A NOVEL APPROACH TO SMOOTHING ON 3D STRUCTURED ADAPTIVE MESH OF THE KINECT-BASED MODELS

Erdal Özbay[1] and Ahmet Çınar[2]

[1,2]Department of Computer Engineering, Fırat University, Elazığ, Turkey
erdalozbay@firat.edu.tr
acinar@firat.edu.tr

ABSTRACT

3-dimensional object modelling of real world objects in steady state by means of multiple point cloud (pcl) depth scans taken by using sensing camera and application of smoothing algorithm are suggested in this study. Polygon structure, which is constituted by coordinates of point cloud (x,y,z) corresponding to the position of 3D model in space and obtained by nodal points and connection of these points by means of triangulation, is utilized for the demonstration of 3D models. Gaussian smoothing and developed methods are applied to the mesh consisting of merge of these polygons, and a new mesh simplification and augmentation algorithm are suggested for the over the 3D modelling. Mesh consisting of merge of polygons can be demonstrated in a more packed, smooth and fluent way. In this study is shown that applied the triangulation and smoothing method for 3D modelling, perform to a fast and robust mesh structures compared to existing methods therewithal no remeshing is necessary for refinement and reduction.

KEYWORDS

Point Cloud, Mesh, 3D Reconstruction, Triangulation, Smoothing, 3D Camera

1. INTRODUCTION

Especially in the last years, 3D models of the objects are utilized as one of the most important features of real world objects in the fields of application such as computerized graphic systems, computer games, virtual and augmented reality, computer-human interaction. The process of identifying shape of the physical objects and demonstrating it in different mediums representatively is called 'modelling'. During modelling, various methods are utilized to identify the position of objects in the real world and to process and identify them, and each method has preferences and imperfections comparing to others [1].

In this study, 3D stereo camera is preferred to obtain depth information of 3D shapes of objects in terms of quality, low cost and real-time feasibility. Thus, 3D utilization has become popular among the inquiring persons and researchers. Features of this 3D sensing camera are utilized to establish 3D model of objects in a closed medium by using Point Cloud Library (PCL). Additionally, process of connecting data that contains more than one depth information of the same object is performed. With the aid of this process, different point cloud sets are connected in order to obtain a more detailed surface. Thus a large number of point clouds are constituted [2,3].

Mostly it is impossible to use the results as they are which are obtained from certain hardware performing 3D scanning, because of their complex geometric structure [4]. For this reason, it is necessary to put these into a certain order by which graphical processes shall be realized as far as possible. One of the methods prevalently used to create 3-dimensional models and perform

certain procedures on these models is the structure of polygonal meshes [5,6]. This method enables to make changes on the space positions of the objects. There are a great number of points and edges that connect these points, surfaces and corners within the structure of the polygon. Different kind of references may be applied by examining the relations among these and by taking into consideration their status with those having different dimensions and measures and their neighborhood. Thus, simplification algorithms constitute an important stage in performance of the methods, and many different types of algorithms are developed [7,8,9,10,11].

2. MATERIAL AND METHOD

In this section mentioned a number of steps required to provide the 3D modelling with stereo camera. After the 3D modelling reconstruction we applied mesh smoothing algorithm on it. Our developed method was applied on the reconstructed 3D model created by ourselves, as well as Stanford's official model's data. Finally, our method results were evaluated and which have been compared with the previous comparable methods.
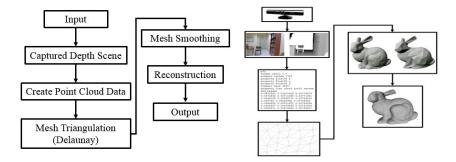

Figure 1. Implementation of Flow Chart

2.1. Three Dimension Modelling

In the first place a method is determined to specify the location of the object in the real world for which modelling process shall be realized. With this aim, 3D stereo cameras are used which find area of usage in the scenarios such as 3 dimensional scanning of the objects. 3D sensing camera is used to obtain the depth information of objects. Camera realizes scanning in the area of 47° vertical and 57° horizontal by means of the laser on it. Reflection time of the light is calculated for each voxel by means of the depth sensor, and distance to the object is determined. Form this viewpoint, distance between the camera and the objects for which model is to be created and their stance positions are important. A mechanism to which camera and object shall be placed is constituted with the aim to obtain more delicate and fixed images and for to prevent shifting in the coordinate points during uniting of images.

Figure 2. Placing the mechanism of 3d cameras and object to be modelled

Information of distance between the object and the camera is reported for 30 times in a second up to the voxel of 640x480x480 by means of the depth sensing camera. 3-dimensional (x,y,z) structure of two dimensional (x,y) pixel values is called voxel. Each voxel carries three dimensional value in itself according to the coordinate point. Besides their known pixel characteristics, voxels contain an additional depth data [12].
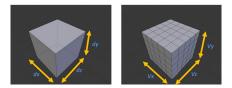

Figure 3. Voxel structure, dx=dy=dz= 3m, Vx=Vy=Vz= 640x480x480 voxel

After the coordinates of the whole surface area of the object are taken, polygonal caging method as one of the prevalent methods utilized to connect the coordinates taken to constitute 3D models, is used. Factors constituting the 3D models are generally corners, edges, surfaces and polygons. Each corner has its own place and each successive two corners constitute an edge. Each three edges that are connected form a closed surface and therefore surfaces are sequence of closed edges. In a polygonal cage, each polygon is composed of three corners, and each corner may be a part of multiple polygons [13]. A point may represent a corner of more than one edge. In the figure below, you can see each face of a polygonal cage consisting of three corners. For example, PA surface is composed of 1, 2 and 4 corners. At the same time, we can see that most of the corners are parts of polygonal faces. For example, corner 5 is a part of PB, PC, PD and PE faces [14]. As it is seen the figure 4 below, nodal points of corners of a surface are connected by means of x, y, and z coordinates and thus polygons are associated with each other. Delaunay triangulation method is utilized in the stage of connecting the point cloud data [15].
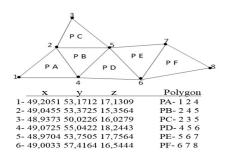

x	y	z		Polygon
1- 49,2051	53,1712	17,1309		PA- 1 2 4
2- 49,0455	53,3725	15,3564		PB- 2 4 5
3- 48,9373	50,0226	16,0279		PC- 2 3 5
4- 49,0725	55,0422	18,2443		PD- 4 5 6
5- 48,9704	53,7505	17,7564		PE- 5 6 7
6- 49,0033	57,4164	16,5444		PF- 6 7 8

Figure 4. Triangulation of data points

2.2. Mesh Smoothing

Simplification and smoothing of polygonal models that constitute the foundation of modelling and contain a great number of points, nodes and thus polygons have a particular importance in terms of computer graphics systems. Geometric objects having a complex structure is the reason for requirement of mesh simplification processes. Additionally, they should be subjected to pre-treatment in the state obtained from hardware realizing scanning before they are set out for graphics process. Because of this reason, variations in the various simplification algorithms suggested are examined and evaluated [16]. Variations recorded in the model having the same cross section and by applying simplification algorithm in different numbers are shown [17].

According to smoothing algorithm while calculating the new coordinate values of the point clouds are checked the neighborhood of that. This smoothing algorithm which used for the

regulation of the revised point clouds with together Delaunay triangulation is to bring in the best position between the respective nodes. We accepted for the best position of the nodes that are close on the imaginary line which related between the nearest and farthest points in the relevant points. Therefore assumed that points to be the closest and farthest distances have high priority.
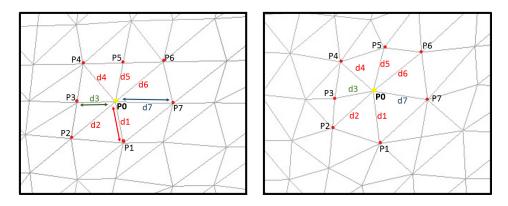

Figure 5. P[i]: ● Points (Relevant), P0: ● Point (Renovated), d[i]: distances, d: min distance, d: max distance.

As a result of the algorithm applied to a section taken from the upper surface of this teapot model, coordinates of the related nodal points after the first and second stages of the original image and differences in the distance between the nodes were examined. As a result, it is seen that the farthest points as the Euclid distance between the related nodes get close to each other more after algorithm is applied. In the same way, the close ones become distanced. Consequently, different algorithms are developed and their results are analysed.
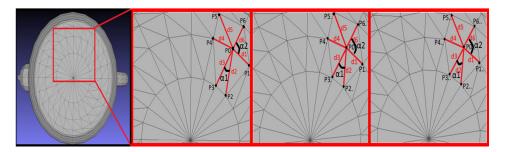

Figure 6. One section of the mesh taken from teapot model that consists of 2256 polygons and for which algorithm is applied (A section of teapot model, Original, Phase1, Phase2)

In the system developed, most distant and closest nodes have the priority in comparison to the other connected nodes by calculating Euclid distances of polygon knots to the all other nodes it is connected. According to the algorithm; new coordinate values are created by calculating these priority nodes that are multiplied with different coefficients numbers with the other nodes left. Below, variations observed in coordinates of one section of teapot model, for which more than one simplification algorithm is applied and in Euclid distances between interrelated nodes are shown.

Exchanges of Points Coordinate	Exchanges of Points Coordinate	Exchanges of Points Coordinate
P0: 1,381370 2,45469	P0: 1,359634 -0,000603 2,448438	P0: 1,337057 0,020553 2,450224
P1: 1,4 0 2,4	P1: 1,363122 0,0741844 2,421876	P1: 1,333855 0,114904 2,432501
P2: 1,35074 -0,375926 2,4	P2: 1,335226 -0,291651 2,421876	P2: 1,317554 -0,242121 2,432501
P3: 1,33276 -0,370922 2,45469	P3: 1,313011 -0,361698 2,448438	P3: 1,296903 -0,333603 2,450224
P4: 1,38426 0 2,4875	P4: 1,36091 0,000834 2,481251	P4: 1,338637 0,001381 2,475001
P5: 1,33555 0,371699 2,4875	P5: 1,314185 0,362255 2,481251	P5: 1,286589 0,37278 2,450224
P6: 1,35376 0,376765 2,49844	P6: 1,33028 0,367708 2,492188	P6: 1,292975 0,354719 2,475001

Exchanges of Euclidean Distance Between Points	Exchanges of Euclidean Distance Between Points	Exchanges of Euclidean Distance Between Points
d1: 0,608564450021565	d1: 0,703925827264045	d1: 0,704900314929089
d2: 1,01094078081888	d2: 0,926660479802087	d2: 0,886171936604949
d3: 0,9107746246766	d3: 0,903789286264646	d3: 0,891905800368812
d4: 0,484658672431632	d4: 0,50476612608753	d4: 0,579329762936723
d5: 0,9107746246766	d5: 0,903430586657506	d5: 0,904509708493443
d6: 1,00242835097346	d6: 0,998352724962898	d6: 0,97233450506312
davg: 0,821356	davg: 0,823487	davg: 0,823192
[davg-d1]: 0,212791	[davg-d1]: 0,119561	[davg-d1]: 0,118291
[davg-d2]: 0,189584	[davg-d2]: 0,103173	[davg-d2]: 0,062979
[davg-d3]: 0,089418	[davg-d3]: 0,080302	[davg-d3]: 0,068713
[davg-d4]: 0,336697	[davg-d4]: 0,318720	[davg-d4]: 0,243862
[davg-d5]: 0,089418	[davg-d5]: 0,079943	[davg-d5]: 0,081317
[davg-d6]: 0,181072	[davg-d6]: 0,174865	[davg-d6]: 0,149142

Exchanges of Angle between Lines	Exchanges of Angle between Lines	Exchanges of Angle Between Lines
$\alpha 1$ (P2-P0-P3): 30°	$\alpha 1$ (P2-P0-P3): 36°	$\alpha 1$ (P2-P0-P3): 39°
$\alpha 2$ (P1-P0-P6): 84°	$\alpha 2$ (P1-P0-P6): 80°	$\alpha 2$ (P1-P0-P6): 78°
[$\alpha 1$- $\alpha 2$] : 54°	[$\alpha 1$- $\alpha 2$] : 44°	[$\alpha 1$- $\alpha 2$] : 39°
Min α : 7°	Min α : 9°	Min α : 10°
Max α : 118°	Max α : 114°	Max α : 112°

$$Pi_x = \frac{\sum_{i=0}^{n} x_1 + (x_{ek}+x_{eb})}{n+2} \ , \ P\,i_y = \frac{\sum_{i=0}^{n} y_1 + (y_{ek}+y_{eb})}{n+2} \ , \ P\,i_z = \frac{\sum_{i=0}^{n} z_1 + (z_{ek}+z_{eb})}{n+2} \qquad (1)$$

n: Total of numbers of nodes related with Pi.

x_{ek},y_{ek},z_{ek}: Node coordinate for which Euclid distance is the smallest between the related nodes.
x_{eb},y_{eb},z_{eb}: Node coordinate for which Euclid distance is the biggest between the related nodes.
P$i_{(x,y,z)}$: Coordinate value found for each node.

New values of (x,y,z) points are calculated during the smoothing process, as the new coordinates are calculated "(1)". As it is seen in the teapot model above, edges farthest to the P0 node get close after algorithms are executed. As it is shown in the change of coordinates of the points, d2 and d3 which are Euclid distances of respectively P2 and P3 of P0 point, get closer by each algorithm.

After the first algorithm is applied, d2 Euclid distance reached 0.926 from 1.0109 and was 0.886 after the second algorithm. In the same way, d4 value that expresses the Euclid distance of the closest point P4 reached 0.504 from the value of 0.484 after the first algorithm and to 0.579 after the second algorithm. The difference between the average value of the Euclid distances and each Euclid distance gradually decreased. In the same way, the difference between the angles also decreased after each algorithm application. The smallest angle value between all nodes increased and the biggest angle value decreased. Briefly, the farthest points between the interrelated nodes get closer and the closest points draw apart from each other. In the same way, angles connecting the lines between the related points are made closer to each other's values after each algorithm

application. For example, angle of lines between P2-P0-P3 that has the acute angle of 30° on the original mesh has reached 36° after the first stage, and 39° after the second stage. While the angle of lines between P1-P0-P6 points that has the most obtuse angle is 84°, it reached 80° after the first stage and 78° after the second stage. A cute angles between the lines widened and obtuse angles narrowed, thus they got closer to each other. In this way, a structure more closer to an equilateral is obtained.

Algorithm

Greatest ➡ 0
Least ➡ 100
for i : 0 to : Number of polygons **do**
        Find all nodes associated with Polygon[i]; **end for**
for i1 : 0 to : Number of polygons **do**
        Remove the nodes that are common interest of the same nodes with Polygon[i1]; **end for**
for i2 : 0 to : Number of polygons
    for j2 : 0 to : i2 **do**
        Euclidean Distance ➡ 0
        Calculate the Euclidian distances of all nodes connected to Polygon [i2];
      If (Euclidean Distance > Greatest) **then**
        P[U_Bt] Grab this node; **end if**
      If (Euclidean Distance <Least) **then**
        P[U_St] Grab this node; **end if**
     end for
end for
for i3 : 0 to : Number of polygons
    for j3 : 0 to : i3 **do**
        Pull of the three-coordinate of this node from memory;
        for each (x, y, z) coordinates;
        p[i3].node = (p[j3] + P[U_Bt] + P[U_St]) / p[i3]+2;
        Delete the old coordinates of the calculated value of the node;
        Write the new coordinate, instead of old one;
        Save;
     end for
end for

As it is shown that, making time affective of complex of algorithm which is applied on data formed by various models was controlled. The number of polygon left is increasing the time of proceeding against being complex O (n^2). At the same time, it affects the time of its achievement by using a different counting method.

Figure 7. Appearance of human model after apply algorithms that consists of 16483 polygons

In the figure 7, it is shown that the model has a smoother structure in each stage after the algorithm application on the model constituted by means of data received from 3D stereo camera. Big number of polygons increases the processing time, even though the application to the data created from different models is 3D as it is shown above. Application time of algorithms changes according to the polygon numbers of the objects, and realization time is affected in line with the utilization of different calculation technique according to the Gaussian method.

Another triangulation technique is fan cloud work with point clouds [18]. It can compute and replace a triangular mesh representation. We observed that the smoothing point clouds are already triangular mesh with regions that are covered by several different triangulations. Therefore we compared our smoothing triangulation algorithm with available this fan cloud method, over the same pattern of data.

Table 1. Computation times for our algorithm, global triangulation and fan cloud triangulation with the different and the same model's data [18]

method	models	# of points	# of nodes	time
Point Cloud generation Algorithm via Delaunay Triangulation	Bunny	35947	69451	13s
	Buddha	25003	49999	9s
	Head	8254	16483	3s
	Body	15697	31324	6s
Global Triangulation	Bunny	35948	69460	8s
	Buddha	20021	40036	4s
Fan Cloud Triangulation	Bunny	100001	193227	17s
	Buddha	47109	94203	5s

Since point cloud triangulation generation is fast, the smoothing algorithm can be applied to large data set. We worked with core i7- GT 650M/2Gb Pc and Visual Studio 2010 /C#. The values of changed point (smoothing algorithm) and approximate values in the table above have calculated by this computer. If we look at the duration for completion of transactions in the table, considering both of the triangulation and smoothing operations, in terms of time appears to be an advantageous operation. Analysing the results of the algorithm on the distances related between points, angle values, and mesh have occurred a demonstrable improvement.
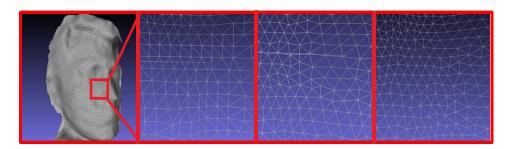
3. CONCLUSIONS

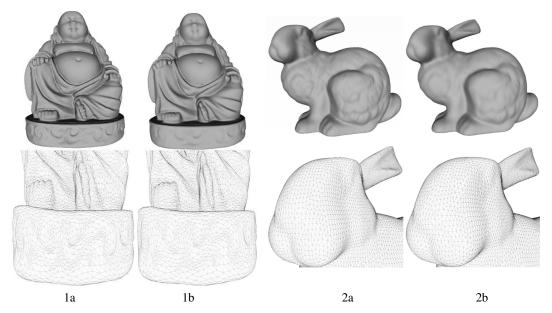Figure 8- A section of mesh taken from human model having 8254 corners, 16483 polygons and for which algorithm is applied

In this study, creation of 3 dimensional models of objects by using 3D sensing cameras; and then 3-dimensional object modelling and imaging system by using mesh simplification technique on these models created are suggested. The point cloud triangulation and smoothing algorithm can be computed efficiently and faster than existing triangular meshes. The number of triangles in a point cloud is less than them. Our method is also useful in less time-critical applications. Because the smoothing and triangulation are totally independent, for such less time-critical applications, it would be possible to use more robust existing methods.

Variations, which are observed in the polygon meshes of models newly created after the algorithms applied to one section of a human silhouette modelled, are given an example in figure 8. It is observed that vertex values in the polygons of the new models gave their place to a more equilateral structure. As it is seen in figure 9, structure of these models in the face and folding points are more resistant against distortions. It is considered that this study shall enable utilization of objects that are modelled and have the same data and a more resistant structure in the areas such as computer games, animation, protection of historical and cultural heritage.

Figure 9- Original (left) and algorithm applied Bunny and Buddha model that have 35947 corners, 69451 polygons (bunny) and 25003 corners, 49999 polygons (buddha).
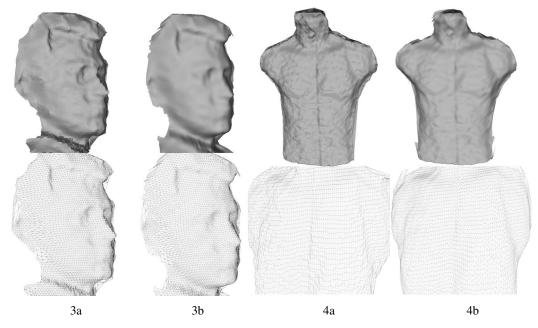
Figure 10- Comparing global triangular mesh visualizations of formal Stanford's (1a, 2a) and own models (3a, 4a) with smoothing point cloud generation models (1b, 2b, 3b, 4b).

We extract and visualize the point cloud with a fine resolution without the region of interest. We proceed as above, update the position of all points in the reverse order of their removal and activate and reinsert the points.

ACKNOWLEDGEMENTS

REFERENCES

[1]   Sohail S., Scott E. D., Mario H., Member, IEEE, and Bernd H., Member, (2012) IEEE "The Topological Effects of Smoothing", *IEEE Transactions on Visualization and Computer Graphics*, V.18, No.1.

[2]   Cui, Yan, et al. (2013). "Algorithms for 3D shape scanning with a depth camera", *Pattern Analysis and Machine Intelligence, IEEE Transactions* on V.35, No. 5.

[3]   Wei, Jiang & Hao, Zhang, et al. (2013). "Skeleton-based intrinsic symmetry detection on point clouds." *Graphical Models,* V.75, I.4, pp. 177–188.

[4]   Ozbay, Erdal & Cinar, Ahmet (2013). "3D Reconstruction Technique with Kinect and Point Cloud Computing", *AWERProcedia Information Technology & Computer Science.* V.3, pp. 1748-1754.

[5]   Ulucay, O., & Sarp, E., (2004). "Resolution adjustable 3-dimensional object modeling." *Signal Processing and Communications Applications Conference, Proceedings of the IEEE 12th. IEEE*.

[6]   Hao Zhang, & Yu-Wei, et al. (2012). "Real-time bas-relief generation from a 3D mesh." *Graphical Models*, V.75, I.1, pp. 2–9.

[7]   Guéziec, André. (1999). "Locally Toleranced Polygonal Surface Simplification". *IEEE Transactions on Visualization and Computer Graphics*, V.5, I.2, pp. 178–199.

[8]   Lindstrom P., & Turk G., (1998). "Fast and memory efficient polygonal simplification". *IEEE Visualization*, pages 279–286. ISBN 0-8186-9176-X.

[9]   Hoppe, Hugues. (1996). "Progressive meshes". *Proceedings of SIGGRAPH* 96, pp 99–108, ISBN 0-201- 94800-1.

[10] Ronfard, Rémi & Jarek, Rossignac. (1996). "Full-range approximation of triangulated polyhedra". *Computer Graphics Forum* V.15, I.3, Proc. Eurographics'96 pp. 67-76.

[11] Schroeder, J.William, Jonathan, A.Zarge, & William, E.Lorensen. (1992). "Decimation of triangle meshes". *In ACM SIGGRAPH Computer Graphics* V. 26, No. 2, pp. 65-70.

[12] Smisek, Jan, Michal Jancosek, & Tomas Pajdla. (2013). "3D with Kinect". *In Consumer Depth Cameras for Computer Vision* pp. 3-25. Springer London.

[13] Power, Stephen C. (2011). "Polynomials for crystal frameworks and the rigid unit mode spectrum", *arXiv preprint arXiv*: 1102.2744.

[14] Rusu, Radu Bogdan, & Steve Cousins. (2011). "3D is here: Point Cloud Library (PCL)". *Robotics and Automation (ICRA)*, 2011 IEEE International Conference on pp. 1-4.

[15] Gao, Zhanheng, Zeyun Yu, & Michael Holst. (2012). "Feature-preserving surface mesh smoothing via suboptimal Delaunay triangulation." *Graphical Models* V.75, I.1, pp. 23–38.

[16] Kovalovs, M., A. Sisojevs, & A. Glazs, (2012). "A Surface Smoothing Method for a 3D Model of a Medical Object". *IFMBE Proceedings*. V.38. International Symposium on Biomedical Engineering and Medical Physics, Latvia, Riga. pp. 74-77.

[17] Avallone, Anna, et al. (2013). "Openness of measures and closedness of their range." *Journal of Mathematical Analysis and Applications* V.404, I.1, pp. 57–63.

[18] Linsen, Lars & Hartmut, Prautzsch (2003). "Fan clouds-an alternative to meshes". *In Geometry, Morphology, and Computational Imaging* V.2616 (pp. 39-57). Springer Berlin Heidelberg.

Authors

Erdal Özbay.. He is a Research Assistant in the Department of Computer Engineering at the University of Fırat where he has been a faculty member since 2012. He completed his MSc at Fırat University. His research interests lie in the area of programming languages, computer graphics, image processing, human computer interaction, mesh generation, ranging from theory to design to implementation, with a focus on improving software quality.

Ahmet Çınar.. He received the PhD degree in Electric-Electronics Engineering in 2003 from Firat University. He has been working on Firat Univ. Department of Computer Engineering, (Assistant Professor). His research is interested in development and improvement of mesh generation methods, and applications of virtual reality, augmented reality, artificial intelligence and game programming.