Dragan Lazarevic[1)]
Milan Misic[1)]
Bojan Stojcetovic[1)]

1) Higher Technical School of Professional Studies Zvecan,
lazarevicddragan@yahoo.com
m.misic@vts-zvecan.edu.rs
bstojcetovic@yahoo.com

3D MESH SEGMENTATION FOR CAD APPLICATIONS

Abstract: *3D mesh segmentation is a fundamental process for Digital Shape Reconstruction in a variety of applications including Reverse Engineering, Medical Imaging, etc. It is used to provide a high level representation of the raw 3D data which is required for CAD, CAM and CAE. In this paper, it is presented an exhaustive overview of 3D mesh segmentation methodologies examining their suitability for CAD models. In particular, a classification of the various methods is given based on their corresponding underlying fundamental methodology concept as well as on the distinct criteria and features used in the segmentation process.*
Keywords: *3D objects, mesh segmentation, shape analysis.*

1. INTRODUCTION

3D mesh segmentation is a crucial step in the pipeline of Digital Shape Reconstruction, for which the exploitation of high level semantics in 3D raw data is addressed, enabling a semantic richness useful for CAD, CAM and CAE purposes. Usually a 3D physical object is sampled with a laser scanner and the output is a set of 3D points which represent the surface of the object. This set is triangulated acquiring by this way a topological structure of the surface which is called a 3D mesh. Segmentation is the process which provides the necessary organization of the data points by partitioning them into connected regions or parts that can be approximated by standard CAD surfaces (e.g. planes, cylinders, etc.) or volumetric primitives (e.g. super-ellipsoids). There is a variety of algorithms for 3D mesh segmentation, which can be grouped in two basic categories:

- Surface-based: The 3D mesh is segmented into regions which represent distinct surfaces of the CAD model and can be approximated by various primitives like planes, cylinders, spheres, polynomials, etc.
- Part-based: The 3D mesh is segmented into volumetric parts which can be approximated by volumetric primitives (e.g. super-ellipsoids).

The quality of segmentation is a crucial issue that is directly related to the corresponding application which imposes particular requirements. For surface-based algorithms it is usually required that (i) the boundaries of the segmented regions should be smooth; (ii) the extracted regions should be able to be approximated by smooth surfaces; and (iii) the boundaries where the regions meet should allow certain types of continuity to hold for the approximating surfaces. For part-based algorithms a variety of criteria can be used in order to be able to extract the meaningful parts of the object. The quality of segmentation is also directly dependent on the type of the CAD object that is being processed. Different algorithms work better for each CAD object type. It is a critical step toward content analysis and mesh understanding. Although some supervised methods exist [1, 2], most existing techniques are fully automatic. According to recent states-of-the-art [3, 4], mesh segmentation techniques can be classified into two categories: surface-type (or geometric) methods and part type (or semantic) methods. In the first case, the algorithms are based on low level geometric information (e.g. curvature [5]) in order to define segments (i.e. regions) with respect to geometric homogeneity, while in the latter case, the algorithms aim at distinguishing segments that correspond to relevant features of the shape, by following higher level notions in human perception theory [6]. This kind of approach is particularly suited for object animation deformation and indexing applications, where the decomposition has to be meaningful. Although development of mesh segmentation algorithms for both approaches has drawn extensive and consistent attention, relatively little research has been done on segmentation evaluation. For the firrst approach (surface-type), some tools exist depending on the end application as texture mapping [7] or medical imaging [8]. Recently, two main works, Benhabiles et al. [9] and Chen et al. [10], have been proposed to study the quality assessment problem of part-type 3D-mesh segmentation. Both works propose a benchmark for segmentation evaluation which is based on a ground-truth corpus.

The corpus is composed of a set of 3D-models grouped in different classes and associated with several manual segmentations produced by human observers. These two benchmarks comprisethe ground-truth corpus and a set of similarity metrics, then the evaluation of a segmentation algorithm consists in measuring the similarity between the reference segmentations from the corpus and that obtained by this algorithm (on the same models). In this kind of benchmark the quality of the evaluation depends on the quality of the corpus but also on the quality of the segmentation similarity measure. This leads to conclude that the choice of an accurate measure is quite critical in order to provide a strict evaluation and to reject the real quality of an automatic segmentation with comparison to a manual one. In this context, less efforts were investigated to propose a reliable measure of mesh segmentation similarity. Indeed, the previous works [9, 10] focused their interests on the design of the ground-truth corpus and presented rather simple metrics su_ering from degeneracies and low discriminative power.

Three dimensional Point Cloud (3DPC) is the collection of three dimensional coordinate system which are obtained by 3D scanners such as Light Detection and Ranging (LIDAR), laser range under and microsoft kinects. Each point usually contains x,y and z coordinates value to the shape and geometry of an object. The intensity value and RGB color value of the scene can be stored in the point cloud for the additional information. 3D point cloud segmentation is the technique of classifying point clouds into multiple isolated regions based on voxels salient characteristics like color or the intensity value [11]. Each region of the point cloud which have same characteristics will have the same label value. In machine vision segmentation of point cloud is important to analyze the scene in various aspects for the purpose of locating and recognizing objects in a scene and separating between interesting geometry (object) and not interesting object (background). In 2D subdividing the pixels or the digital images is called image segmentation. The image segmentation is usually more applicable in medical imaging for the purpose of locating tumors and treatment planning.

2. METHODS OF 3D MESH SEGMENTATION

The segmentation of point cloud can be performed by two methods, one based single point technique and another one considering neighboring points attributes. There are various techniques and algorithm being developed and investigated for the segmentation of point cloud. There is neither a universal approach nor objective criteria to judge the state of the art technique for the segmentation method as the different methods are used for different purposes. Normally in order to segment a 3D point cloud, the point cloud are projected into 2D panorama images and the 2D images are segmented applying certain image segmentation method. The segmented 2D coordinates are then mapped to corresponding 3D coordinates. In this paper all the results are based on segmentation of equi-rectangular panorama images projected from the point clouds. All the images are intensity images. The experiments was performed in 3D toolkit which is an client point cloud processing software.

Mesh segmentation has become an important and challenging problem in computer graphics, with applications in areas as diverse as modeling [6], metamorphosis [9, 10], compression [11], simplification [5], 3D shape retrieval, collision detection [3], texture mapping [2] and skeleton extraction [3]. Mesh, and more generally shape, segmentation can be interpreted either in a purely geometric sense or in a more semantics-oriented manner. In the first case, the mesh is segmented into a number of patches that are uniform with respect to some property (e.g., curvature or distance to a fitting plane), while in the latter case the segmentation is aimed at identifying parts that correspond to relevant features of the shape. Methods that can be grouped under the first category have been presented for example in [8, 5], and may serve as a pre-processing for the recognition of meaningful features. Semantics-oriented approaches to shape segmentation have gained a great interest recently in the research community [4, 11, 12], because they can support parametrization or re-meshing schemes, metamorphosis, 3D shape retrieval, skeleton extraction as well as the modeling by composition paradigm that is based on natural shape decompositions. It is rather difficult, however, to evaluate the performance of the different methods with respect to their ability to segment shapes into meaningful parts. This is due to the fact that the majority of the methods used in computer graphics are not devised for detecting specific features within a specific context, as for example is the case of form-feature recognition in product modeling and manufacturing [2]. Also, the shape classes handled in the generic computer graphics context are a broadly varying category: from virtual humans to scanned artefacts, from highly complex free-form shapes to very smooth and feature-less objects. Moreover, it is not easy to formally define the meaningful features of complex shapes in a non-engineering context and therefore the comparison of the different methods is mainly qualitative. Finally, shape segmentation methods are usually devised to solve a specific application problem, for example retrieval or parametrization, and therefore it is not easy to compare the efficacy of different methods for the shape segmentation itself.

Mesh decomposition using fuzzy clustering and cuts [9, 10]. The key idea of this algorithm is to first find the meaningful components using a clustering algorithm, while keeping the boundaries between the components fuzzy. Then, the algorithm focuses on the small fuzzy areas and finds the exact boundaries which go along the features of the object.

Mesh segmentation using feature point and core extraction [13]. This approach is based on three key ideas. First, *Multi-Dimensional Scaling (MDS)* is used to transform the mesh vertices into a pose insensitive representation. Second, *prominent feature points* are extracted using the MDS representation. Third, the core component of the mesh is found. The core along with the feature points provide sufficient information for meaningful segmentation.

Tailor: multi-scale mesh analysis using blowing bubbles [9, 10]. This method provides a segmentation of a shape into clusters of vertices that have a uniform behavior from the point of view of the shape morphology, analyzed at different scales. The main idea is to analyze the shape by using a set of spheres of increasing radius, placed at the vertices of the mesh; the type and length of the sphere-mesh intersection curve are good descriptors of the shape and can be used to provide a multi-scale analysis of the surface.

Plumber: mesh segmentation into tubular parts [9, 10]. Based on the *Tailor* shape analysis, the *Plumber* method decomposes the shape into tubular features and body components and extracts, simultaneously, the skeletal axis of the features; tubular features capture the elongated parts of the shape, protrusions or wells, and are well suited for articulated objects.

Hierarchical mesh segmentation based on fitting primitives (HFP) [9, 10]. Based on a hierarchal face clustering algorithm, the mesh is segmented into patches that best fit a pre-defined set of primitives; in the current prototype, these primitives are planes, spheres, and cylinders. Initially each triangle represents a single cluster; at each iteration, all the pairs of adjacent clusters are considered, and the one that can be better approximated with one of the primitives forms a new single cluster. The approximation error is evaluated using the same metric for all the primitives, so that it makes sense to choose which is the most suitable primitive to approximate the set of triangles in a cluster. The set of models examined in this work are medical models, CAD models, models of human figures in various postures, models of animals, and a miscellanea class of shapes.

In the current state-of-the-art, integrating the images with 3D data like range images incorporates computer vision and photogrammetry in this area of research. Automatic 3D reconstruction from 3D point clouds or range image is still one of the active research areas that has many applications in forestry, urban planning, tourist information systems and so on. 3D data measurements are grouped with respect to their similarity measures to define meaningful, coherent and connected segments. Acquisition of the high accurate and dense 3D data can assist us in the direction of automatic extraction of building models. Image matching can be considered a renaissance in the modern photogrammetry due to generating dense and high accurate point clouds with low price in comparison with LiDAR data. Recently, many algorithms are focused on the extraction of planar surfaces especially extraction of roof facets for 3D building reconstruction that related works will be reviewed in the next section. However, building extraction is still challenging issue due to the complex building roofs, occlusions and shadows.

Availability of the 3D structured data for each pixel and intensity values from the high resolution aerial images in addition to usage of high performance computer can assist us to deal with the challenging issues and achieving better segmentation results with low price. Thus, this combination can be considered the important step to reach the goal of automatic 3D reconstruction and object recognition. The goal of this work is to improve and extend surface growing based segmentation in the X-Y-Z image in the form of 3D structured data with combination of spectral information of RGB and grayscale image to extract building roofs, streets. The advent of affordable and accurate range sensing hardware has had a tremendous impact on the development of fundamental skills of mobile autonomous robots. Basic capabilities such as navigation, obstacle avoidance, and localization and mapping in indoor and outdoor environments can nowadays be considered solved thanks to the availability of 2D laser range finders and the development of powerful algorithms that process this kind of data. As mobile robotics is gradually moving towards more complex scenarios such as mapping and interpretation of 3D environments, researchers are only now beginning to harvest

the rich information contained in 3D range data. The quest for algorithms that efficiently process, abstract and interpret this data involves, among other things, a vivid knowledge exchange with several related fields, such as computer graphics and computer vision. Relevant applications in mobile robotics encompass a wide spectrum, that ranges from 3D mapping of buildings and surface reconstruction of architectural heritage to model extraction for manipulation and semantic mapping in household environments. Given the available sensing modalities, a common problem is the detection and interpretation of medium-sized objects, such as living-room furniture, kitchen appliances etc. Existing works often make use of extremely dense point clouds on the order of millions of points, recorded from several view points. This data is then processed to extract very detailed geometric models of objects in the environment and finally use the generated models to do something useful. While this is possible for an initial offline learning phase, in which the robot is allowed to survey an environment un-interrupted, the acquisition of such detailed data is infeasible for everyday operation. In many cases, a robot that is to perform some useful task in a domestic setting must be able to repeatedly assess the state of its environment by performing a quick 3D scan and interpreting it. In this article, we consider the scenario of a robot using an actuated 2D laser range finder to acquire a relatively sparse point cloud within several seconds. A second issue that has been neglected in most works using actuated 2D laser range finders to acquire 3D data, is the fact that algorithms are usually only developed and tested for the sampling characteristics encountered in the specific scanning setup. The many different ways to actuate or sweep a 2D scanner may yield scans with very different densities and distinct sampling characteristics. The applicability of these algorithms to data acquired with a different setup is therefore not guaranteed. The main contribution of this article is that it addresses both of the aforementioned issues: the segmentation framework operates on relatively sparse data that is acquired and segmented within several seconds. More importantly, it yields consistent results across point cloud data obtained with different characteristic scan trajectories using one set of parameters for both simulated as well as real data.

3. COMPARING SHAPE MODELS OF CAD

Most CAD models are solid models that are defined parametrically. Due to the development of rapid prototyping and visualization areas, approximate shape models represented by a polygonal mesh and dense point clouds are becoming another useful alternative to CAD representations. Shape model representations of 3D objects are approximate models characterized by a mesh of polygons or a cloud of points for presentation or rendering purposes in computer graphics. Rather than exact parametric equations, polygons or densely sampled points are used to approximate curved surfaces. Only the geometry of triangles and points are stored without any topological information. In contrast to proprietary solid model formats, open mesh file formats such as VRML, STL, and ASCII point clouds are widely available. Although shape models are not suitable for many tasks in CAD/CAM systems, polygonal meshes can serve as the lowest common denominator in comparing CAD models. CAD mesh models can be generated by faceting solid models from different modeling systems. Shapemodels of objects can also be acquired easily by using 3D scanners or CT to enable comparison of digital and physical artifacts. From the polygon mesh, different transformation invariant attributes can be extracted as the means of similarity among 3D models. Thompson et al. [12] examined the reverse engineering of designs by generating surface and machining feature information off of range data collected from machined parts. The method of Osada et al. [13] creates an abstraction of the 3D model as a probability distribution of samples from a shape function acting on the model. Novotni and Klein [14] demonstrated the use of 3D Zernike descriptors. Kazhdan et al. [15] compared 3D models with spherical harmonics. While these techniques target general 3D models, Ip et al. [16, 17] focused on comparing shape models of CAD with shape distributions. Iyer et al. [18] presented a CAD oriented search system, based on shape, voxelization and other approaches. Pal et al. [19] extracted features from CAD models using genetic algorithms. Cardone et al. [20] compared prismatic machined parts by using machining features. Various database techniques for CAD are discussed in [6, 7, 12]. Recently, research efforts in industry and academia are examining the use machine learning techniques to train a 3D shape recognition system with CAD data. Work in industry has explored the use of neural networks to identify parts based on multiple 2D views [21]. Hou et al. [22] attempted to use shape information to cluster the semantics of parts with SVMs. In the context of shape model matching, Elad [23] used linear SVMs to adjust retrieval results from a 3D shape database according to users' feedback. Ip et al. [24] classified models according to manufacturing processes by a curvature descriptor and SVMs.

Figure 1 - 3D meshes of CAD models: example of a model that consists of free-form surfaces, the dual graph a mesh denoted by red lines and segmentation of a mechanical part

There are recent approaches that employ partial matching of models. Bespalov et al. [25] used scale-space representations to segment different features of meshes. Funkhouser et al. [26] partially matched shape features according to different priorities. More extensive surveys and literature reviews in this area can be found in references [5], [18], and [16]. The availability of 3D scanning technologies (Laser, white light, and CT scanners) has stimulated the interest in 3D point cloud alignment and registration. Given two point clouds with overlapping regions, registration based on iterative closest points (ICP) aims to rotate and translate a point cloud to match the other one. Because laser scanners and range finders often come with limited measure volume, registration becomes a critical process when acquiring 3D images of large scale parts in the industry. Since Besl et. al [27] published the original ICP algorithm, there have been many variations with different kind performance improvements in some of the recent work. Rusinkiewicz et al. [28] published a survey of the ICP techniques and demonstrated a fast variant that registers point clouds in real time. Mitra et al. [28] optimized the registration according to the point cloud geometry. Research in partitioning triangular meshes into separated meaningful surface patches is of great interest for many applications, such as, shape simplification, compression, analysis, and recognition. Segmentation of Point Cloud and CAD Mesh into Surface Patches Point clouds and CAD meshes are segmented into surface patches using an identical algorithm. It is important to apply the same approach to both the point cloud and the CAD mesh to ensure the similar surface patches are produced from the matching point cloud and CAD mesh. Partial matching of 3D models is a challenging problem for many global shape descriptors. The shape of a partial scan often differs from its complete scan counterpart, e.g. the change of total length, width, and height. Hence, many global shape descriptors will discriminate a model against its own fragments. In addition, many 3D scans are imperfect. Hence, lengthy post-processing is often required to fill holes and remove noises from the point cloud. In attempt to alleviate these issues, we first segment the point clouds and CAD meshes into local patches and use them as matching units. This approach removes the gross shape dependency problem by separating both the partial scan and the CAD meshes into similar local surface patches that can directly be compared. Any extra patches from the CAD mesh will be ignored during evaluation. The segmentation procedure also allows us to discard insignificant patches, which are possibly noise, from the scanned point cloud. The surface patches of point clouds and meshes are created according to their surface curvature values. This simple method is generally sufficient to partition CAD surfaces. For complex freeform surfaces, more sophisticated or semantic based segmentation algorithm may be required in future. Curvature defines the variation of surfaces patches and it is a popular criterion among many previous segmentation approaches. The identical segmentation algorithm is applied to both point clouds and CAD meshes. This allows similar patches to be generated on corresponding point clouds and CAD meshes. It is very important to ensure the patches of the matching point clouds and meshes are close enough. These patches will be used as matching primitives and they will be compared with one another. Since the surface patches are similar, it is not necessary to perform many-to-many matching on the surface patches. Total curvature is computed from the normal vectors distribution of local neighborhoods on the surface. Normal vectors on the mesh model are sampled according to the mesh connectivity, for smooth meshes, normal vectors in a 1-ring neighborhood are sufficient for curvature computation. At the same time, normal vectors on the point cloud are estimated by normals of the best fitted planes of small neighborhoods of points. Following the method described in [21], the total curvature of a small neighborhood can be estimated by the norm of the covariance matrices of its normal vectors. Neighboring points and triangles that share similar curvature are grouped into patches.

For CAD models, the segmentation of the surface into patches of simple geometry is usually considered a pre-processing for the more complex recognition of form-features; in this context it is possible, or easier, to define precise geometric and morphological rules to detect certain configurations, even if the problem is not fully solved [2].

Methods like Plumber are also based on an a-priori knowledge about the features we want to extract, that are in this case defined as generalized sweep-like features. Plumber, indeed, performs better on features with elongation axis larger than section axis: in the tiger model, for instance, only the tail is recognized correctly as a tubular feature while the body is not identified as a tubular feature because its section is almost equivalent to its length. For articulated objects that are used in applications such as skeleton extraction, metamorphosis and retrieval, it is expected that the meshes be segmented at their joints. In this case, deep concavities as well as the size of the components, indicate the locations of segment boundaries. As discussed in [14], a segmentation can be partitioned into two sub-problems: the extraction of the segments and the smooth refinement of the cuts. There are, however, methods, such as Plumber that guarantee by definition a smooth boundary. For other methods, that do not inherently produce smooth boundaries, a post-processing stage that refines the boundaries can be added. This was done, for instance in [14] and [13], where a minimum cut algorithm was applied to the initial segmentation. It is important to mention, however, that not all applications require smooth boundaries. Multi-scale segmentations can be exploited to get a global segmentation. For example, the segmentation into patches of uniform behavior provided by Tailor highlights well detail-features rather than bigger shape components, but the persistence of the labeling across different scales give less sparse clusters [22].

Segmentation as an important part of data processing is applied on the various kinds of data sets like 3D point clouds or range image to partition data sets into meaningful, disjoint and connected segments with homogenous property. Hence, the purpose of the segmentation is to group points or pixels with similar features into segments. Segments are smooth surfaces that are achieved by grouping neighboring points or pixels with similarity measures, such as the direction of a locally estimated surface normal or intensity values of each pixel. Generally speaking, point cloud segmentation can be considered a difficult subject, especially in presence of the noise. In addition the gaps between point clouds (mismatch points from image matching for each image pixel) and varying point densities make it more problematic. Methods of surface extraction can be categorized in two main groups. Firstly, surface parameters can be estimated directly by clustering or finding maximum parameter in the parameter space. Secondly, point clouds can be segmented on the basis of proximity of the point clouds or similarity measures like locally estimated surface normals [3]. In other word, range segmentation problem can be categorized in two main approaches: region-based and edge-based segmentation problem. Furthermore, region-based segmentation problem can be divided in two main groups: parametric model-based segmentation algorithms and region-growing algorithms [4].

Figure 2 - Adaptive segmentation, segmented corridor and refining segmentations by different criteria (the initial curves are in red and the resulting curves are in blue)

4. EDGE-BASED SEGMENTATION MODELS IN CAD

Edge based segmentation is a special case of region growing algorithms since points are bounded within the closed boundaries and connectivity between them is not on the basis of spatial relations [5]. The algorithm starts with extracting the edges along the boundaries of different regions. Edges can be detected while changes in local surface properties (surface normals, gradients, principal curvatures) exceed a pre-defined threshold. This procedure followed by grouping the points inside the boundaries and result the segmented regions. As a drawback of edge-based segmentation algorithm, in many cases, they generate non-closed boundaries. Moreover, it is difficult to detect discontinuity in the curved surfaces due to smoothness in this kind of surfaces which lead to under-segmentation in the range image [4]. Furthermore, only measurements close to the edges are taking into account and other available measurements are not considered as well [6]. In addition, these algorithms are very sensitive to the noise in the range data. Clustering represents another approach of range image segmentation in the feature space and can be considered one of the main categories of the region-based segmentation problem. In clustering approach, parameters of the surfaces are introduced and then point clouds are grouped based on surface parameters. This algorithm is performed by subdividing the points into disjoint regions with homogeneous property. The points inside each region share similar property which is different from other regions and therefore distinguish each region from other regions. Clustering is similar to Hough transform approach in case of working in the feature space. However, unlike the Hough transform, the attributes are calculated locally and the risk of grouping points that are not connected is very low [7]. Moreover, unlike the Hough transform, in clustering approach, with consideration of proximity, just points with similar surface normals are clustered together [9]. 3D Hough transform can be considered the subset of clustering approach. The purpose of the Hough transform technique is to find objects by a voting procedure. This voting procedure is carried out in a parameter space (Hough space). In 3D Hough transform, high accumulator value gives the hypotheses for detecting planes. As a drawback of this approach, points' connectivity is not taken into the account and many spurious plane surfaces may be extracted from those points that are not in the same plane with the given point [9]. Moreover, precision or resolution of parameter space seems problematic in this approach. Geometric primitives-based segmentation belongs to clustering approach. This method is suited for fitting higher-order surfaces to the range data and not just planar regions. In [14] it is described parametric surface model-based segmentation from the range image based on Surface Selection Criterion (SSC) that is carried out by minimizing the strain energy of the thin surface. In order to choose the appropriate surface model, surface is fitted to the measurements and sum of squared residuals between the fitted surface and range data will be small. The perfect fitness of the surface to the measurements is achieved by bending and twisting the surface till it will be closer to the measurements recognition is carried out at the same time [10]. As a drawback of the geometric primitives-based segmentation is that the number of surface types directly affects on the segmentation results [11]. Surface growing in object space is the equivalent of region growing in image space. Surface growing in object space can be performed by grouping point clouds which are spatially close and share similar measure properties like the direction of a locally estimated surface normal, gradient and the principal curvatures. As a result of surface growing method, point clouds are segmented into multiple surfaces. Surface growing algorithm starts from the optimal seed points and surfaces extend to neighboring point clouds based on pre-defined criteria. In this fashion, selection of the seed point is important step and final segmentation results are dependent on it. In order to find optimal seed point, firstly, plane equation is defined for each seed point and its neighboring point clouds. Then, the residuals (orthogonal distances of the point clouds to the best fitted plane) are computed. The point within the fitted plane with lowest square sum of residuals is considered the optimal seed point. Outliers would affect on the results of surfaces normal and consequently on the square sum of residuals which leads to a failure of detecting proper seed points of surfaces.

Therefore, robust least squares adjustment is applied to detect the optimal seed point even in the presence of outliers. In this approach, firstly, the plane fit to the surface points and their neighborhood. The candidate point that its orthogonal distance to the fitted plane is below the predefined threshold is accepted as a new surface point. Due to increase the efficiency of the program while using low accurate point clouds, the plane equation is updated after adding the new candidate point to the corresponding surface points. The neighborhood threshold and residual threshold are used to determine the smoothness of the fitted plane. Secondly, the local surface normal at each point is compared with its neighboring points. The neighboring points are accepted if the angle between its surface normal and normal at the neighboring point is below the pre-defined threshold. As a privilege of surface based algorithms in comparison with edge based algorithm is no need to identify the surface boundaries at the preliminary step. Due to their easy implementation and well time performance, surface growing algorithms are the prevailing method for point cloud segmentation. However, the weakness of this approach is connected with a proper choice of the seed points. Furthermore, selecting different seed points may results different segmentation regions [12]. In addition, this algorithm tends to generate distorted boundaries due to segment objects in the region level instead of pixel level [4].

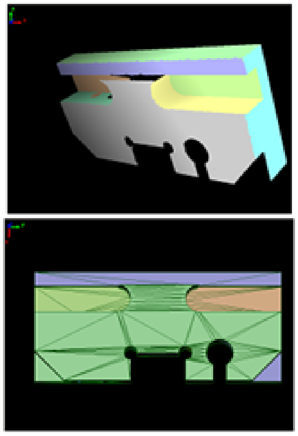

Figure 3 - Corresponding segmented point cloud and CAD mesh model

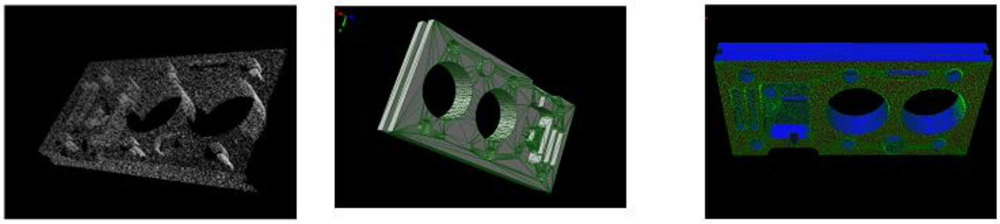

Figure 4 - Aligned point cloud with matching CAD model

5. CONCLUSION

Point clouds acquired by 3D scanners can immediately be used as targets in CAD database queries This approach brings together 3D scanning and shape based CAD models matching and retrieval ideas.. General shape matching challenges like rotational variance and incomplete shape information are resolved by the segmentation and local surface patches alignment processes. This shows that it is plausible to efficiently look up matching CAD models using 3D scanning. To further accelerate the matching process, more rotational invariant attributes may be included during the patch matching stage. One alternative approach is to include a shape descriptor for each surface patch, while this is suitable for complex surface patches, it may be too complicated for engineering artifacts with only specific classes of surfaces. By including more discriminating attributes for specific surfaces, we believe the system's performance can further be tuned. On the other hand, as oppose to a fully automatic system one may allow the users to interactively rank the importance of surface patches generated from point cloud. This changes the alignment order and may lead the system to discover an appropriate alignment faster. Robust estimation has been improved the result of surface normals from least squares plane fitting by eliminating the effects of the noises and outliers. However, it also increased the runtime of the program with recalculating the surface normals. First methods of segmenting surface normals worked well for the dense and accurate point clouds and furthermore, was utilized to extract streets due to not varying the surface normals in the flat areas. The second methods were carried out for the extraction of building roofs and worked well in most cases and especially applicable in case of low accurate point clouds. Availability of the 3D coordinates for each image pixel in the form of X-Y-Z image in addition to intensity values from RGB image or grayscale image can assist us to better interpret and recognize the objects and it can be considered the good combination in order to reach the goal of automatic 3D city reconstruction and object recognition. Vegetation extraction has been led to failure in some image pixels that their intensity values differed too much from their realistic color in the nature. The novelty of this work comparing to previous work is proceeding of surface growing based approach with unlimited neighboring points and there of no need of merging surface patches. In addition, robustness of the computation of surface normals assists us to discard outliers. Furthermore, segmentation in object space is applied in two different methods to increase the run time of the program in the procedure. As a recommendation for the future work, performing the segmentation procedure in object space and image space simultaneously by the usage of 3D point clouds in the form of X-Y-Z image in company of gray value from grayscale image may lead to better results of the segmentation and reach closer to the goal of automatic 3D city modeling and object recognition. In addition, performing segmentation in object space and thereafter in image space was also successful.

REFERENCES:

[1] Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., Dobkin, D., *Modeling by example,* ACM Transactions on Graphics (Proc. SIGGRAPH) 2004

[2] Ji, Z., Liu, L., Chen, Z., Wang, G., "Easy mesh cutting" *Computer Graphics Forum* 25 (3) (2006) 283-291

[3] Attene, M., Katz, S., Mortara, M., Patane, G., Spagnuolo, M., Tal, A. "Mesh segmentation, a comparative study" *IEEE International Conference on Shape Modeling and Applications* 7(7) (2006)

[4] Shamir, A. "A survey on mesh segmentation techniques". *Computer Graphics Forum* 27 (6) (2008) 1539-1556

[5] Lavoue, G., Dupont, F., Baskurt, A., "A new cad mesh segmentation method, based on curvature tensor analysis" *Computer Aided Design* 37(10) (2005) 975-987

[6] Biederman, I. "Recognition-by-compenents: A theory of human image understanding" *Psychological Review* 94 (1987) 115-147

[7] Sander, P.V., Snyder, J., Gortler, S.J., Hoppe, H. "Texture mapping progressive meshes" *SIGGRAPH* (2001) 409-416

[8] Gerig, G., Jomier, M., Chakos, A., Valmet: A new validation tool for assessing and improving 3d object segmentation " *MICCAI 2001: Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*, (2001) 516-523

[9] Benhabiles, H., Vandeborre, J.P., Lavoue, G., Daoudi, M., "A framework for the objective evaluation of segmentation algorithms using a ground-truth of human segmented 3d-models", *IEEE International Conference On Shape Modeling And Application (SMI)* (2009)

[10] Chen, X., Golovinskiy, A., Funkhouser, T., "A benchmark for 3d mesh segmentation. ACM Transactions on Graphics" *SIG-GRAPH* 28 (3) (2009)

[11] Berretti, S., Bimbo, A.D., Pala, P., "3d mesh decomposition using reeb graphs" *Image Vision Comput* 27 (10) (2009) 1540-1554

[12] Thompson, W.B., Owen, J. Germain, H.; Stark, S., Henderson, T., "Feature-based reverse engineering of mechanical parts" *IEEE Transactions on Robotics and Automation*, 12 (1) (1999) 57-66.

[13] Osada, R.; Funkhouser, T., Chazelle, B., Dobkin, D., "Shape distributions" *ACM Transactions on Graphics*, 21 (4) (2002) 807-832.

[14] Novotni, M.,; Klein, R., "Shape retrieval using 3d Zernike descriptors" *Computer Aided Design* 36 (11) (2004) 1047-1062.

[15] Kazhdan, M,, Funkhouser, T., Rusinkiewicz, S., "Rotation invariant spherical harmonic representation of 3d shape descriptors" *Symposium on Geometry Processing* (2003) 156-164.

[16] Ip, C.Y., Lapadat, D., Sieger, L., Regli, W.C., "Using shape distribution to compare solid models" *Symposium on Solid Modeling and Applications* (2002)

[17] Ip, C.Y., Sieger, L., Regli, W. C., Shokoufandeh, A., "Automated learning of model classifications" *Symposium on Solid Modeling and Applications* (2003).

[18] Iyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y., Ramani, K., "Shape-based searching for product lifecycle applications" *Computer Aided Design* 37 (13) (2005) 1435-1446

[19] Pal, P., Tigga, A., Kumar, A., "Feature extraction from large cad databases using genetic algorithm" *Computer Aided Design* 37 (5) (2005) 545-558
[20] Cardone, A., Gupta, S. K., Deshmukh, A., Karnik, K., "Machining feature-based similarity assessment algorithms for prismatic machined parts", *Computer Aided Design*, 38 (9) (2006) 954-972.
[21] Smith, S.D.G., Escobedo, R., Anderson, M., Caudell, T.P., "A deployed engineering design retrieval system using neural networks" *IEEE Transactions on Neural Networks* 8 (4) (1997), 847-851.
[22] Hou, S., Lou, K., Ramani, K., "Svm-based semantic clustering and retrieval of a 3d model database" *Computer Aided Design and Application* 2(1-4) (2005) 155-164
[23] Elad, M., Tal, A., Ar, S., "Content based retrieval of VRML objects: an iterative and interactive approach" *Eurographics workshop on Multimedia* (2001)
[24] Ip, C.Y., Regli, W.C., "A 3D object classifier for discriminating manufacturing processes" *Computer & Graphics* 30 (6) (2006) 903-916.
[25] Funkhouser, T., Shilane, P., "Partial Matching of 3D Shapes with Priority-Driven Search" *Symposium on Geometry Processing* (2006)
[26] Besl, P., McKay, N., "A Method for Registering 3D Shapes" *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (1992) 239-256.
[27] Rusinkiewicz, S., Levoy, M., "Efficient variants of the ICP algorithm" *3DIM* (2001) 145-152.
[28] Mitra, N. J., Gelfand, N., Pottmann, H., Guibas, L., "Registration of point cloud data from a geometric optimization perspective" *Symposium on Geometry Processing* (2004).