Single view 3D structure estimation using Generic Fitted Primitives (GFP) for service robotics

Tiberiu T. Cocias, Sorin M. Grigorescu, Florin Moldoveanu

Departament of Automation, Transilvania University of Brașov, Brașov, Romania
{tiberiu.cocias, s.grigorescu, moldof}@unitbv.ro

Abstract. This paper presents a method for surface estimation applied on single viewed objects. Its goal is to deliver reliable 3D scene information to service robotics application for appropriate grasp and manipulation actions. The core of the approach is to deform a predefined generic primitive such that it captures the local geometrical information which describes the imaged object model. The primitive modeling process is performed on 3D *Regions of Interest* (ROI) obtained by classifying the objects present in the scene. In order to speed up the process, the primitive points are divided into two categories: *control* and *regular points*. The control points are used to sculpt the initial primitive model based on the principle of active contours, or snakes, whereas the regular points are used to smooth the final representation of the object. In the end, a compact volume can be obtained by generating a 3D mesh based on the newly modified primitive point cloud. The obtained *Point Distribution Model*s (PDM) are used for the purpose of precise object manipulation in service robotics applications.

Keywords: Robot vision, 3D object reconstruction, Object structure estimation, Primitive modeling, Service robotics.

1 **Introduction**.

Nowadays most service robotics applications use depth perception for the purpose of environment understanding. In order to precisely locate, grasp and manipulate an object, a robot has to estimate as good as possible the pose and the structure of that object of interest. For this reason different visual acquisition devices, such as stereo cameras, range finder or structured light sensors, are used [14].

For online manipulation, together with the pose of the object, it is needed to determine the 3D particularities of the viewed structure in order to estimate its shape [5].

There are several types of methods that focus on the 3D reconstruction of objects using multiple perspectives. Such methods try to reconstruct the convex hull of the object [11], or to recover its photo-hull [9]. Other algorithms explore the minimization of the object's surface integral with a certain cost function over the surface shape [10].

On the other hand, the reconstruction can be addressed also from a single view. This technique is usually efficient when applied to regular surface objects. An early approach for this challenge was investigated for piecewise planar reconstructions of paintings and photographs [6]. Subsequent improvements of the technique [3], [13] increased the geometric precision especially for scenes with multiple vanishing planes.

In terms of reconstruction resolution and accuracy, range images (e.g. from laser scanners) provide one of the best surface estimations data. However, it has speed deficiency, sensor dimension and power consumption [8]. The main challenge encountered during 3D reconstruction is the automatic computation of the 3D transformations that align the range data. Thus, the registration of different perspective point clouds into one common coordinate system represents one of the most researched topics in the computer vision community [8], [12].

The rest of the paper is organized as follows. In Section 2 a brief description of the image processing chain is provided. The main contribution of the paper, that is the 3D shape modeling approach, is given in Section 3. Finally, before conclusions, performance evaluation results are presented in Section 4.

2 Machine Vision apparatus

The block diagram of the proposed scene perception system can be seen in figure 1.

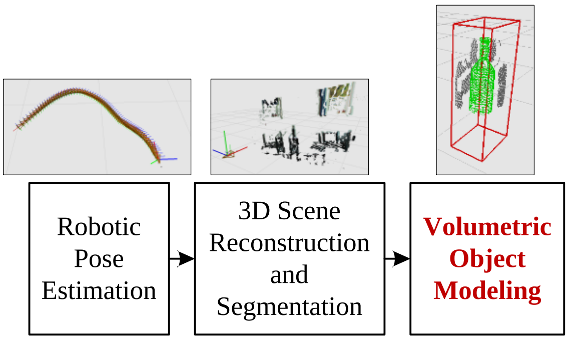

Robotic Pose Estimation
3D Scene Reconstruction and Segmentation
Volumetric Object Modeling

Fig. 1. Block diagram of the proposed scene perception system.

The reference coordinates of the obtained visual information is related to the online determined *position and orientation* (pose) of a robot which perceive the environment through a stereo camera configuration [4] or using and RGBD sensor [17]. Since the robot can operate both in indoor and outdoor environments, the use of stereo-vision, for outdoor scenes, and MS Kinect® [17] for indoor scenarios, is well justified. Once a certain pose is determined, the imaged scene can be reconstructed and segmented for the purpose of environment understanding. The main objective here is to get the depth, or disparity, map which describes the 3D structure of an imaged scene. This information is further used by the final *object structure estimation* algorithm, which is actually the main focus of this paper. One of the main algorithms used in the proposed vision system is the object classification method which delivers to the volumetric modeling method the object class and the 2D object ROI. The classification procedure is based on color and depth information. A detailed description of the approach can be found in [4]

3 Modeled based object structure estimation

The object structure estimation system is based on the active contour principle used to manipulate a set of pre-defined *Point Distribution Models* (PDM) by stretching them over a point cloud describing an object in a given 3D *Region of Interest* (ROI). In the considered process, three main challenges arise: the sparse nature of the disparity maps, for the case of stereo-vision configuration, the calculation of the 3D ROI and the nonlinear object modeling.

3.1 The Generic Fitted Primitive (GFP)

In the presented work, a Generic Fitted Primitive (GFP) is defined as a PDM model which serves as a backbone element for constructing a particular object, or shape. The generic PDM primitive is represented by a data structure that has as background component a shape vector $\boldsymbol{X}$ which contains 3D feature points describing the model of an object class. In order to keep the initial structure compact, a second vector Y is used to store the indexes of the un-deformed primitive triangulation such that after the modeling process the moving points can be easily followed. Such example models are shown in figure 2. Additionally, the structure contains a scale factor s, a rotation matrix R and a translation matrix t that relates the PDM to a canonical reference coordinate system.

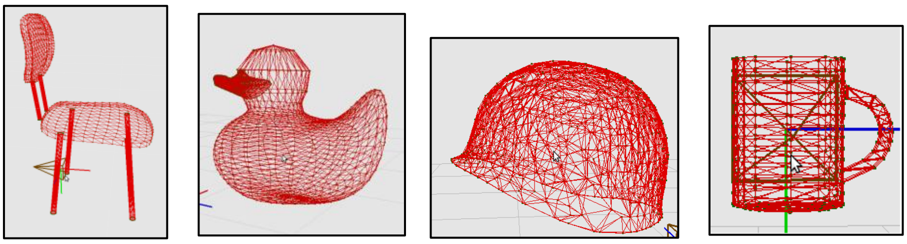


Fig. 2. Examples of meshed GFPs; (from left to right) chair, duck, helmet and mug.

Since in the considered source of visual information, that is disparity images or depth maps, only one perspective of the imaged object is available, the PDM model is actually used to augment the missing information. In this case, we consider objects that have a symmetrical shape. Nevertheless, the proposed approach can be applied on irregular shaped entities. Depending on the complexity and regularity of the surface object, the primitive model can be defined either by a low or a high number of 3D feature points. For example, the mug shown in figure 2 is described by 412 feature points. Since the PDM describing such an object represents an almost regular surface, not all these points are important for the object modeling process. In this sense, primitive points can be divided into two main categories. The first category is represented by *regular points*, or points with low discriminative power which usually form constant geometrical surfaces. The second category assembles the so-called *control points*, namely those points that define the shape of an object. Furthermore, control points can be automatically determined based on three main characteristics [2]:

1. Points marking the center of a region, or sharp corners of a boundary;
2. Points marking a curvature extreme or the highest point of an object;
3. Points situated at an equal distance around a boundary between two control points obeying rule 1.

In the same time, control points can be determined manually under the guidance of a human [16]. This last method captures the features of an object more efficiently but suffers from subjectivity on features definition since the process is controlled by a human person. Depending on the modeled object, in our approach we used both the automatic and the manual techniques to determine control points. Using the introduced points, the computation time is increased since the number of points describing the shape of an object is usually much lower than the total number of points from the GFP. The 3D positions of the GFP points are actually directly dependent on the positions of the control points, as it will later be shown in this section. For example, from a total of 1002 points describing the helmet primitive from figure 2, only 473 of them (marked with red dots) are considered to be control points. On the other hand, for a complex object, this number can be equal to the initial PDM features number, meaning that all points from the primitive are considered to be control points since all of them are needed to capture a specific feature. Taking into account a lower number of control points will considerably increase the computational speed of the modeling process.

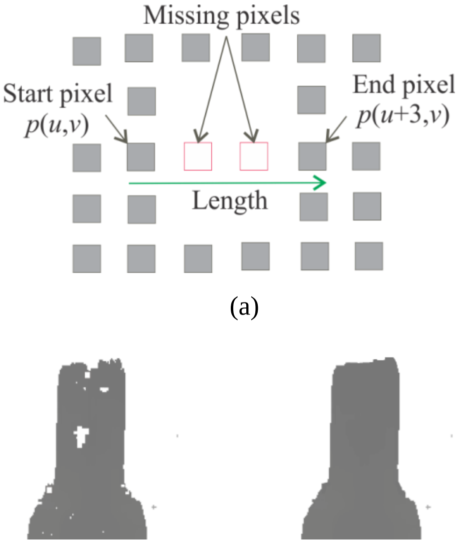
3.2 Disparity Map Enhancement

The presented modeling principle accepts as input information a dense point cloud of the object which structure will be estimated. In this sense, the MS Kinect® [17] sensor has no problem in providing such dense information, whereas the stereo camera configuration outputs a sparse disparity maps. Namely, it contains “holes” or discontinuities in areas where no stereo feature matching exists [1]. Such discontinuities are present in low textured regions or constant color areas from the environment.

To overcome this issue we propose an enhancement method which deals with disparity maps discontinuities. Basically, the idea is to scan each point from a disparity image and determine if there is a gap between the considered point and a neighboring point situated at a certain distance, as shown in figure 3(a). Since we apply the principle on disparity maps, which are defined on the 2D domain, there are only 5 main neighboring directions from a total of 8 in which we search for discontinuities. The untreated 3 directions refer to the back of the centered point and it is assumed that are no discontinuities in that direction since the position is already searched.

The disparity map is actually a grey scale image with pixel intensities inverse-proportional to the distance between the considered 3D world point and the camera’s position. Having in mind that the disparity image is represented using 8 bits, we sample it using 256 layers, each corresponding to certain intensity in the disparity domain. The enhancement process searches for discontinuities only in one layer at a time, since there is no information about the connectivity of the intensities. In this sense, the layers are stored using a 256 bins histogram. For each pixel in each layer the number of the same intensity along a direction is calculated. In order not to merge two different objects, the search area is reduced to a finite value, dynamically calculated. The search process starts from the lowest neighboring distance value, which has a two pixels length, and ends when a discontinuity is found or the maximum length is reached. The discontinuity is determined by comparing the length of the direction with the number of the same intensity pixels found along this direction. If the number of pixels found is below the length of the considered direction, the missing positions are filled with pixels with the same intensity as the ones already found.

There is a slight chance that two closely positioned objects are merged by the algorithm. In order to overcome this challenge, a variance driven image of the disparity map has been used [15]. From the variance image only object contours are extracted. In this way it can be determined if the points which take part in the fill process belong to one single region or to a neighboring region. The result of the presented method will be a compact and dense representation of the disparity image, as shown in figure 3(c). On the other hand, it is needed to connect the layers which are very close (in terms of disparity) in the 3D model. This can be achieved by diffusing the gradient separating two neighboring intensities. In order to preserve the 3D features of the object, the diffusing process will occur only for regions with translation of intensity no grater then 5 intensities layers. In this way, the obtained layers are smoothly connected.



(a)

(b) (c)

Fig. 3. Disparity enhancement algorithm. (a) Missing pixels searching principle. (b)Original disparity map. (c) Enhanced disparity image.

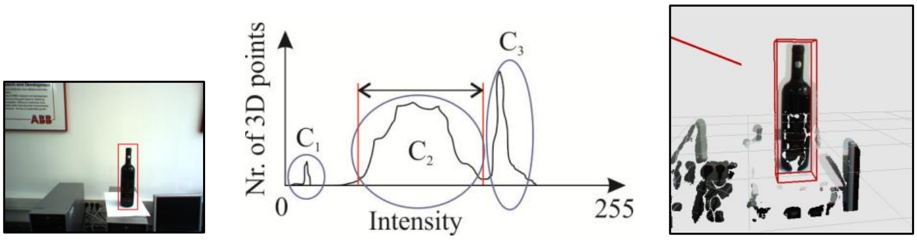
3.3 3D ROI definition

Because of the complexity of the scene, it is difficult to apply the modeling process directly on the entire scene. To overcome this issue, we propose the definition of a local frame attached directly to the object which we want to model, rejecting from the scene all the redundant information. This process starts in the 2D domain by segmenting and classifying the objects from the scene and providing, besides the class to which the object belongs, 2D ROI's feature vectors $[p_l\ , p_R]$, $i = 1,2,3,4$. This description restricts the object search area to a quadratic region of interest. For the stereo-vision configuration the 3D ROI is determined by computing the disparity between the left and right ROI points. In this way only a planar representation (slice) in the 3D space is obtained. The volumetric property is evaluated starting with the assumption that the pixels inside the 2D ROI describe only the object. The depth is determined statistically by finding the highest density of 3D points which lie inside the planar ROI along the Z 3D Cartesian axis. A 3D representation of the ROI can be seen in figure 4. However, there is a possibility that the highest density of 3D points belongs to a noise entity outside the object border but still inside the ROI.

To overcome this problem, a histogram of the disparity image is calculated. Instead of searching only the top density value of the intensities in histogram, we check also the highest aperture of the histogram for the considered top density. Basically we determine the highest distribution of connected points by summing all the densities from the slices of the aperture belonging to a top value of the histogram as:

$$d = \sum_{i+a}^{i+b} \max\big(h(i)\big)\,, \tag{1}$$

where d represent the highest cluster of 3D points, $h(i)$ is the number of pixels for a certain bin i and a, b are the closest and farthest non zero $h(i)$ relatively to the considered intensity i, respectively. The margin of the aperture is actually defining the first and last planes of the 3D ROI volume along the Z axis, respectively.



(a) (b) (c)

Fig. 4. 3D ROI computation. (a) Input image together with the calculated 2D ROI. (b) Histogram cluster segmentation. (c) 3D ROI re-projection.

For the case of the MS Kinect sensor, the ROI formulation is trivial. The final bounding box can be easily extracted by a simple selection, from the depth map, of the points laying inside the segmented 2D ROI.

3.4 PDM Shape Alignment

The 3D alignment process deals with the calculation of the rotation and translation of the primitive shape with respect to the point cloud distribution inside the 3D ROI. Because each PDM, that is primitive and point cloud, is defined in its own coordinates system, a similarity transformation is used to align the two models. Since the ROI's PDM is related to the same coordinate system as the 3D virtual environment, we have chosen to bring the primitive's PDM into a reference 3D environment coordinate system. The reference coordinate system is given by the on-line determined pose of the stereo camera [4]. In this sense, the primitive is considered to be a translational shape, while the 3D ROI is marked as a static cube. The similarity transformation is described by:

$$X_{new} = sR(X_{old} - t), \tag{2}$$

where, x_{old} and x_{new} represent the 3D coordinate of a point before and after the similarity transformation, s is a scale factor, while R and t are the matrices defining the rotation and translation of a point, respectively. These coefficients represent the Degrees of Freedom (DoF) of a certain shape.

The scale factor is determined based on the 3D point cloud information inside the ROI. Since a disparity enhancement is considered before the 3D re-projection process, it can be presumed that inside the ROI exist one or more large densities of points which describe the object of interest. For each model or point cloud is determined the radius of a centered sphere which embeds the respective point cloud. By computing a ration between these radiuses, a scale factor can be determined.

The translation of the moving shape is easily determined by adding the center of gravity of the points inside the 3D ROI from the center of gravity of the primitive PDM, as follows:

$$t = \frac{1}{n_p}\sum_{i=1}^{n_p} a_i + \frac{1}{n_p}\sum_{j=1}^{n_p} b_j, \tag{3}$$

where, n_p represent the number of points of the model and a and b are the two densities of points, that is of the object's shape (primitive model) and of the fixed point cloud inside the 3D ROI. The rotation between the two models is not so trivial to obtain. Because one of the models represents only a perspective of the second one, the rotation can be determined by matching these two entities. This operation starts by computing the shape descriptors for each model based on the FPFH descriptor [18]. Next, a Sample Consensus Initial Alignment (SAC-IA) algorithm [18] is used to determine the best transformation between the two entities. This transformation contains also the rotation of the primitive relative to the reference point cloud, which is the scene object.

By using the proposed alignment method, a rough object volumetric estimation is obtained based on the fitting primitive principle. An example result of the similarity transformation can be seen in figure 5, where, the red silhouette represents the PDM primitive shape whereas the blue model represent the object from the scene.

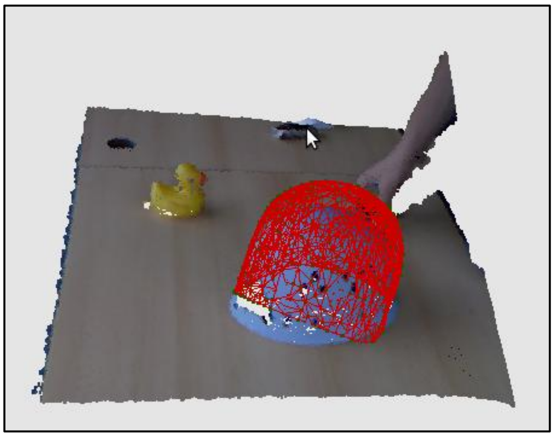


Fig. 5. Primitive PDM shape alignment example.

3.5 PDM Primitive modeling

The points which drive the modeling process are the control points described in a previous subsection. The behavior of the other points in the PDM model is automatically derived from the movement of the control points. The modeling process is achieved by dragging after each control point the neighbors from the surrounding area. Each of the neighbor point is moved based on a physical relation describing the property of the considered object. This relation can be either linear, as in equation 4, or non-linear for more complex surfaces. For simplicity if explanations, we have considered in this work a linear relation between control and the rest of the PDM points:

$$x_{new} = x_{old} \left(1 + \frac{d_{curr}}{d_{max}}\right), \tag{4}$$

where, x_{new} and x_{old} represent the new and old 3D coordinates of the considered neighboring points, respectively, d_{max} is the distance between the control point and the farthest neighbor within the affected area and d_{curr} represent the distance between the control point and the translated neighbor. The results of such a linear modeling are shown in figure 6, where control points are marked with red, their neighbors are labeled with blue the rest of the PDM points are green. The surrounding dragged area has the shape of a cube centered on the control point 3D coordinate and has its area defined as double the distance between the initial and the new position of the control point, as depicted in figure 6.

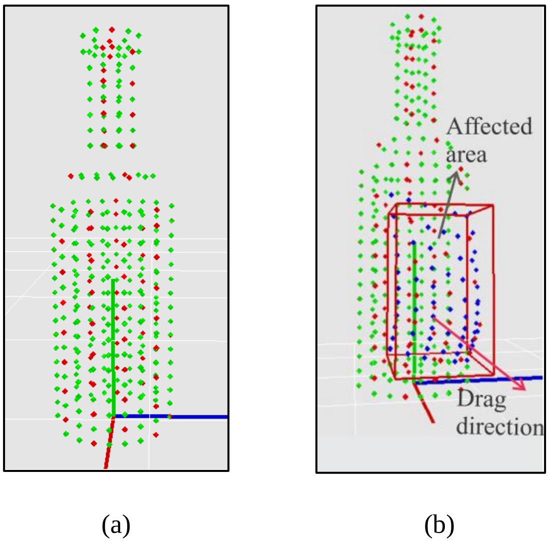


Fig. 6. A linear dependency between a control point and its neighbors. (a) Initial PDM shape model.(b) Point deformation along a considered direction.

The proposed approach for estimating an object's volume starts with a generic object PDM model, namely a primitive, and ends by capturing by each primitive control point the local features of the modeled object of interest. As explained, this is achieved by minimizing the distance between the control point and the PDM in a respective neighborhood. The minimization procedure is based on the *active contours* principle, better known as *snakes* [7]. This approach represents a deformable contour model of the considered object.

In an image, an active contour is a curve parameterized by the minimization of and energy functional:

$$\varepsilon(c) = \varepsilon_{int}(c) + \varepsilon_{ext}(c) = \int_0^1 \left[E_{int}\big(c(s)\big) + E_{ext}\big(c(s)\big)\right]ds, \tag{5}$$

where, E_{int} and E_{ext} are the internal and external energies, respectively and $c(s) = [x(s), y(s), z(s)]$ represents the curve describing the object's surface, while $s \in [0,1]$. By defining an initial contour within an image, it will move under the influence of internal forces computed based on the derivatives from the active contour an also under the influence of the external forces captured from the image.

In the presented 3D object modeling approach, the image domain is equivalent to the 3D representation of the scene. For this reason the same energy minimization principle has been used to model the shape of an object. Instead of using an initial active contour, as in the original method, we propose the use of a 3D generic primitive PDM model. The movement of the contour surface is thus described by the direction of the lowest functional energy, that specific region actually corresponding to a probable contour in the image [19]. In the considered 3D case being the highest density of points from the 3D scene.

The idea of using forces to move the primitive points is that the primitive PDM must be attracted and fitted on the border of the object. The internal forces, which refer exclusively to the primitive PDM, are responsible for supervising the shape smoothness and continuity. As described in the equation 6, the continuity property is controlled by the first derivate while smoothens is define by the second derivate of the surface.

$$E_{int} = E_{cont} + E_{curv} = \frac{1}{2}(\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2), \tag{6}$$

where, E_{cont} is the energy responsible for the continuity of the surface and E_{curv} deals with the bending property of the hull of the object. α and β are two parameters which influence the E_{cont} and E_{curv} forces, while $v(s) = [x(s),\ y(s),\ z(s)]$ represent the coordinates of a point from the shape vector $\boldsymbol{X}$. In the discreet domain the two energies can be rewritten as:

$$E_{cont} = (x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2, \tag{7}$$

$$E_{curv} = (x_{i-1} - 2x_i + x_{i+1})^2 + (x_{i-1} - 2x_i + x_{i+1})^2 + (x_{i-1} - 2x_i + x_{i+1})^2 \ , \tag{8}$$

where, $x_i, y_i, z_i \in \mathbf{X}$, $i = 0, \ldots . n_p$ and n_p is the number of points in the shape PDM vector. In the original formulation of the principle, each point from the shape can be moved in one of the eight possible 2D directions. In current 3D approach, because of the third dimension, a number of 24 directions are taken into account.

The correct moving direction is mainly influenced by the external energy E_{Ext} which evaluates for each direction the highest density of 3D points. Because this density can be spatially very close, a weight factor for the external energy is introduced. Thus, if these candidate positions have an appropriate number of points, the weight factor will be considered zero.

Since we have only one perspective of the object, there are large object areas with no 3D point cloud description needed to drive the contour energies. The un-imaged back side of an object represents such an example. In the proposed approach, the missing information is filled by the generic data introduced by the PDM primitive model.

4 Experimental results

The main objective of the presented approach is robust 3D object structure estimation for appropriate object grasping and manipulation in service robotics. In order to prove the reliability of the concept, different types of objects were estimated, starting from more simple models (e.g. mug and bottle) and ending with high irregular geometrical shapes (duck, chair or helmet). Since the robot can operate in any environment, the experiments were conducted both outdoor, using a Point Grey Bumblebee® stereo camera system [20], and indoor using a MS Kinect® RGBD sensor [17]. The sensing devices were mounted on a Pioneer P3 – DX robotic platform [21] equipped with a 7 DOF Cyton 2 robotic arm [22]. The grasp plan action was performed using *GraspIt!* library [23].

The *Ground Truth* (GT) data against which the proposed method has been tested is composed of a number of manual measurements conducted on the objects: width, height, thickness, translation and rotation. The translation and rotation was measure with respect to a fixed reference coordinate system represented by an ARToolKit® [24] marker.

4.1 Mug modeling

The mug primitive structure is defined by 412 points out of which only 258 are defining the actual structure of the shape (control points). Three different mugs were placed on a flat surface, in and indoor scenario. The scene was perceived using the MS Kinect sensor. The structure estimation method was applied independently on each segmented mug. The time consumed for the entire estimation process, including segmentation, classification, initial raw alignment and modeling process of all considered shapes, was 11.76 seconds on an Intel Pentium I5 2.30 GHz platform. The result obtained is depicted in figure 7.

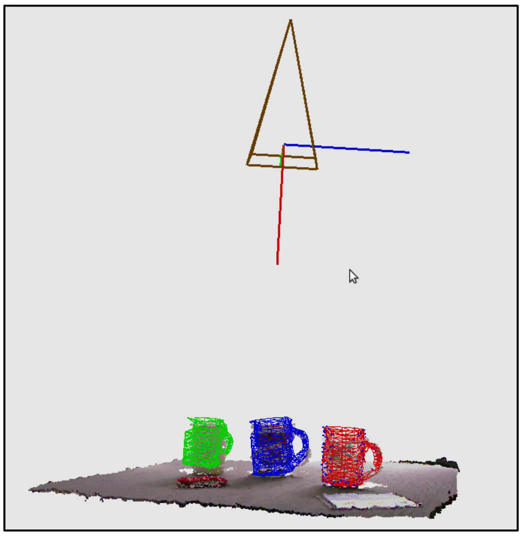


Fig. 7. Segmented modeled mugs

The first two mugs (red and blue) were accurately positioned, with a pose error below 3%, while the green mug positioning produced an increased error because of the occluded part of the handle. Lacking distance information, the handle was positioned randomly in the unseen part of the model, producing in the end a 3D volume with a low confidence for grasping. Regarding the first two models, more than 65% of primitive points were repositioned during the structure estimation process, leaving the rest of 35% points to generally estimate the occluded parts of the mugs. Based on the new

3D definition of the model, the grasp action had a successfully grasp rate of over 95%. In figure 8, the difference between the initial 3D structure of the primitive (presented as a green mesh) and the modeled one (red mesh) can be intuitively observed.

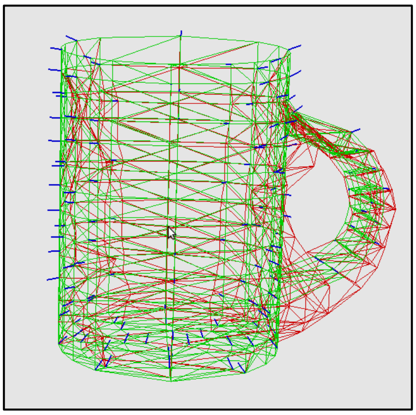


Fig. 8. Mug GFP meshes before (green) and after (red) the fitting process. The blue lines represent the directions of the features normals.

4.2 Helmet modeling

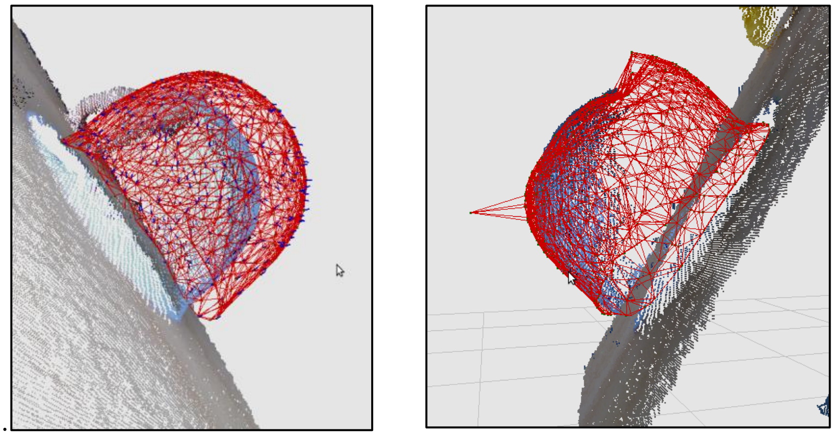
Representing a large volume in reality, a helmet can be difficult to estimate because of the elongated shape and also since it produces large occluded regions. The primitive is defined by only 1002 feature points whereas the sensed visible part is described through 15378 feature points. In order to efficiently align and model the primitive, the scale ratio between the primitive and the object cloud, together with a raw transformation matrix, must be correctly determined. For the cases where the object is seen directly from the front it is almost impossible to correctly estimate the scale since the centered sphere will embed a small cloud representing only a part of the object. In this sense, the object must be observed from a more distanced view point. An initial raw alignment for a helmet can be observed in figure 5.

After the modeling process, 734 out of 1002 primitive points have captured the local geometry of the object of interest, creating in the end a meaningful compact volume. The result of such reshape action can be observed in figure 9. Because the primitive helmet (war helmet) is rather different from the imaged helmet (climbing helmet), the occluded part is more protuberant and can be easily observed. During tests, the presented modeling method has proved its efficiency and robustness as long as the observed object does not have more than 60% of its surface occluded. Above this value, the scale and the translation of the primitive are ambiguous.

The statistical measures of achieved errors in all the experiments are given in Table 1.

Table 1. Performance evaluation results for the proposed GFP estimation approach.

Object	Alignement						Modeling			
	Rotation (%)			Translation (%)			Primitive Points		Volumetric error (%)	Ocluded region (%)
	ϕ	θ	φ	x	y	z	Total	Modeled		
Mug	4.1	1.2	2.4	0.1	0.74	1.7	412	258	2.78	47
Helmet	2.5	0.7	2.4	0.25	0.14	1.2	1002	734	3.12	28
Duck	5.1	3.2	4.2	1.3	2.1	3.1	2329	1452	5.72	45
Chair	0.7	1.3	1.5	2.01	0.1	0.8	842	548	1.35	19
Bottle	0.1	0.4	0.3	0.5	0.6	1.15	382	305	2.71	36



(a)

(b)

Fig. 9. Object structure estimation. (a) Initial primitive alignment between the primitive (red mesh) and object point cloud (blue point cloud). (b) Final modeled primitive.

5 Conclusions

In this paper, an object volumetric modeling algorithm for objects of interest encountered in real world service robotics environments has been proposed. The goal of the approach is to determine as precisely as possible the 3D particular surface structure of different objects. The calculated 3D model can be further used for the purpose of visually guided object grasping. As future work the authors consider the time computation enhancement of the proposed procedure through parallel computational devices (e.g. Graphic Processors), as well as the application of the method to other computer vision areas, such as 3D medical imaging.

Acknowledgements

This paper is supported by the Sectoral Operational Program Human Resources Development (SOP HRD), financed from the European Social Fund and by the Romanian Government under the projects POSDRU/107/1.5/S/ 76945 and POSDRU/89/1.5/S/59323.

References

1. Brown, M., Burschka, D., Hager, G.: Advances in Computational Stereo. In: IEEE Transaction on Pattern Recognition and Machine Intelligence, vol. 25(8), pp. 993–1008 (2003)
2. Cootes, T., Taylor, C., Cooper, D., Graham, J.: Active Shape Models-Their Training and Application. In: Comp. Vision and Image Understanding, vol. 61(1), pp. 38-59, New York (1995)
3. Criminisi, A., Reid, I., Zisserman, A.: Single View Metrology. In: International Journal on Computer Vision, vol. 40(2), pp. 123–148,MA, USA (2000)
4. Grigorescu, S., Cocias, T., Macesanu, G., Puiu, D., Moldoveanu, F.: Robust Camera Pose and Scene Structure Analysis for Service Robotics. In: Robotics and Autonomous Systems, vol. 59(11), pp. 899-909, Nederland (2011)
5. Hartley,R., Zisserman, A.: Multiple View Geometryin Computer Vision, Cambridge University Press, Great Britain (2004)
6. Horry, Y.,Anjyo, K., Arai, K.: Tour into the Picture: Using a Spidery Mesh Interface to Make Animation from a Single Image. In: Proc. ACM SIGGRAPH, pp. 225 – 232, New York (1997)
7. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active Contour Models. International Journal of Computer Vision, vol. 1(4), pp. 321-331(1998)
8. Kim,T., Seo,Y., Lee,S., Yang,Z., Chang,M.: Simultaneous Registration of Multiple Views with Markers. In:Computer-Aided Design, vol. 41(4), pp. 231–239 (2009)
9. Kutulakos, K., Seitz, S.: A Theory of Shape by Space Carving. In: International Journal of Computer Vision, vol. 38(3), pp. 199–218, New York (2000)
10. Lhuillier, M., Quan, L.: A Quasi-Dense Approach to Surface Reconstruction from Uncalibrated Images. In: IEEE TPAMI, vol. 27(3), pp. 418–433, New York (2005)
11. Matsuyama, T., Wu, X., Takai, T., Wada, T.: Real-Time Dynamic 3-D Object Shape Reconstruction and High-Fidelity Texture Mapping for 3-D Video. In: IEEE Transactions on Circuits and Systems for Video Technology, vol. 14(3), pp. 357–369, New York (2004)
12. Stamos, I., Liu, L., Chao, C., Wolberg, G., Yu, G., Zokai, S.: Integrating Automated Range Registration with Multi-view Geometry for the Photorealistic Modeling of Large-Scale Scenes. In: International Journal of Computer Vision, vol. 78(2/3), pp. 237–260, Springer, Heidelberg (2008)
13. Sturm, P., Maybank, S.: A Method for Interactive 3D Reconstruction of Piecewise Planar Objects from Single Images. In: BMVC, United Kingdom (1999)
14. Trucco, E., Verri, A.: Introductory Techniques for 3-D Computer Vision, Prentice Hall PTR (1998)
15. Turiac, M., Ivanovici, M.,Radulescu, T.: Variance-driven Active Contours. In*:* International Conference on Image Processing, Computer Vision, and Pattern Recognition, Las Vegas, Nevada, USA (2010)

16. Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., and Comaniciu, D.: Four-Chamber Heart Modeling and Automatic Segmentation for 3D Cardiac CT Volumes Using Marginal Space Learning and Steerable Features. In: IEEE Transaction on Medical Imaging, vol. 27(11), pp. 1668–1681 (2008)
17. Microsoft motion sensing devices, `http://en.wikipedia.org/wiki/Kinect`
18. Rusu, R. B., Blodow, N., Beetz, M.: Fast Point Feature Histograms (FPFH) for 3D Registration. In: International Conference on Robotics and Automation, pp. 3212 - 3217 Kobe, Japan (2009)
19. Nixon, M. S., Aguado, A. S.: Feature extraction and image Processing, Academic Press, (2002)
20. Point Grey Research's, `http://www.ptgrey.com/products/bumblebee2/bumblebee2_stereo_camera.asp`
21. Autonomous mobile robots, software and accessories, `http://www.mobilerobots.com`
22. Autonomous mobile robots, manipulators, `http://www.mobilerobots.com/accessories/manipulators.aspx`
23. Miller, A., Allen, P. K.: Graspit!: A Versatile Simulator for Robotic Grasping. In: IEEE Robotics and Automation Magazine, vol. 11(4), pp. 110-122 (2004)
24. Augmented Reality (AR) research department, `http://www.hitl.washington.edu/artoolkit/`