

RESEARCH

Open Access

A method of 3D object recognition and localization in a cloud of points

Jerzy Bielicki* and Robert Sitnik

Abstract

The proposed method given in this article is prepared for analysis of data in the form of cloud of points directly from 3D measurements. It is designed for use in the end-user applications that can directly be integrated with 3D scanning software. The method utilizes locally calculated feature vectors (FVs) in point cloud data. Recognition is based on comparison of the analyzed scene with reference object library. A global descriptor in the form of a set of spatially distributed FVs is created for each reference model. During the detection process, correlation of subsets of reference FVs with FVs calculated in the scene is computed. Features utilized in the algorithm are based on parameters, which qualitatively estimate mean and Gaussian curvatures. Replacement of differentiation with averaging in the curvatures estimation makes the algorithm more resistant to discontinuities and poor quality of the input data. Utilization of the FV subsets allows to detect partially occluded and cluttered objects in the scene, while additional spatial information maintains false positive rate at a reasonably low level.

Keywords: 3D object recognition, Point cloud

Introduction

Prevalence of 3D data acquisition optical systems demands development of dedicated algorithms for an efficient analysis of such data in various fields of civil engineering, entertainment, and industry. Automated monitoring systems more and more often require end-user applications that are able to perform basic analysis or to detect unexpected object presence or behavior. Monitoring of large amount of visual data (e.g., representing terrain after flood, hurricane or another natural catastrophe) to find predefined objects (e.g., human bodies or generally 3D objects) that may be only partially visible is a difficult task for people. It demands constant focus, which usually varies in time. Efficiency of such analysis depends on human factor, and may lead to serious mistakes, that may cost human beings. Proposed method is suitable for such tasks due to automatic processing of large datasets, and specifying areas (with shape similar to requested; e.g., with higher probability than defined threshold), that should be analyzed precisely by human

operator (to avoid false-positive mistakes). Such a solution allows to apply robust and automated analysis for whole dataset, while only the limited areas require more efficient and expensive analysis performed by well-trained human operator.

Most of the existing algorithms are based on either local or global object description. The local approach allows to effectively detect partially occluded objects. Utilization of 3D curves to describe edges of the objects and splashes to describe variations of surface normals within the local neighbourhood is described in [1]. Other local representations are point signatures (similar to splashes, but avoiding first-order derivative calculations) describing neighbour surface by analysis of sphere-surface intersections [2], point fingerprints (where geodesic circles are projected onto surface tangent planes) [3], or keypoints with quality measure, based on values of principal curvatures (more sensitive to presence of noise and discontinuities, as it requires second-order derivative calculations) [4]. Locally based recognition is relatively fast, as it does not require computation of any features dependent on the size of the model. The disadvantage is that information provided only by a small part of the

* Correspondence: j.bielicki@mchtr.pw.edu.pl
Institute of Micromechanics and Photonics, Warsaw University of Technology,
8 Sw. Andrzeja Boboli, 02-525, Warsaw, Poland

object may easily be corrupted by, e.g., noise, and mislead the detection process.

The global approach may be less dependent on the local quality of analyzed surface. Utilization of oriented surface–point pair (point and surface normal at this point) histograms and review of their statistical comparison criteria are presented in [5]. Other methods apply 3D modification of Hough transform for object similarity retrieval [6], or exploit reference object information from range image for efficient, low-level comparison implemented on GPU [7]. Important drawbacks include sensitivity to the loss of information (e.g., occlusion), misinformation (e.g., clutter) and time of data processing usually proportional to the object size.

Some existing algorithms require segmentation of the analyzed scene, which uses either *a priori* knowledge about input data or simple segmentation rules. Treating horizontally oriented planes as background to easily extract objects [8], or clustering scene using distance-related feature [9] limits the generality of the algorithm. It often leads to under or over segmentation, what may result in significant errors, what is considered in the smoothness constraint segmentation problem [10].

Popular approach to 3D recognition problem is to exploit range images, also known as depth maps. These maps make data processing significantly faster, as they convert the most time-consuming problems (e.g., neighbourhood finding) from 3D into 2D space. Such a conversion into deeply studied image comparison problem allows to calculate more efficiently local surface descriptors (2D local histograms describing distribution of shape index [11] versus scanning angle) [12], to match multidimensional object histograms (distributions of surface normals and shape indexes) [13], or to focus on implementation of efficient algorithms directly on the GPU [7]. Interesting, but requiring preceding conversion from range image data into mesh model, is tensor-based representation of local surface patches, with voting scheme using 4D hash table [14]. However, the drawback of this approach is that it cannot analyze surfaces representing complete volumes. A range image is a projection of the single, directional point cloud (which is point cloud acquired from single direction—so-called 2.5D data) onto a plane, therefore uniqueness cannot be maintained for full 3D objects (formed from numerous directional clouds of points integrated in a single model).

Another approach converting 3D recognition problem into 2D image correlation task is to exploit spin-images, presented in [15,16]. Spin-image is a 2D accumulator located at oriented point (point with surface normal n_s at this point), which collects all points, that are intersection of analyzed object and rectangle rotated around surface normal. Only such points are taken into consideration, which surface normals are deviated from surface normal

(n_s) direction less than certain angle threshold. Such a solution is time-efficient regarding scene-model point matching. Drawback of this approach is the fact, that for optimal object description, setting the angle threshold requires knowledge about shape complexity of reference objects and objects analyzed in the scene.

Topology of the analysed data is exploited by another group of algorithms related to *Non-rigid shapes* [17], where statistical significance measure uses geodesic metrics as partial similarity criterion [18], or where 3D object is characterized by a set of signatures (histograms of geometric distances, diffusion distances, the ratio of diffusion and geodesic distances, and two curvature-related histograms) which allows to determine similarity between objects as a multiplication of the pair wise histogram comparison (with χ^2 measure) results [19]. Unfortunately, they are not suitable for data in the form of single directional point clouds. Topology of an object represented by a directional point cloud may easily be changed due to the presence of noise or occlusion. This would introduce significant errors in the recognition process that use such type of algorithms.

In the process of development of the proposed algorithm, advantages of local and global approaches have been considered. Proposed algorithm exploits locally calculated descriptors based on point cloud parameters (which are equivalent to curvatures [20,21] often utilized in the problem of face recognition [22–24]) on the purpose of scene-model matching and then considers combinations of spatially distributed scene-model matched candidates. Segmentation, as a step which may introduce errors and decrease algorithm efficiency, has been avoided. Considering various types of data representation and processing, it has been decided to keep whole 3D information, although for a purpose of analysis of single, directional point cloud the range image representation would suffice. Since input data for each reference object is a set of directional point clouds (acquired from different directions), there is no need to perform any preliminary segmentation or to use iterative closest point (ICP-type) [25] algorithms, to obtain full 3D reference object, like in [4]. This assumption allows to expand reference objects library quite easily, simply by scanning new 3D object and performing training phase, without using any additional software and with practically no effort. As a result, the proposed algorithm is more general, and may successfully be applied in various fields of industry (automatic machine part recognition), civil engineering (large area monitoring) and entertainment (applications for stereo imaging mobile phones), where 3D data are exploited. In cases mentioned above, the proposed algorithm points out areas geometrically similar to reference objects, which may significantly improve detection speed and effectiveness of search and rescue. As a part of built-in 3D scanner software, may be

optional, but in result significant operator support in critical situations.

Main text

Method

The proposed algorithm is divided into two phases (Figure 1). First is the training phase, performed off-line, when reference object database is created. Second phase is the recognition process, performed on-line. Both phases use common pre-processing (PP) in the beginning of the data processing flow. PP extracts geometrical features from the input data, which are utilized in the further steps. Detection results are presented as a probability distribution of the object presence in the scene.

PP

First, the common stage of both phases of the proposed algorithm consists five steps, which are preceded by simple point cloud average distance calculation to allow further automatic calculations on 3D input data.

Average point-to-point distance calculation

This step precedes PP and is used to estimate the average distance between points in the input data. A set of random points (approximately 1000, or less if model is smaller than 1000 points) is selected and for each point the distance to the closest neighbour point is calculated and the average distance is found. It allows to set a value of neighbourhood radius r for each calculation step as a multiplication of an average point distance, so it is not

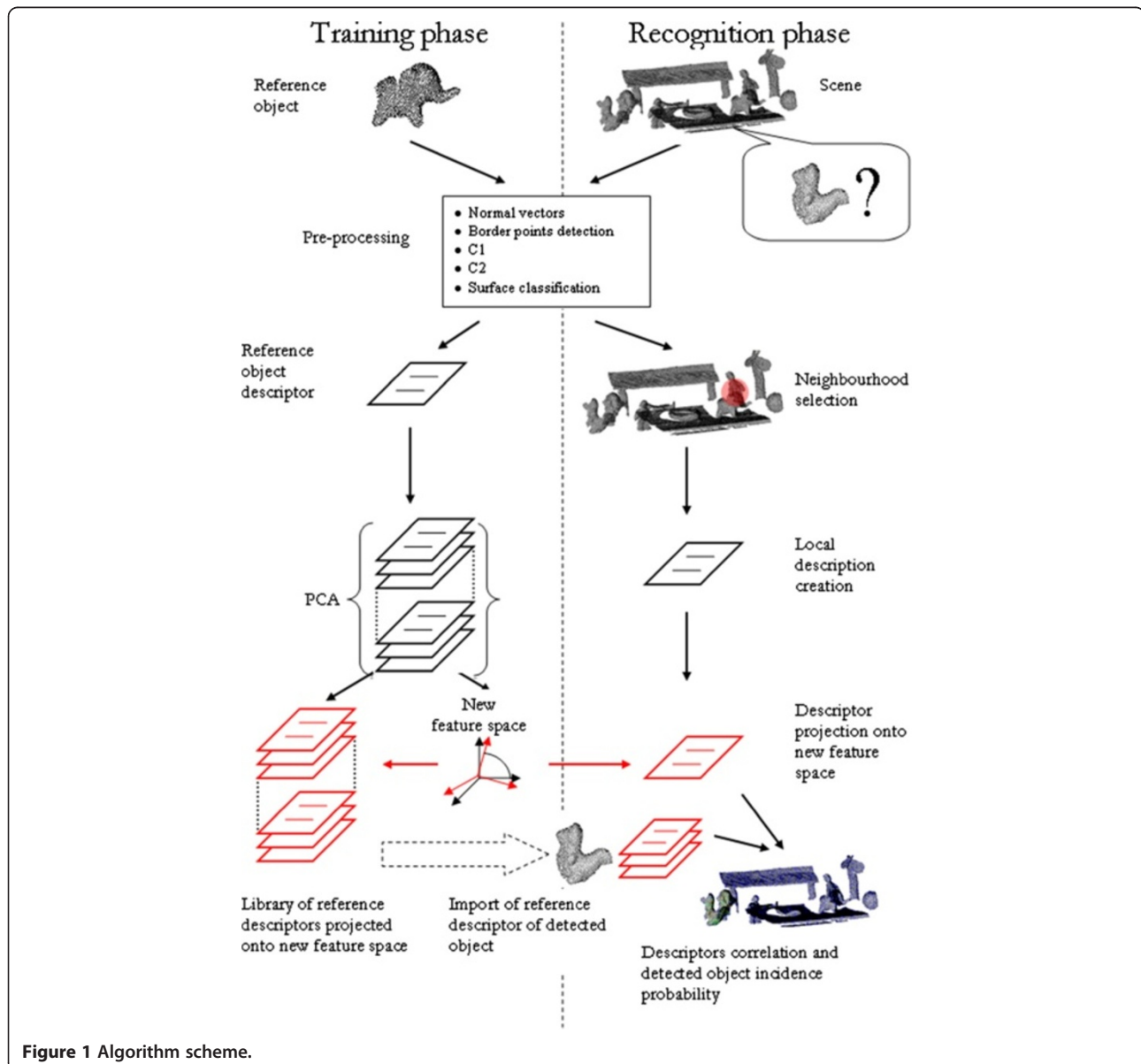


Figure 1 Algorithm scheme.

necessary to know the dimensions of the measured objects. Value of neighbourhood radius is set arbitrary, assuring non-zero neighbourhood even in the presence of fluctuation of the local cloud density. In all calculations presented in the article neighbourhood radius $r = 6$.

Data PP consists of following steps:

- surface normal vectors estimation,
- border points detection,
- C1 parameter computation [26]
- C2 parameter computation,
- surface-type estimation.

Surface normal vectors estimation

In the next step, normal vectors for the directional point cloud data are calculated. For each point, a best fit plane (BFP) to the surface within the neighbourhood is estimated, using Root Mean Square (RMS) minimization criterion [27]. Obtained BFP is described by coefficients A, B, C and D in plane Equation (1). Normal vector is denoted as a set of normalized coefficients (A, B, C) from Equation (1), and its orientation is determined by the location of the scanning system (normal vector is oriented towards scanning system, which is equivalent to $C > 0$).

$$BFP : Ax + By + Cz + D = 0 \quad (1)$$

Border points detection

Surface of the BFP is radially divided into six equal zones (angle of each zone is approximately 60°) with the projection of currently analysed point p_p used as division centre (Figure 2) [28]. BFP division is made using vector v_1 , which connects projections of the arbitrary neighbour point p_1 with division centre p_p , and using vector v_2 , which is a vector product of BFP normal and vector v_1 . Each zone accumulates orthogonal projections of every neighbour point p_i onto the BFP. Lack of points in any of the zones indicates that the considered point belongs to an edge. Division orientation for each BFP is arbitrary,

but it does not affect the result of the border detection stage.

C1

Parameter C1 is defined as follows [26].

For each sampling point p_p signed, weighted, average distance to BFP is calculated. Distance with sign means that the sign of the value depends on the location of neighbour point p_i with respect to the BFP normal vector (Figure 3). Points located on the same side of the BFP as its normal vector contribute with positive value d_i whereas points on the opposite side contribute with negative value d_j . Weight w_i for each neighbour point is proportional to the Gaussian distribution of the distance s_i between orthogonal projections of sampling point p_p

and neighbour point p_i : $w_i \sim e^{-\frac{s_i^2}{2r^2}}$, where r is the neighbourhood radius. Value of C1 is greater than zero for convex surfaces, smaller than zero for concave surfaces and equal to zero for planes or saddle points. C1 parameter corresponds qualitatively to mean curvature, which is arithmetical mean of main surface curvatures.

C2

Parameter C2 is defined as below [26].

For each sampling point p_p , a best plane (BFP_N) is fitted to all its neighbour normal vectors (such as n_i), and average distance between BFP_N and tips of the normal vectors is calculated (Figure 4c,f). As one can see, for cylindrical surfaces (Figure 4d-f) C2 value is equal to zero, while for spherical surfaces (Figure 4a-c) value of C2 is positive. Negative value of C2 denotes saddle-like surface. For each neighbour point p_i , inner product of normal n_i and vector v_i is also calculated. Vector v_i connects orthogonal projections of sampling point p_p and neighbour point p_i onto BFP (Figure 5). Sign of each inner product $v_i \cdot n_i$ gives information whether surface surrounding neighbourhood point p_i is concave or convex. Presence of noise can cause local fluctuations of direction of normal vectors and change sign of particular inner

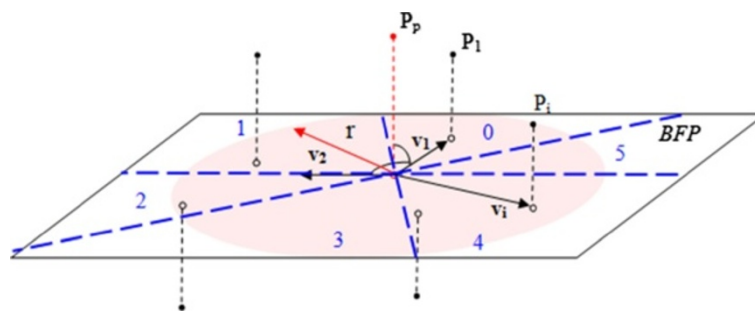


Figure 2 Border points estimation. BFP is equally divided into six zones, all of which “collects” orthogonal projections of neighbour points. Empty zone indicates that analysed point lies on the border.

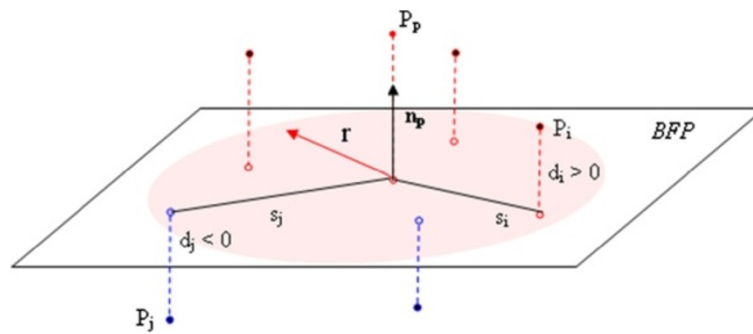


Figure 3 C1 parameter estimation. C1 value is a signed, weighted average point distance to BFP, where weight is proportional to Gaussian function of point distance s_j and neighbourhood radius r .

products. Taking it into consideration, the presence of saddle-like surface is indicated when number of the inner products with opposite signs exceeds 35%. C2 parameter corresponds qualitatively to Gaussian curvature, which is a multiplication of main surface curvatures.

Local surface type

Local surface type is estimated using parameters C1 and C2 (Table 1). Positive signs of values of both parameters indicate convex ellipsoidal surface. Negative sign of C2 parameter value indicates that one of the main curvatures

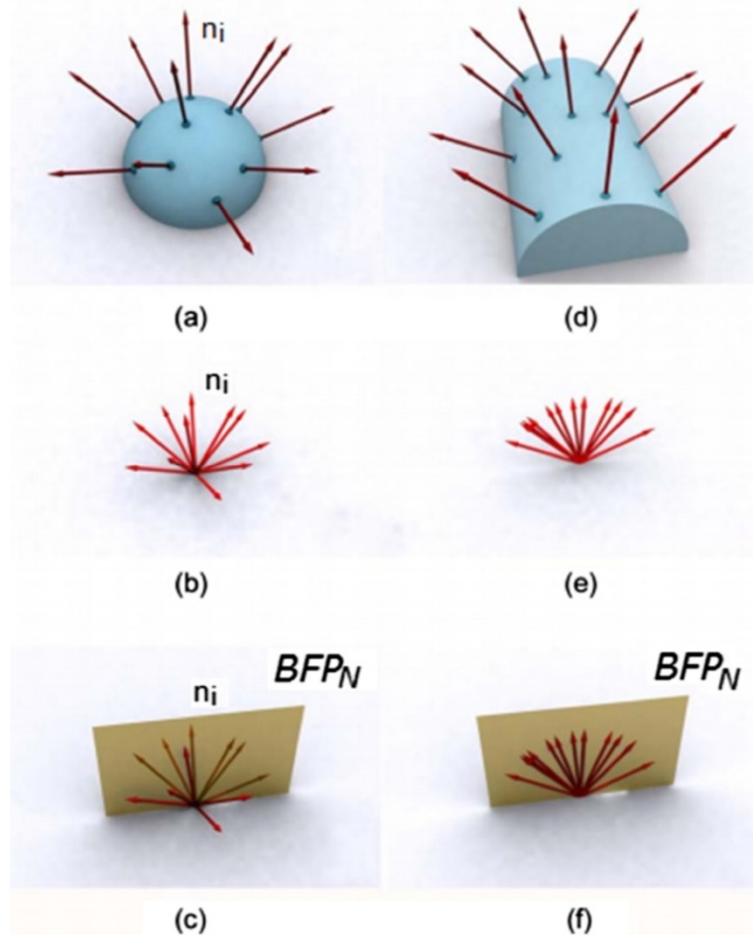


Figure 4 C2 parameter calculation. (a) Normal vectors on a sphere; (b) normal vectors from a sphere translated into the origin of the coordinate system; (c) normal vectors from a sphere and their BFP_N ; (d) normal vectors on a cylinder; (e) normal vectors from a cylinder translated into the origin of the coordinate system; (f) normal vectors from a cylinder and their BFP_N [26].

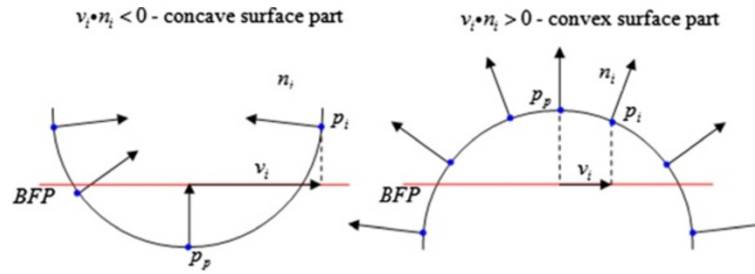


Figure 5 Estimation of surface convexity. Positive inner product denotes convex surface, otherwise surface is concave.

is negative, which means that the surface is saddle-like. When C2 parameter value is zero, then surface is planar for C1 parameter equal to zero, and cylindrical otherwise. Sign of the C1 parameter then indicates cylinder convexity (or ellipsoid convexity for positive C2 value). Local surface type corresponds to shape index introduced by Koenderink and van Doorn [11] utilized in [6,12,13], which is basically a mapping of main curvatures into polar system, where angle denotes the surface type, and distance from origin denotes the values of the curvatures. Shape index s is denoted with equation:

$$s = \frac{2}{\pi} \arctan \frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1} \quad (2)$$

where κ_1 and κ_2 are main curvatures. As it can be seen in Table 1, local surface type parameter can have eight different values, utilized in further calculations.

Training phase

In this off-line phase, a description of the reference object is created. Based on the PP results the local feature vectors (FVs) are computed, and then collected to form a global descriptor for each reference object (Figure 6). Further, reduction of the feature space dimensionality using principal component analysis (PCA) is performed [29].

Table 1 Local surface type estimation using H and K (mean and Gaussian curvature, respectively) or using $C1$ and $C2$ parameters, equivalently

Surface type	H	K	$C1$	$C2$
Plane	0	0	0	0
Convex cylinder	>0	0	>0	0
Concave cylinder	<0	0	<0	0
Convex ellipsoid	>0	>0	>0	>0
Concave ellipsoid	<0	>0	<0	>0
Saddle-like surface	>0	<0	>0	<0
	=0	<0	=0	<0
	<0		<0	

Descriptor of a reference object consists of a set of local FVs, each of which is calculated for a sampling point p_n . Since the local FV is calculated for every n th point of the data, the number of FVs depends on sampling resolution of the reference model. The descriptor also contains the radius R of the smallest bounding sphere for the model. The radius R is utilized in recognition phase to specify neighbourhood for local FV calculations.

Each local FV contains two histograms. First one is 2D distribution C1 versus C2 (Figure 7).

$$H_1 : D_1 \mapsto (i, j) \in \{1, 2, \dots, d_{C1} = 10\} \times \{1, 2, \dots, d_{C2} = 10\}, \quad (3)$$

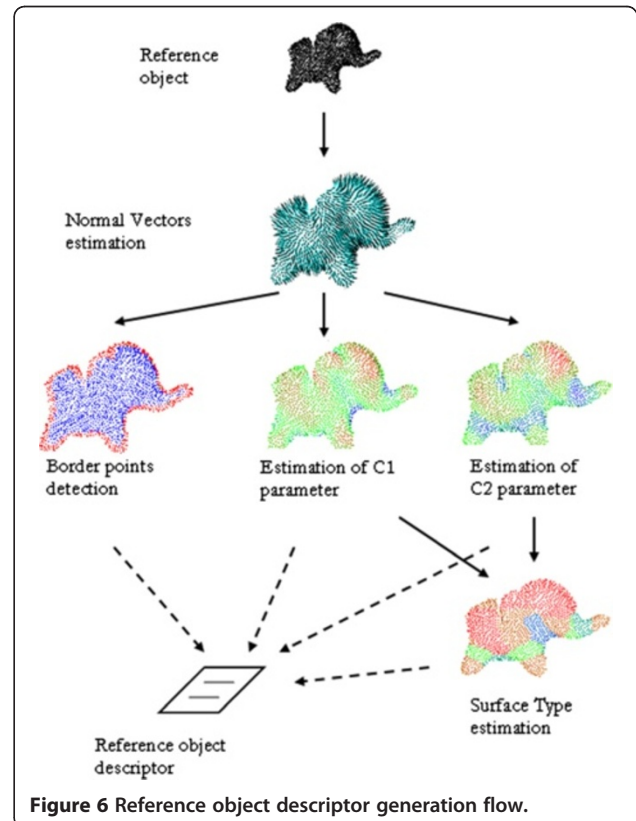


Figure 6 Reference object descriptor generation flow.

where d_{C1} is a number of intervals for C1 distribution and d_{C2} is a number of intervals for C2 distribution. The other one is a local surface type distribution:

$$H_2 : D_2 \mapsto (i) \in \{1, 2, \dots, d_S = 8\} \quad (4)$$

where d_S is a number of intervals. Dimensionality of the feature space FS is then $N = d_{C1} \cdot d_{C2} + d_S$. Each local FV contains also minimal and maximal values of parameters C1 and C2 and coordinates of the sampling point p_n , for which the vector is calculated. During this process all edge points are ignored, as the information obtained for these areas may be misleading. For each local FV only the points located closer than 60% of radius R are considered. It implies that only approximately 30% of reference model surface is considered for a single FV and partial occlusion or clutter affects only few FVs. Due to this fact the algorithm is more resistant to these adverse conditions.

The last step of the training phase is PCA, which reduces dimensionality of the created feature space. Feature space is reduced from initial $N = 108$ dimensions into significantly smaller space, usually less than 10D. Condition taken into consideration during PCA is to preserve 99% of initial energy. It guarantees that only the most meaningful linear combinations of initial features will be considered in the recognition phase.

Recognition phase

In this on-line phase, scene-model matching is performed. Based on the PP results, local FV is created. Then the local FV is correlated with reference object descriptors. The thresholding step leads to the final decision about object presence.

Local model-scene correlation

The step following the PP stage is introduced to compute the local FV for each sampling point. Similarly to the

training phase, every n th point is sampled. The size of neighbourhood (denoted by the radius R) utilized for the calculation of local FVs depends on the size of the reference object that is looked for (see 1.2 paragraph for details). Each of the scene's local FVs is projected onto feature space FS and then correlated with all FVs of the reference object. Correlation degree is denoted as a local probability P of object existence, and is calculated as follows:

$$P = e^{-\frac{\|d - d_{ref}\|^2}{\|d_{ref}\|^2}} = e^{-\frac{\sum_{o=1}^N (d_o - d_{ro})^2}{\sum_{o=1}^N d_{ro}^2}} \quad (5)$$

where $d = [d_1, \dots, d_N]$ is the projection of the local FV_n onto feature space FS , $d_{ref} = [d_{r1}, \dots, d_{rN}]$ is the projection of one of the reference model's local FV_{kl} onto feature space FS .

As a result a set of probability distributions P_{ikl} is obtained, where i is point index, k is the index of the reference object and l is the index of one of the reference model's local FV_{kl} .

Thresholding

At this stage, final decision about the presence of one or multiple objects is made. Sensitivity of the algorithm is modified using three parameters:

T_p – probability threshold; all points with probability P_{ikl} below T_p are ignored,

D_p – distance difference threshold; a pair of scene sampling points is accepted only if the difference between distance $|P_i P_j|$ and $|P_{kl} P_{km}|$ is smaller than the threshold value, where $|P_i P_j|$ – distance between neighbour points p_i and p_j , with probabilities P_{ikl} and P_{jkm} , respectively, $|P_{kl} P_{km}|$ – distance between reference object sampling points p_{kl} and p_{km} for which local FV_{kl} and FV_{km} were calculated.

N_p – number of accepted points threshold; for $N_p = 3$ similar triangles, and for $N_p = 4$ similar quadrangles, are searched.

Thresholding algorithm (Figure 8) is shown below.

1. Point p_1 with probability P_{1kl} higher than T_p is searched. If found, k (object index) and l (FV index) are remembered (Figure 8a).
2. Point p_2 with probability P_{2km} higher than T_p is searched. Additional condition to satisfy is

$$||p_1 p_2| - |p_{kl} p_{km}|| < D_p \cdot |p_{kl} p_{km}| \text{ for } l \neq m \quad (6)$$

which corresponds to the fact that a line segment $|P_i P_j|$ similar to the reference model segment $|P_{kl} P_{km}|$ is found in the scene (Figure 8b).

3. If $N_p = 3$, then last, third point p_3 is searched. That point has to satisfy condition (6) against both points

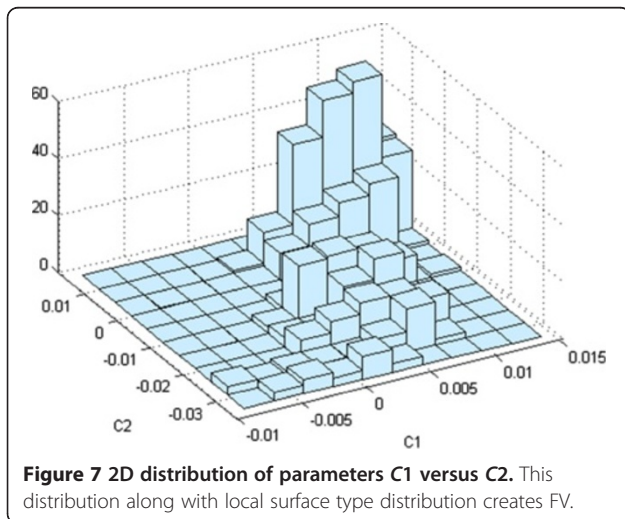


Figure 7 2D distribution of parameters C1 versus C2. This distribution along with local surface type distribution creates FV.

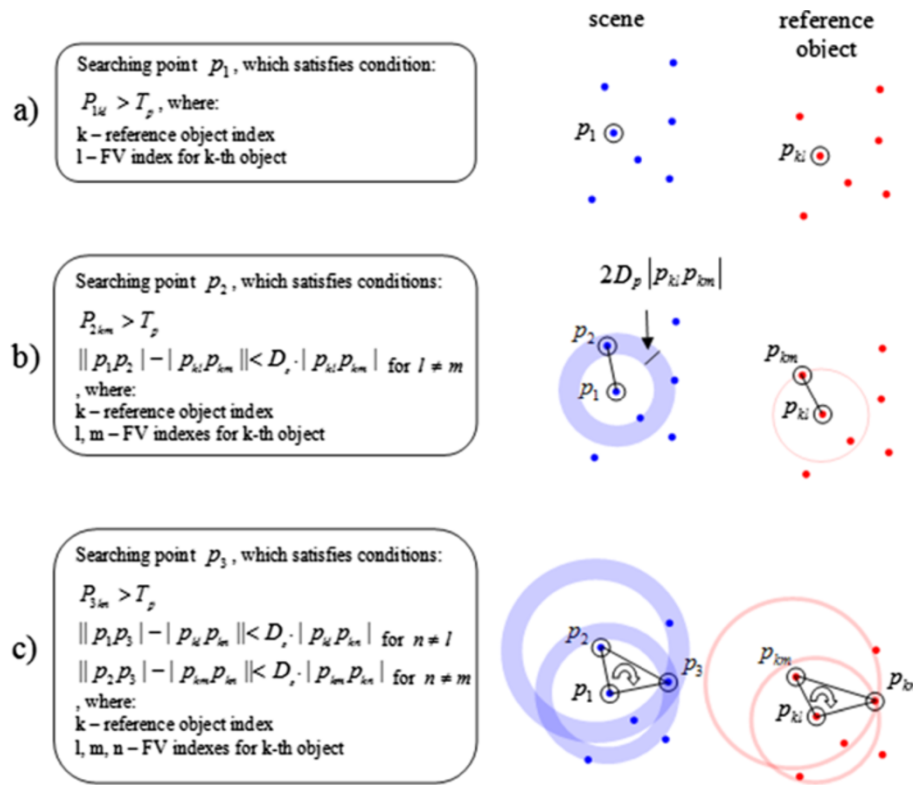


Figure 8 Thresholding algorithm. (a) First point with probability greater than threshold is found (with highest probability, as all probabilities points are sorted top to bottom); (b) second point with enough high probability, satisfying additional distance requirement, is found; (c) similar triangle is found, when exists third point which satisfies both distance requirements and guarantee that triangle vertexes permutation is maintained. If needed ($N_p = 4$), fourth point is searched to obtain similar quadrangle.

p_1 and p_2 (Figure 8c). To guarantee similarity between detected and referenced triangle, direction of vector product in relation to average direction of normal vectors is checked. If succeeded, a triangle similar to the triangle from reference model is found in the scene and average probability is calculated. If $N_p = 4$, a fourth point is searched to find a similar quadrangle. All probabilities are sorted top to bottom, so even if the search does not succeed, performance of the algorithm is high.

At the end of the recognition phase all small, adjacent groups of selected points are converted into consistent objects.

Results

Assessment of the proposed method was performed using synthetic and real input data. Synthetic data have mostly been used to evaluate the influence of data quality on detection results. Real data were captured using the OGX|3DMADMAC scanner [30,31].

For the purpose of algorithm evaluation, an object database was created (red objects in Figure 9). Each reference object is presented as a set of directional point clouds

acquired from various points of view. Direction of data acquisition changes horizontally. Angles between any two adjacent scans are approximately 45° . All objects apart from two human figures and a dog, located right bottom in Figure 9, were scanned with OGX|3DMADMAC scanner. Human figures and the dog are imported from Princeton Shape Benchmark (called further: human 1 PSB, human 2 PSB, dog PSB) [32], re-sampled and converted into directional point cloud to maintain consistency within input data. Conversion from full 3D model usually properly imitates point surface acquired during scanning process. If a 3D model consists of a set of volumes (i.e., computer designed model as a set of sub volumes), then created surface may be similar to one that has poor quality, but still allows to perform the calculation. All presented results (Figures 10, 11, 12) are made using colour map with red colour denoting probability close to 100%, through green denoting 50% to blue colour denoting 0%. Grey colour denotes probability below threshold.

In Figure 10a, one of the test scenes with objects from five different classes is presented. Consecutive figures present detection results (after the thresholding procedure, where colour of the object depends on the detection probability) of the following objects: (b) ball, (c) plasticine



Figure 9 Reference objects database. Object marked as black (plasticine figurines) are not present in the created library.

figures, (d) elephant, (e) dinosaur and (f) dog. All objects apart from the plasticine figures are detected with probability of nearly 100%. Plasticine figures are detected with lower, 80–90% probabilities. It is important to notice that only one of these figures was scanned as a reference object, and that all the figures differ from each other in shape, as they were hand-made.

Figure 11 shows one of the worst case detection results for (a) torus, (b) ball, (c) plasticine figures, (d) elephant, (e) dinosaur and (f) dog. Figure 11a,c,d shows that in some cases local descriptors may not be descriptive enough to distinguish between similar objects. On the other hand, such a property reduces algorithm sensitivity to clutter and occlusion.

In Figure 12, a synthetic scene is presented. All the complex objects, such as (a) Porsche, aircraft and (b–d) Mercedes models have been converted from full 3D models (composed of multiple solids), so the obtained surfaces may locally be confusing for the algorithm. In Figure 12c, detection of the dog PSB is presented. Figure 12a,b,d presents detection results of human 1 PSB. Figure 12b,d shows the influence of distance difference threshold D_p value on detection results for probability threshold $T_p = 90\%$. For Figure 12b, with $D_p = 0,04$ only the standing pose is detected. Increasing parameter value to $D_p = 0,1$ results in detection of both human figures and in one false positive detection (Figure 12d). It can be seen that presence of obstacles in the neighbourhood does not

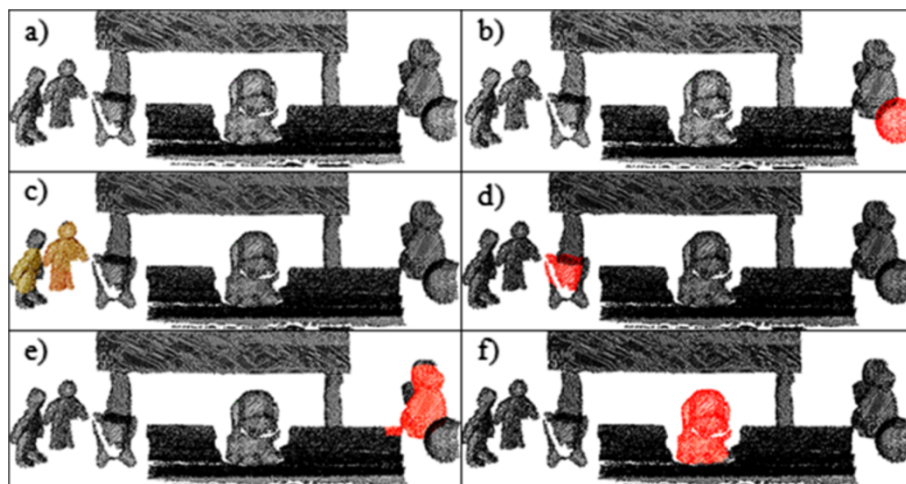


Figure 10 Recognition results in scenes scanned with OGX|3DMADMAC system.

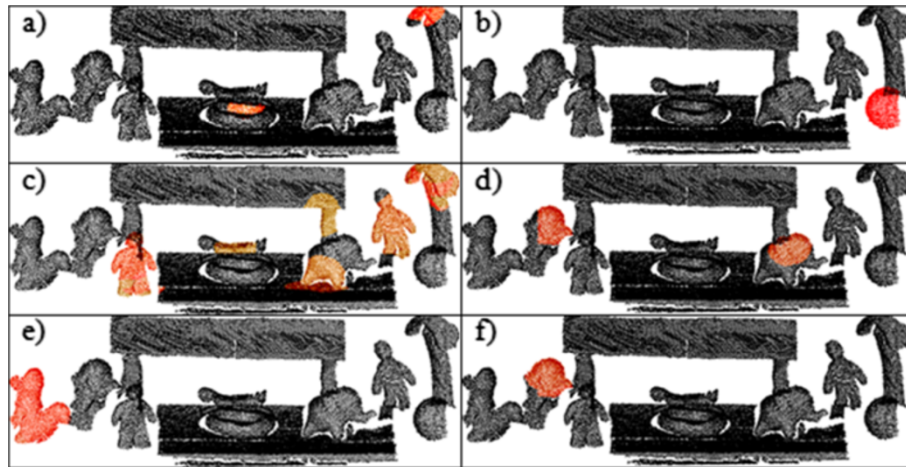


Figure 11 Recognition results in scenes scanned with OGX|3DMADMAC system.

strongly affect detection, since local descriptors are utilized. From the other hand it is noticeable that, with constant value of D_p threshold, the bigger is maximum distance between reference points (which is a result of value of radius R of the analysed neighbourhood—see 1.2 for details) the bigger is tolerance field for real difference between reference points distance and candidate points distance. For better performance, value of D_p threshold should vary as a function of distance, to avoid linear increase of distance difference toleration with increase of the distance between reference points.

Figure 13 presents the relation of recognition rate versus occlusion for all the analysed objects. For most of the objects recognition rate above 80% is reached for 70% occlusion, since detection process utilizes a combination of a small number of local FVs. Simultaneously, spatial information (analysis of distribution of at least three descriptors for each reference object) allows to keep *false positive ratio* at a reasonably low level. The achieved

results show that the proposed algorithm is resistant to occlusion. More sensitive to clutter and occlusion seem to be the algorithms, based on histogram comparison [5], where incompleteness of data may change comparison result significantly. Better performance present algorithms based on keypoints [4] with further ICP-type surface registration, what may be less effective, when considering objects, that are in general similar, but differ from each other on the whole surface (i.e., plasticine figures), which is the most common case in real environment.

Robustness to clutter results from the fact that varying number of points representing other objects/background does not affect the locally calculated FVs, nor the detection results. Nevertheless, the impact of spatial distribution of clutter points is not easy to estimate.

Another important factor to consider is noise. Presence of the noise was simulated by adding to each object (synthetic or real) randomly generated noise of increasing amplitude. To maintain consistency within all reference

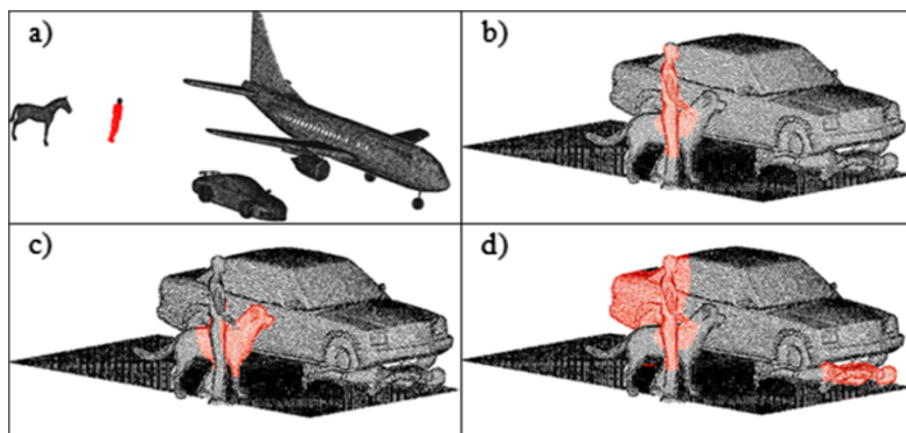
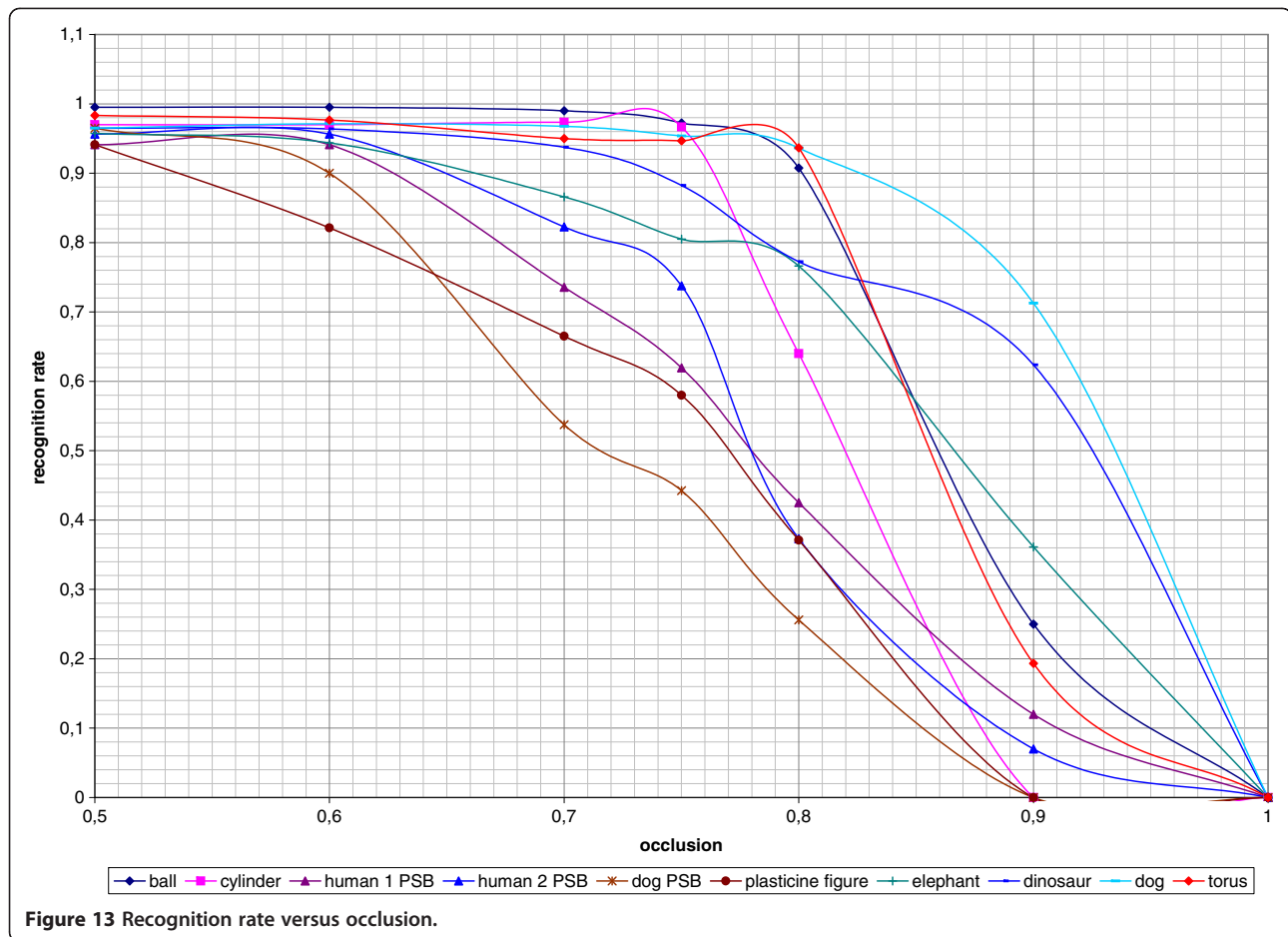


Figure 12 Recognition results in synthetic scenes.



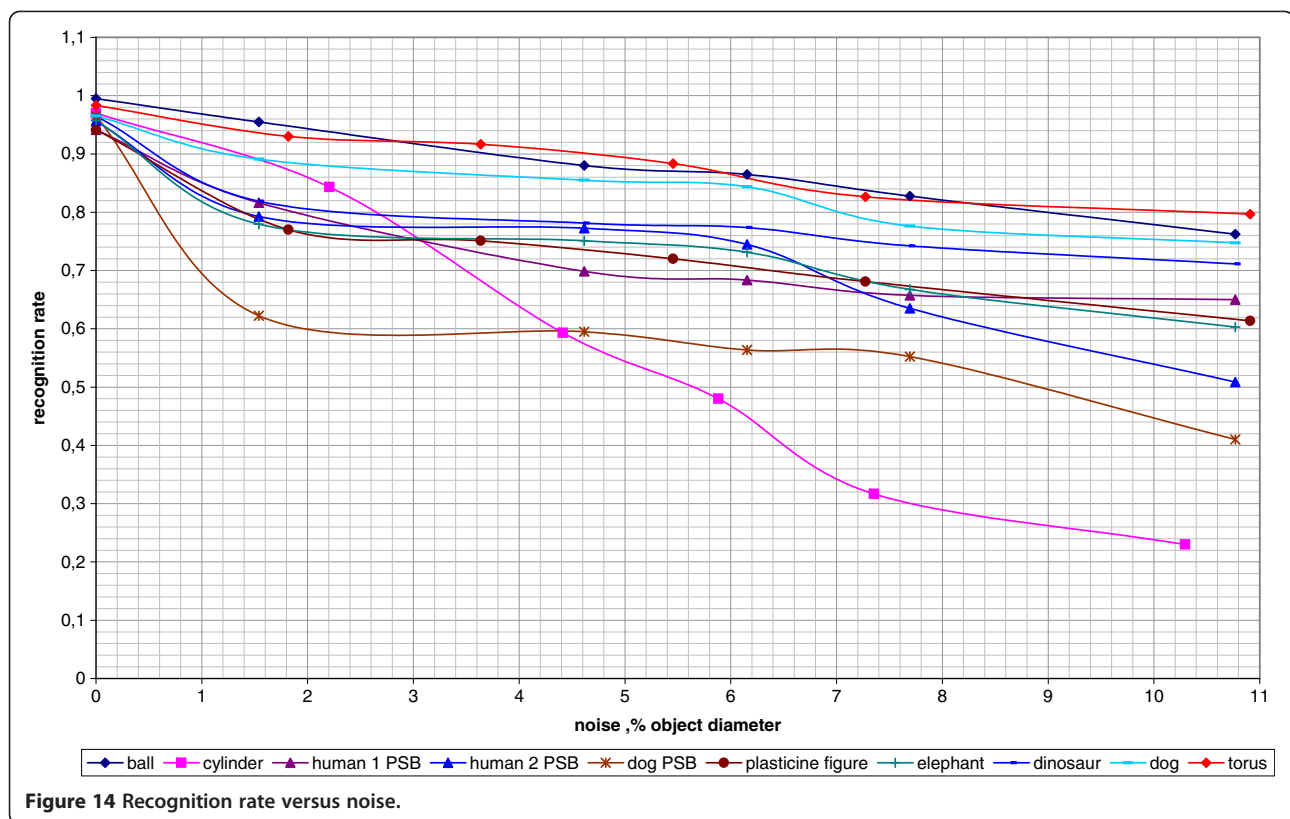
objects, amplitude value depends on object's diameter. Values of all detection parameters are the same as in previous assessments. Influence of the noise on recognition efficiency is presented in Figure 14. Most of the objects are recognized with efficiency above 75% for amplitude noise reaching 5% of object diameter. Presence of noise seems to be crucial for descriptiveness of histograms presented in [5], as they require whole data of good quality for proper representation of the object. Sensitive to the presence of noise appear to be also point fingerprint approach [3], as it utilizes geodesic curves, which length and position may strongly be disrupted by noisy points. For the ICP-type algorithms [4], presence of noise may slightly increase RMS error value (quality of surface registration), but should not strongly affect recognition rate.

The worst case presented in Figure 14 is recognition of the dog PSB. The reason seems to be the fact that even small amplitude noise added to, e.g., thin legs of the dog changes rapidly their shape (two legs may be joined into a single one). The next to the worst case is the cylinder. In this case, poor recognition result is caused by high ratio between length and radius of the cylinder. Large length implies high amplitude of random

noise, which, compared with small radius, significantly modifies the shape of the object. The rest of the objects are detected efficiently and increasing noise amplitude decreases the recognition accuracy at an acceptable rate. Utilization of the BFPs instead of spatial derivatives in the process of local surface type estimation improves algorithm's resistance to noise or discontinuities within the input data.

Conclusion

In this article, efficient and robust to noise algorithm have been presented. Performed comparisons in the recognition process are based on an approach avoiding noise-sensitive calculations, to make an algorithm suitable for low-quality data. The assumption was to exploit minimum number of parameters, which completely describe local geometry of the surface. Main step of the algorithm is to consider set of spatially distributed surface parts as a representation of reference object stored in database. Such an approach should allow to detect even strongly occluded objects, where detection is a result of finding similar structure during scene-model correlation process. Thanks that there is no significant difference



for the algorithm between occlusion and clutter, as long as noise is not similar in shape to the missing parts of detected objects.

Main drawback is, in some cases, descriptiveness of the proposed descriptor, which may result in high false positive ratio. It may need to be minimized by improvements of geometrical representation of the reference object, and by utilizing quadrangles instead of triangles in thresholding process.

The results show that the proposed algorithm can be used with incomplete and noisy data. Impact of clutter on the recognition rate is also minimized because of the utilization of local FVs.

In the future, a sampling algorithm similar to *Farthest Point Sampling* [33] will be implemented to assure uniform data sampling. Also modification of the spatial representation of the reference objects may improve descriptiveness and thanks to that reduce false positive recognition ratio. The most promising seems to be an implementation of two-level hierarchy model representation, which will be suitable for high detailed and point dense objects. At the moment, thresholding process results in finding in scene a triangle (or quadrangle), where its vertexes represent surface parts similar to the surface of reference object. Adding top level hierarchy, resulting in detection of “triangle of triangles” of similar surface parts, will increase the accuracy of detection, especially of large models (i.e. huge animals)

which consist high detailed elements (i.e. head, legs, tail) which may be significant for distinction one from another (i.e. horse from rhinoceros). Another advantage of proposed modification is that it can represent each object as a spatial distribution of basic shapes, which is much more intuitive and easy to analyse for humans, than, i.e. set of keypoints [4], spin-images [16] or histograms [5], and can further be exploited in more abstract way (i.e. cognitive systems). Calculation time should not increase significantly, as the number of the most time-consuming calculations (scene-model similarity comparison) remains same, with change of the thresholding process only.

One of the advantages is that the same algorithm can add reference objects to the existing database, which allows to develop such a database whenever it is necessary. Presented method may efficiently assist the operators of the automated maintenance monitoring systems.

Competing interests

The authors declare that they have no competing interests.

Received: 12 February 2012 Accepted: 28 January 2013

Published: 21 February 2013

References

1. GM Fridtjof, Stein, Structural indexing: efficient 3-D object recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 125–145 (1992)
2. RJ Chin, Seng Chua, Point signatures a new representation for 3D object recognition. *Int. J. Comput. Vis.* **25**(1), 63–85 (1997)

3. JP Yiyong Sun, K Andreas, DL Page, MA Abidi, The point fingerprint. A new 3-D object representation scheme. *IEEE Trans. Syst. Man Cybern.* **33**(4), 712–717 (2003)
4. A Mian, M Bennamoun, R Owens, On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *Int. J. Comput. Vis.* **89**, 348–361 (2009)
5. E Wahl, U Hillenbrand, G Hirzinger, Surflet-pair-relation histograms: a statistical 3D-shape representation for rapid classification, in *Proc. Fourth International Conference on 3-D Digital Imaging and Modeling*, 2003, pp. 474–481
6. T Zaharia, F Preteux, Hough transform-based 3D mesh retrieval. *Proc. SPIE* **4476**, 175–185 (2001)
7. MDB Marcel Germann, *Automatic Pose Estimation for Range Images on the GPU* (Hanspeter Pfister, Kyu Park, 2005)
8. RB Rusu, A Holzbach, M Beetz, G Bradski, Detecting and segmenting objects for mobile manipulation, in *Proc. IEEE 12th International Conference on Computer Vision Workshops*, 2009, pp. 47–54
9. K Klasing, A clustering method for efficient segmentation of 3D laser data, in *Proc. IEEE International Conference on Robotics and Automation*, 2008, pp. 4043–4048
10. T Rabbani, FA van den Heuvel, G Vosselman, Segmentation of point clouds using smoothness constraint. *ISPRS XXXVI*(5), 248–253 (2006)
11. JJ Koenderink, AJ van Doorn, Surface shape and curvature scales. *Image Vis. Comput.* **10**(8), 557–564 (1992)
12. BB Hui, Chen, 3D free-form object recognition in range images using local surface patches. *Pattern Recognit. Lett.* **28**(10), 1252–1262 (2007)
13. B Leibe, G Hertz, P Levi, B Schiele, 3D object recognition from range images using local feature histograms, in *Proc. CVPR(II)*, 2001, pp. 394–399
14. AS Mian, M Bennamoun, RA Owens, 3D recognition and segmentation of objects in cluttered scenes, in *Proc. ACV*, 2005, pp. 8–13
15. E Andrew, MH Johnson, Surface matching for object recognition in complex 3-D scenes. *Image Vis. Comput.* **16**, 635–651 (1998)
16. E Andrew, MH Johnson, Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(5), 433–449 (1999)
17. M Ovsjanikov, Q-X Huang, L Guibas, A condition number for non-rigid shape matching. *Computer Graphics Forum* **30**(5), 1503–1512 (2011)
18. AM Bronstein, MM Bronstein, Y Carmon, R Kimmel, *Partial Similarity of Shapes Using a Statistical Significance Measure* (Department of Computer Science, Technion Israel Institute of Technology, Haifa, 2009)
19. M Mahmoudi, G Sapiro, Three-dimensional point cloud recognition via distributions of geometric distances. *Graphical Models* **71**(1), 22–31 (2009)
20. J Goldfeather VI, A novel cubic-order algorithm for approximating principal direction vectors. *ACM Trans. Graph* **23**, 45–63 (2004)
21. A Mustafa, L Shapiro, M Ganter, 3D object identification with color and curvature signatures. *Pattern Recognit.* **32**, 339–355 (1999)
22. A Colombo, C Cusano, R Schettini, 3D face detection using curvature analysis. *Pattern Recognit.* **39**, 444–455 (2006)
23. P Szeptycki, M Ardabilian, L Chen, A coarse-to-fine curvature analysis-based rotation invariant 3D face landmarking, in *Proc. International Conference of Biometrics: Theory, Applications and Systems*, 2009, pp. 1–6
24. F Crosilla, D Visintini, F Sepic, Reliable automatic classification and segmentation of laser point clouds by statistical analysis of surface curvature values. *Appl. Geomat.* **1**(1–2), 17–30 (2009)
25. C Li, D Shaoyi, Z Nanning, A fast multi-resolution iterative closest point algorithm, in *Proc. CCPR*, 2010, pp. 1–5
26. M Witkowski, R Sitnik, Locating and tracing of anatomical landmarks based on full-field four-dimensional measurement of human body surface. *J. Biomed. Opt.* **13**(4), 044039 (2008)
27. A Björck, *Numerical Methods for Least Squares Problems* (SIAM, Philadelphia, 1997)
28. E Kalogerakis, D Nowrouzezahrai, S Patricio, S Karan, Extracting lines of curvature from noisy point clouds. *Elsevier J. Comput.-Aided Design* **41** (Special Issue on Point-Based Computational Techniques), 282–292 (2009)
29. IT Jolliffe, *Principal Component Analysis*, 2nd edn. (Springer, New York, 2002)
30. R Sitnik, M Kujawinska, J Woźnicki, Digital fringe projection system for large-volume 360-deg shape measurement. *Opt. Eng.* **44**(41), 443–449 (2002)
31. OGX(Optographx). <http://ogx.mchtr.pw.edu.pl/projects/3dmdmac/>
32. Princeton Shape Benchmark. <http://shape.cs.princeton.edu/benchmark/>
33. AM Bronstein, MM Bronstein, R Kimmel, *Numerical Geometry of Non-Rigid Shapes* (Springer Verlag, New York, 2009)

doi:10.1186/1687-6180-2013-29

Cite this article as: Bielicki and Sitnik: A method of 3D object recognition and localization in a cloud of points. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:29.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com