RECOGNIZING 3D OBJECTS IN CLUTTERED SCENES USING PROJECTION IMAGES

Dimitris Zarpalas, Georgios Kordelas and Petros Daras

Informatics and Telematics Institute
1st Km Thermi-Panorama Road, P.O. Box 60361, 57001 Thessaloniki, Greece
Email: {zarpalas, kordelas, daras}@iti.gr

ABSTRACT

This paper presents a novel descriptor for recognizing objects in highly occluded and cluttered 2.5D scenes produced by range scans. This new compact regional shape descriptor, called "projection images", is designed to be robust against noise, partial occlusion and clutter. Projection images are formed by "projections" of points onto the plane centered at the basis point which is perpendicular to the viewing axis. Multiple experiments were performed on a dataset of 50 range scans, each one comprised of multiple objects, which proved that the proposed method is robust and efficient to a satisfactory degree of occlusion and clutter, while it compared favorably against descriptors previously introduced in the literature.

Index Terms— Projection Image, range scan, feature extraction, local shape descriptor

1. INTRODUCTION

In recent years, significant progress has been made toward the recognition of free-form 3D objects from their 2.5 D counterparts. The aim of object recognition systems is to correctly identify objects in a scene. Noise, partial occlusion and clutter are the main obstacle such a system should overcome. Approaches utilizing plain 2D cameras are fast and low cost, yet they are very sensitive to illuminations, shadows and occlusions and do not provide accurate estimation of object's pose.

Thus, there is an increasing number of approaches that utilize range scanners in order to limit down such effects. Algorithms that extract local descriptors, such as surface curvatures [1], are proven to be unstable and sensitive to noise [4]. Moreover, the method in [5], which is based on point signatures, is unstable on noisy data, and sensitive to surface sampling [4]. Johnson and Hebert proposed the spin images [2], which where very influential in this field. However, spin images are vulnerable to sampling and resolution (level-of-detail) of the models and when spin images are compressed, the average recognition rate decreases significantly. In [3] an enhancement of the spin images algorithm is presented by using vertex interpolation. Although this change resolved the sensitiveness of spin images to variations in resolution, it's discriminative power was not improved significantly. Other works, [7, 8], enhanced spin images by performing some post-processing or other matching methods. Spherical harmonics [10] and locality-sensitive hashing [11] are exploited in [9] to perform recognition by 3D shape information obtained from laser range scanners. Mian et al. [4], proposed a tensor-based surface representation defined on pairs of oriented points. Their descriptors are 3D tensors that measure the variation of surface position. Correspondence between 3D tensors is established using a voting process to find pairs of tensors with high overlap ratio. The method in [15] use distance maps to perform the object recognition task. Matching between scene's and object's distance maps is established achieved using the SIFT algorithm [14] on greyscale images that are generated from the distance maps. The algorithm presented in [12] calculates the local surface properties of patches, which are defined at the extracted feature points. By comparing local surface patches of a model and a test image, and casting votes for the models containing similar surface descriptors, the potential corresponding local surface patches and candidate models are hypothesized. The evaluation experiments were simple, since at most two objects existed in the scene. In [13], the generalized Hough transform is extended to detect instances of an object model in laser range data, independently to the scale and orientation of the object. However, this method is restricted to simple objects that can be represented with few parameters, such as planes, spheres and cylinders.

The plethora of the existing algorithms [7, 8, 3] either modify the spin image [2] or integrate it with other components so as to improve its performance. Additionally, most methods [4, 2, 3, 15] require the point cloud, which is generated from a range scanner, to be converted to mesh before object recognition takes place. Thus, there is a need for the development of new methods that address the object recognition problem in a more direct way.

The rest of this paper is organized as follows: in Section 2 *Projection Images* are introduced. Section 3 presents the object recognition procedure. Experimental results are given in Section 4. Finally, conclusions are drawn in Section 5.

This work was supported by the EU funded project "3D VIVANT", GA-248420.

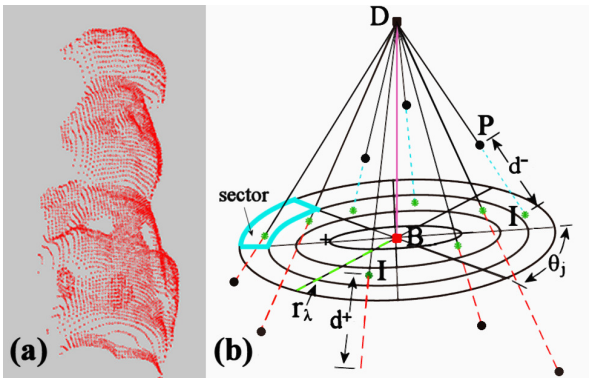

Fig. 1. (a) Generated point cloud from a specific viewpoint and (b) Illustration of projection images computation: $\boldsymbol{D}$ is the cameras viewpoint, $\boldsymbol{B}$ the point that serves as the basis point, $\boldsymbol{P}$ the neighboring points, where the dotted red segments represent the d^+ and the blue the d^- .

2. PROJECTION IMAGES

A point cloud of a scanned object $\boldsymbol{V}_o$ (Fig.1(a)), is created when a range scanner is placed at point $\boldsymbol{D}$. Having $\boldsymbol{D}$ and a basis point $\boldsymbol{B} \in \boldsymbol{V}_o$, a circular support region Π_B of radius R, is defined on the plane that is perpendicular to line segment $\boldsymbol{DB}$. For each neighboring point $\boldsymbol{P}$, the "projection" point is found, i.e. the point $\boldsymbol{I}$ where line $\boldsymbol{DP}$ intersects Π_B. Then, the distance between $\boldsymbol{I}$ and $\boldsymbol{P}$ is computed, which is considered positive d^+ or negative d^- depending on which side of the projection image plane the point lies (Fig.1(b)).

Arbitrary axes are defined on Π_B, centered at $\boldsymbol{B}$. The circular support region is divided into sectors by defining angular and radial divisions. The points $\boldsymbol{P}$ whose projections belong to each sector are found, and two values are computed; one representing the average of the their positive distances d^+ and one representing the average of their negative ones d^- (Fig. 1(b)), depending on which side of the plane they lie. Therefore, the *Projection Images* are formed denoted with $PI^{\pm}(\rho_i, \theta_j)$, $i = 1, ..., RD$ and $j = 1, ..., AD$, where RD and AD are the number of radial and angular divisions respectively.

In order to remove the degree of freedom along the angular coordinate, the counterpart of the *Projection Images* in the frequency domain will be used. The amplitudes of the Fourier coefficients of $PI^{\pm}(\rho_i, \theta_j)$ for each ρ_i are calculated, producing the final form of the descriptor vector:

$$PrIm^{\pm}(\rho_i, w) = \|F[PI^{\pm}(\rho_i, \theta_j)]\| \quad (1)$$

where w ($w = 0, 1, .., W$) indexes the first W fourier coefficients, thus the dimensionality of $PrIm^{\pm}$ is $2 \cdot RD \cdot W$.

The degrees of similarity between two projection images $PrIm_x^{\pm}(\rho_i, w)$ and $PrIm_y^{\pm}(\rho_i, w)$ are:

$$dist^{\pm} = \sum_{i=0}^{RD} \sum_{w=0}^{W} \frac{|PrIm_x^{\pm}(\rho_i, w) - PrIm_y^{\pm}(\rho_i, w)|}{PrIm_x^{\pm}(\rho_i, w) + PrIm_y^{\pm}(\rho_i, w)} \quad (2)$$

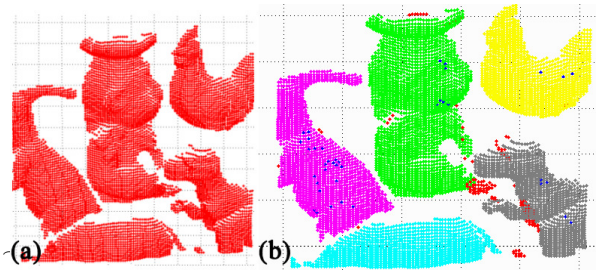

Fig. 2. Illustration of a scene range scan: (a) before and (b) after point could clustering. Red dots depict unclustered points, while blue dots in (b) depict the correspondences between the "Parasaurolophus" and the scene's points. The majority are concentrated on the magenta object, which indeed depicts the "Parasaurolophus".

The final distance $d_{x,y}$ is the mean of $dist^+$ and $dist^-$.

3. RECOGNITION PROCEDURE

The projection images of known objects are extracted and stored in the database. Given a scene's scan, *Projection Images* are extracted and compared with those in the database. Assume that there are M model's descriptors $PrIm_m$ ($m = 0, 1, .., M$), each one extracted for basis point P_m of one object and N scene's descriptors $PrIm_n$ ($n = 0, 1, .., N$) each one extracted for basis point P_n of the scene. For every $PrIm_n$ the $\min_{m=1,...,M}(d_{n,m})$ is found. In order to yield a point correspondence, neither $dist^+$ nor $dist^-$ can be too large.

After finding the point correspondences, a straightforward process is followed in order to verify when an object is positively identified in the scene. Firstly, a clustering approach is applied in the scene's point cloud (Fig.2(a)), exploiting the fact that the points depicting one object are close to each other in terms of their Euclidean distance. On this scope, starting from a random point, its neighbors that are within distance $\boldsymbol{\mu}$ are computed. Then, for each of these neighbors their corresponding neighbors are computed and so on. Thus the point set gets expanded until there is no other point within distance $\boldsymbol{\mu}$ from even one point of the set. This way point clusters are created that realize an $\boldsymbol{\mu}$ connectivity. Fig.2(b) shows the different clusters extracted. Obviously this simple clustering technique successfully grouped the points that correspond to every object into different clusters. Small clusters were not assumed as objects's parts since they probably correspond to severely occluded regions or isolated surface patches (unclassified points are depicted in red in Fig.2(b)).

The cluster with the highest correspondence ratio, i.e. the ratio of the number of correspondence points within the cluster over the cluster's population (in Fig.2(b) the magenta colored cluster), is considered to be the correct match.

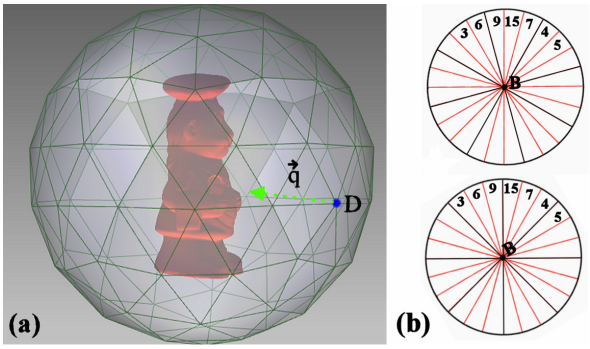

Fig. 3. (a) Viewpoint grid and (b) Circular disk alignment.

4. EXPERIMENTAL RESULTS

4.1. Dataset

In order to be able to compare the performance of the proposed method with existing ones, the dataset available in [16] was used, since there are already published results on it. This dataset includes 50 real 2.5D scenes, produced from a range scanner, each of them comprised of 3 or 4 objects, that occlude both themselves and their neighbors, while there is also a reasonable amount of clutter. Both objects and scenes are given in PLY file format. The presented algorithm is directly applied on point clouds though. Therefore, a range scanner simulator that extracts the point clouds of 3D objects when observed from a specific viewpoint was implemented for the needs of this paper.

In order to generate the point clouds of a 3D model a grid of viewpoints is created around the center of the model so as to scan the object from every angle. The bounding grid is generated by iteratively bisecting the edges of an icosahedron and projecting the new vertices onto a sphere, in order to avoid oversampling near the poles. The vertices define the viewpoints, while their orientations $\overrightarrow{q}$ are from the viewpoint to model's center (Fig.3(a)). The icosahedron used to extract object's viewpoints was tessellated 14 times. This parametrization allowed for the creation of a sufficiently dense grid of viewpoints, which fostered the performance of the algorithm. On the contrary, in order the setup to be actually realistic, for every scene only one point cloud is generated, corresponding to the viewpoint from where the scene is actually observed. One third of the points comprising these point clouds were randomly chosen and served as basis points.

4.2. Pre & Post-processing

During the experimentation phase several pre- and post- processing steps were used to fine tune the Projection Images concept into this specific dataset.

In order to remove points that lie far from Π_B, since they may correspond to a different object than the one where the viewing axis is centered, the mean value of distances and their standard deviation is calculated for both positive and negative distances. In the case the projection distance of a point deviates more than 1.5 times the computed standard deviation then this points is excluded from Π_B's support region.

In order to disregard projection images that come from points observed from slantwise views, thus their region topology could not be accurately captured, a plane is fitted to the points that comprise the support region. In case the plane's perpendicular orientation, $\overrightarrow{u}_B$, is more than $50°$ apart the viewing axes, i.e. $acos(\overrightarrow{u}_B \cdot \overrightarrow{q}) \leq 50°$, $\boldsymbol{B}$'s Projection Image is not taken further into account.

The support region disc was discretized after taking into account the fundamental issue of discretization, depicted in Fig.3(b). The trade-off between a small number of angular divisions versus large is that on the first case information could get distributed among adjacent angular sectors, in case of small geometrical transformation (consider each triplet in Fig.3(b) inbetween black radii to be one sector and the values to be mean values of projection distances of the sub-sectors), versus of having a lot of sectors with none or really few "projection" points in each of them which is highly affected from noise and clutter, and adds to the processing efforts and storing needs. The balance this work proposes is to have few angular divisions with two constraints. First of having more non-zero valued sectors than zero valued (otherwise this basis point is disregarded) ensuring that the projection image will contain sufficient information of the surface. Second, after further diving each sector into three more, the disc is rotated in order the sub-sector with the largest value to be the first (in clockwise direction) of the three that compose one of the sectors as in the low part of Fig.3(b).

4.3. Specification of parameters used

The number of angular and radial division was set to $AD = 10$ and $RD = 10$, respectively. Also, the first $W = AD$ coefficients were used to produce the final descriptor vector. Regarding the size of the support region, it is desired to be large enough so that the projection images contain sufficient information of the topology. At the same time, if R is set to be very large scene's projection images would be more affected by occlusion and clutter. Experimentally, R was set to $R = mean(H_\gamma)/10$, where γ is the index for each library model and H_γ is the maximum inter-distance between the points of it. The variable μ used for the scene's point clustering (Section 3) was set to $\mu = 2 \cdot mean(H_\gamma)/100$.

4.4. Experimental Comparison

The proposed algorithm was compared against uncompressed "spin images" [2], an extensively used state-of-the art algorithm, and "distance maps" [15], a recent and efficient method, using the testing data of [4]. The proposed algorithm

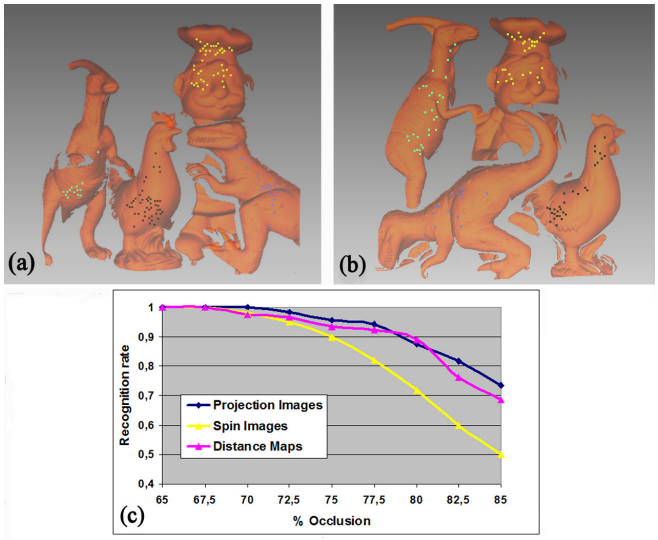

Fig. 4. (a),(b) Experimental results and (c) Recognition rate against occlusion.

was not compared against [4] though, since apart of using the "3D tensors", it further utilizes a registration scheme to allow for recognition, which segments out the parts of the scene that align perfectly with the reference objects, boosting the performance of the algorithm. Thus, comparing to this method would be unfair. The recognition results of "spin images" were copied from [4], and towards a fair comparison the exact same experimental setup as in [4] was used in this work too.

In total 168 recognition tasks were executed while processing the 50 range scan scenes. The overall recognition rate was 89.8%, since 151 out of 168 objects were successfully identified. The recognition rate with respect to occlusion is indicated in Fig.4(c), where the proposed method is compared with "spin images" and "distance maps". The occlusion in the scene was defined as:

$$occlusion = 1 - \frac{object's\ visible\ part}{total\ object's\ surface} \qquad (3)$$

It is clear that the proposed method compares favorably against both "spin images" and "distance maps". Fig. 4(a-b) depicts the results for two experiments. The correspondences of each model in the scene are displayed with dots of different colors, i.e yellow for the "Chef", brown for the "Chicken", green for the "Parasaurolophus" and blue for "T-rex" model. The present method requires about 50 minutes per scene while distance maps required 65 minutes and spin images needed multiple computation time.

5. CONCLUSION

This paper proposes a new regional shape descriptor able to handle clutter and occlusion. Despite the fact that the proposed descriptor depends on the viewpoint, it contains high discriminative power, since it captures information about the object's shape in a sophisticated way.

Experiments conducted on real range scan scenes, proved the robustness of this method to a satisfactory degree of clutter and occlusion. These tests indicated that this method is advantageous to the spin image and the distance map algorithms.

6. REFERENCES

[1] C. Dorai and A. K. Jain, "A Representation Sheme for 3D Free-Form Objects", *IEEE Trans. on PAMI*, vol. 13, No. 2, pp. 1115-1130, 1997.

[2] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes", *IEEE Trans. on PAMI*, Vol. 21, No. 5, pp. 433-449, 1999.

[3] O. Carmichael, D. Huber, and M. Hebert, "Large Data Sets and Confus- ing Scenes in 3-D Surface Matching and Recognition", In 3DIM, pp. 358-367, 1999.

[4] A. S. Mian, M. Bennamoun, and R. Owens, "Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes", *IEEE Trans. on PAMI*, vol. 28, No. 10, pp. 1584-1601, 2006.

[5] C. S. Chua and R. Jarvis, "Point Signatures: A New Representation for 3D Object Recognition ", *IJCV*, vol. 25, No. 1, pp. 63-85, 1997.

[6] R. O. Duda and P. E.Hart, "Pattern Classification and Scene Analysis", *A Wiley-Interscience Publication*, Stanford Research Institute, Menlo Park, California, 1973.

[7] S. Ruiz-Correa, L. G. Shapiro, and M. Meila, "A new paradigm for recognizing 3-d objects from range data", *In Proc. ICCV*, vol. 2, pp. 1126-1133, 2003.

[8] D. Huber, A. Kapuria, R. Donamukkala, and M. Hebert, "Parts-based 3d object classification", *In Proc. ICCV*, vol. 2, pp. 82-89, 2004.

[9] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik, "Recognizing objects in range data using regional point descriptors", *In Proc. ECCV*, 2004.

[10] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors", In Proc. Eu- rographics/ACM SIGGRAPH symposium on Geometry processing, pp. 156-164, 2003.

[11] P. Indyk and R. Motwani, "Approximate nearest neighbor-towards removing the curse of dimensionality", In Proc. Symposium on Theory of Computing, pp. 604-613, 1998.

[12] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches", Pattern Recognition Letters, 28(10):1252-1262, 2007.

[13] K. Khoshelham, "Extending generalized hough transform to detect 3D objects in laser range data", In Proc. International Society for Photogrammetry and Remote Sensing Workshop, pp. 206-210, 2007.

[14] D. Lowe, "Distinctive image Features from Scale-Invariant Keypoints", *In Proc. IJCV*, 2:91-110, 2004.

[15] G. Kordelas and P. Daras, "Viewpoint independent object recognition in cluttered scenes exploiting ray-triangle intersection and SIFT algorithms", Pattern Recognition, 43(11):3833-3845, 2010.

[16] http://www.csse.uwa.edu.au/ ajmal/recognition.html