A Resolution Control Method of an Object's Shape Applying Tolerance Caused by Micro Motion

Daisuke Wakatsuki[1], Ikuo Ishii[1] and Akira Takahashi[2]
[1]Niigata University
waka@jura.niigata-u.ac.jp, ishii@ie.niigata-u.ac.jp
[2]Nagaoka National College of Technology
ataka@nagaoka-ct.ac.jp

Abstract

In this paper, we proposed a new resolution control method of a moving object's shape applying observers' perceptual characteristics. The method applies a tolerance caused by micro motion of the moving objects in order to achieve an effective resolution control for representing a detailed object's shape which is based on contour shape error and brightness error. The experiment with the above method shows better results in processing time and in the quality of the object's images in comparison with previous methods, such as to control shape resolution in direct proportion or inverse proportion to the object's velocity.

Key words: Virtual Reality, Level of Detail, Perceptual Criterion, Octree

1. Introduction

Efficient representing methods of objects with detailed structure are important technologies to improve the reality of various VR contents. These technologies can be applied to representing VR environments that require high realistic images such as VR museum. Generally, an enormous processing power is required to represent a detail structured object with computer graphics. In order to hold down its processing cost, various methods to reduce the number of polygons for a high-speed representation have been developed [1].

Some resolution changing methods corresponding to observational distance have been developed. In these methods, the resolution of an object is controlled by the feature of object's itself such as shape, color, material, texture and so on [2][3]. Availability to reduce the number of polygons that are invisible from a visual point has been insisted [4][5]. There are also explored development methods which can be applied to observers' perceptual characteristics. For example, these methods apply the contour shape and the brightness change of objects, the visual acuity of observers and so on [6][7][8]. The above methods control the resolution of an object by using static parameters of visual point and light sources. However, since there are many cases that moving objects are observed, the environment can be represented more efficiently by using dynamic parameters about movements of visual point and objects.

Referring to the fact that the recognition of moving object is difficult, a number of shape resolution control methods taking account of the object's movement have been proposed. Some previous methods control the object's shape resolution in inverse proportion [9][10] or direct proportion [11][12] to its velocity. These are simple and high-speed processing methods because they apply uniform velocity against the whole object. However, it is not enough to represent general movement with combination of parallel and rotational movement, and partial object's resolution control is necessary in order to obtain a close-up observation.

Although images of each frame are represented finely by rendering objects completely, their movements are discontinuous because of dropping to a lower fps when moving objects are represented. Conversely, if objects are represented by lower quality, their movements are smooth. Therefore, there is trade-off relation between the quality of images and the smoothness of movements. Our approach is the appropriate representation of moving objects with keeping suitable quality of images for their movements

In this paper, we propose a resolution control method to represent a detail structured object in consideration of observers' perceptual characteristics with new tolerance parameters for an object's movement. The method aims to achieve an efficient object's shape resolution control by using the tolerances for object's contour and its brightness caused by reflection which takes account of the micro motion of the object. We have assumed this method in order to obtain better results in comparison with previous methods such as using direct proportion or in inverse proportion to the object's velocity.

2. Data Structure of Object

The target object's shape is divided recursively into cubes. Their cubes are stratified with an octree which is one of hierarchical data structures as in Fig.1 in order for partial shape resolution control. The octree is a data structure in which each node has eight child nodes as in Fig.2 and can refer simply and quickly a target node representing partial shape. In this experiment, we set a root node as level 0, and set levels increasing in order of descendant nodes. The part requiring a high resolution refers to the level of nodes which are closer to leaves (higher level nodes). The part requiring a low resolution refers to the node data at a level close to a root (lower level nodes). These nodes of different levels are selected according to a method described in section 4 following.

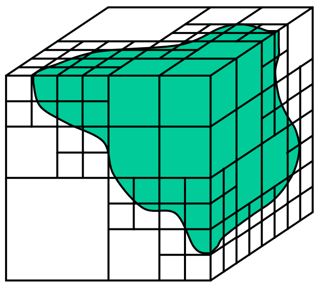


Fig.1: Division of an Object

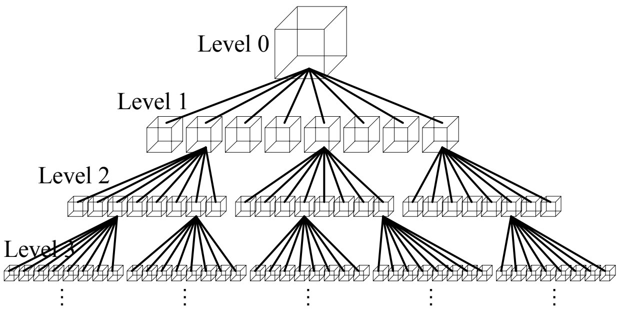


Fig.2: Octree

The data shown in Table 1 are located in each node. $\boldsymbol{p}$ stands for a coordinate value which represents surface contained in each node. $(\boldsymbol{n},\theta)$ indicates a cone (*normal cone*) which encompasses normal vectors of surface contained in each node. $\boldsymbol{n}$ is a normal vector of $\boldsymbol{p}$ and θ is a maximum angle between $\boldsymbol{n}$ and an each normal vector in descendant nodes. E_s stands for maximum distance error of $\boldsymbol{n}$ direction component between the $\boldsymbol{p}$ and $\boldsymbol{p}$(s) of descendant nodes. $\boldsymbol{M}$ includes a set of maximum and minimum values of diffuse and specular reflectance contained in each node.

The $\boldsymbol{p}$ is obtained by an average coordinate value of vertices which form polygons included by the node's cube. It is also applied to join adjoining $\boldsymbol{p}$(s) of nodes to form surfaces for the purpose of representing objects [13].

Table 1: Data Located in a Node

$\boldsymbol{p}$	Coordinate value
$(\boldsymbol{n},\theta)$	Normal vector cone
E_s	Shape approximation error
M	Material data

3. Movement Parameter

3.1 Movement Velocity Vector

Those parts of a detailed structural object which move at low velocity are recognized easily. Parts of the object which move at high velocity can be simplified because it is difficult to recognize detailed structure. In order to control shape resolution in consideration of an object's movement, movement velocity vector $\boldsymbol{v}_v$ is obtained by

$$\boldsymbol{v}_v = \frac{\boldsymbol{v}_m}{t_m}, \quad \ldots(1)$$

where $\boldsymbol{v}_m$ stands for movement vector of $\boldsymbol{p}$ in each node between previous and current frames and t_m indicates the rendering time of previous frame here. All vectors are given in object's coordinates.

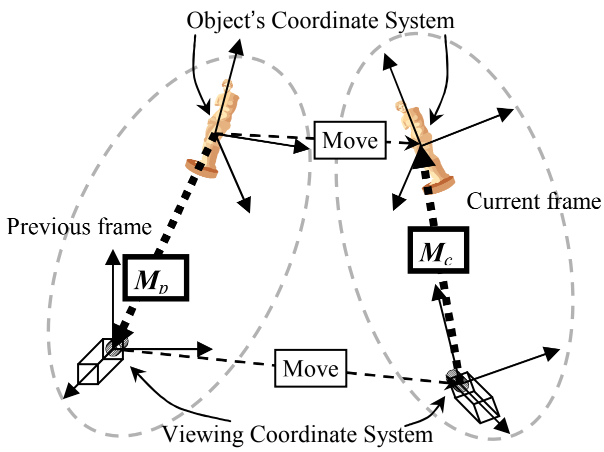


Fig.3: Coordinate Systems

The $\boldsymbol{v}_m$ is related to a movement of visual point. We use $\boldsymbol{M}_c$ for a coordinate transformation matrix from an object's coordinate systems to a viewing coordinate system in current frame, and use $\boldsymbol{M}_p$ for a coordinate transformation matrix from a visual point coordinate system to an object's coordinate system in previous frame as in Fig.3. $\boldsymbol{v}_m$ of each $\boldsymbol{p}$ is acquired by

$$\boldsymbol{v}_m = \boldsymbol{p} - \boldsymbol{M}_p \boldsymbol{M}_c \boldsymbol{p}. \quad \ldots(2)$$

3.2 Movement Vector in a Frame Interval

Unlike previous methods applying direct proportional or inverse proportional parameters to movement velocity, our method aims to control an object's shape resolution by using condensable movement vectors in a frame interval time Δt.

We set standard Δt as 1/60 second referring to the field rate of TV and HDTV(60Hz) in order to represent capably an object's Movement in general display devices. Movement vector $\boldsymbol{v}_f$ in a frame is obtained by

$$\boldsymbol{v}_f = \boldsymbol{v}_v \Delta t \,. \quad ...(3)$$

$\boldsymbol{v}_f$ stands for an unrecognizable movement vector. In other words, $\boldsymbol{v}_f$ is supposed as a tolerance for object's shape errors. The object's shape resolution is controlled efficiently by $\boldsymbol{v}_f$.

4. Principle of Shape Resolution Control

4.1 Our Previous Method

We proposed object's shape resolution control in consideration of shape recognition characteristics of observers [13]. This method applies static parameters of a visual point, light sources and objects, and controls object's shape taking account of contour shape and brightness of object's surface when rendering objects. We apply $\boldsymbol{v}_f$ to this method, and realize object's shape resolution control method in consideration of its movement.

4.2 Flow of Whole Process

When representing an object, the procedure of our method selects suitable nodes for representation from an octree, and represents an object by forming polygons from selected nodes. In short shape resolution control means selecting suitable nodes from an octree. The following procedures are executed starting from the root in an octree :

1. Nodes contained in a view volume are selected by a 3D clipping algorithm.
2. Nodes with suitable resolution in an object's contour are selected by resolution control by contour shape error (see 4.2).
3. Nodes with suitable resolution on an object's surface except for contour are selected by resolution control by brightness error due to reflectance (see 4.3).

In the procedure 1, it is possible to select quickly the cube of a node by applying a general 3D clipping algorithms. An efficient shape resolution control of a moving object is achieved by applying $\boldsymbol{v}_f$ to the procedure 2 and 3.

4.3 Resolution Control by Contour Shape Tolerance

4.3.1 Fundamental Principle

Resolution control of an object's contour shape selects suitable nodes by using data shown in Table 1, because in order to reconstruct the contour within a suitable shape error that the observer will be almost indistinguishable from original shape. The procedure selects suitable nodes for representation from following two steps :

1. Extraction of nodes which lie in object's contour part when representing an object.
2. Selection of suitable nodes for representation by using the shape approximation error Es and the tolerance $\boldsymbol{v}_f$ for object's shape error.

4.3.2 Extraction of Contour Nodes

In order to extract a node which represent an object's contour, it tests whether the normal cone ($\boldsymbol{n}$,θ) of the node is subvertical to the view directional vector $\boldsymbol{v}_e$ from $\boldsymbol{p}$ to a visual point as in Fig.4. Namely nodes with

$$\frac{\pi}{2} - \theta < \arccos(n \cdot \boldsymbol{v}_e) < \frac{\pi}{2} + \theta \,, \quad \ldots(4)$$

are extracted.

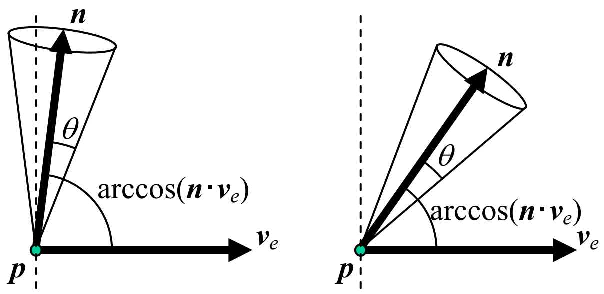


Fig.4: Extraction of Contour Nodes

4.3.3 Resolution Control of Object's Contour

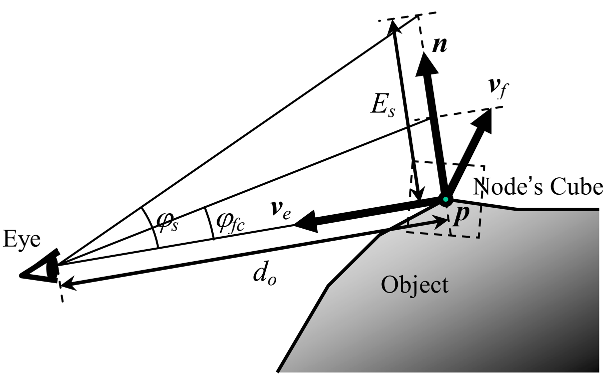


Fig.5: Error of Contour Taking Account of $\boldsymbol{v}_f$

The suitability of this extracted node is tested as shown in Fig.5. The tangent of φ_s which is a visual angle of shape approximation error E_s is obtained by

$$\tan \varphi_s = \frac{E_s}{d_o}, \quad \ldots(5)$$

where d_o is a distance from $\boldsymbol{p}$ to a visual point. This is a visual angle for the shape error of an object which gets still. The tangent of a visual angle φ_{fc} of $\boldsymbol{n} \cdot \boldsymbol{v}_f$ which is a movement of $\boldsymbol{v}_f$ in $\boldsymbol{n}$ directional component is obtained by

$$\tan \varphi_{fc} = \frac{\boldsymbol{n} \cdot \boldsymbol{v}_f}{d_o}. \quad \ldots(6)$$

This is supposed as a movement which observers can not recognize and is tolerance against $\tan\varphi_s$. Therefore, $\tan\psi_s$ which stands for a visual angle for the shape error of a moving object is obtained by

$$\tan \psi_s = \tan \varphi_s - \tan \varphi_{fc}. \quad \ldots(7)$$

We force $\tan\psi_s$ to keep more than 0. Threshold $\tan\psi_t$ of $\tan\psi_s$ to tolerate shape error is given in order to represent a suitable resolution of an object's contour. Nodes with $\tan\psi_s$ smaller than $\tan\psi_t$ are selected. Since nodes with $\tan\psi_s$ larger than $\tan\psi_t$ are not selected, resolution control by contour shape tolerance is executed recursively in their children.

4.4 Resolution Control by Brightness Tolerance

4.4.1 Fundamental Principle

An object's shape, except for the contour, is recognized by partial brightness changes on the object's surface represented on a screen. Resolution control selects suitable nodes according to partial brightness changes caused by reflectance.

In this procedure, suitable nodes for representation are selected partially by using brightness change D_b caused by shading and a visual angle φ_b of a polygon size in each node. Especially, the resolution control by brightness tolerance for moving objects becomes possible by applying $\boldsymbol{v}_f$ to φ_b.

4.4.2 Brightness Change D_b

Maximum and minimum brightness values of a node are obtained by light source, $(\boldsymbol{n},\theta)$, and $\boldsymbol{M}$ from data shown in Table 1. Brightness change D_b is set as the difference between the maximum and minimum brightness value.

We use Lambert diffuse reflection and Phong specular reflection as a reflection model expressed by

$$\boldsymbol{I} = \boldsymbol{L}_d \boldsymbol{M}_d \cos\alpha + \boldsymbol{L}_s \boldsymbol{M}_s \cos^n \beta, \quad \ldots(8)$$

where $\boldsymbol{I}$ is an intensity of reflected luminance, $\boldsymbol{M}_d$ and $\boldsymbol{M}_s$ are diffuse and specular reflectance respectively, $\boldsymbol{L}_d$ and $\boldsymbol{L}_s$ are diffuse and specular luminous intensity respectively, $\boldsymbol{n}$ is highlight coefficient, α is an angle between normal vector and $\boldsymbol{v}_l$, and β is an angle between normal vector and $\boldsymbol{v}_l$+$\boldsymbol{v}_e$. $\boldsymbol{v}_l$ stands for a unit vector from $\boldsymbol{p}$ to a illuminant point and $\boldsymbol{v}_e$ for a unit vector from $\boldsymbol{p}$ to a visual point here.

Referring the equation (8), maximum $\boldsymbol{I}^{max}$ and minimum $\boldsymbol{I}^{min}$ are acquired by using maximum and minimum of reflectance given by $\boldsymbol{M}$, and maximum and minimum of $\cos\alpha$ and $\cos\beta$ which are obtained by geometric relation of $(\boldsymbol{n},\theta)$, $\boldsymbol{v}_l$ and $\boldsymbol{v}_e$. Brightness change D_b is given as

$$D_b = I^{max} - I^{min}. \quad \ldots(9)$$

If considering a color space such as RGB, D_b is acquired by a weighted average, based on perceptual characteristic, about D_b(s) of each component from the equation (9).

4.4.3 Polygon's Visual Angle ψ_b

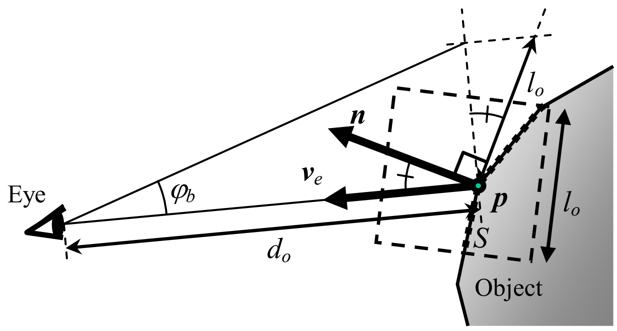


Fig.6: Visual Angle when an Object Stands Still

A visual angle φ_b is acquired by a surface S formed by a node. l_o which is the length of an edge of a node's cube is assumed as the size of S as in Fig.6. By considering the inclination of normal vector $\boldsymbol{n}$ relative to $\boldsymbol{v}_e$ which is a unit vector from $\boldsymbol{p}$ to a visual point, $\tan\varphi_b$ is given by

$$\tan \varphi_b = \frac{l_p(\boldsymbol{v}_e \cdot \boldsymbol{n})}{d_o}, \quad \ldots(10)$$

where d_o is a distance from $\boldsymbol{p}$ to a visual point. This stands for a visual angle for the size of S when an object stands still.

In order to control shape resolution of moving object, the tolerance $\boldsymbol{v}_f$ for object's shape error is applied to $\boldsymbol{v}_d$ which is a direction with largest brightness change. $\boldsymbol{v}_f$ is obtained as a unit vector which is $\boldsymbol{n}$-$\boldsymbol{v}_e$ projected on a plane perpendicular to $\boldsymbol{v}_e$. Namely, $\boldsymbol{v}_f$ is expressed by

$$\boldsymbol{v}_d = \text{normalize}\{\boldsymbol{v}_{en} - (\boldsymbol{v}_e \cdot \boldsymbol{v}_{en})\boldsymbol{v}_e\}, \qquad \ldots(11)$$
$$\boldsymbol{v}_{en} = \boldsymbol{n} - \boldsymbol{v}_e.$$

As shown in Fig.7, $\tan\varphi_{fs}$ is the tangent of a visual angle for movement distance of $\boldsymbol{v}_f$ along $\boldsymbol{v}_d$ direction and is expressed by

$$\tan\varphi_{fs} = \frac{\boldsymbol{v}'_d \cdot \boldsymbol{v}_f}{d_o}. \quad \ldots(12)$$

This is supposed as a movement which observers can not recognize and is tolerance against $\tan\varphi_b$. Therefore, $\tan\psi_b$ which stands for a visual angle for the shape error of an moving object is obtained by

$$\tan\psi_b = \tan\varphi_b - \tan\varphi_{fc}. \quad \ldots(13)$$

We force $\tan\psi_b$ to keep more than 0.

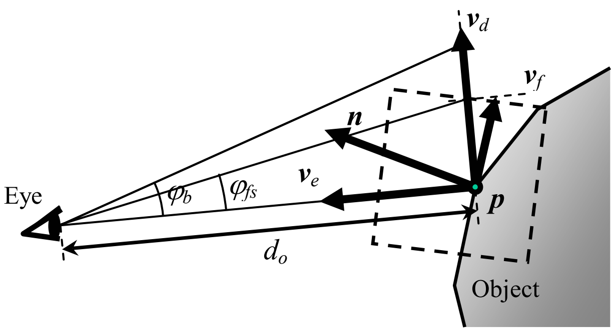


Fig.7: Visual Angle Taking Account of $\boldsymbol{v}_f$

4.4.4 Resolution Control by D_b and ψ_b

Shape resolution is supposed in proportion to D_b and $\tan\psi_b$ here, because the possibility of incorrect object's shape recognition increases according to increase of D_b and $\tan\psi_b$.

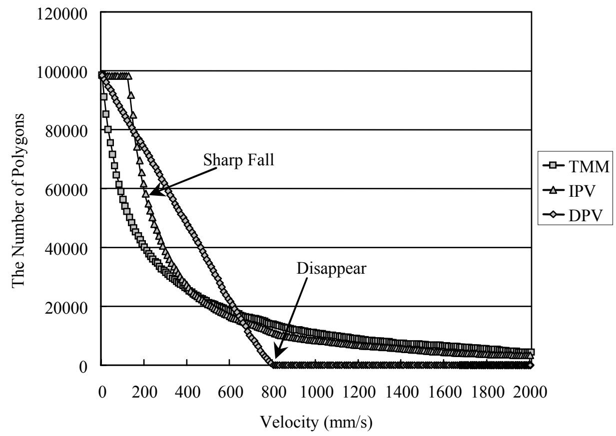
The product of D_b and $\tan\psi_b$ is considered as brightness error. Threshold P_t of $D_b\tan\psi_b$ to tolerate brightness error is given in order to represent a suitable resolution of object's surface. Nodes with $D_b\tan\psi_b$ smaller than P_t are selected. Since nodes with $D_b\tan\psi_b$ larger than P_t are not selected, resolution control by brightness tolerance is executed recursively in their children.

5. Experimental Results

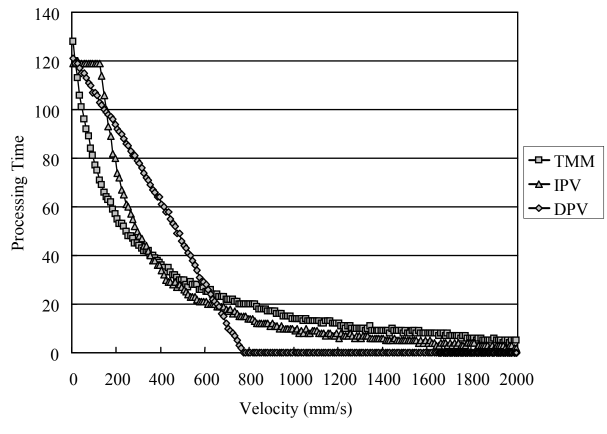
5.1 Environment for Experiments

The processing speed and quality of images were examined in order to establish the effectiveness of our method. In following experiments, we used a general personal computer (CPU: Pentium4 3.0GHz, Graphics: QuadroFX1000), LCD (Width: 340mm, Height: 270mm, Resolution: XGA) as a display device, OpenGL for rendering objects, and *dragon* and *lucy* of *Stanford Computer Graphics Laboratory* as target objects. Each polygon model was stratified with an octree. The space defined by the octree was cube 256(mm) on a size, and the octree's maximum levels of *dragon* and *lucy* were set as 9 and 10 respectively. Our method of applying tolerance caused by micro motion (TMM) was evaluated in comparison with a method applying inverse proportional criteria to velocity (IPV) and a method applying direct proportional criteria to velocity (DPV).

5.2 The Number of Polygons and Processing Time



(a) The Number of Polygons



(b) Processing Time

Fig.8: The Number of Patches of Processing Time vs. Velocity (*dragon*).

Fluctuations in the number of polygons and processing time of each method versus velocity of the object (*dragon*) which moves in parallel direction to the screen are shown in Fig.8. The distance between a visual point and the object's center was set at 500mm. IPV and DPV were controlled in order that means of each processing time were equal to one of TMM. In each method, the number of polygons and processing time reduced according to an increase of velocity as shown in Fig.8 (a) and (b). The data of each result are sampled every 10mm/s.

DPV exposed the fatal disadvantage by which objects disappear caused by the number of polygons becomes 0 when a certain velocity is exceeded. IPV exposed inefficiency which the number of polygons does not reduce until an object's velocity achieves a certain velocity and falls sharply when the certain velocity is exceeded. On the other hand, TMM succeeded in representing objects efficiently according to an object's velocity. Almost the same results are obtained when applying other objects, other movements and also different parameter Δt(s).

5.3 Quality of Images

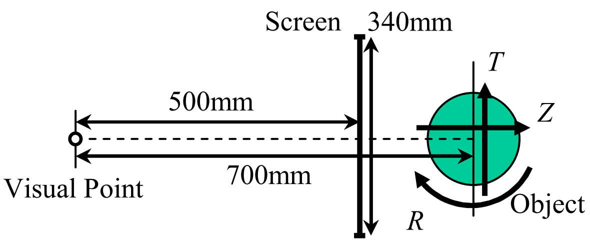


Fig.9: Environment for evaluation of image quality

The quality of image of a moving object was evaluated by PSNR. Experimental environment is shown in Fig.9. An object's movements of translation of parallel direction to a screen *T* (Translation in the image), translation of vertical direction to a screen *Z* (Zoom in the image) and rotation of object itself *R* (Rotation) were examined. Table 2 indicates means and standard deviations of PSNR(s) between the moving images of objects with each movement applying each method and not applying any methods. In almost all cases, it was shown that TMM represents a higher quality moving object's image from the means and achieves smoother shape resolution controls from the standard deviations than other methods. Almost the same results were obtained when using other objects.

Still images which constitute the frames of the aforementioned moving images of TMM and IPV with movement *T* and Δt=1/60(sec.) are shown in Fig.10 and 11. Although IPV represented a detailed object's shape with low velocity shown in Fig.11 (b), delay for representation was large than TMM. In the actual moving image, the quality of images between IPV (Fig.11 (b)) and TMM (Fig.10 (b)) in which the object moves with 250mm/s are almost equal. Shape resolution of the object with high velocity was simplified excessively as shown in Fig.11 (a). Our TMM method (as shown in Fig.10 (a)) focuses on the partial micro motion and takes into account movement direction. It is preferable to the IPV method in that the latter only incorporates the criterion of velocity and thereby ignores contours and shading, thus producing an over simplified image (as shown in Fig.10 (a)).

6. Conclusion

This paper describes a resolution control method of object's shape applying tolerance caused by micro motion in order to represent efficiently detail structured objects. This method applies tolerance for object's contour and its brightness caused by a reflection in

Table 2: Image Evaluation by PSNR

			Δt=1/30(sec.)			Δt=1/60(sec.)			Δt=1/120(sec.)		
Object	Movement	Mean & S.D.	TMM	IPV	DPV	TMM	IPV	DVP	TMM	IPV	DPV
dragon	*T*	Mean	13.28	9.72	2.00	15.28	12.02	3.68	16.17	15.91	12.40
		S.D.	2.68	3.75	5.88	3.36	3.45	7.21	5.38	5.16	9.11
	Z	Mean	19.04	11.36	3.63	18.45	13.85	5.15	19.04	17.97	16.07
		S.D.	3.55	5.82	8.86	4.73	5.32	9.81	5.85	4.63	8.80
	R	Mean	16.15	14.45	2.51	15.85	14.06	12.00	14.99	13.76	13.59
		S.D.	5.94	6.03	6.87	7.11	7.63	7.43	8.12	7.75	7.85
lucy	*T*	Mean	11.81	9.09	3.34	12.84	11.98	3.33	13.62	12.00	11.64
		S.D.	3.32	4.14	6.74	3.21	3.96	6.70	4.39	4.70	4.73
	Z	Mean	14.65	9.90	4.22	15.77	13.43	4.21	16.03	14.82	14.58
		S.D.	3.55	5.35	8.20	3.47	4.57	8.17	4.20	4.21	4.22
	R	Mean	15.42	14.28	9.66	14.84	13.63	13.63	14.20	13.56	14.25
		S.D.	4.14	4.64	3.55	5.14	5.88	5.46	5.90	6.21	6.72

consideration of object's micro motion.

In order to examine feature of TMM which is our method, the number of polygons, processing time for representation and quality of images were evaluated in comparison with IPV and DPV. As a result, it was shown that the number of polygons and processing time versus object's velocity of TMM are more efficient than IPV and DPV. Moreover, TMM achieved higher quality of images and smoother shape resolution change of objects than IPV and DPV.

References

[1] D. Luebke, M. Reddy, J. D. Cohen, A. Varshney, B. Watson and R. Huebner, "Level of Detail for 3D Graphics," Morgan Kaufmann Publishers, 2002.

[2] T. He, L. Hong, A. Varshney and S. Wang, "Controlled Topology Simplification," IEEE Trans. on Visualization and Computer Graphics, Vol.2, No.2, pp.171-183, 1996.

[3] H. Hoppe, "Progressive meshes," Computer Graphics (SIGGRAPH'96 Proceedings), pp.99-108, 1996.

[4] J. El-Sana, N. Sokolovsky and C. T. Silva, "Integrating Occlusion Culling with View-Dependent Rendering," Proc. of IEEE Visualization 2001, pp.371-378, 2001.

[5] W. V. Baxter III, A. Sud, N. K. Govindaraju and D. Manocha, "GigaWalk: Interactive Walkthrough of Complex Environments," UNC-CH Technical Report TR02-013, University of North Carolina at Chapel Hill, 2002.

[6] H. Hoppe, "New Quadric Metric for Simplifying Meshes with Appearance Attributes," Proc. IEEE Visualization '99, pp.59-66, 1999.

[7] D. Luebke and C. Erikson, "View-Dependent Simplification Of Arbitrary Polygonal Environments, "Computer Graphics (SIGGRAPH'97 Proceedings), pp.199-208, 1997.

[8] J. C. Xia, J. El-Sana and A. Varshney, "Adaptive Real-Time Level-of-Detail-Based Rendering for Polygonal Models, "IEEE Trans. on Visualization and Computer Graphics, Vol.3, No.2, pp.171-183, 1997.

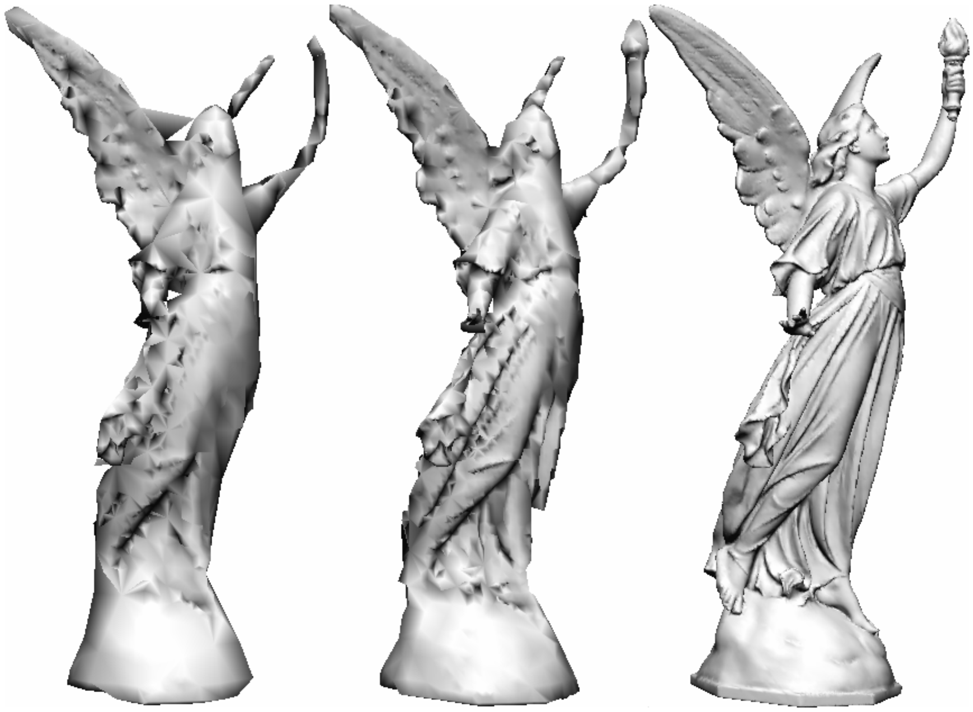
[9] J. H. Clark, "Hierarchial Geometric Models for Visible Surface Algorithms," Communications of the ACM, vol.19 (10), pp.547-554, 1976.

[10] L. E. Hitchner and M. W. McGreevy, "Methods for User-Based Reduction of Model Complexity for Virtual Planetary Exploration," Proceedings of the SPIE, The International Society for Optical Engineering, Vol.1913, pp.622-636, 1993.

[11] T. A. Funkhouser and C. H. S'equin, "Adaptive Display Algorithm for Interactive Frame Rates During Visualization of Complex Virtual Environments," Proceedings of SIGGRAPH 93, pp.247-254, 1993.

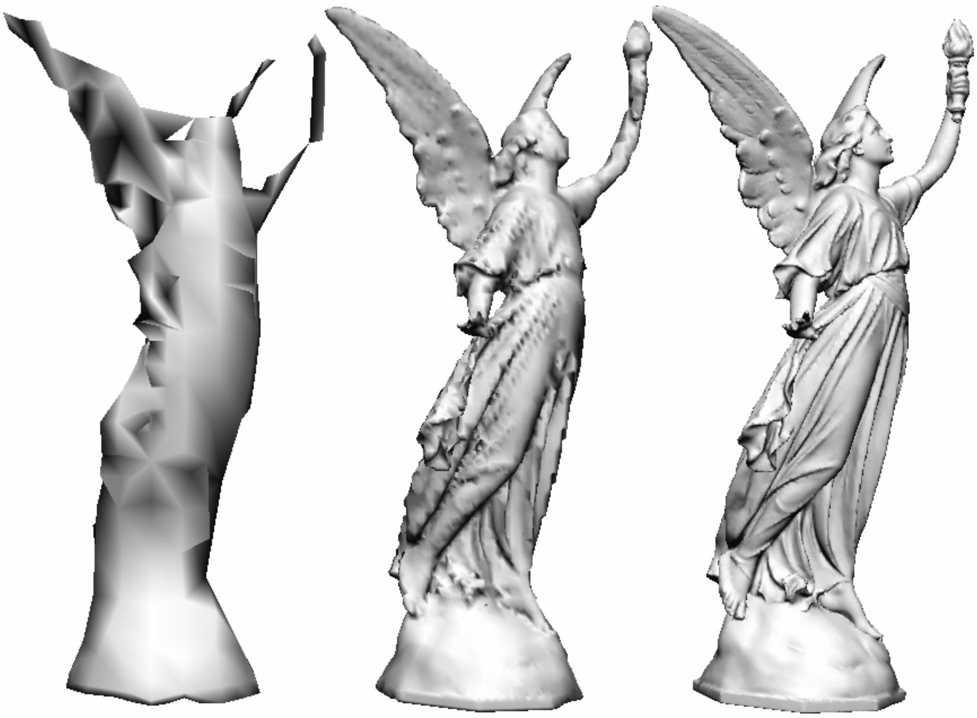
[12] T. Ohshima, H. Yamamoto and H. Tamura, "Gaze-Directed Adaptive Rendering for Interacting with Virtual Space," Proceedings of 1996 IEEE Virutal Reality Annual International Symposium, pp.103-110, 1996.

[13] D. Wakatsuki, I. Ishii, H. Imai, A. Takahashi and H. Makino, "A High Reality Drawing Method for VR Objects with Details Structure in Consideration of Shape Recognition Characteristics of Observers," The Transactions of the Institute of Electronics, Information and Communication Engineers A, Vol.J86-A, No.7, pp.780-791, 2003.



(a) 600mm/s (b) 250mm/s (c) 0mm/s

Fig.10: Results of Rendering of *lucy* by using TMM (Δt=1/60(sec.))



(a) 600mm/s (b) 250mm/s (c) 0mm/s

Fig.11: Results of Rendering of *lucy* by using IPV (Δt=1/60(sec.))