TEXTURE CLASSIFICATION AND RESOLUTION CONTROL FOR 3D URBAN MAP

Satoshi Ueno†, Hiroyuki Inatsuka‡, Makoto Uchino, Masahiro Okuda†*

†The University of Kitakyushu, ‡Zenrin Co., Ltd, * NHK, JAPAN

ABSTRACT

This paper proposes a method to control texture resolution for rendering large-scale 3D urban maps. Recently, performance of computers has been developed rapidly, but it is still hard to directly handle the large-scale 3D maps on PCs, because of its high requirement for hardware capability. Thus the data reduction is essential for rendering of the large map. Since on the 3D maps, texture data, in general, tend to be far larger than geometry data, it is more effective to reduce the texture data by exploiting the LOD (Level of Detail) in order to decrease whole data size. To achieve this, we propose a method to control the resolution of the texture. In our method, the textures are classified into some classes. The appropriate texture resolutions are decided according to their rendered sizes on a display and their importance, and the classes they belong to. We verify the validity of our resolution control method by some experiments.

Keywords: 3D urban map, LOD, texture, classification

1. INTRODUCTION

A three-dimensional map (3D map) is widely used, as the computer technology have developed and spread, and the use of the visual simulation that uses the virtual environment and a 3D navigation have expanded furthermore in recent years as well. Moreover, the technology that acquires 3D data from the real world have advanced, and it has become easy to construct the 3D urban model. The 3D map is often huge and its data size becomes a problem when it is treated in devices including PCs, car-navigation systems, and portable devices. Therefore, the reduction in the volume of data becomes important to make the 3D map applications user-friendly.

Many methods on the LOD control of the 3D terrain data have been proposed so far [1-3]. While the terrain data are the surfaces of the ground with geographical features like mountain district, the city model consists chiefly of buildings. Since the data in the 3D map mostly consist of many small, already simplified meshes (like buildings made of some cuboids), these simplification methods, in which they assume that a single mesh contains a large number of vertices, cannot be applied to it. Although some methods on modeling and rendering of the 3D maps have been already proposed [4-6], there exist few methods concerning the reduction of the texture data for the 3D map. In the 3D map, the proportion of the volume of data in texture tends to be much higher than that of geometrical information such as vertices and triangles. Thus the LOD of the texture can effectively reduce the whole data size when rendering. In this paper we propose the technique for controlling the resolution of the texture to achieve this.

2. PROPOSED METHOD

2.1. Outline

In the framework of our rendering system, first of all, a set of images at multiple resolutions are prepared for all the texture images beforehand. Our strategy is that we classify the texture images into the classes such as the signboards and walls, and we select a resolution for each texture based on the classes. When rendering the 3D map is actually done, only the texture at a suitable resolution is loaded from among the multiple resolution levels.

Now we introduce a representative viewpoint (RV), which is a discrete point in the 3D map used for reference points of rendering. The objects seen from the RV on the map are found, and then only the viewable objects from it are rendered. To select the RVs, the entire map is divided to rectangle parts, we set the point on the grids as candidates for the RVs. Fig.1 shows an example (note that in actual cases the distance between the lattice points is shorter and the viewpoints are more densely located). Next, the points in the area thought to be paths when the user actually walks through in the 3D map are selected as the RVs from the candidates. This is done by excluding the candidate points in a high places such as the tops of the buildings.

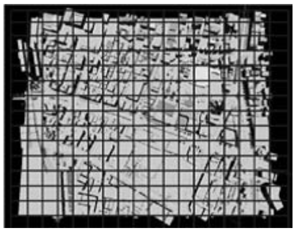

Fig. 1. Candidates for Representative Viewpoints

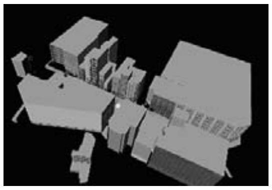

Fig. 2. Objects Visible from a viewpoint

A set of the objects that are visible from each of the RVs are determined by using the Z-buffer. For example, Fig.2 depicts all the objects judged as "visible" from the RV denoted by the sphere in the middle. The process is done for all the RVs and we store the indices of the visible objects in a list. The accuracy of the determination depends on the resolution of the Z-buffer. Increasing the resolution, both of the accuracy and the computational cost increase, and vice versa. Note that one needs to do this processing to a 3D map only once as pre-processing.

2.2. Texture Resolution

In general there is a trade-off between the quality of the rendered texture and its computational cost to load and render it. It is necessary to find an appropriate resolution of the texture that keeps an adequate image quality and low computational cost in order to enhance the efficiency in rendering the 3D map. The determination of the resolution is done for all the textures on the visible objects from each RV. Our strategy is that we define the ”Level” of the importance for each texture by the criteria described below, and then based on the level we select an appropriate resolution for the texture among several resolutions prepared in advance. We determine the resolution by two criteria: (1) the size of the texture in a display and (2) the ”importance” of the texture. First of all, the size of the texture rendered is decided by the three elements: (a) the distance d between the RV and the surface where the texture are mapped, (b) the area S of the texture in the 3D map, and (c) the number of repeats r. The distance d can be calculated by the barycentric coordinates of the surface where the texture is mapped and the xyz coordinates of the RV. When the surface is inclined to the RV, the area of the texture is narrower than the one seen from the front. To take this into account, we first consider a virtual plane P that is perpendicular to the line that connects the center of the plane and the RV. The area S in the new surface made by projecting each vertex of the surface, where the texture is mapped, to the plane P is calculated. Sometimes one texture is repeatedly mapped on a large surface. The number of the repeats r can be found by the texture coordinate (s, t). Finally the size of the texture is roughly estimated by using d, S, and r.

The criteria for the importance of the texture consist of: (1) The ratio of sharp edges in the texture, and (2) The number of colors used in the texture.

Fig. 4 shows some examples of the texture with sharp edges. As for such an object, one can easily identify what it is due to its salience feature in a real scene, and is often used as markers in some applications such as a navigation system. Thus in our framework, we consider that such a texture has high level of importance.

Oppositely, the texture without sharp edges like Fig.3 is judged that the level of importance is low and low resolution is enough to express such textures.

To actually evaluate the edges, first we calculate the intensity Y from RGB values. Then, the edges are detected by applying a simple Laplacian filter to Y. Two thresholds t_l and t_s $(t_s < t_l)$ are set here, and the number of pixels whose absolute values exceed each of the threshold is counted. Denoting the numbers of pixels that exceed t_l and t_s as $c(t_l)$ and $c(t_s)$ respectively, the ratio of the sharp edges R_e are given by

$$R_e = c(t_l)/c(t_s) \tag{1}$$

Next we consider the colors of the texture. In our method, the texture composed of the wide variety of colors (such as the upper left image in Fig.6) is considered as ”less important” texture, since in the many applications including the navigation system, too complex information seldom plays an important role, and it may be even unrecognizable from a distance.

To evaluate the color complexity, we use the variance σ^2 of the RGB histograms. Note that as each bin of the histogram means its frequency, a low variance indicates that the texture contains a wide variety of colors, and a small number of colors induces a high variance. In practice the variance is normalized by the image size.

Finally we simply combine the measures

$$V = \frac{S}{d \cdot r} \cdot \min\{\beta R_e, \sigma^2\}, \tag{2}$$

where β is a normalization factor. We use this value V for judging the level of importance of the texture. A suitable resolution level of the texture is determined by V for each of texture. And the procedure is done for all of the RVs.

3. TEXTURE CLASSIFICATION

In the previous section, we define the value V that is obtained from the display size and the features of the texture. When actually rendering the 3D map, according to V of each texture, one of the multiple resolutions is assigned. In our simulation, we prepare four resolutions for each textures, which is simply done by reducing the resolution of an original texture by 1/2, 1/4 and 1/8. However with this method, salient texture such as road traffic signs is treated similarly to unremarkable texture like the walls of buildings. To address the problem, we introduce a new LOD control based on the texture classification. We make it possible to control the resolution more reasonably based on image features by classifying the texture into some classes, and then changing the reduction ratio based on the classes. By using this method, for example, it becomes possible to make the resolution of the image in one class smaller than the one in other class even if it has larger V in (2).

3.1. Definition of Class

We first define the classes. Fig.3 shows some examples of the texture with soft edges. As these types of textures have the same mere pattern, they rarely have some meaning like letters and marks. Therefore a low resolution is often enough to maintain its visual quality in the 3D map.

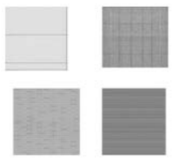

Fig. 3. Walls with soft Edges (Class3)

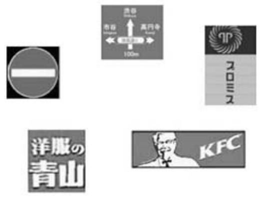

Fig. 4. Billboards and Directional Signs (Class 1)

On the other hand, the textures shown in Fig.4 often have salient features, and contain important information. A human easily perceives blurring and ringing artifacts when such a texture is rendered at low resolution, which results in the decrease in the quality of the 3D map. It can be said that a higher resolution should be allocated for this kind of the textures, since they are often used as eyemarks in the application of navigation.

The texture with some sharp and soft edges simultaneously in Fig.5 is located in the middle of those in Fig.3 and Fig.4, that is, those are less simple than those in Fig.3, but also less significant than those in Fig.4. The low resolution display will somewhat decrease the quality of such textures. However it will not become a serious problem since those seldom contain significantly salient features.

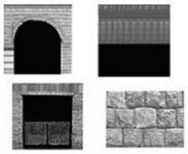

Fig. 5. Walls with Sharp Edges (Class 2)

Fig. 6. Detailed Texture (Class 4)

In addition, as is shown in Fig.6, the textures with minute and complex structures are not often used as markers and less important. And they are not very much affected by reducing the resolution, since a masking effect is higher than smooth images according to the psychophysical aspects [11]. Moreover this kind of textures is less compressible due to its granularity, which is inefficient in terms of overall trade-off between data sizes and its visual quality.

In the end, considering these, we define the four classes:

Class 1: Texture with some clearly outlined objects (such as characters) with flat background.

Class 2: Texture with sharp and soft edges.

Class 3: Texture only with soft edges.

Class 4: Detailed texture

In our framework, we select the resolution based on these classes as well as (2). We consider the case that four resolutions for each texture are prepared, where the reduction ratios are 1, 1/2, 1/4, and 1/8, and all the textures are labeled as each of the Levels 1 to 4. Then we classify all of the original texture to Class 1 to 4. For the textures in Class 1, their original images are allocated to the Level 1 and 2 and the images shrunk to a half are allocated to other levels, while for the Class 3 the images shrunk to 1/8 are used for all the levels as in Table 4. By this strategy, more efficient LOD control of the texture is made possible than the case of only taking (2) into account.

3.2. Classification Method

For the classification, we use K-Nearest Neighbor (K-NN) algorithm is used to automatically classify a large amount of textures. In general the selection of feature vectors greatly affects the accuracy of the classification. Here we introduce two feature vectors, (1) the second moment in color and (2) the energy of edges derived by the anisotropic diffusion filter [7], which performs best in our experiments.

We adopt the second moment with respect to its mean in HSV color space as the color moment.

The anisotropic diffusion filter is a iterative nonlinear lowpass operator that flattens smooth regions while keeping sharp edges. After the operation, we calculate the difference between an original image and its smoothed version, and then we adopt the energy of the difference as the second feature. The role of the feature is to take into consideration only soft edges.

4. EXPERIMENTAL RESULTS

By applying the pre-processing of Sec. 2 and 3 are done, all the objects that are visible from each RV are selected, and the classes and levels of all the texture for the RVs are determined. Then we construct the database of the texture using the levels and the classes. When walking through the 3D map, a current position is found and then loaded are all the textures at the appropriate resolutions that are visible from the nine RVs, that is closest RV to the current position and eight neighboring RVs as well. By loading the eight neighboring RVs, loading cost can be reduced when the user moves to the area of an other RV.

4.1. Precision of Classification

In our experiment, we use a 3D map with more than 5000 textures shown in Fig.7. For the learning of the K-NN method, we use 100 textures for each of the four classes and we set $K = 30$. To evaluate the validity of our classification method, we manually select 50 textures for each class, use them as ground truth. Then, we count the numbers of correctly classified textures, and calculate the precision of the classification by

Fig. 7. 3D map used for experiment

$$P = (\sum_i R_{C_i})/(\sum_i N_{C_i}), \tag{3}$$

where C_i is the classes, R_{C_i} and N_{C_i} are the number of the textures that are correctly classified and are manually classified as the ground truth (that is, in our experiment, $N_{C_i} = 50$), respectively. The precision for each class is defined as

$$P_{C_i} = R_{C_i}/N_{C_i}, \tag{4}$$

First of all we verify the validity of the two criteria: the moment of the color and the edge energy based on the anisotropic diffusion. The feature vectors in our method are compared with features that are often used for general image retrieval [8, 10]: Table 1 shows the comparison in color. In Table 1 and 2 we show the comparison in color and the features of edges, respectively. Among the feature vectors in colors, our method gives the highest score. For the edge features, our method and the wavelet coefficients perform better than others. In Table 3, we show the precision of the combination of the two feature vectors. After experiments for every combination of the vectors in Table 1 and 2, we have confirmed our method outperforms others. Note that although it is possible to increase the dimension of vectors by adding some features, we have not seen any improvements with more dimensions.

4.2. Data size and Quality of 3D Map

We investigate the data size and the quality of the 3D map using textures with and without our resolution control technique. In this experiment, we set the resolution of the original texture as Level 1, and then define the reduction ratio of the resolutions as in Table 4.

Fig.8 illustrates snapshots of the 3D map from a viewpoint. At left of the figure, we show the original 3D map, that is, the textures at original resolutions are used. Fig.8 (right) shows our results with the reduction ratio of Table 4. We adopt the Visual Difference Predictor (VDP) [11] to quantitatively evaluate the visual quality. The VDP is an image assessment tool that models some properties of the HVS, such as nonlinearity, frequency selectivity, direction selectivity, and masking. The VDP outputs a probability map that predicts a probability of error visibility for each pixel. Thus higher values represent that errors are more perceivable. The numerical comparison is shown in Table 5 when it is displayed at the resolutions of 1024x768 and 640x480. The values in VDP75 and VDP95 of this table indicate the ratio (in %) of pixels that have higher probability than 0.75 and 0.95 in the probability map. Thus only a few percentages of pixels may be perceptually different while the amount of the data is reduced to one-quarter. The actual rendering time is also reduced to about 16 %.

Fig. 8. (left)Original Scene, (right) Scene with Data Reduction

5. CONCLUSION

In this paper, we proposed the method that controls texture resolutions based on their features. By allocating low resolutions to visually unimportant textures, we reduce the data size to load for rendering without much degradation of quality.

6. REFERENCES

[1] Renato Pajarola, “ Large Scale Terrain Visualization Using The Restricted Quadtree Triangulation ”, Proceedings of IEEE Visualization ’98, pp. 19-24, October. 1998.

[2] Frank Losasso, Hugues Hoppe, “ Geometry Clipmaps: Terrain Rendering Using Nested Regular Grids ”, ACM SIGGRAPH 2004, pp. 769-776, 2004

[3] Hugues Hoppe, “ Smooth View-Dependent Level-of-Detail Control and its Application to Terrain Rendering ”, IEEE Visualization 1998, pp. 35-42, October 1998.

[4] Paolo Cignoni, Fabio Ganovelli, Enrico Gobbetti, Fabio Marton, Federico Ponchio, Roberto Scopigno. “ Interactive Out-of-core Visualization of Very Large Landscapes on Commodity Graphics Platform ”. International Conference on Virtual Storytelling 2003, pp. 21-29. November 2003.

[5] J. Dollner and H. Buchholz,“ Continuous Level-Of-Detail Modeling of Building in 3D City Models, ” ACM international workshop on Geographic Infromation Systems, pp.173-181, 2005

[6] J. Hu, S. You, U. Neumann,“ Approaches to Large-Scale Urban Modeling, ” IEEE Computer Graphics and Applications, 23(6):pp.62-69

[7] Pietro Perona, Jitendra Malik.“ Scale-Space and Edge Detection Using Anisotropic Diffusion ”. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol12, no7, pp629-639, July, 1990.

[8] Long, F., Zhang, H.J. and Feng, D.D., Fundamentals of content-based image retrieval. In: Feng, D. (Ed.), Multimedia Information Retrieval and Management, Springer, Berlin

[9] G. Pass, R. Zabih, and J. Miller, “ Comparing Images Using Color Coherence Vectors ”, ACM Conference on Multimedia, pp.65-74, 1996.

[10] A. Goodrum,“ Image Information Retrieval: An Overview of Current Research, ” Informing Science 3(2), pp.63-67, 2001.

[11] S. Daly, ”The Visible Difference Predictor: An algorithm for the assessment of image fidelity,” in A.B. Watson, editor, Digital Image and Human Vision, pages 179-206, Cambridge, MA, MIT Press, 1993.

Table 1. Precision of Features on Colors (Var. Hist.: The variance of the histogram, Hist(512): Vectors composed of the histogram quantized to 512 bins, CCV: Color Coherent Vector proposed in [9] Moment: Our feature vector)

	Class 1	Class 2	Class 3	Class 4	All
Var. Hist.	0.18	0.60	0.96	0.82	0.64
Hist(512)	0.54	0.12	0.46	0.98	0.53
CCV	0.36	0.42	1.00	0.84	0.66
Moment	**0.66**	**0.66**	**0.90**	**0.74**	**0.74**

Table 2. Precision of Features on Edges (Sharp Edges: The cost in (1) that we use to determine the texture level, Wavelet Coeffs.: Wavelet coefficients (high pass outputs of the dyadic wavelet), Directionality: The quantized direction of the edges that obtained by Sobel filter, Anisotropic Diffusion: Proposed feature vector)

	Class 1	Class 2	Class 3	Class 4	All
Sharp Edges	0.62	0.38	0.92	0.52	0.61
Wavelet coeffs	0.54	0.60	0.92	0.76	0.71
Directionality	0.44	0.34	0.88	0.74	0.60
Anisotropic Diffusion	**0.66**	**0.38**	**0.88**	**0.86**	**0.70**

Table 3. Precision of Two Features

	Class 1	Class 2	Class 3	Class 4	All
Moment+Wavelet	0.76	0.68	0.96	0.84	0.81
Hist(512)+Wavelet	0.40	0.56	0.96	0.74	0.67
Moment+ Anisotropic Diffusion	**0.92**	**0.82**	**0.96**	**1.00**	**0.93**

Table 4. Reduction Ratio of Texture Resolution

	Level1	Level2	Level3	Level4
Class1	1	1	1/2	1/2
Class2	1/2	1/4	1/8	1/8
Class3	1/8	1/8	1/8	1/8
Class4	1/2	1/4	1/8	1/8

Table 5. VDP and Data Size

	Original	Our Method (1024x768)	Our Method (640x480)
Data Size(MB)	41.1	10.6	10.6
VDP75(%)	0	5.17	0
VDP95(%)	0	2.8	0