3D Recognition and Segmentation of Objects in Cluttered Scenes

A. S. Mian, M. Bennamoun and R. A. Owens
School of Computer Science and Software Engineering
The University of Western Australia
{ajmal,bennamou,robyn}@csse.uwa.edu.au

Abstract

In this paper we present a novel view point independent range image segmentation and recognition approach. We generate a library of 3D models off-line and represent each model with our tensor-based representation. Tensors represent local surface patches of the models and are indexed by a 4D hash table. During the online phase, a seed point is randomly selected from the range image and its neighbouring surface is represented with a tensor. This tensor is simultaneously matched with all the tensors of the library models using a voting scheme. The model which receives the most votes is hypothesized to be present in the scene. The model from the library is then transformed to the range image coordinates. If the model aligns accurately with a portion of the range image, that portion is recognized, segmented and removed. Another seed point is picked from the remaining range image and the matching process is repeated until the entire scene is segmented or no further library objects can be recognized in the scene. Our experiments show that this novel algorithm is efficient and it gives accurate results for cluttered and occluded range images.

1. Introduction

The aim of range image segmentation is to accurately identify the boundaries of 3D objects or regions of interest and separate them from the rest of the dataset. However, the problem of segmentation is ill posed and the existing definitions of segmentation do not guarantee a unique segmentation of an image. A major application of range image segmentation includes object recognition and classification. In another paradigm, object recognition in a cluttered scene can be used for 3D segmentation of a range image. We adopt this paradigm in our approach which results in the simultaneous segmentation of a range image and the recognition of free-form [3] objects. The main challenges in 3D object recognition and range image segmentation are the presence of occlusions (including self occlusions and occlusions caused by other objects) and clutter (due to noise and unwanted objects) which are usually present in real scenes.

The following is a brief review of some of the existing segmentation techniques. Deformable models [5][14][19] based segmentation techniques are computationally expensive and require a manual interaction to position an initial model in the dataset. These techniques also require the manual selection of their initial parameters. Moreover, to the best of our knowledge, deformable models have been used for 2D and 3D volumetric image segmentation only and have not been applied to range image segmentation. Atlas guided segmentation techniques, for instance [7][24], have only been used in the case of 3D volumetric medical images. The range image segmentation algorithm of Pulli and Pietikainen [25] is limited to planar and smoothly curved objects. 3D edge detection based segmentation techniques [2] suffer from the difficulty of extracting the regions of interest from the detected edge maps. Moreover, these algorithms do not perform well in case of low depth variations between regions. LEGION (Locally Excitory Globally Inhibitory Oscillator Networks) [30] based segmentation techniques [18] are computationally expensive. Mathematical morphology [26] based segmentation algorithms [1][27] are not automatic and require a user defined criterion to control the erosion and dilation operations. Region growing segmentation algorithms [16] are sensitive to the selection of seed points and occlusions. Different seed points may result in different segmentations of the same range image and occlusions can cause the segmented region to include false holes. Most of these techniques have not been tested on scenes containing free form objects [3]. Moreover, these techniques only perform segmentation and are not applicable to object recognition.

A brief review of existing work in the area of 3D object recognition includes the following. Dorai's COSMOS representation [8] does not work in occluded scenes and calculates the principle curvatures which are sensitive to noise. Stein's structural indexing algorithm [28] is not applicable to free-form objects. SAI matching [11] is only applicable to objects which are topologically equivalent to a sphere. B-Spline curve matching techniques [6][29] suffer from the knot problem i.e. the positions of knots for a given B-Spline curve are not unique. HOT curves [15] based recognition is not robust to noise as it relies on the accurate localiza-

Proceedings of the Seventh IEEE Workshop on Applications of Computer Vision (WACV/MOTION'05)
0-7695-2271-8/05 $ 20.00 IEEE

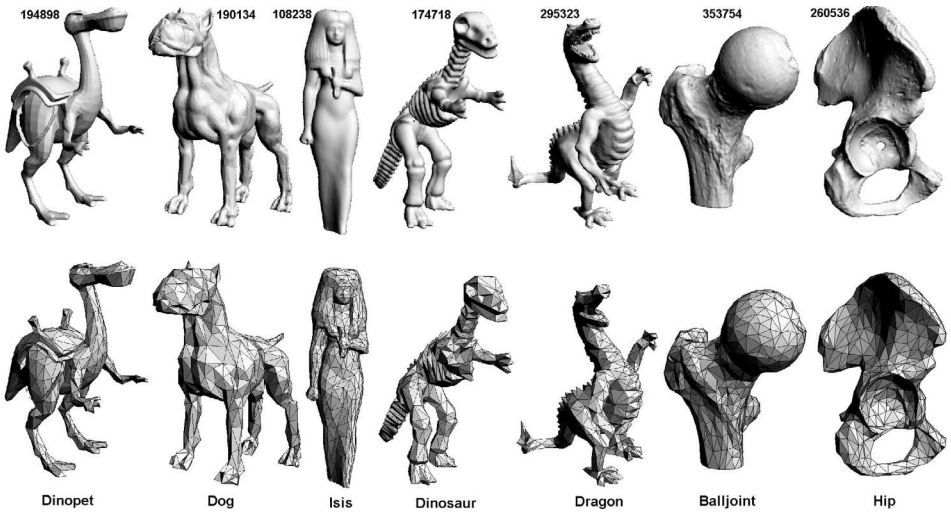

Figure 1. First row: High resolution models. The number of faces of each model in the first row is written on its top. Second row: Low resolution models at 1200 faces per model.

tion of inflection points which are themselves sensitive to noise. Moreover, all these techniques assume the presence of a single object in the scene and have not been tested under clutter. Johnson's recognition algorithm [13] works well in cluttered and occluded scenes however it requires a uniform mesh resolution for the library models and the scene. Moreover, most of these algorithms use a one-to-one matching strategy in which case the recognition time grows linearly with the size of the model library.

In this paper, we present a novel automatic 3D recognition based segmentation algorithm applicable to complex range images containing clutter and occlusions. The algorithm is fully automatic and does not require any manual intervention. Briefly our algorithm proceeds as follows. During an offline phase a 3D model library of objects is built along with their tensor representations [20]. A 4D hash table (variant of [17]) is also constructed from the model tensors for quick indexing. During the online phase, a seed tensor is generated at a randomly selected point in the range image. The seed tensor is used along with the hash table to cast votes to matching tensors in the model library. The model receiving maximum votes is transformed to the coordinates of the range image. If the model aligns accurately with a portion of the range image, that portion is recognized and segmented. The segmented portion is removed and the process is repeated for further segmentation and recognition of the remaining range image.

2. Construction of the Model Library

3D models of objects which are likely to be present in the scene are stored in a model library along with their tensor representations. Our algorithm is equally effective with high and low resolution models. Therefore, in order to gain memory and computational efficiency, models are stored at a significantly low resolution in the database. Fig. 1 shows the high resolution models and their corresponding low resolution models which are obtained by applying a mesh simplification algorithm [10] to the high resolution models. The high resolution models were built with our automatic 3D modeling algorithm [20][21] (range data courtesy of the University of Stuttgart [12]). Only the low resolution models were stored in our model library. The high resolution models are shown only for illustration purposes.

Along with the 3D models, their tensor representations (Section 2.1) and a hash table (Section 2.2) are also stored in the model library. Each tensor represents a local surface patch of a model by quantizing the surface area into a 3D cubic grid. These tensors are then used to build a 4D hash table for quick indexing. [22] contains all the necessary details of our tensor representation and the construction of the hash table used in the context of multiview correspondence for 3D modeling. However, we still believe that a brief discussion is also necessary here for completeness and in order to fully describe our algorithm.

2.1. The Tensor Representation

First, normals are calculated for each vertex of a model (which is in the form of a triangular mesh Fig. 2(a)). Next, its vertices are paired such that two vertices in any pair satisfy the following two constraints. First, their mutual distance should be between d_{min} and d_{max}. Second, the angle θ_{def} between their normals should be less than 60^o. There are two advantages to these constraints. First, they ensure that the vertices in any pair are visible from a single viewing angle. Second, they avoid the C_2^n combinatorial explosion of vertices (where n is the total number of vertices in a

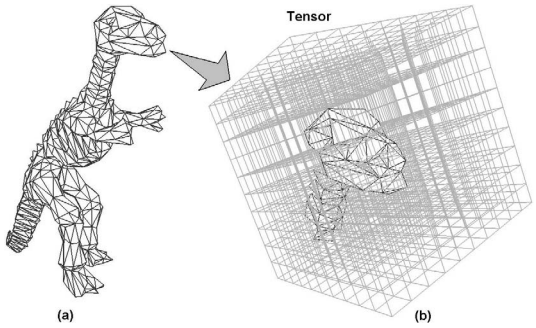

Figure 2. A 10^3 grid defined on the head of the dinosaur of Fig. 1. Only the visible triangular faces contribute toward the tensor corresponding to this grid.

mesh). d_{min} should be selected large enough to reduce the sensitivity of the tensor computation to noise. In our experiments we selected d_{min} equal to twice, and d_{max} equal to three times the average resolution of the models.

A vertex is allowed to participate in a maximum of three pairs and a total of $n_t = 800$ vertex pairs per model are selected such that they uniformly cover the 3D model. Each pair is then used to define a local 3D basis at their middle point (the origin). The average of the normals of the two vertices makes the z-axis, their cross product makes the x-axis and the cross product of the z-axis with the x-axis makes the y-axis. This basis is used to define a 3D grid centered at the origin (see Fig. 2(b)). Two parameters need to be chosen for this purpose: the number of bins in the grid and the size of each bin. Choosing more bins will enclose more surface inside the grid. The size of an individual bin governs the level of granularity at which the surface is represented. We selected a 10^3 grid based on an extensive number of surface matching experiments [21]. The bin size was set to half the mean resolution of the library models.

Next, the surface area of the mesh crossing each bin of the grid is computed and stored in a third order tensor. Each element of the tensor is equal to the mesh surface area that is present inside its corresponding bin in the 3D grid (Fig. 2(b)). The area of intersection of the mesh with the grid bins is calculated using Hodgman's polygon clipping algorithm [9]. A polygon inside the grid is considered for contributing toward the computation of a tensor only if its normal makes an angle of less than 90^o with the z-axis of the grid. Most of the elements of the tensors are zeros therefore these tensors are reduced to sparse arrays in order to reduce memory utilization by approximately 85%.

2.2. Hash Table Construction

The tensors of the models are used to fill up a 4D geometric hash table (variant of [17]). Three dimensions of the hash table correspond to the i, j, k indices of the tensor elements whereas the fourth dimension is defined by θ_{def} of the tensors. θ_{def} is quantized into bins of $\Delta\theta_{def}$. Choosing a lower $\Delta\theta_{def}$ will reduce the number of possible matches for a tensor but will also increase the risk of missing a correct match due to noise and sampling errors in θ_{def}. During our multiview correspondence experiments [23] we found that a $\Delta\theta_{def} = 5^o$ gives good results. The 4D hash table is filled up as follows. For each tensor of every model, the tuple (tensor number, model number) entry is made in all the bins of the hash table corresponding to the i, j, k indices of the non-zero elements of the tensor and its θ_{def}.

3. 3D Recognition and Segmentation

During the online phase, the range image of the scene is converted into a triangular mesh and normals are calculated for all its vertices. A list of seed points (approx. 500) is then selected from the scene on a uniform 2D grid. Next, a seed point is picked up at random from this list and is paired with another seed point which satisfies the distance and angle constraints of Section 2.1. A tensor $\mathbf{T}_s$ is then calculated from these points and the i, j, k indices of its occupied bins and its θ_{def} are used to cast votes to the tuples present at the i, j, k,θ_{def} index position in the 4D hash table. The tuples that receive less votes than half the total occupied bins of $\mathbf{T}_s$ are discarded. Next, the correlation coefficient C_c of the tensors of the remaining tuples with $\mathbf{T}_s$ is calculated using Eqn. 1. C_c is calculated in the region of overlap of $\mathbf{T}_m$ and $\mathbf{T}_s$ to cater for occlusions.

$$C_c = \text{correl coeff}(\mathbf{T}_m(I_{ms}), \mathbf{T}_s(I_{ms})) \quad (1)$$

In Eqn. 1, I_{ms} is the intersection of the non-zero elements of the model tensor $\mathbf{T}_m$ and the scene tensor $\mathbf{T}_s$. $\mathbf{T}_m(I_{ms})$ and $\mathbf{T}_s(I_{ms})$ are the values of the model and scene tensors respectively, in their region of overlap. The tuples whose tensors' C_c are below a threshold t_c are discarded and the remaining tuples are then sorted according to their decreasing values of C_c with the $\mathbf{T}_s$. t_c can either be calculated dynamically from the C_c values of the tuples or chosen to be a constant value. We experimented with both a fixed value of $t_c = 0.5$ as well as calculated it dynamically (mean C_c of all tuples). The latter approach gave better results by eliminating many false positives at this stage.

The remaining list of tuples are hypothesized as the possible matches of the scene tensor and verified one by one as follows. The model tensor $\mathbf{T}_m$ (in each tuple) and the scene tensor $\mathbf{T}_s$ are used to transform the corresponding model to the scene coordinates using the rotation matrix $\mathbf{R}$ (Eqn. 2) and the translation vector $\mathbf{t}$ (Eqn. 3).

$$\mathbf{R} = \mathbf{B}_m^{\top}\mathbf{B}_s \quad (2)$$

$$\mathbf{t} = \mathbf{O}_s - \mathbf{O}_m\mathbf{R} \quad (3)$$

In Eqn. 2, $\mathbf{B}_x$ is the matrix of coordinate basis of the model or scene tensor. In Eqn. 3, $\mathbf{O}_x$ is the origin vector of tensor $\mathbf{T}_x$. After transforming the model to the scene coordinates, the surface match is verified by refining the regis-

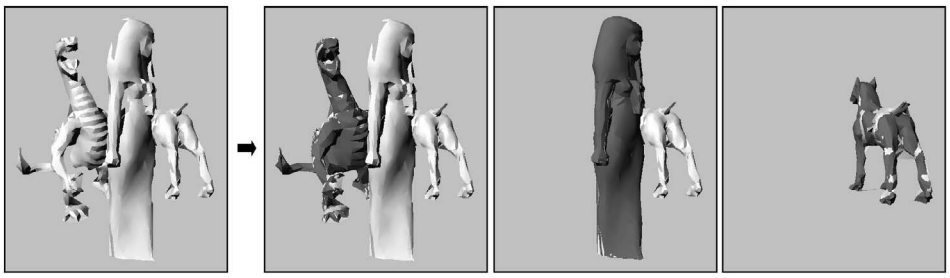

Figure 3. Trace of the automatic 3D segmentation and recognition algorithm. First the dragon is recognized and its model (shown in dark shade) is transformed from the model library and aligned with the dragon in the scene. This results in the 3D segmentation of the dragon. Its data points are removed from the scene in the next iteration. Then the isis and finally the dog are recognized and segmented in the range image.

tration with the ICP algorithm [4]. If the model and scene surfaces have a significant overlap, the algorithm proceeds to the next verification step otherwise the next hypothesis is tested. In case all the hypotheses fail, another seed point is randomly selected from the list and the above process is repeated. In the next verification step, the number of model points that are transformed in the region between the scene surface and the sensor or into the free space of the sensor are counted. If a significant number of such points are found, the hypothesis is rejected. If the number of such points is small, the hypothesis is accepted. This results in the 3D segmentation of the object in the scene as well as its recognition. Additionally, the pose (location and orientation) of the object in the scene is also calculated from $\mathbf{R}$ and $\mathbf{t}$.

After the recognition and segmentation of an object in the range image, all its data points are removed and another set of seed points is picked up from the remaining scene. This process is repeated until the range image is completely segmented or no further library objects can be recognized in the scene. Fig. 3 shows the trace of our algorithm for a range image of three objects namely the dragon, the isis and the dog. The recognized model (shown in dark shade) from the library is aligned with the object in the scene. This results in the 3D segmentation of the object in the scene. The data related to a segmented object is removed from the range image in the next iteration in order to facilitate the recognition of the remaining objects in the scene.

4. Results

We used the low resolution model library of Fig. 1 (second row) in our experiments. We generated synthetic scenes by placing different views of the models in a z-buffer. We were able to generate scenes with varying topologies including disjoint objects, touching objects, objects occluding other objects etc. Next, we applied our automatic segmentation algorithm to these objects. Fig. 3 and Fig. 4 show the results of our algorithm when applied to a typical cluttered scene. The segmented objects are shown in dark shade. All the objects are accurately segmented in each case by transforming and aligning their corresponding 3D models (shown in dark shade) from the library. Notice that the segmentation is performed in 3D i.e. the boundaries of objects are identified in 3D. These figures also illustrate that the range images of the scenes have missing data due to self occlusions and occlusions caused by other objects, however after the segmentation using our approach, this missing data is completed by super-imposing the 3D models over their corresponding objects in the scene. To illustrate the 3D segmentation, the segmented range images have been shown from three different angles in Fig. 4 (row 2, 3 and 4).

In addition to high resolution range images (Fig. 4), we also tested our algorithm at a low resolution (Fig. 3 and Fig. 5). The resolution of the models in the library was also different from the range images in each case. The results show that our algorithm is independent of the resolution and surface sampling of the range images.

5. Conclusion

We presented a novel 3D recognition and segmentation algorithm. Our algorithm is memory efficient as it requires models at a significantly low resolution. Efficiency in terms of time is achieved by matching a single tensor simultaneously with all the tensors in the database using a hash table. Our algorithm is independent of the surface sampling and resolution of the range images since it matches surface patches (represented with tensors) as opposed to data points. We performed 3D segmentation of the range images as opposed to a 2D segmentation. The algorithm was tested on complex scenes containing clutter and occlusions and our results show that it is accurate, efficient, applicable to free-form objects and is robust to clutter and occlusions.

6. Acknowledgment

The authors would like to thank Carnegie Mellon University for providing mesh reduction software. This research is sponsored by ARC grant number DP0344338.

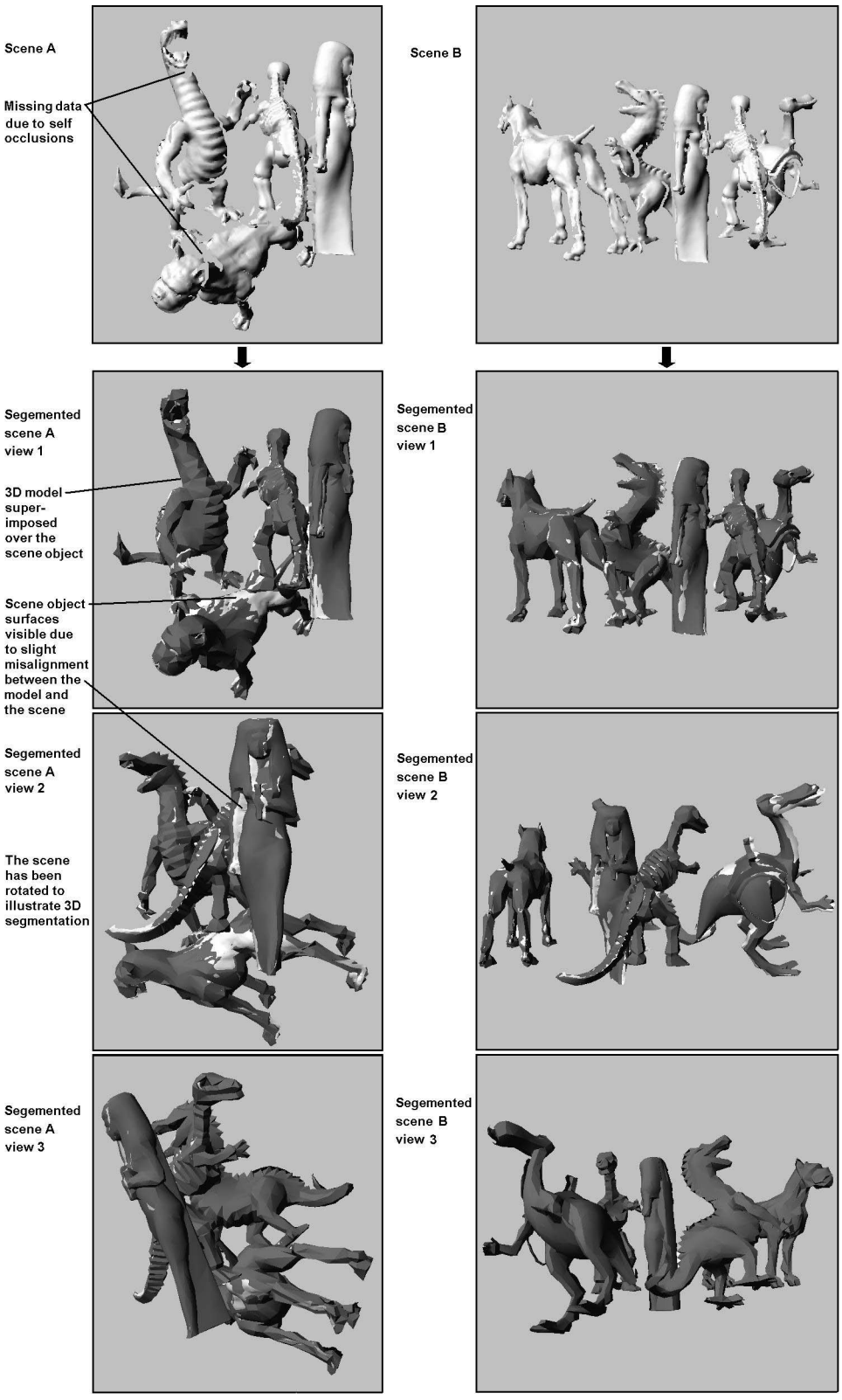

Figure 4. Two different range images to be segmented (first row). Each segmented range image is shown from three different angles (row 2, 3 and 4).

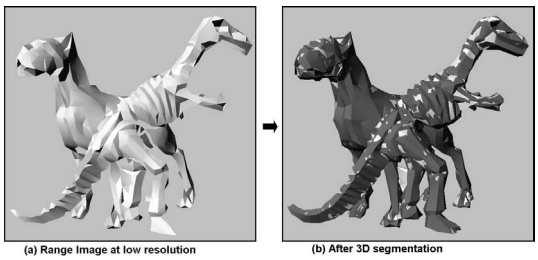

Figure 5. (a) A scene range image at low resolution (1200 faces). (b) The models (shown in dark shade) from the library are accurately aligned with their corresponding objects (shown in light shade) in the scene.

References

[1] M. Baccar, L. Gee, R. Gonzalez and M. Abidi, "Segmentation of Range Images via Data Fusion and Morphological Watersheds," *Pattern Recognition*, Vol. 29(10), pp. 1673-1687, 1996.

[2] O. R. P. Bellon and L. Silva, "New Improvements to Range Image Segmentation by Edge Detection," *IEEE Signal Processing Letters*, Vol. 9(2), pp. 43-45, 2002.

[3] P. Besl, "Machine Vision for Three-dimensional Scenes," *Academic Press*, pp. 25-71, 1990.

[4] P. J. Besl and N.D. McKay, "Reconstruction of Real-world Objects via Simultaneous Registration and Robust Combination of Multiple Range Images," *IEEE TPAMI*, Vol. 14(2), pp. 239-256, 1992.

[5] L. D. Cohen and I. Cohen, "Finite Element Methods for Active Contour Models and Balloons for 2D and 3D Images," *IEEE TPAMI*, Vol. 15(11), pp. 1131-1147, 1993.

[6] F. S. Cohen and J. Wang, "Part I: Modeling Image Curves Using Invariant 3-D Object Curve Models - A Path to 3-D Recognition and Shape Estimation from Image Contours," *IEEE TPAMI*, Vol. 16(1), pp. 1-12, 1994.

[7] M. B. Cuadra, C. Pollo, A. Bardera, O. Cuisenaire and J. P. Thiran, "Atlas-Based Segmentation of Pathological Brain MR Images," *IEEE ICIP*, pp. 573-576, 2003.

[8] C. Dorai and A. K. Jain, "COSMOS: A Representation Scheme for 3D Free-Form Objects," *IEEE TPAMI*, Vol. 19(10), pp. 1115-1130, 1997.

[9] J. Foley, A. van Dam, S. Feiner and J. Hughes, "Computer Graphics-Principles and Practice," *Addison-Wesley*, 1990.

[10] M. Garland and C. Heckbert, "Surface Simplification Using Quadric Error Metrics," *SIGGRAPH*, pp. 209-216, 1997.

[11] M. Hebert, K. Ikeuchi and H. Delingette, "A Spherical Representation for Recognition of Free-Form Surfaces," *IEEE TPAMI*, Vol. 17(7), pp. 681-690, 1995.

[12] G. Hetzel, B. Leibe, P. Levi and B. Schiele, "3D Object Recognition from Range Images using Local Feature Histograms," *IEEE CVPR*, Vol. 2, pp. 394-399, 2001.

[13] A. E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," *IEEE TPAMI*, Vol. 21(5), pp. 674-686, 1999.

[14] S. Joshi, S. Pizer, P. T. Fletcher, P. Yushkevich, A. Thall and J. S. Marron, "Multiscale Deformable Model Segmentation and Statistical Shape Analysis Using Medical Descriptions," *IEEE TMI*, Vol. 21(5), pp. 538-550, 2002.

[15] T. Joshi, J. Ponce, B. Vijayakumar and D. Kriegman, "Hot Curves for Modeling and Recognition of Smooth Curved 3D Objects," *IEEE CVPR*, pp. 876-880, 2002.

[16] K. Koster and M. Span, "MIR: An Approach to Robust Clustering - Application to Range Image Segmentation," *IEEE TPAMI*, Vol. 22(5), pp. 430-444, 2000.

[17] Y. Lamdan and H. Wolfson, "Geometric Hashing: A General and Efficient Model-based Recognition Scheme," *IEEE ICCV*, pp. 238-249, 1988.

[18] X. Liu and D. L. Wang, "Rang Image Segmentation Using a Relaxation Oscillator Network," *IEEE TPAMI*, Vol. 10(3), pp. 564-573, 1999.

[19] T. McInerney and D. Terzopoulos, "A Dynamic Finite Element Surface Model for Segmentation and Tracking in Multidimensional Medical Images with Application to Cardiac 4D Image Analysis," *Computerized Medical Imaging and Graphics*, Vol. 19(1), pp. 69-83, 1995.

[20] A. S. Mian, M. Bennamoun and R. A. Owens, "Matching Tensors for Automatic Correspondence and Registration," *ECCV*, part 2, pp. 495–505, 2004.

[21] A. S. Mian, M. Bennamoun and R. A. Owens, "Performance Analysis of an Improved Tensor Based Correspondence Algorithm for Automatic 3D Modeling," *IEEE ICIP*, 2004.

[22] A. S. Mian, M. Bennamoun and R. A. Owens, "A Novel Algorithm for Automatic 3D Model-based Free-form Object Recognition," *IEEE SMC*, 2004.

[23] A. S. Mian, M. Bennamoun and R. A. Owens, "From Unordered Range Images to 3D Models: A Fully Automatic Multiview Correspondence Algorithm," *TPCG*, pp.162–166, 2004.

[24] K. Pohl, S. Bouix, R. Kikinis and W. E. Grimson, "Anatomical Guided Segmentation with Non-stationary Tissue Class Distributions in an Expectation-Maximization Framework," *IEEE Int. Symposium on Biomedical Imaging*, 2004.

[25] K. Pulli and M. Pietikainen, "Range Image Segmentation Based on Decomposition of Surface Normals," *Scandinavian Conf. on Image Analysis*, Vol. 2, pp. 893-899, 1993.

[26] J. Serra, "Image Analysis and Mathematical Morphology," *Academic Press*, 1982.

[27] D. Shiwei and Y. Baozong, "Range Image Segmentation Using Mathematical Morphology," *IEEE Tecon*, Vol. 2, pp. 1009-1011, 1993.

[28] F. Stein and G. Medioni, "Structural Indexing: Efficient 3-D Object Recognition," *IEEE TPAMI*, Vol. 14(2), pp. 125-145, 1992.

[29] J. Wand and F. S. Cohen, "Part II: 3-D Object Recognition and Shape Estimation from Image Contours Using B-Splines, Shape Invariant Matching, and Neural Network," *IEEE TPAMI*, Vol. 16(1), pp. 13-23, 1994.

[30] D. Wang and D. Terman, "Locally Excitory Globally Inhibitory Oscillator Networks," *IEEE TNN*, Vol. 6(1), pp. 283-286, 1995.