RECOGNIZING 3D OBJECTS USING RAY-TRIANGLE INTERSECTION DISTANCES

Georgios Kordelas and Petros Daras, Member IEEE

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

ABSTRACT

A novel method for recognizing 3D objects in an occluded, cluttered and noisy 2.5D scene, is presented. A ray-triangle intersection algorithm is used to compute distances between a circular sector that does not belong to the object and a triangulated surface. Firstly, for each sector's point its distance from the object is calculated and stored in a distance map. Secondly, a 2D histogram that counts the distance map's points whose corresponding distance falls within its distance bins, is formed. Then, the percentages of the bin points that fall within each bin are calculated forming the final descriptor vector. The same procedure is followed for the 2.5D scene. The number of the extracted descriptor vectors is independent to the number of the object's or scene's vertices. Experiments proved that the proposed method is fast, robust to noise, occlusion and clutter.

Index Terms— Object recognition, Ray tracing, Feature extraction

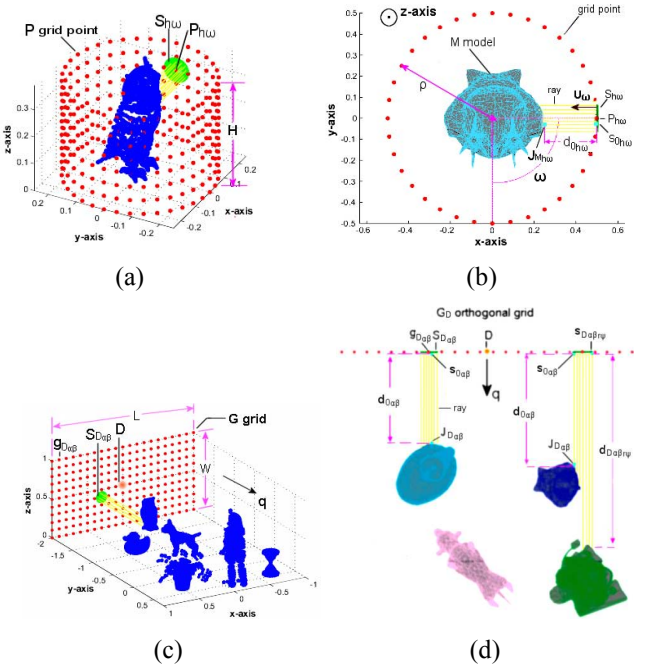
1. INTRODUCTION

In the recent years, significant progress has been made toward the recognition of free-form 3D objects. The aim of object recognition systems is to correctly identify an object in a scene of objects in the presence of noise, clutter and occlusion and to estimate its position and orientation (pose estimation). Several methods have been implemented so far to deal with 3D object recognition. Algorithms that extract local descriptors, such as surface curvatures [2], are proven to be unstable and sensitive to noise [4]. Moreover, the method in [5], which is based on point signatures, is unstable when faced with noisy data, and sensitive to surface sampling [4]. The spin image method, proposed by Johnson and Hebert [3], is vulnerable to sampling and resolution (level-of-detail) of the models [4]. In object-centered methods [3, 4] the extracted descriptors don't depend on the possible views, however the total number of the extracted descriptors does depend on the number of vertices of the model. Thus, the total time needed for the extraction and the comparison of the descriptors is high. Spin image [3] is applied to every vertex, therefore the number of the descriptors increases as the number of vertices does. When the number of descriptors is compressed, using Principal Component Analysis (PCA), the average recognition rate decreases significantly (almost 10%). Spin images have influenced many other researchers after this paper was published. Some papers have copied the idea of spin images and used them in the exact same way, but they have performed some other post-processing or matching methods, like in [7, 8]. Others have created different descriptors but in a somehow similar way [9, 10].

In this paper a novel method is presented that can be characterized as viewer-centered since the descriptors differ each time a different view of the surface is taken into account. However, the total number of the extracted descriptors is significantly less than those extracted when object-centered methods are applied. The presented method is based on the extraction of distance maps that characterize the local topology of a surface. The model descriptors are extracted from 3D triangulated objects, while the scene descriptors are extracted from a 2.5D triangulated surface. The distance maps are created using ray-triangle intersection algorithms, where distances between a circular sector of points (which express the origins of oriented rays) lying away from the surface, and the triangulated surface are computed. These distances, per sector, are arranged in ascending order, and a 2D histogram is created which expresses the number of sector points within each distance bin. From the 2D histogram a descriptor vector is formed as the normalized percentage of the number of each bin's points to the total number of the sector's points. By doing so, the number of the extracted descriptors is independent to the number of the vertices an object or a scene contains, thus the number of the descriptors for a 3D object, at different levels-of-detail, remains the same, in contrast to the methods presented in [3, 4]. In this way, simpler and faster comparison of the descriptors is achieved.

The presented algorithm is semi-automatic meaning that there is a need for user's intervention during the parametrization of the scene, in the sense that the user defines the boundaries of the grid to be used later for the extraction of scene's descriptors. However, when the object recognition task takes place in a specified place and the recognition system is adapted in a fixed position, this procedure needs to be done only once.

This work was supported by VICTORY and CATER EC projects and by altaB23D GSRT project.



(a) (b) (c) (d)

Fig. 1. Illustration of distance map computation for model (a),(b) and scene (c),(d).

The rest of this paper is organized as follows: In Section 2 the algorithm for the extraction of the distance map and the creation of the descriptor vector for both the scene and the models, is presented. Section 3 describes the object recognition procedure. Experimental results are presented in Section 4. Finally, the conclusions are drawn in Section 5.

2. PROPOSED METHOD

In this section the proposed procedure, from the creation of the distance map for the 3D objects and the 2.5D scene, up to the final descriptor vector extraction, is explained.

2.1. Extraction of 3D object's distance map

Let M be a triangulated 3D object. Firstly, M is rotated to its physical position in the space. Then, a bounding cylindrical grid is created around the object, so that its z-axis coincides to M's z-axis. The parametrization parameters of the bounding cylindrical grid are defined as: $h \in \{k \cdot\ i; i = 0, 1, 2, ..., H/k\}, \omega \in \{b \cdot j; j = 0, 1, 2, ..., 360°/b\}, \lfloor H/k \rfloor = H/k$ (Fig. 1 (a)). The height parameter h varies on z-axis and the angle parameter ω varies on the xy plane. k and b are the height and the angle grid intervals, respectively. Value $\omega = 0$ corresponds to x-axis. A cylindrical grid point is defined as $\boldsymbol{P}_{h\omega} = [\rho \cdot \cos(\omega),\ \rho \cdot \sin(\omega),\ h]^T$, where ρ is the cylinder's radius. Then, a circular sector S with radius R is created using the parametrization variables r, ψ, where $r \in \{c \cdot\ i; i = 0, 1, 2, ..., R/c\}$, $\psi \in \{f \cdot\ j; j = 0, 1, 2, ..., 360°/f\}$ and $\lfloor R/c \rfloor = R/c$. c is the radius interval and f is the angle interval. The parameters r, ψ define each point on S. The centered at $\boldsymbol{P}_{h\omega}$, sector $S_{h\omega}$ (Fig.1 (a),(b)) is given by the equation:

$$S_{h\omega} = \boldsymbol{R}_z(-\theta)\boldsymbol{R}_y(-(\pi/2 - \phi))S + \boldsymbol{P}_{h\omega} \qquad (1)$$

where $\boldsymbol{R}_z$ and $\boldsymbol{R}_y$ are the rotation matrices about the y and z axis, respectively. Each point $\boldsymbol{s}_{h\omega_{r\psi}} \in S_{h\omega}$ is the origin of an oriented ray with direction $\boldsymbol{v}_\omega$, where $\boldsymbol{v}_\omega = [-cos(\omega),\ -sin(\omega),\ 0]^T$. $\theta(longitude)$ and $\phi(latitude)$ in (1) are the spherical coordinates of $\boldsymbol{v}_\omega$ (Fig. 1 (b)).

Using the ray-triangle algorithm described in [1], the distance $d_{h\omega_{r\psi}}$ between $\boldsymbol{s}_{h\omega_{r\psi}}$ and the triangulated surface of M is computed. It should be noted that when the ray does not intersect any triangle the distance is set to "-1". If the ray intersects M more than once, the smallest distance is kept. The point $\boldsymbol{s}_{h\omega_{r\psi}}$ for which $d_{h\omega_{r\psi}}$ is minimum, is denoted as $\boldsymbol{s}_{0_{h\omega}}$ and it is stored for the sector $S_{h\omega}$, as well as its corresponding distance $d_{0_{h\omega}}$ (Fig. 1 (b)). Having as input $\boldsymbol{s}_{0_{h\omega}}$, $d_{0_{h\omega}}$ and $\boldsymbol{v}_\omega$, a ray that intersects the 3D object's surface at point $\boldsymbol{J}_{M_{h\omega}}$ is defined (Fig. 1 (b)). $\boldsymbol{J}_{M_{h\omega}}$ is given by the equation:

$$\boldsymbol{J}_{M_{h\omega}} = \boldsymbol{v}_\omega \cdot d_{0_{h\omega}} + \boldsymbol{s}_{0_{h\omega}} \qquad (2)$$

The computed distances for all $\boldsymbol{s}_{h\omega_{r\psi}}$ are used to extract a distance map per $S_{h\omega}$, where a distance map point is defined as $\boldsymbol{\Phi}_{h\omega_{r\psi}} = [r \cdot \cos(\psi),\ r \cdot\ sin(\psi),\ d_{h\omega_{r\psi}} - d_{0_{h\omega}}]^T$. The number of the distance maps per object equals to the total number of $\boldsymbol{P}_{h\omega}$. Distance maps for grid points of all models are created and stored (off-line) in a model library. The maximum distance $(d_{h\omega_{r\psi}} - d_{0_{h\omega}})$ of distance maps of all models is defined as D_{max}.

2.2. Extraction of scene's distance map

Let us assume a synthetic 2.5D scene that simulates the reconstructed scene from a stereo pair of cameras placed in a pre-specified room where multiple 3D objects exist (Fig. 1 (c)). The scene is observed from a specified viewpoint $\boldsymbol{D} = [x_d,\ y_d,\ z_d]^T$ (cyclopean eye of stereo pair) and the view-direction is defined by the normal vector $\boldsymbol{q} = [q_1,\ q_2,\ 0]^T$. An orthogonal grid G of length L and width W is created (Fig. 1 (c)). G's parametrization variables are $\alpha \in \{-W/2 + i \cdot \gamma; i = 0, 1, 2, ..., W/\gamma\}$, $\beta \in \{-\mathcal{L}/2 + j \cdot \gamma; j = 0, 1, 2, ..., \mathcal{L}/\gamma\}$ and $\lfloor W/\gamma \rfloor = W/\gamma, \lfloor \mathcal{L}/\gamma \rfloor = \mathcal{L}/\gamma$, where γ is the distance between the grid points. The centered at $\boldsymbol{D}$, grid G_D, is given by the equation:

$$G_D = \boldsymbol{R_z}(-\theta_1)\boldsymbol{R_y}(-(\pi/2 - \phi_1))G + \boldsymbol{D} \qquad (3)$$

where $\boldsymbol{R}_z$ and $\boldsymbol{R}_y$ are the rotation matrices about the y and z axis, respectively and $\boldsymbol{q}$'s (Fig. 1 (c)) spherical coordinates are $\theta_1(longitude)$ and $\phi_1(latitude)$. Each point $\boldsymbol{g}_{D_{\alpha\beta}} \in G_D$ is the center of a circular sector $S_{D_{\alpha\beta}}$ given by the equation:

$$S_{D_{\alpha\beta}} = \boldsymbol{R_z}(-\theta_1)\boldsymbol{R_y}(-(\pi/2 - \phi_1))S + \boldsymbol{g}_{D_{\alpha\beta}} \qquad (4)$$

Fig. 2. Library models.

A distance map is created per $\boldsymbol{S}_{D_{\alpha\beta}}$. Each point $\boldsymbol{s}_{D_{\alpha\beta_{r\psi}}} \in S_{D_{\alpha\beta}}$ is the origin of an oriented ray with direction $\boldsymbol{q}$. The distance $d_{D_{\alpha\beta_{r\psi}}}$ between $\boldsymbol{s}_{D_{\alpha\beta_{r\psi}}}$ and the 2.5D triangulated scene is computed. The point $\boldsymbol{s}_{D_{\alpha\beta_{r\psi}}}$ for which $d_{D_{\alpha\beta_{r\psi}}}$ is minimum, is denoted as $s_{0_{\alpha\beta}}$ and it is stored for the sector $S_{D_{\alpha\beta}}$, as well as its corresponding distance $d_{0_{\alpha\beta}}$ (Fig. 1 (d)). Having as input $\boldsymbol{s}_{0_{\alpha\beta}}$, $d_{0_{\alpha\beta}}$ and $\boldsymbol{q}$, a ray that intersects the object surface at point $\boldsymbol{J}_{D_{\alpha\beta}}$ is defined (Fig. 1 (d)). $\boldsymbol{J}_{D_{\alpha\beta}}$ is given by the equation:

$$\boldsymbol{J}_{D_{\alpha\beta}} = \boldsymbol{q} \cdot d_{0_{\alpha\beta}} + \boldsymbol{s}_{0_{\alpha\beta}} \tag{5}$$

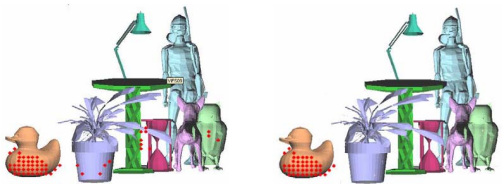
The distance map is defined by the points $\boldsymbol{\Phi}_{\alpha\beta_{r\psi}} = [r \cdot \cos(\psi),\ r \cdot sin(\psi),\ d_{D_{\alpha\beta_{r\psi}}} - d_{0_{\alpha\beta}}]^T$. The variables $W, \mathcal{L}$ must have a large enough value so that if the plane, that is defined by G's grid points, is moved perpendicular to $\boldsymbol{q}$, then the 3D objects of the scene are contained in the swept volume (Fig. 1(c)).

2.3. Descriptor vector

In order to create the descriptor vector $\boldsymbol{\Delta}$ from a distance map, distance map's stored distances are arranged in ascending order. Then, a 2D histogram that counts the number of sector points whose distance ($(d_{h\omega_{r\psi}} - d_{0_{h\omega}})$ or $(d_{D_{\alpha\beta_{r\psi}}} - d_{0_{\alpha\beta}})$ for a model or a scene sector point, respectively) is falling within each of N distance bins, is created (the range of the 2D histogram is [0, D_{max}] and $N = 60$). The more the bins, the more precise is the histogram. $\boldsymbol{\Delta}$ is formed as the normalized percentage of the number of each bin's sector points to the total number of the sector points that fall within 2D histogram's range. $\boldsymbol{\Delta}_{D_{\alpha\beta}}$ is created from $S_{D_{\alpha\beta}}$ and $\boldsymbol{\Delta}_{M_{h\omega}}$ is created from $S_{h\omega}$ of the object M. Supposing that $\boldsymbol{\Delta}_{D_{\alpha\beta}}$, $\boldsymbol{\Delta}_{M_{h\omega}}$ are the N dimensional descriptor vectors, their percentage of dissimilarity is given by the equation: $K_{D_{\alpha\beta}M_{h\omega}} = \sum_{i=1}^{N} \left(\frac{|\boldsymbol{\Delta}_{D_{\alpha\beta}}(i) - \boldsymbol{\Delta}_{M_{h\omega}}(i)|}{N} \right)$. When 80% of the total number of points of a distance map has distance equal to "-1", the created descriptor vector may lead to misleading dissimilarities. Thus, it is not taken into account. This limitation further reduces the number of the created descriptor vectors for the scene and the 3D models.

3. OBJECT RECOGNITION AND POSE ESTIMATION

Prior to recognition procedure, the $\boldsymbol{\Delta}_{M_{h\omega}}$ descriptors are created and stored in the model library. Afterwards, the dissimilarity for a $\boldsymbol{\Delta}_{D_{\alpha\beta}}$ is computed for all $\boldsymbol{\Delta}_{M_{h\omega}}$. The $\boldsymbol{\Delta}_{M_{h\omega}}$ for which the computed $K_{D_{\alpha\beta}M_{h\omega}}$ is below a threshold determines the object M to whom $\boldsymbol{\Delta}_{D_{\alpha\beta}}$ corresponds. Then, the point $\boldsymbol{J}_{M_{h\omega}}$ (which lies on M surface), corresponds to the point $\boldsymbol{J}_{D_{\alpha\beta}}$ (which lies on scene's surface). Thus, a point correspondence between $\boldsymbol{J}_{D_{\alpha\beta}}$ and $\boldsymbol{J}_{M_{h\omega}}$ is established. The pairs of all $\boldsymbol{J}_{M_{h\omega}}, \boldsymbol{J}_{D_{\alpha\beta}}$ result in a sizeable set of point correspondences between the surface of object M and a scene surface. This set is used to align the surface of the library object M to the surface of the scene.



(a) (b)

Fig. 3. Found $g_{D_{\alpha\beta}}$(red dots) (a) before clustering (b) after clustering

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed recognition system a model library consisted of 8 synthetic models was created (Fig. 2). During the experimental procedure the performance of the proposed approach was tested for noisy, cluttered and occluded scenes and it was compared to spin image method [3]. The recognition success was verified by computing the rates of true positive (TP), false positive (FP) and false negative (FN) [3]. The definitions of clutter and occlusion are given in [4].

4.1. Generating parameters

The library models had comparable size. The k interval (section 2.1) is obtained by dividing the height of the shortest object by a number equal or higher than 5, in order to have an adequate number of descriptors for the smallest object. The angle interval b (section 2.1) of the cylindrical grid was set to be about seven degrees. The variable R that defines the S radius is crucial for the method. It is desired that the rays starting from the sectors $S_{h\omega}$, $S_{D_{\alpha\beta}}$ intersect the surface in a large enough area so that the distance map will contain sufficient information for the topology of the surface in order to discriminate different surfaces. At the same time, if radius R is set to be very large, a greater computation time will be required and scene's distance maps will have more chances to store distances for rays that intersect surfaces from different objects due to occlusion and clutter. The optimum R belongs to $k \leq R \leq 1.5 \cdot k$. Experimentally, R was set to $R = 1.4 \cdot k$. It is desired that the points are dense on sector S in order to exist many origins. For that reason the S's angle interval is set to

be f=6° degrees and the radius interval is given by $c = R/40$. The distance interval γ of G was set to $\gamma \leq k/2$. The spin image parameters were defined according to [3].

4.2. Analysis

The models were placed randomly in the scene and the number of models per scene varied from 4 to 8 in order to estimate the recognition rate at different clutter and occlusion rates. It was assumed that only one object was present in the scene and the other models were assumed unknown. Thus, there were more candidate $\mathbf{\Delta}_{D_{\alpha\beta}}$ to be matched to $\mathbf{\Delta}_{M_{h\omega}}$ of the searched object M. Per recognition trial, the $\mathbf{\Delta}_{D_{\alpha\beta}}$ that corresponded to descriptor vectors of object M, were found. $\mathbf{\Delta}_{D_{\alpha\beta}}$ were created from $S_{D_{\alpha\beta}}$, whose center was $\boldsymbol{g}_{D_{\alpha\beta}}$. $\boldsymbol{g}_{D_{\alpha\beta}}$ points were lying on the plane that G_D defines. Using the ISODATA algorithm [6], the $\boldsymbol{g}_{D_{\alpha\beta}}$ points were classified into clusters based on their inter-distances. The largest cluster was chosen (Fig. 3) as the one that contains the points $\boldsymbol{g}_{D_{\alpha\beta}}$ whose descriptor vectors will be used for the pose estimation (section 3). Seven scenes were created using 8 (Fig. 2) models and totally 37 recognition experiments were performed for the proposed method and spin image. For each scene, the number of the recognition experiments was equal to the number of models that were present to the scene. The average number of $\mathbf{\Delta}_{D_{\alpha\beta}}$ per scene was 1814, while the average number of spin images was 8465. Moreover the average number of extracted $\mathbf{\Delta}_{M_{h\omega}}$ per model was 712, while the spin images per model were 2665. It is obvious that the number of descriptor vectors was significant less than number of spin images. As a result the extraction and comparison of descriptor vectors using the proposed method is quicker. From Fig. 4 (a) is concluded that the average recognition rate was 82% with up to 76% occlusion, while spin image recognition rate was 73.5%. The recognition rate with respect to clutter was 87.9% at 80% clutter and 80.5% for the method in [3] (Fig. 4 (b)). Experimentally, it was shown that the rate was mainly affected by occlusion. The average recognition rate was 81.1% (since 30 out of 37 recognition trials were successful) and 73% for the method in [3].

Finally, the method was tested under the presence of noise. The noise was added to the scene along the view-direction $\boldsymbol{q}$ (Fig. 4 (d)), in order to simulate the noise that a range scanner placed at D with viewing direction $\boldsymbol{q}$ would cause. The proposed method is satisfactory to noise of $\sigma = 1.75\ cm$, since the percentage of correct recognition was 84.7% (while spin image recognition rate was 81.5%) as it is depicted in (Fig. 4 (c)). At noise $\sigma = 2.43\ cm$ the rate was 76.3% for our method and 71.5% for spin image.

5. CONCLUSION

In this paper a novel object recognition algorithm was presented. It is proved to be simultaneously fast and robust to a satisfactory degree of noise, occlusion and clutter. Comparison to the spin image algorithm proved that our algorithm is superior to spin image recognition.

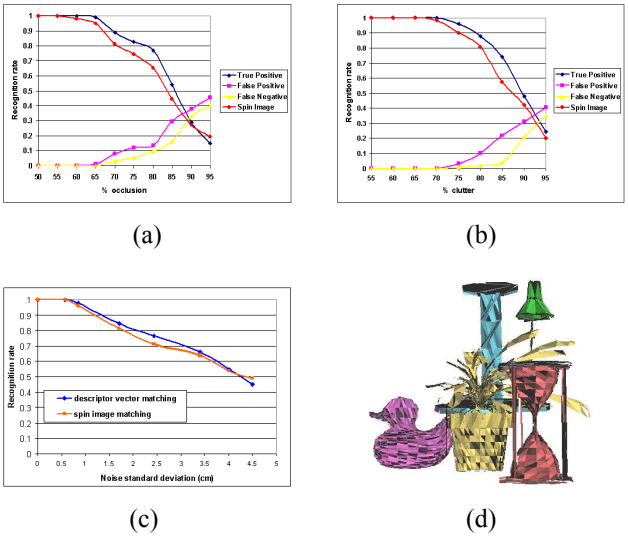


Fig. 4. Recognition rate against (a) occlusion (b) clutter and (c) noise. A scene with σ=1.75cm Gaussian noise (d).

6. REFERENCES

[1] T. and B. Trumbore, "Fast, Minimum Storage Ray-Triangle Intersection", *Journal of Graphics Tools*, vol. 2, No 1, pp. 21-28, 1997.

[2] 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.

[3] 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.

[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] H. Chen and B. Bhanou, "3d free-form object recognition in range images using local surface patches", *In Proc. ICPR*, vol. 3, pp. 136-139, 2004.