An Effective Approach to the Use of 3D Scanning Technology which Shortens the Development Time of 3D Models

Georgia Constantinou, George Wilson, Shabnam Sadeghi-Esfahlani, Marcian Cirstea
Department of Computing and Technology
Anglia Ruskin University
Cambridge, UK
georgia.constantinou@anglia.ac.uk, george.wilson@anglia.ac.uk, shabnam.sadeghi-esfahlani@anglia.ac.uk, marcian.cirstea@anglia.ac.uk

***Abstract*— The development process of creating 3D models such as characters, objects and landscapes is very time-consuming. This paper explores some new alternative methodologies (algorithms, software, techniques) to shorten this development time. This has relevance to more generic applications in 3D virtual environments, particularly through the use of 3D target models and 3D optical scanning technology. The work shows how high-resolution 3D models acquired by software or scanning can be analysed in a common format for practical use as imports into a game or other 3D virtual applications. An application for loading a 3D polygon mesh from a file has been developed as a case study to explain the methodology. This is generic enough to handle the processing of 3D data derived either from graphics design software or exported from optically scanned objects.**

***Index Terms*— 3D Scanner, OBJ file format, 3D model.**

I. INTRODUCTION

In computer graphics, 3D models can be expressed as numerical representations of the real as well as the imaginary world [1]. They contain very large data sets comprising geometrical and appearance attributes [2] and they are used to generate complex polygonal meshes to represent the models in virtual space. The 3D models can be produced using interactive software development packages (e.g. 3D Studio Max, Maya) but these introduce a major bottleneck in the design and development process and often produce 3D models containing too much detail for practical use. The work shows how high-resolution 3D models acquired by software or scanning can be analysed in a common format for practical use as imports into a game or other 3D virtual applications. This paper is organized as following: we provide appropriate theoretical background in Section II, an overview of the 3D scanning technology in section III and present the research methodology and case study results in section IV. Section V concludes with some planned future work.

II. THEORETICAL BACKGROUND

The computational representation of surfaces has been widely described in literature [3]. The surfaces are usually represented by a collection of vertices, edges and faces sometimes known as a polygonal mesh and defines the shape of an object. Every face of a mesh is an addition of triangles and it consists at least three vertices respectively [4]. In some surface interpolation geometries, a three dimensional mesh could be compared to a Triangular Irregular Network (TIN) locally. There is a range of commercial platforms which have the ability to create surface models from point clouds. These platforms can be open source and they offer innovative solutions for three dimensional mesh generation, visualization, interpretation and analysis, such as Leica Cyclone, 3DReshaper or Geomagic Wrap ([5], [3]). There are also open-source solutions which are robust and easy to use, such as MeshLab and CloudCompare. In particular, the MeshLab software [6] is a free, open-source software for mesh processing and editing and which generates a triangular 3D-mesh. This software works with the most common 3D file formats, such as OBJ, 3DS, PLY, STL, COLLADA, XYZ, ASC, X3D, PTX, PTS, XYZ, ASC, X3D and VRML. MeshLab is a principal software application package used in the current research along with 3Ds Max. MeshLab has a range of algorithms which can be used to reconstruct surfaces from point clouds (a point cloud is a data structure which is used to represent a collection of multi-dimensional points and used to represent three-dimensional data). In a 3D point cloud, the points usually represent the X, Y and Z geometric coordinates of an underlying sampled surface. The point clouds can be acquired from hardware sensors such as stereo cameras, 3D scanners, or generated from a computer program synthetically [7].

Current computer graphics tools allow the design and visualization of very realistic and precise 3D models. These devices can generate complex meshes with a high level of detail. This leads to a mandatory simplification procedure since the management and representation of these meshes can be very complicated and demanding of resources. This applies especially to the geometry which is produced by 3D scanning devices such as the Mephisto Extreme used in the present work which can produce surface meshes composed of 10 - 100 M faces. These large size meshes have to be simplified to a smaller size for ease of management for use in real applications. Some techniques offer efficient processing time to produce simplified meshes but are visually undesirable [8]. Two common simplification operations are the edge collapse ([9], [10]) and vertex collapse ([11]).

III. 3D SCANNING TECHNOLOGY OVERVIEW

3D scanning technologies are potential tools for increasing productivity whilst at the same time securing quality in product development. Generally, 3D scanning can be of great benefit in resolving the issues concerning ways of creating 3D data for objects that do not have pre-existing computer models. Creating good digital representations is often of crucial importance when using today's manufacturing methods.

3D scanners are available to digitize objects from microscopic to large constructions in size. Data points are captured with speeds ranging from a few points per second to more than a million points per second. There are handheld manual devices available as well as large size automatic scanning equipment ([12]).

There are mainly two methods for obtaining the coordinates of an object's geometrical shape. The first approach is a mechanical method, which uses mechanical arms where the object is fixed on a table; the coordinates of the points picked by the inspector by means of touch-probes are transferred to the computer. With this system, measurement of formed and large surfaces may take hours or even days depending on the details of the object and accuracy of the measurement required. Accuracy levels up to 1 μm can be achieved using this method. This level of sensitivity depends on the experience of the inspector and type of the equipment used. The second approach involves non-contacting scanning methods which can be classified in to three main categories optical, acoustic and magnetic.

Optical scanning systems are based on techniques such as laser scanning, fringe projection, and photogrammetry. These systems are being applied successfully for the 3D measurement and virtual reconstruction of object surfaces in many areas. Fringe projection scanning systems generally work with white structured light where the light pattern is projected on the object's surface while one or two cameras record the reflected light while laser scanning systems can obtain data by sending laser light onto the object and processing the data obtained from the returning light ([13], [14], 15]). The advantages of these scanners are that they are more portable compared to contact systems and their sensitivity levels are partially independent of the inspector.

Optical scanning systems such as laser or fringe projection can obtain a large amount of point data in a short period of time and the accuracy of laser systems vary typically from 1 μm up to 20 μm, whereas fringe projection systems have the capability of 10 μm up to 60 μm. Since the accuracy of the non-contact systems are continually improving, they are now widely adopted for many applications in the industry [16].

Optical technology is generally a preferred method because it gives a greater flexibility in the digitization of surfaces and provides higher resolution and accuracy when compared to mechanical technology ([17], [18]) and because of speed of measurement ([19], [20], [21]). The advantage of contacting devices is that they do not depend on the colour and reflective characteristics of the surfaces to be scanned, which might be the case with optical scanners.

IV. RESEARCH METHODOLOGY AND CASE STUDY RESULTS

The 3D scanner used in this study is the Mephisto Ex-PRO Optical scanner. The Mephisto Extreme is a 3D optical scanner that has the capability to achieve fast, accurate and high quality 3D scanning results with a minimum of processing time and user interaction. It is suitable for a variety of scanning jobs from the very small to the scanning of large animated as well as static objects. The system has a deep scan, turntable mode, HDTV (1920 x1080) resolution machine vision camera with large, high quality Kodak CCD sensor , high quality Nikon mount lens with aspheric optics, high contrast ratio high resolution projector. The Mephisto Extreme produces results compatible with any of the highest range comparable scanners, whilst remaining an affordable and extremely flexible solution. The 3D scanning uses DLP type projectors for binary and fringe pattern projection. The DLP projectors have many advantages when compared to other projection technologies and provide an excellent base for structured light systems. The DLP technology is based on a micro-mirror device where every pixel is a tiny mirror that can be switched on and reflect light or switched off. DLP projectors are therefore true digital devices, are very fast in operation, and produce high brightness and high contrast images. These are important factors for a structured light 3D scanning system. Brightness and contrast are two main factors influencing scanning quality and robustness.

Most modern DLP projectors are able to produce 2000-3000 ANSI lumens with 2000:1 contrast ratio. The Mephisto Extreme system is compatible with Infocus, Acer, Benq and other multimedia projectors with wide screen format and 1280x768 resolution. All components are synchronized and linked together by simple cable assembly and Mephisto software. System assembly and connection requires no special skills or tools and takes less than five minutes. Individual components are lightweight and compact and this makes it easy to pack and unpack system for travelling and off site scanning. The Mephisto Extreme is therefore a truly mobile scanning solution and can be used in most extreme conditions and places.

The Mephisto 3D scanning engine is based on three core components – Calibration, Processing and IO Interface. The calibration is a very simple yet powerful solution where a geometric system calibration is performed using a flat calibration board with a checker pattern. Five or six shots with different calibration board orientation angles are needed to compute system intrinsic and extrinsic parameters, and robust calibration code uses bundle adjustment to optimise the solution to produce ideal results.

Workflow use of the Mephisto Extreme system is project based utilising a Project Wizard to help the user define the scanning system, select cameras to be used for scanning and high-resolution texture, projection screens and other properties. Once a project is created and the system is calibrated, it can be saved and repeatedly reloaded according to requirements. The project structure allows the saving of scanned data, images and other information in an organized manner. Even when the user scans the most complex object from hundreds of different angles, the Mephisto system will organise the data within minutes.

A. Identification of 3D scanner configuration issues

Initial setup trials of the 3D scanner did not produce good results largely due to incorrect configuration (geometric, including calibration board positioning; radiometric, including environmental lighting). Fig. 1 for example shows a target object of a scanning process (figurine) and Fig. 2 shows the initial results of the scanning process.

Fig. 1. Target object (figurine)

It is clear from Fig. 2 that the target object image contains a lot of noise and inaccurate colour representation.

B. 3D scanner hardware configuration and optimization

Following recalibration in consultation with the 3D scanner suppliers, crucial issues preventing the successful operation of the 3D scanner were identified and rectified (repositioning of the calibration board and use of the Canon 400D camera in a monochromatic mode. A software upgrade along with the aforementioned operational changes in procedure has produced much more successful scanning results and outputs of dramatically improved quality (Fig. 3).

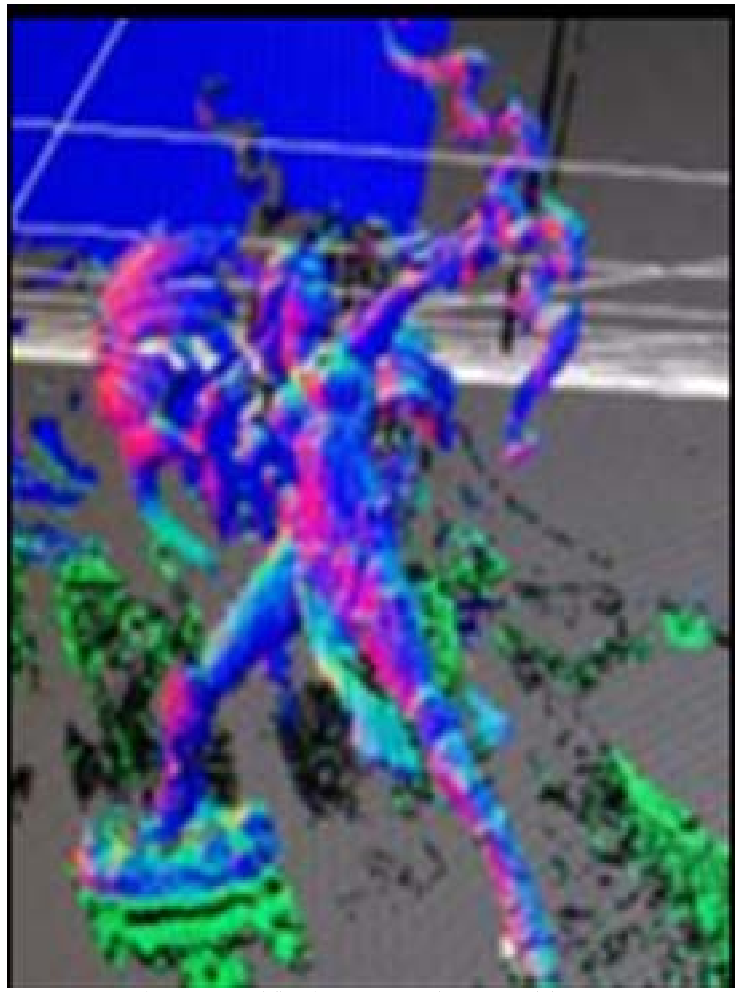


Fig. 2. Scanned figurine represented as 3D model

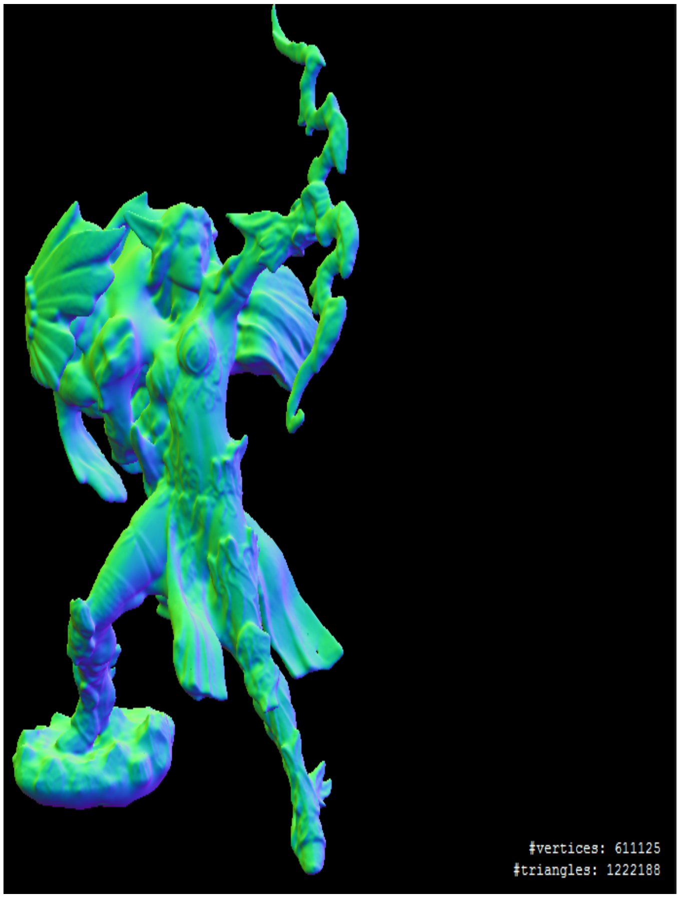
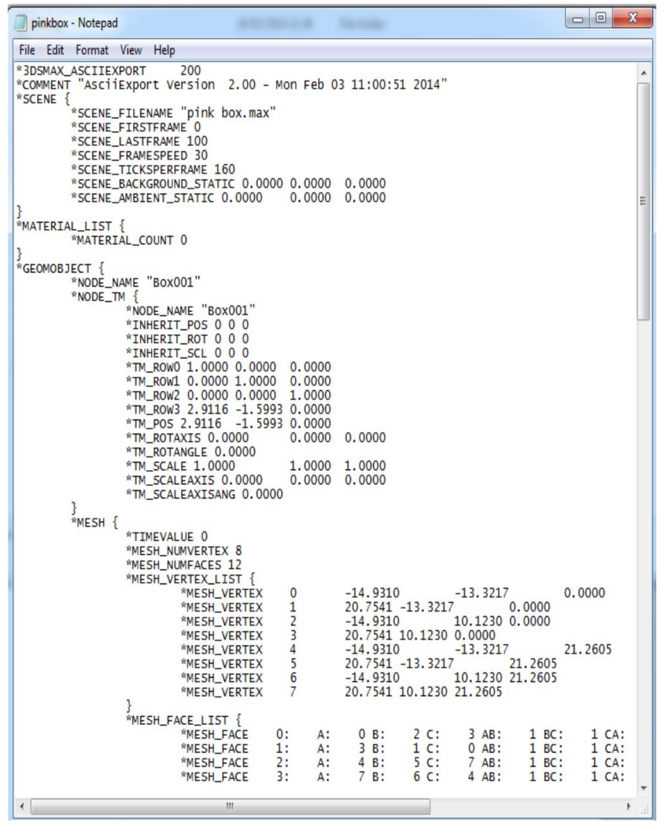


Fig. 3. Scanned 3D model of object using improved approach.

C. *Algorithm and software development*

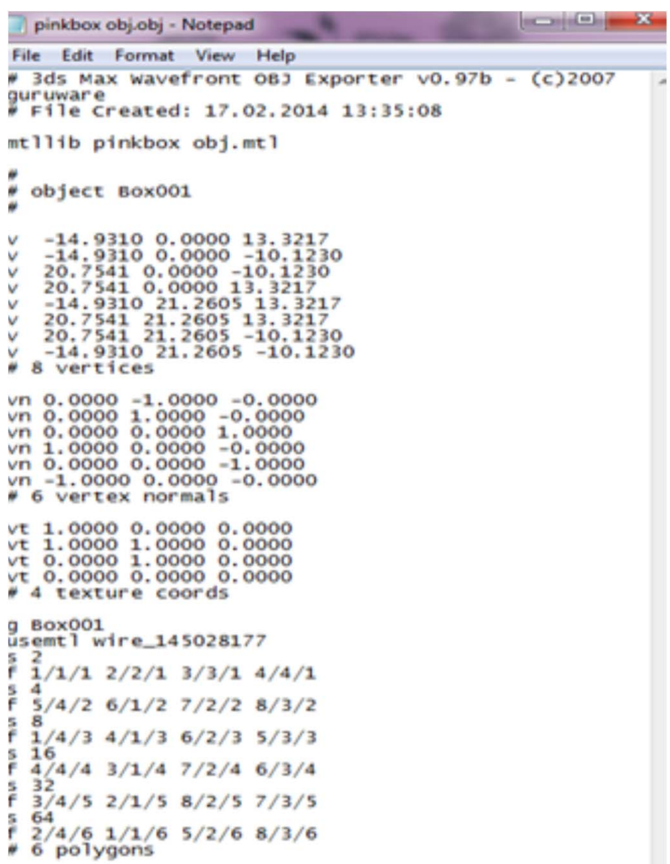
3D Studio Max is a software development package that includes tools for constructing 3D objects. In particular, objects can be saved to file (for export and input to other applications) and 3D data can be imported either directly from a 3D scanner or from certain external file formats. A software application was developed to facilitate the portability of 3D data between the Mephisto 3D Scanner and 3D Studio Max. In order to establish ‘proof-of-principle’, initially a simple 3D object was constructed in 3D Studio Max (a cube) and exported as an ASE file (a common 3D Studio Max file format). The ASE files are text files (ASCII files) and they can be read by many applications including a user- written program for the purpose of implementing 3D data simplification algorithms. However, there is very little documentation available on how to read and interpret ASE files; thus, importing the ASE files back into 3D Studio Max proved difficult.

For this reason, the use of ASE files was abandoned and instead the OBJ file type has been adopted as the type of input/output file used in this research. Like ASE files, they are ASCII-based and are easily readable in a text editor. They are readily compatible with 3D Studio Max for input/output purposes. Also, unlike ASE files, there is good documentation available on the type structure and the types of data and its format. Fig. 4 shows a comparison of the content of the two file formats, each containing the same information for an identical object.



```
pinkbox - Notepad
File  Edit  Format  View  Help
*3DSMAX_ASCIIEXPORT	200
*COMMENT "AsciiExport Version  2.00 - Mon Feb 03 11:00:51 2014"
*SCENE {
	*SCENE_FILENAME "pink box.max"
	*SCENE_FIRSTFRAME 0
	*SCENE_LASTFRAME 100
	*SCENE_FRAMESPEED 30
	*SCENE_TICKSPERFRAME 160
	*SCENE_BACKGROUND_STATIC 0.0000 0.0000	0.0000
	*SCENE_AMBIENT_STATIC 0.0000	0.0000	0.0000
}
*MATERIAL_LIST {
	*MATERIAL_COUNT 0
}
*GEOMOBJECT {
	*NODE_NAME "Box001"
	*NODE_TM {
		*NODE_NAME "Box001"
		*INHERIT_POS 0 0 0
		*INHERIT_ROT 0 0 0
		*INHERIT_SCL 0 0 0
		*TM_ROW0 1.0000 0.0000	0.0000
		*TM_ROW1 0.0000 1.0000	0.0000
		*TM_ROW2 0.0000 0.0000	1.0000
		*TM_ROW3 2.9116 -1.5993 0.0000
		*TM_POS 2.9116	-1.5993 0.0000
		*TM_ROTAXIS 0.0000	0.0000	0.0000
		*TM_ROTANGLE 0.0000
		*TM_SCALE 1.0000	1.0000	1.0000
		*TM_SCALEAXIS 0.0000	0.0000	0.0000
		*TM_SCALEAXISANG 0.0000
	}
	*MESH {
		*TIMEVALUE 0
		*MESH_NUMVERTEX 8
		*MESH_NUMFACES 12
		*MESH_VERTEX_LIST {
			*MESH_VERTEX	0	-14.9310	-13.3217	0.0000
			*MESH_VERTEX	1	20.7541 -13.3217	0.0000
			*MESH_VERTEX	2	-14.9310	10.1230 0.0000
			*MESH_VERTEX	3	20.7541 10.1230 0.0000
			*MESH_VERTEX	4	-14.9310	-13.3217	21.2605
			*MESH_VERTEX	5	20.7541 -13.3217	21.2605
			*MESH_VERTEX	6	-14.9310	10.1230 21.2605
			*MESH_VERTEX	7	20.7541 10.1230 21.2605
		}
		*MESH_FACE_LIST {
			*MESH_FACE	0:	A:	0 B:	2 C:	3 AB:	1 BC:	1 CA:
			*MESH_FACE	1:	A:	3 B:	1 C:	0 AB:	1 BC:	1 CA:
			*MESH_FACE	2:	A:	4 B:	5 C:	7 AB:	1 BC:	1 CA:
			*MESH_FACE	3:	A:	7 B:	6 C:	4 AB:	1 BC:	1 CA:
```

Fig. 4.a ASE file format for a cube



```
pinkbox obj.obj - Notepad
File  Edit  Format  View  Help
# 3ds Max Wavefront OBJ Exporter v0.97b - (c)2007 guruware
# File Created: 17.02.2014 13:35:08

mtllib pinkbox obj.mtl

#
# object Box001
#

v  -14.9310 0.0000 13.3217
v  -14.9310 0.0000 -10.1230
v  20.7541 0.0000 -10.1230
v  20.7541 0.0000 13.3217
v  -14.9310 21.2605 13.3217
v  20.7541 21.2605 13.3217
v  20.7541 21.2605 -10.1230
v  -14.9310 21.2605 -10.1230
# 8 vertices

vn 0.0000 -1.0000 -0.0000
vn 0.0000 1.0000 -0.0000
vn 0.0000 0.0000 1.0000
vn 1.0000 0.0000 -0.0000
vn 0.0000 0.0000 -1.0000
vn -1.0000 0.0000 -0.0000
# 6 vertex normals

vt 1.0000 0.0000 0.0000
vt 1.0000 1.0000 0.0000
vt 0.0000 1.0000 0.0000
vt 0.0000 0.0000 0.0000
# 4 texture coords

g Box001
usemtl wire_145028177
s 2
f 1/1/1 2/2/1 3/3/1 4/4/1
s 4
f 5/4/2 6/1/2 7/2/2 8/3/2
s 8
f 1/4/3 4/1/3 6/2/3 5/3/3
s 16
f 4/4/4 3/1/4 7/2/4 6/3/4
s 32
f 3/4/5 2/1/5 8/2/5 7/3/5
s 64
f 2/4/6 1/1/6 5/2/6 8/3/6
# 6 polygons
```

Fig. 4.b OBJ file format for the same cube.

A program was successfully developed in C++ that could read an OBJ file and export (save) it as a separate OBJ output file. The original input file data and the output file data are the same, but the program also includes a count of vertices, normal, texture coordinates and faces.

The read / write 'proof-of-principle' aim has been successfully achieved through use of freely available web based source code which was configured for the specific requirements of this research. Any object of any complexity can now be read in, the appropriate data extracted for processing, and that data (potentially processed) then written back to an output file. This is very helpful when there is a need to count the geometric data of a 3D model. The output of the OBJ Loader is presented in Fig. 5.

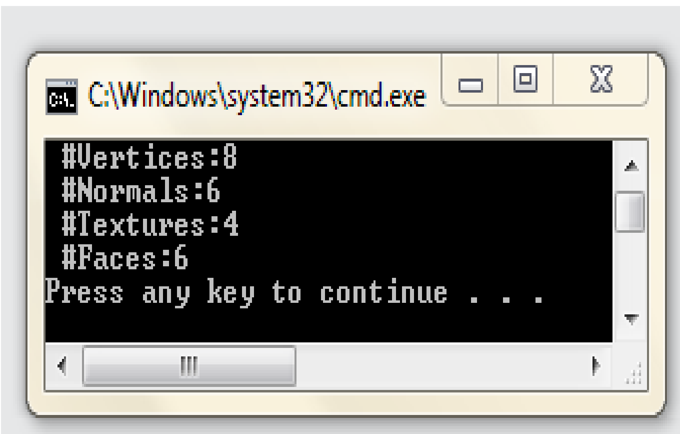


Fig. 5. The output of the program for the file pinkbox.obj

Similarly, when selecting a 3D model retrieved from the 3D scanner, the output of the program is shown in Fig.6.

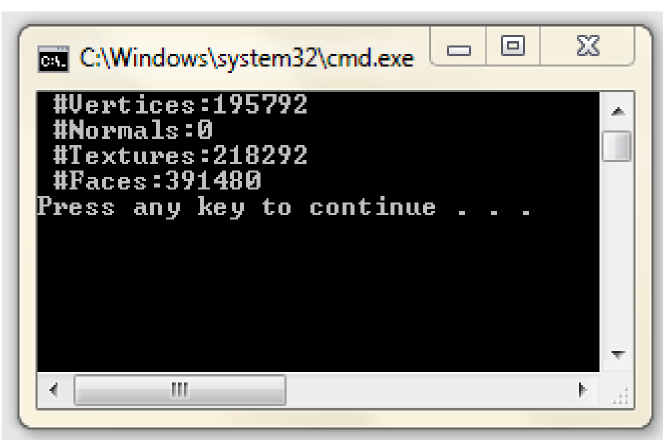


Fig. 6. The output of the program for the file model.obj

V. Conclusions and Future Work

This paper explored an alternative methodology to shorten development time of 3D model construction. A program was successfully developed in C++ which reads an OBJ file and exports it as a separate OBJ output file, generating additional property information about the object. The program introduced a step in the 3D model workflow process which potentially will easily facilitate the implementation of various mesh processing algorithms as required.

Further work will explore the use of the 3D scanner to scan facial expressions with the aim of establishing more effective expression recognition based on 3D geometrical features rather than solely 2D features.

VI. References

[1] Z. Zhao, K. Hwang, J. Villeta, *Proceedings of the 3rd workshop on Game cloud design with virtualized CPU/GPU servers and initial performance results*, 2012.

[2] K. Finney. 3D game programming all in one. p.26-39, 2013.

[3] A. Roncat, Y. Dublyansky, C. Spötl, P. Dorninger, "Full-3D surveying of caves: a case study of Märchenhöhle (Austria)", in Marschallinger R. & Zobl F., Mathematical geosciences at the crossroads of theory and practice, *Proceedings of the IAMG2011 Conference*, September 5-9, Salzburg, Austria, pp. 1393-1403, 2011.

[4] R. Tobler, S. Maierhofer, "A mesh data structure for rendering and subdivision.", in: Jorge J., Skala V. (Eds.), *The 14th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG'2006) Short Communication Papers Proceedings*, pp. 157-162, 2011.

[5] B. Pucci, A. Marambio, "Olerdola's Cave, Catalonia: a virtual reality reconstruction from terrestrial laser scanner and GIS data." in: 3D virtual reconstruction and visualization of complex architectures, *Proceedings of ISPRS International Workshop 3D-ARCH*, 2009.

[6] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli and G. Ranzuglia, "Meshlab: an opensource mesh processing tool." in: Scarano V., De Chiara R. & Erra U. (Eds) – *Eurographics Italian Chapter Conference*, pp. 129-136, 2008.

[7] B. Radu, R. Cousins, S. Cousins, "3D is here: Point Cloud Library (PCL)", *IEEE International Conference on Robotics and Automation (ICRA*), Shanghai, China, 2011.

[8] R. Alvarez, J.-V. Noguera, L. Tortosa and A. Zamora. "A mesh optimization algorithm based on neural networks", *Information Sciences: an International Journal*, Volume 177, Issue 23, December 2007.

[9] R. Zhang, R. "Edge collapse improved model-based three dimensional surface model data compression algorithm", *Journal of Chemical and Pharmaceutical Researc*h, www.jocpr.com, 2014.

[10] T. Ma, G. Gong, J. Yan, "A 3D model simplification algorithm based on edge-collapse", *IEEE International Conference in Industrial Informatics (INDIN)*, 2012.

[11] D. Attali, A. Lieutier, and D. Salinas, "Collapsing rips complexes", *29th European Workshop on Computational Geometry (EuroCG),* 2013.

[12] D. Schodek, M. Bechthold, K. Griggs, K.M. Kao, M. Steinberg, CAD/CAM Applications in Architecture and Design.John Wiley & Sons, Inc; New York, 2005. Digital design and manufacturing.

[13] A.A. Bernard, "Review of State-of-the-Art Reverse Engineering." *Proceedings of TCT Conference*, Nottingham, U.K.. pp. 177–188, October 1999.

[14] T. Varady, R. Martin, J. Coxt, "Reverse engineering of Geometric models – an introduction." Computer aided Des. vol. 29, pp: 255–268, 1997.

[15] J. Peipe, H.J. Przybilla, "Modeling The Golden Madonna." *Cipa 2005 XX International Symposium*, Torino, Italy. 26 Sept. – 1 Oct. 2005.

[16] S. Seokbae, P. Hyunpung, H.L. Kwan, "Automated laser scanning system for reverse engineering and inspection.", *Int. J. Mach. Tools Manuf.*, vol. 42, pp. 889–897, 2002.

[17] G. Tognola, M. Parazzinia, C. Sveltob, P. Ravazzania, F. Grandoria, "A fast and reliable system for 3D surface acquisition and reconstruction", *Image Vision Computing*, vol. 21, pp.295–305, 2003.

[18] M. Sokovic, J. Kopac, “RE (reverse engineering) as necessary phase by rapid product development.” *Journal Mater. Process*. *Technol.* Vol. 175, pp.398–403, 2006.
[19] F. Blais, “A review of 20 years of range sensors development.” *Videometrics VII, SPIE IS&T Electronic Imaging*, pp. 62–76, 2003.
[20] M. Milroy, D.J. Weir, G.W. Vickers, C. Bradley, “Reverse engineering employing a 3-D laser scanner: a case study”, *Int. J. Adv. Manuf. Tech.*, vol. 12, pp. 111–120, 1996.
[21] L. Li, N. Schemenauer, X. Peng, Y. Zeng, P. Gu, “A reverse engineering system for rapid manufacturing of complex objects.”, Robot.Comput-Integr. Manuf., vol.18, pp.53–67, 2002.