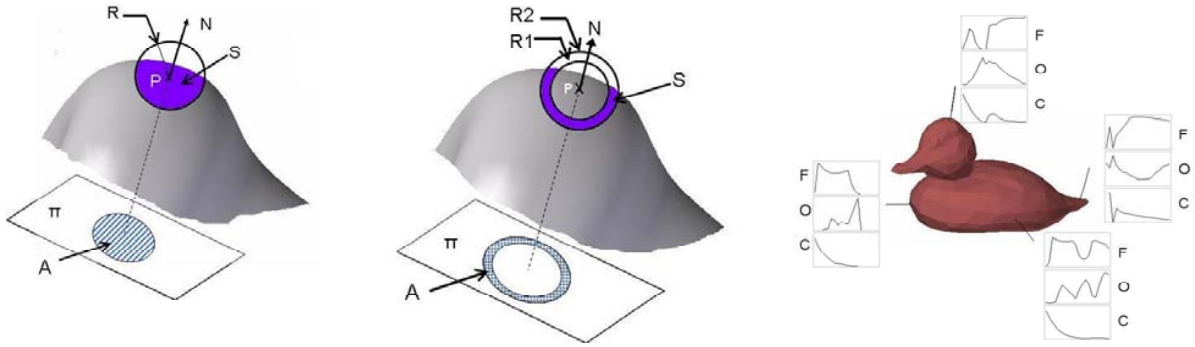

Fig. 1 Modeling. Left: Accumulated method. Middle: Differentiated method. Right: Sample signatures

fluctuates from its original position, N, as R increases, creating another curve, the orientation signature. For a symmetric surface, the orientation signature is a horizontal line if P is set on the symmetrical point on the surface. The combination of orientation signature and flatness signature models the object on point P on the surface of an object. The collection of the signatures is used to model the entire object. Each signature is a vector of *d* elements, in which *d* is the number of intervals used to generate the signature. For each interval, the radius of the sphere is set to $R_i = R_{i-1} + \Delta R$.

To find the signature for each oriented point on a vertex, the normal of the vertex, N, is first calculated by averaging the normal of the surfaces around the vertex. To decrease noise effect, the normal of the patches is averaged inside a sphere with a specific radius called the *base radius*. By finding the normal, N, and Π, the plane normal to N, the two vectors

$$S = [s_i \geq 0 \mid s_1,...,s_d]$$
$$A = [a_i \geq 0 \mid a_1,...,a_d] \quad (2)$$

can be calculated.

The flatness signature, $F = [1 \geq f_i \geq 0 \mid f_1,...,f_d]$, of the oriented point P is then calculated from $F = A/(S+\varepsilon)$. Here, ε is added to each element of *S* to avoid division by zero.

Concurrently, the oriented signature $O = [o_i \geq 0 \mid o_1,...,o_d]$ is found. The collection of *F* and *O* signatures models the surface of the object at the selected vertex. This method is called *accumulative method* [6].

The main problem with this method is that as R increases, the signatures reach a relatively steady state. Consequently, the disparity of the signatures decreases as R increases. To overcome this problem, instead of using the surface of the object circumscribed inside a sphere as R increases, two co-centered spheres are used, as shown in the middle column of Fig. 1. The area circumscribed between two spheres within radius R_2-R_1 is then used for signature creation. This method is called *differentiated method* [10]. The flatness signature in this method is calculated by $F = A/(S+\varepsilon)$ where

$$S = [\Delta s_i = s_i - s_{i-1}, \Delta s_i \geq 0 \mid \Delta s_1,...,\Delta s_d]$$
$$A = [\Delta a_i = a_i - a_{i-1}, \Delta a_i \geq 0 \mid \Delta a_1,...,\Delta a_d] \quad (3)$$

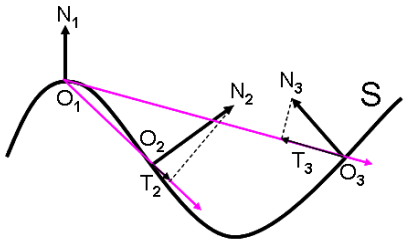

Fig. 2 Convexity and concavity of a sample surface

Experiments show that the signatures created by differentiated method are sensitive to noise. When Δs_i and Δa_i are too small, a small perturbation on the surface of the object caused by noise affects the flatness signature. To remedy this problem, the flatness signatures are smoothed with a high-pass half a bell-shape filter with parameters $\mu = mean(S)$ and $\delta = std(S \mid for\ all\ \Delta s_i < \mu)$ for all the values of S<μ.

B. Convex and Concave Surfaces

The method introduced so far creates identical signatures for both concave and convex surfaces. To distinguish between concave and convex surfaces, the surface is divided into positive and negative patches (convex and concave surfaces respectively) based on the direction of their normal relative to an oriented point. Consider the cross-section of surface S in Fig. 2. O_1 is a point on the surface, and N_1 is its normal. Let us assume we are interested in finding the signatures of the surface relative to oriented point O_1. To find the convexity and concavity of the surface on points O_2 and O_3 relative to O_1, connect O_1 to O_2 and O_1 to O_3 to create two vectors, O_1O_2 and O_1O_3. Then find the projection of N_2 and N_3 on O_1O_2 and O_1O_3, T_2 and T_3, respectively. If the direction of O_1O_x and T_x are the same, then the surface at point O_x is convex relative to point O_1. If the direction of O_1O_x and T_x are opposite, then the surface at point O_x is concave relative to O_1. For example, relative to the oriented point O_1, the surface at O_2 and O_3 are convex and concave, respectively.

In addition to flatness and orientation signatures, the total convex and concave area at each step of signature creation are

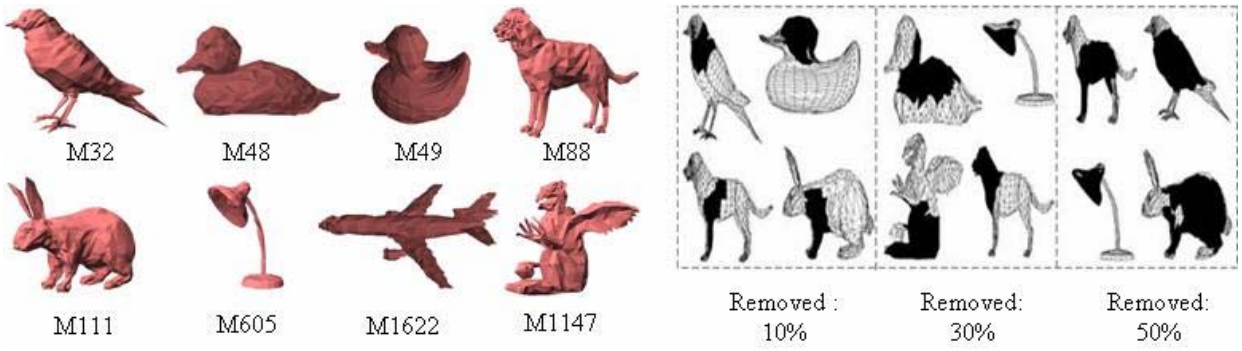

Fig. 3 Left: Test objects. Right: Surface of the test objects removed form 10% up to 50% of their total surface

stored separately. The calculation of convex and concave areas has no major effect on processing time since the same procedures are used for calculating flatness and orientation signatures.

The convexity signature is calculated by dividing the convex area to the total area at each step of signature creation. The convexity can hold a value between one and zero. A value of one for convexity means that the area being considered for signature creation is a complete convex area. A value of zero means that the area is a complete concave or a flat area. Convexity signatures, along with flatness and orientation signatures, are used to model the surface around an oriented point. A similar filter used to smooth fatness signatures is applied to convexity signatures to reduce the noise effect. Please refer to the right-hand column of Fig. 1 for sample FOC signatures on the surface of an object.

C. Models and Setting

The Princeton benchmark [11] models (Fig. 10) were chosen as experimental models to test our approach. The models were scaled to have a maximum dimension equal to 200 units for consistency. Normal noise with standard deviation equal to 1 was added to each vertex in the normal direction of the surface of the models to create noisy test models (please refer to the left-hand column of Fig. 3).

To demonstrate the robustness of the signatures to clutter, a large portion of the noisy test models were removed by selecting a random patch as seed patch on the surface of the test object, and surface patches totaling up to 50 percent of the total surface of the object were removed. This is a valid assumption for cluttered surfaces, since subtracting and adding extra surface have similar effect on the signatures created by the proposed method.

$$\text{Clutter} = \frac{\text{surface of the object cluttered}}{\text{total surface of the object}}$$

The right-hand column of Fig. 3 shows sample test objects that their surfaces were removed up to 50 percent of their total surface area.

Signatures created with the proposed modeling technique depend on two parameters that limit the modeling area of a surface around an oriented point. The supporting distance limits the value of R. Small values of R model the local deformation around point P, and large values of R model the global deformation of the surface relative to the oriented point P. Due to occlusion, the entire object cannot be seen from a single viewpoint. It is logical to assume that if the normal of a patch makes an angle greater than a threshold angle with the orientation of point P, it cannot be seen from the same angle that sees point P. Therefore, those patches cannot be used for modeling the object from that viewpoint. These two parameters are called *support distance* and *support angle*, respectively [12].

The direction of the normal of the patches is an important factor in creating the signatures. The normal should point to outside of the object to create consistent signatures. We determined the direction of the normal of the patches by assuming different view-points far from the object, and then the directions were calculated by a method similar to ray tracing method [13].

D. Effect of Sampling Intervals on Signatures

It was shown in [10] that FOC signatures are robust to scale, orientation, occlusion and clutter, patch resolution (sampling rate), and they tolerate noise. Our experiments indicate that FOC signatures are also robust to sampling intervals (ΔR).

Different sampling intervals (ΔR) create different flatness and concavity signatures. To deal with this problem, instead of storing flatness and concavity signatures, their components were stored as modeling parameters, and then the signatures were calculated from these components. Flatness and convexity signatures are calculated as

$$Flatness = \frac{A}{S} = \frac{A_{convex} + A_{concave}}{S_{convex} + S_{concave}}$$

and

$$Convexity = \frac{S_{convex}}{S_{convex} + S_{concave}}$$

where A_{convex} , $A_{concave}$, S_{convex} , and $S_{concave}$ are A and S components of convex and concave surfaces. Fig. 4 shows the four components of a sample vertex on the surface of M48.

In turn, A and S components of differentiated signatures can be calculated from A and S components of accumulative signatures (please refer to equations 2 and 3). As a result,

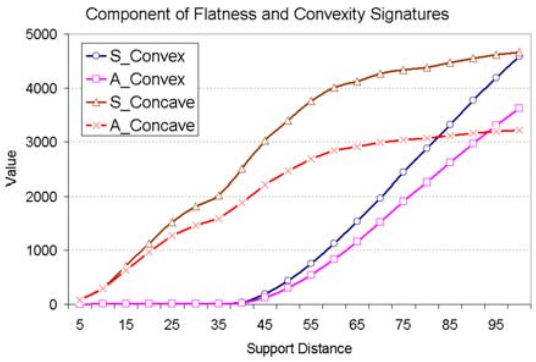

Fig. 4 Components of flatness and convexity signatures of a sample vertex on the surface of M48

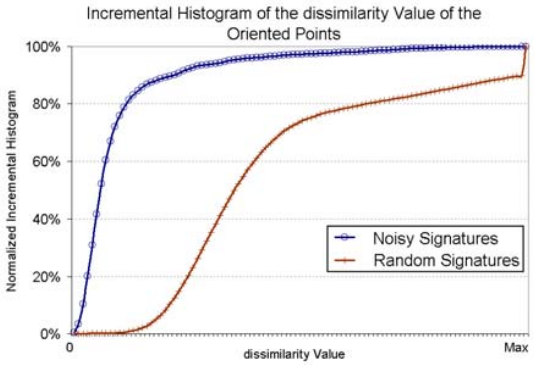

Fig. 5 Dissimilarity of the oriented points

instead of storing flatness and convexity signatures, four discriminative vectors of the accumulative method, A_{convex} , $A_{concave}$, S_{convex} , and $S_{concave}$ were stored as components of the signature of the model on each oriented point. Since the values of the components increases continuously, they can be re-sampled accurately within any intervals. The flatness and convexity signatures of both accumulative and differentiated methods are then calculated from these components. The components of orientation signatures cannot be stored. However, experiments indicated that orientation signatures can be down-sampled safely. Note that orientation signatures cannot be up-sampled.

III. Proposed Method

The authors in [10] used normalized cross correlation to compare signature of the models. However, the experiments indicate that the simple proposed distance measurement works much faster and provide better measurement to compare the signatures of the models.

The distance of two signatures (S_1 and S_2) is calculated as their normalized Euclidean distance

$$dis(s_1,s_2) = \frac{1}{d}\|S_1 - S_2\| \quad (4)$$

where d is the dimension of the vectors S_1 and S_2.

The dissimilarity of the oriented points is calculated from;

$$dissimilarity = (dis(f_1, f_2)^2 + dis(o_1,o_2)^2 + dis(c_1,c_2)^2)^{1/2} \quad (5)$$

where f_x, o_x, and c_x are the FOC signatures of the oriented point x.

A. Discriminatory Power of the Signatures

To study discriminatory power of the signatures, random vertices were selected on the surface of the test objects, and then their signatures were compared with the signatures of the original model on the same oriented points using equation (5). For comparison purposes, random vertices were also selected on the surface of the library models and their signatures were compared with the signatures of other randomly selected vertices.

Fig. 5 shows the incremental histogram of the *dissimilarity* of the oriented point where *Noisy Signatures* are the signatures of the noisy-test models compared with the signatures of the original models, and *Random signatures* are the signatures of randomly selected vertices and compared with other randomly selected signatures.

As indicated in the figure, the *dissimilarity* creates a distinctive measurement for comparison. For example, while about 80% of the signatures of the noisy models compared with the signatures of the original models have a dissimilarity less than a threshold value, only about 0.7% of the *Random signatures* have a dissimilarity less than the same threshold value.

B. Sharing Signatures

Experiments indicate that oriented points close to each other on the surface of an object have similar FOC signatures. Furthermore, similar surfaces have similar FOC signatures. As a result, FOC signatures can be shared to model a variety of objects. The signatures were clustered by using a method similar to BIRCH [14]. Since we do not know the number of the clusters, we set a threshold for grouping similar signatures in a cluster. To calculate the threshold value, we referred to our experiments in the previous Section. The threshold value to cluster the signatures was set to the median distance of the noisy signatures compared with the original signatures for each set of F, O, and C signatures.

$$threshold = median(dis(noisy\_signaures, original\_signatures))$$

The threshold was used as the maximum inter-cluster distance between the signatures of the clusters [15].

Here the common covariance matrix (Σ) was used for the clusters. The shared covariance matrix was calculated by the following equation

$$\Sigma = \frac{\sum_i n_i \sigma_i}{\sum_i n_i} \qquad (\forall \sigma_i \mid \det(\sigma_i) \neq 0)$$

where σ_i is the covariance matrix of cluster i, and n_i is the number of the signatures in the cluster.

The top-row of Fig. 6 shows the number of the clusters versus the number of the signatures for each set of F, O, and C signatures. As indicated in the figure, the number of clusters does not increase linearly as more signatures are added to the list of the signatures.

The middle-row of the figure shows the histogram of the signatures per clusters. As indicated in the figure, half of the signatures were grouped in clusters with 22 signatures and more, and another half of the signatures were grouped into clusters with 22 signatures per cluster or less. In our experiments, since objects were symmetrical, clusters with one signature were deleted as outliers.

Finally, the bottom-row of the figure shows the number of the clusters versus number of the distinctive models in each cluster. For example, as shown in the figure, there were 2482 clusters that each cluster was shared by signatures of four different models.

C. Object Recognition

The probability of an object given signature *s* from a list of *n* models is calculated by Bayes' rule:

$$P(O_n \mid s) = \frac{P(s \mid O_n)P(O_n)}{\sum_n P(s \mid O_n)P(O_n)} \tag{6}$$

Since each signature may come from any cluster, then

$$P(s \mid O_n) = \sum_i P(s \mid C_i)P(C_i)$$

where C_i is cluster i, and

$$P(s \mid C_i) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(s-\mu_i)^{\mathrm{T}}\Sigma^{-1}(s-\mu_i)\right]$$

$$P(C_i) = \frac{r_{mi}}{N_m}$$

r_{mi} is the number of the signatures of model m in cluster i, and N_m is the total number of the signatures of model m in all clusters. Since the same probability can be assumed for all objects, then equation (6) becomes

$$P(O_n \mid s) = \frac{\sum_i P(s \mid C_i)P(C_i)}{\sum_n \sum_i P(s \mid C_i)P(C_i)} \tag{7}$$

Since there are three different types of signatures and clusters, the probability of model O_n given signature *foc* is equal to

$$P(O_n \mid foc_j) = \log \prod_{s=1}^{3} P(O_n \mid s_j) = \sum_k \log P(O_n \mid s_j) \tag{8}$$

where s_j is one of the *f*, *o*, and *c* signatures of sample *j*.

Since one outlier with the probability equal to zero in equation (8) affects all further calculations, the zero probability was substituted by the lowest probability calculated for all clusters in equation (8).

For N sample points selected on the surface of the test object, the probability of object O_n given signature set *foc* is equal to

$$P(O_n) = \log \prod_{j=1}^{N-1} P(O_n \mid foc_j) = \sum_j \log(P(O_n \mid foc_j))$$

Since the lowest probable sample point was eliminated as a possible outliner for each model, the number of the sample points reduced to *N-1*. The model with the highest probability was chosen as the possible candidate model.

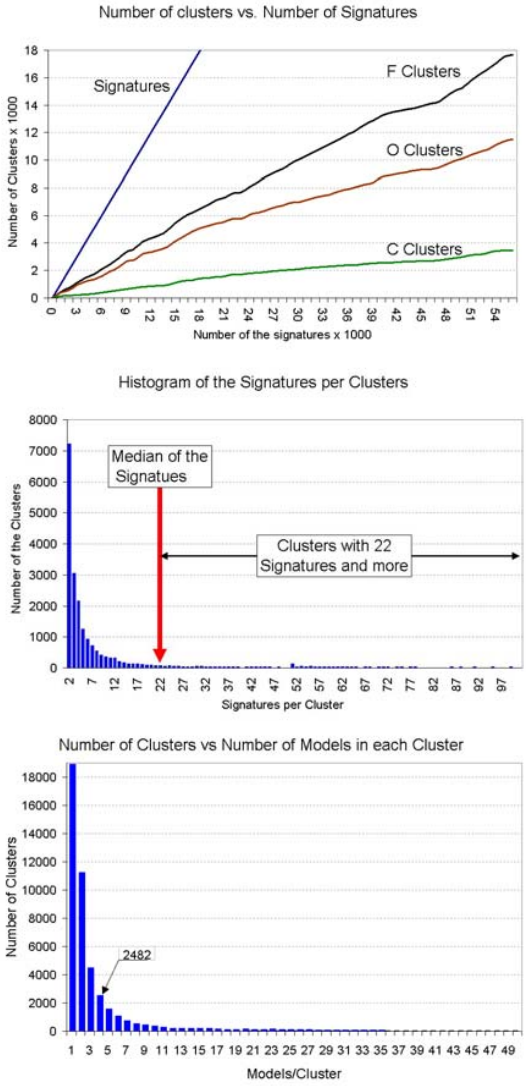

Fig. 6 Top: Number of the clusters versus number of the signatures. Middle: Histogram of number of the signatures per clusters. Bottom: Number of the clusters vs. number of the models shared each cluster

D. Geometric Verification

Because geometric consistency between the object and the candidate model is not being checked, there is possibility that the positive candidate model selected in the previous section may not be the correct match. To make the recognition more flexible, the top M most probable candidates were selected and passed on for geometric verification.

For the geometric verification process, corresponding oriented points on the surface of the candidate models are needed for each set of *foc* signatures. Since the flatness signatures are more descriptive than the orientation and convexity signatures, the corresponding oriented points were selected from the two most probable clusters of the flatness signatures. In this way, there may be a handful of corresponding points (r_{mi} may be large) for each candidate

model, so the corresponding oriented points were clustered based on their distance and the angle of their oriented points for geometric verification process. This process is necessary to limit number of the samples particularly if the models were sampled with a fine resolution. The clustering process reduces the number of the sample points to a manageable number. In our experiments, we have clustered the sample points only if their distance was less than 5 units and their angle was less than $\pi/18$.

To verify the oriented points geometrically, the same method used to verify spin images [16] is used. Assume that S_x and m_y represent a match between the point *x* on the scene to the point *y* on the surface of a model. Now, let $C_1 = [s_1, m_1]$ and $C_2 = [s_2, m_2]$ be two correspondences between the scene and the model. The geometric consistency between two sets of the matches is calculated as follows:

$$d(P_1, P_2) = \frac{\left\| S_{m_2}(m_1) - S_{s_2}(s_1) \right\|}{\left\| S_{m_2}(m_1) + S_{s_2}(s_1) \right\|}$$

where

$$S_p(q) = (\alpha, \beta) = \left(\sqrt{|p-q|^2 - (n \cdot (p-q))^2}, n \cdot (p-q) \right)$$

where *p* and *q* are two points in 3D, and *n* is normal at point *p*.

Since *d* is not symmetric, the maximum of the $d(p_1, p_2)$ is used to define the geometric consistency, *D*.

$$D = \max(d(P_1, P_2), d(P_2, P_1))$$

When D is small, P_1 and P_2 are geometrically consistent that scene and model points in P_1 and P_2 are the same distance apart, and their surface normal form the same angle with each other.

All the matched oriented points with their D greater than a threshold, *k*, were eliminated as outliners. In our experiments *k* was set from 0.25 to 0.11.

E. Registration

For registration process, only three oriented points are needed to estimate the transformation matrix, since three points and three independent normal are enough to calculate transformation [17].

The remaining matches between test object and model were then triangulated using their D value from the previous section. Since, this is a simple look-up table, the process is fast and efficient. The oriented points, which are triangulated, are sorted based on their D value.

$$D_{triangulated} = D_{12} + D_{13} + D_{23}$$

Several random triangulated matches with lowest *D* values were selected and used for initial estimation of transformation matrix. Since calculation of transformation while keeping $|R| = 1$ and $RR^T = 1$ is a known problem, quaternion [18] was used to calculate the rotation matrix, R. From there, the k-nearest neighbor method [19] was used to register the test object on the scene to the model.

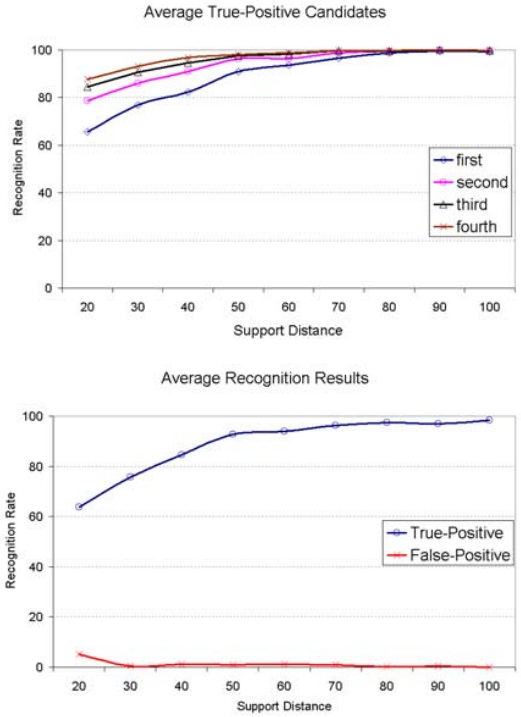

Fig. 7 Average of recognition results for noisy test models. Top: Positive candidate selection. Bottom: True-positive recognition rate

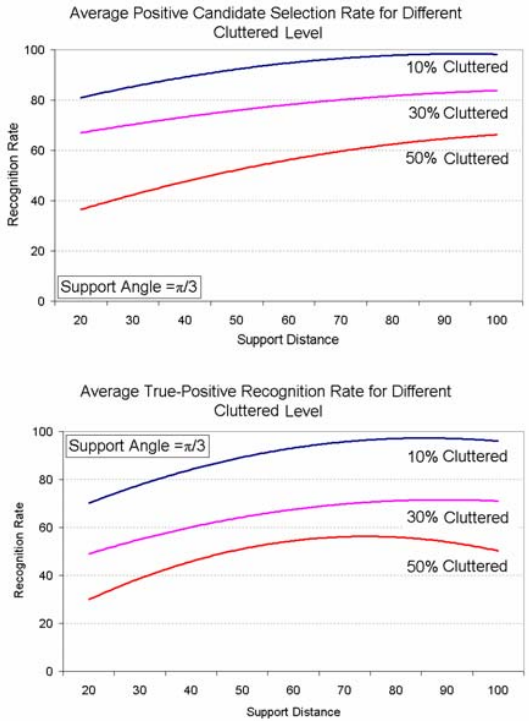

Fig. 8 Average of recognition results for cluttered objects. Top: Positive candidate selection. Bottom: True-positive recognition rate

IV. Experiments

In our experiments the support distance was set from 20 to 100 with intervals of 10, the support angle was set to $\pi/3$, and the base-radius was set to 10. Then six random oriented points were selected on the surface of the noisy models, and their FOC signatures were created. The signatures were then compared and matched to the library models shown in Fig. 10 that consists of 69 toys including 55406 vectors for each F, O, and C descriptive signatures. The signatures were grouped into 9906, 7060, and 2523 clusters for F, O, and C signatures respectively. Each noisy model was tested for 50 recognition times for each support distance.

A. Results

The top-row of Fig. 7 shows the average of positive candidate selection from first to fourth candidates. As indicated in the figure, by extending the candidate selection to second and more, the positive recognition rate increases. The bottom-row of the figure shows average true-positive recognition results when the average alignment error after registration process falls below a threshold value which was set to 10 units in our experiments. The false-positives are the cases where similar objects were recognized as the target model.

Fig. 8 shows the same results for cluttered objects. The top-row of the figure shows the average of positive candidate selection for different levels of clutter, and the bottom-row of the figure shows the average true-positive recognition results when the alignment error falls below 10 units. As indicated in the figure, by increasing clutter, the recognition rate decreases.

Since sample points near the cluttered area are not reliable for matching purposes, in recognition experiments of the cluttered objects, nine sample points were selected randomly to eliminate the effect of the sample points selected near the cluttered area.

In our experiments parameter M was set to 4. By increasing value of M, the recognition rate of the cluttered objects significantly improves.

B. Comparison

Since spin images and original differentiated method were used by a similar approach for matching and recognition process (pleaser refer to [10] and [12]), the spin images of the test objects and models were created with exactly the same parameters for both modeling techniques. The support distance was set to $\pi/3$, and the bin of spin images set to ΔR. Each spin image was a histogram of 20 x 40 bins.

The recognition results of the proposed method were compared with the recognition results of the original differentiated method and those of spin images. The sample points selected were the same in all experiments, and experiments conducted with exactly the same parameters.

The top-row of Fig. 9 shows the recognition rate gained by the proposed method comparing with the original differentiated method and the method used by spin images.

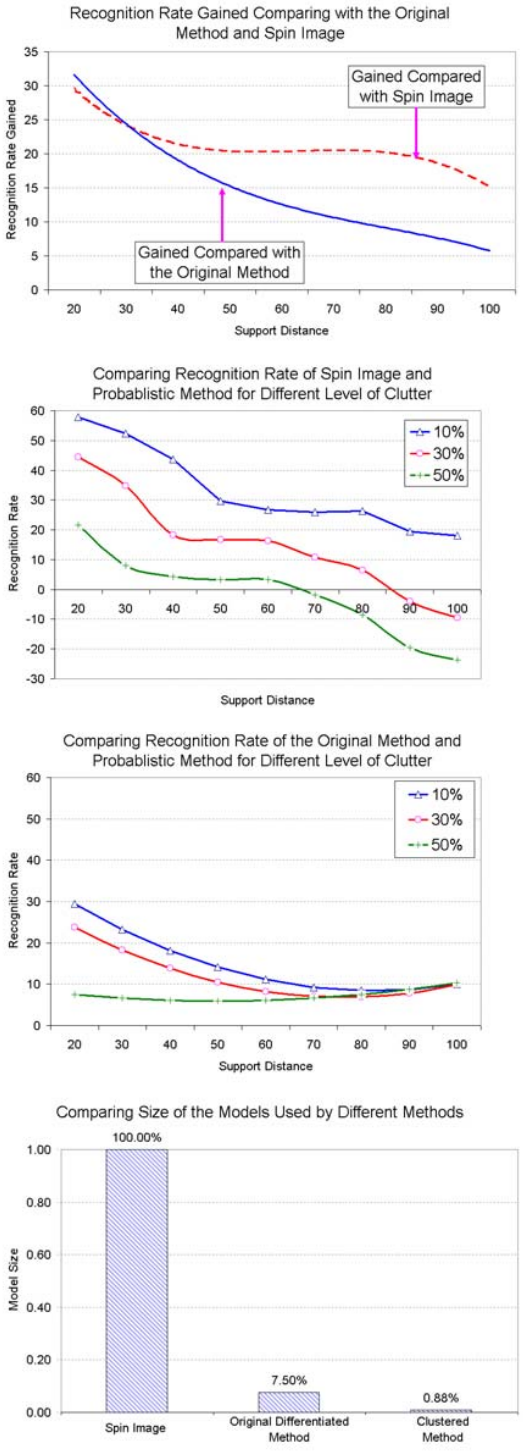

Fig. 9 Top-row: Recognition gained by clustered signatures. Second-row: comparing recognition rate of the proposed technique and spin image for different level of cluttering. Third-row: comparing recognition rate of the proposed technique and the original model for different level of cluttering. Bottom-row: Comparing model size used by different modeling techniques

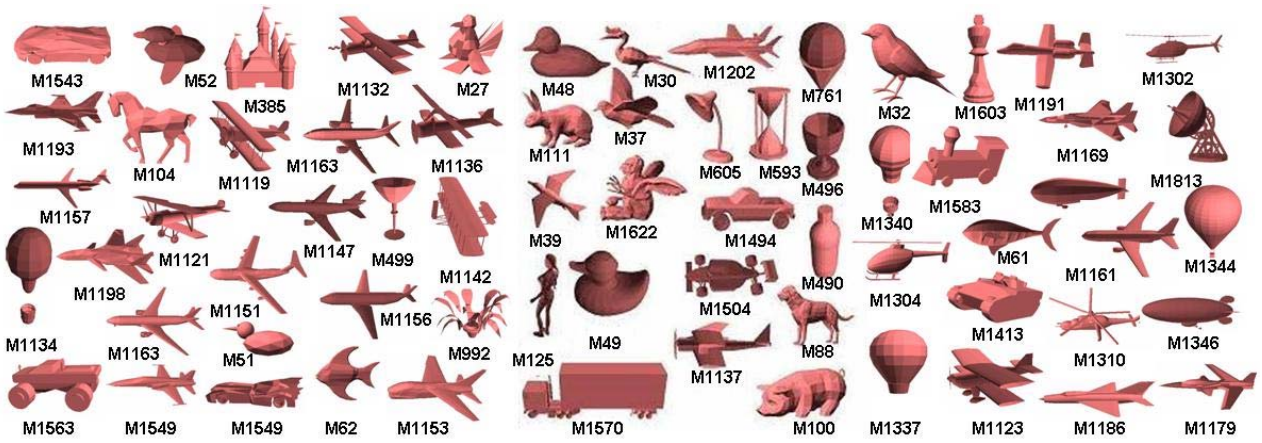

Fig. 10 Library models used in the experiments

Each graph is the result of subtracting the recognition rate of the differentiated method and spin images from recognition rate of the proposed method. The solid curve in the figure shows gained achieved in recognition rate comparing with the recognition rate of the original differentiated method, and the dotted-curve shows the recognition rate gained by proposed method comparing with the recognition rate of the spin images.

The figure in the second-row shows difference of the true-positive recognition rate between the proposed method and the spin image for different level of clutter. As indicated in the figure, the recognition rate of the proposed method is considerably better than the recognition rate of the spin image for low clutter level and for small support distances. However, the recognition rate of the spin image is better for higher clutter level and large support distance. The reason is that, the FOC signatures created near cluttered area are not descriptive enough for matching purpose. One possible solution is to select sample points far from cluttered area rather that random selection.

The figure in the third-row shows difference of the true-positive recognition rate between the proposed method and the original method for different level of cluttering. As indicated in the figure, the proposed method shows better recognition rate for cluttered area than the original differentiated method.

Finally, the bottom-row of the Fig. 9 compares the model size created and used by each method. As indicated in the figure the size of the original differentiated method was 7.5% of the size of the spin image, and the size of the clustered signatures was only 0.88% of the size of the spin image. However, the recognition rates of the clustered signatures are better than the recognition rates of the differentiated method and those of the spin image.

V. Conclusion

In this paper it is shown that a variety of 3D objects can be modeled with shared data, and that objects can be recognized reliability during the matching process. We chose only six sample points on the surface of the test object and nine sample points on the surface of the cluttered objects for recognition process. By increasing the number of the sample points on the surface of the test objects recognition rate increases.

The method presented in this paper is suitable for multiple object recognition. However, if the target object is known, then the recognition results increases further than the results shown in this paper. Our experiments indicate that by adding more objects to the library models, the number of the clusters increases as a polynomial of order 2, in which tends to reach to a saturated level as number of the signatures tends to infinity.

Our experiments also indicate that the FOC signatures are suitable for rigid object classification. However, this finding needs further investigation.

References

[1] Bennamoun, G.J. Mamic, "Object Recognition: Fundamentals and Case Studies," Springer, 2000.
[2] A.E. Johnson, Spin Image: "A Representation for 3D Surface Matching. PhD Thesis," Carnegie Mellon University, 1997.
[3] S. Correa, L. Shapiro, "A New Signature-Based Method for Efficient 3D Object Recognition," Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1: 769-776, 2001.
[4] Stein, F.; Medioni, G.; "Structural indexing: Efficient 3D object recognition of a set of range views," IEEE transactions on pattern analysis and Machine Intelligence, 17(4):344-359, 1995.
[5] S.M. Yamany, A. Farag, "Freeform Surface Registration Using Surface Signatures," Proc. Int. Conf. on Computer Vision, 2: 1098-1104, 1999.
[6] H.B. Darbandi, M.R. Ito; J. Little, "Flatness and Orientation signature for Modeling and Matching 3D Objects," Third International Symposium on 3D Data Processing, Visualization and Transmission, 2006.
[7] Mian, A.S.; Bennamoun, M.; Owens, R. "Pattern Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes," Analysis and Machine Intelligence, IEEE Transactions on Volume 28, Issue 10, Oct. 2006 Page(s):1584 – 1601.
[8] R.J. Campbell, P.J. Flynn, "A Survey of Free-Form Object Representation and Recognition Techniques," Computer Vision and Understanding, vol. 81, pp. 166-210, 2001.
[9] B.M. Planitz, A.J. Maeder, J.A. Williams, "The correspondence framework for 3D surface matching algorithms," Computer Vision and Image Understanding, 2005.

[10] H.B. Darbandi, M.R. Ito, J. Little, "Surface Signature-Based Method for Modeling and Recognizing Free-Form Objects," 3rd International Sysposium on Visual Computing, 2007.
[11] http://shape.cs.princeton.edu/benchmark/
[12] A. Johnson; M. Herbert, "Using spin images for efficient object recognition in cluttered 3D scenes," IEEE Tr. on Pattern Analysis and Machine Intelligence. Vol. 21, N0 5. May 1999.
[13] J.D. Foley, A. van Dam; S.K. Feiner; J.F. Hughes, "Introduction to Computer Graphics," Addison-Wesley, 1993.
[14] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: "An Efficient Data Clustering Method for Very Large Databases," SIGMOD Conference 1996: 103-114.
[15] E. Alpaydin, "Introduction to Machine Learning," MIT Press, 2004.
[16] A. Johnson, M. Herbert, "Using spin images for efficient object recognition in cluttered 3D scenes," IEEE Tr. on Pattern Analysis and Machine Intelligence. Vol. 21, N0 5. May 1999.
[17] D.A. Forsyth, J. Ponce, "Computer Vision, a Modern Approach," Prentice Hall, 2003.
[18] J. B. Kuipers, "Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace and Virtual Reality," Princeton University Press, 1993.
[19] H. Samet, "Depth-first k-nearest neighbor finding using the Maximum Nearest Distance estimator," IEEE Transactions, Image Analysis and Processing, 2003.