3D Solid Texture Shape Descriptors based on Multi-Resolution Pyramids

MOTOFUMI T. SUZUKI, YOSHITOMO YAGINUMA, NORITAKA OSAWA, YUJI Y. SUGIMOTO

National Institute of Multimedia Education, Japan
2-12 Wakaba, Mihama-ku, Chiba-shi, Chiba 2610014
JAPAN

Abstract: 3D solid textures are important data for computer graphics applications, and the amount of the data is increasing. When a large number of the 3D solid textures are stored as databases, similarity search techniques and classification techniques are essential for effective use of the databases. We have applied 3D HLAC (Higher Order Local Autocorrelation) masks to 3D solid textures for extracting shape descriptor indices. By using these indices, the database systems can search for similar 3D solid textures based on texture patterns. Since we have extended the 3D HLAC masks to handle multi resolutions of 3D solid textures by using pyramid structures, the search systems can search for various types of 3D solid textures. We have implemented an experimental system to test search and classification efficiency of 3D HLAC masks using multi resolution pyramidal structures, and the system successfully classified 3D solid textures in the databases based on similarity patterns.

Key-Words: HLAC, Mask pattern, Solid texture, Resolution pyramid, Similarity search

1 Introduction

Recent personal computers are powerful enough to display 3D computer graphics models interactively. Various computer graphics applications have been developed for education, business, and entertainment. The use of 3D models is gaining wide popularity for such applications. Texture mapping is an important method to increase the details of a 3D model without increasing its geometric complexity. Textures can be divided into two types based on their dimensional representation which are (1) 2D image textures and (2) 3D solid textures. Unlike 2D images textures, 3D solid textures are represented in voxels which use x, y, and z coordinates to represent patterns. The number of available 2D and 3D textures is increasing rapidly on various web sites and databases. To utilize such texture data effectively, various search engines have been developed for 2D image textures, but not for 3D solid textures. In previous research [18], we have extended popular 2D HLAC (Higher Order Local Autocorrelation) masks to 3D HLAC masks to analyze 3D solid textures. The system has been developed to search for similar patterns of 3D solid textures from databases based on a 3D HLAC mask shape feature classification technique. Although the previous methods can classify 3D textures based on similarity patterns, there are limitations for increasing classification rates for some textures because 3D HLAC captures local pattern similarity. Occasionally, this characteristic of the 3D HLAC causes low classification rates. To solve the problem, we have modified the system to handle multi resolutions of 3D HLAC mask patterns. The multi resolution pyramids are used as a data structure for increasing classification rates.

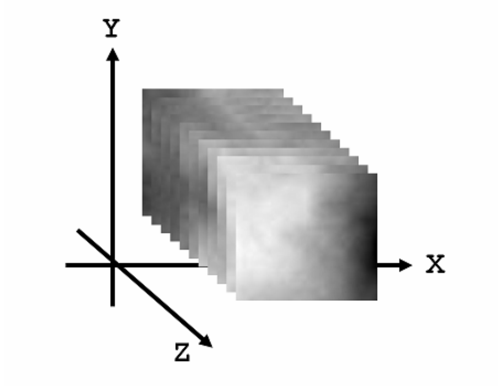

Figure 1: 3D solid textures

To test the classification efficiencies of 3D HLAC shape descriptors with multi resolution pyramids, a 3D solid texture generation program has been implemented. The system uses two types of 3D solid textures generated from noise functions which include (a) Perlin’s noise functions [12] and (b) Perlin’s noise functions with fractal noises. Type (a) of the solid texture shows stochastic texture with various intensities. Type (b) of the solid texture is stochastic texture, but irregular patterns are included by reflecting the fractal functions. Slices of these two types of textures are shown in Figure 2.

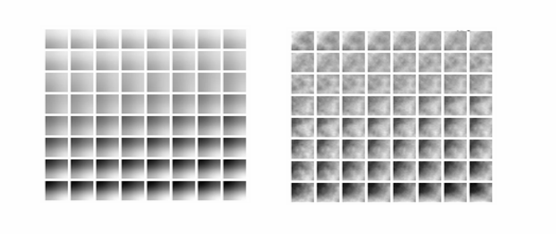

Figure 2: Textures based on simple noise functions (left). Textures based on fractal noise functions (right)

2 3D Solid Textures

3D solid textures are important data for computer graphics applications. Examples of artificially generated 3D solid textures are stone, wood, clouds, water and smoke. To generate such 3D solid textures, procedural methods are used in 3D computer graphics. In the procedural methods, texture functions are involved in order to synthesize various patterns. A variety of research has been conducted to generate 2D and 3D textures, including Fourier synthesis [16], fractal methods [15] statistical models and stochastic texture models [17]. For realistic textures, stochastic functions have been applied to generate textures with irregular patterns. Often, these functions are referred to as noise and return random numbers. Lattice noises are important noises for typical procedural texture synthesis. Figure 1 shows slices of the 3D solid textures based on the lattice noises.

3 3D HLAC and Multiple Resolution Pyramids

In this section, 3D HLAC and multi-resolution pyramids are discussed.

3.1 3D HLAC

HLAC has been used as a feature descriptor for various pattern recognition methods including gesture recognition [7], image retrieval [9] and face recognition [3]. For such kinds of applications, HLAC mask patterns are essential for computing features of 2D images. In our previous research [18], we extended the 2D HLAC mask patterns to 3D HLAC mask patterns for handling 3D solid textures and 3D models [18][19].

The Nth order autocorrelation functions with N displacements are defined by the following equation.

$$x^n(a_1,\cdots,a_n) = \int f(r)f(r+a_1)\cdots f(r+a_n)dr$$

where

n: Order of the autocorrelation functions

r: Coordinate of reference point

a: Direction of shift / translation

The number of autocorrelation functions increases rapidly for the large number of N and mask sizes. Thus, the size of N is limited to 0 =< N =<2, and the mask size is set to 3 x 3 x 3 for practical applications. 3D HLAC mask patterns are generated based on the functions. Information about 3D HLAC masks for 3D HLAC is available at the following web site: (http://open.nime.ac.jp/). This site contains links to patterns which can be downloaded, and it also contains 251 3D HLAC mask patterns which can be used for extracting 3D HLAC shape features. Figure 3 shows the 3D HLAC mask patterns in string sequence forms and corresponding mask positions.

N=0 (1): a
N=1 (13): ab, ak, am, an, ad, ae, aj, km, ks, kv, ms, mt, ns
N=2 (237): akl, akm, akn, ako, aks, akt, aku, akv, akw, akx, abk, abl, aln, abm, abn, abo, abc, amn, amp, amq, ams, amt, amv, amw, amy, amz, abd, ano, anp, anq, anr, ans, ant, anu, anv, anw, anx, any, anz, anA, abe, abf, abj, ack, acn, ace, adk, adm, adn, adp, adq, ade, adg, adh, adj, aek, ael, aem, aen, aeo, aep, aeq, aer, aef, aeg, aeh, aei, aej, afk, afn, agm, agn, ahm, ahn, ain, ajk, ajm, ajn, ajs, ajt, ajv, ajw, klm, kls, klv, kmn, kmo, kmp, kmq, kms, kmt, kmu, kmv, kmw, kmx, kmy, kmz, knp, kns, knv, kny, kos, kov, kpv, kpw, kqv, kst, ksu, ksv, ksw, ksx, ktv, kuv, kvw, kvx, kvy, kvz, kwy, bks, bkv, lmn, lmt, lmw, lnp, lns, lnv, lny, lpw, lst, lsw, ltv, lvw, bms, bmt, bmv, bmw, lwy, bmy, bmz, bns, bnv, bny, mns, mnt, mnu, mot, mps, mpt, mqs, mqt, mst, msv, msw, msy, msz, mtu, mtv, mtw, mtx, mty, mtz, muw, nos, nps, npt, npu, nqs, nrs, nst, nsu, nsv, nsw, nsx, nsy, nsz, nsA, ntv, nty, nuv, nuy, ost, osw, otv, psv,psw,ptv,ptw,puw, qsv, qsw, qtv, rsw, bjs, bjt, bjv, bjw, cks, ckv, cns, cnv, cny, dks, dkt, dku, dkv, dkw, dkx, dms, dmt, dns, dnt, dnu, djs, djt, djv, djw, eks, ekv, ems, emt, ens, ejs, ejt, ejv, ejw, fks, fkv, fns, gms, gmt, gns, gnt, gnu, hms, hmt, hns, ins

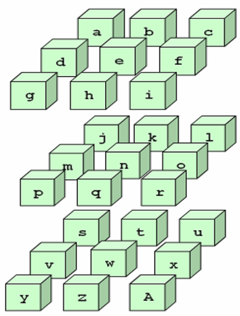

Figure 3: The 3D HLAC mask patterns in string sequence forms and corresponding mask positions

Basically, application methods for 3D HLAC mask patterns to 3D solid textures are similar to the method used in previous research in 2D texture images. The 3D HLAC mask patterns are applied to 3D solid textures as shown in Figure 4. In the figure, there are two cubes. The outer large cube represents a portion of a 3D solid texture and the inner small cube represents a 3D HLAC mask. Intuitively, the 3D HLAC mask moves inside of the large cube randomly. When the 3D HLAC mask moves to a new position, the intensity values of the 3D solid texture are extracted at the point where the mask and voxel cross. The intensity values are multiplied by 1 if mask patterns are on, and multiplied by 0 if the mask is off in order to calculate the HLAC features. These multiplied values are added to one of the shape descriptor slots based on the mask categories. Since there are 251 unique mask patterns, 251 shape descriptors are calculated for each 3D solid texture. These shape descriptors are compared by using a histogram for classifying 3D solid textures. Each 3D solid texture has unique shape feature signatures, and can be compared by using the histogram. 3D solid textures with similar signatures are classified into similar groups. The classification decision is made by using statistical methods such as linear discriminant analysis (LDA) in which the method optimally separates objects into one of a number of groups.

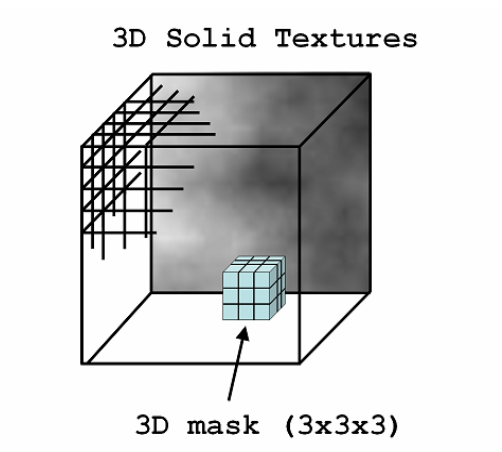

Figure 4: A 3D solid texture and a 3D HLAC mask

3.2 Multi-Resolution Pyramids

It is known that shape descriptors extracted from extremely high or extremely low resolution 2D images are not suitable for some pattern recognition applications because HLAC shape descriptors capture local shape patterns rather than global shape patterns. In pattern recognition applications, often multiple resolutions of 2D images are generated for extracting shape descriptors to avoid low recognition rates. Generation of multi-resolution images can be done by each pixel in the 2D images combined with neighboring pixels, and these images have a low resolution. By repeating the recursive processes, multi resolutions of 2D image data are generated as shown in Figure 5. Often, the set of images are referred to as an image pyramid. The base part of the pyramid represents 2D images with higher resolutions. The top part of the pyramid represents 2D images with a low resolution. In a similar manner, 3D solid texture pyramids are generated as shown in Figure 6. In each voxel in the 3D solid texture, eight neighboring voxel values are averaged and combined to form the lower resolution 3D solid texture.

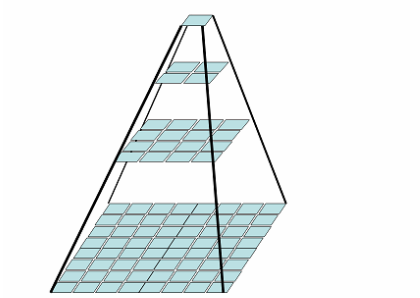

Figure 5: An image pyramid

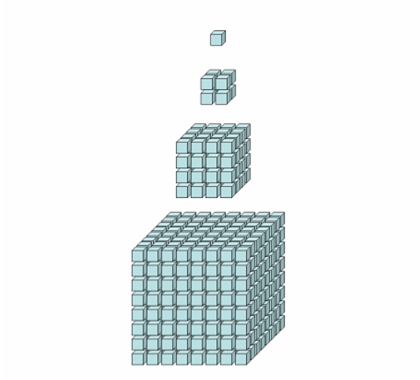

Figure 6: 3D solid texture pyramids

Once the multi resolution pyramid is created for each 3D solid texture, 3D HLAC features are extracted from each level of the pyramid. Different HLAC feature values are computed by reflecting the resolutions of the 3D solid textures. The system calculates HLAC feature values for all levels of the pyramid. The classification rates vary depending on each level of the HLAC features, and the system chooses one of the levels for optimal classification outputs.

4 Experiments and Results

Experiments are conducted for testing 3D HLAC masks using multi resolution pyramidal structures. Examples of similarity search and classification rates have been calculated.

4.1 System

A Pentium IV 2.66 MHz CPU with 1024 Mbytes of memory is used to perform an experimental system. The system was implemented by using C language running on a Linux operating system. An Apache httpd server is used to accept users' search query requests. Figure 7 shows the snap shot of the search system. Users can select a texture database or combinations of texture databases for retrievals. The system returns 3D solid textures with similar patterns in ascending order against query key similarity values. Each 3D solid texture is represented in VRML (Virtual Reality Modeling Language) 3D model data and as slices of 2D images.

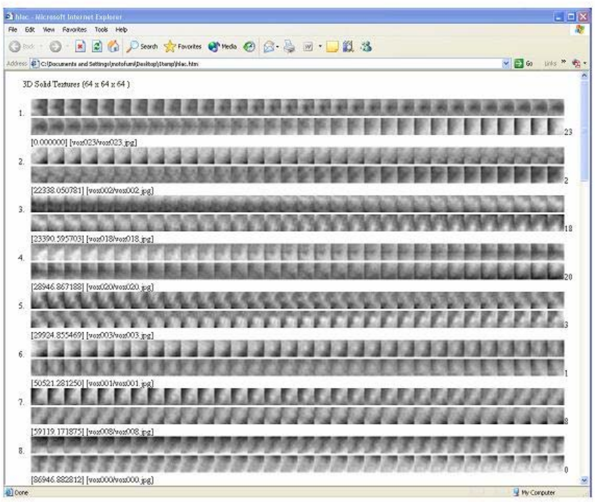

Figure 7: Snap shot of search system.

4.2 Examples of Similarity Search

3D solid textures of a 64 x 64 x 64 voxel grid are generated based on Perlin noise function which was described in the previous section. A total of 1000 3D solid textures are synthesized as a database. The database is composed of type (a) and type (b) textures. Each 3D solid texture is processed to form multi-resolution sizes and levels of 64 (level 0), 32 (level 1), 16 (level 2), 8 (level 3), and 4 (level 4). Figures 8a, 8b and 8c show similarity search examples of 3D solid textures. Each 3D solid texture is represented as 16 slices of 2D images. The number under the 2D images represents the similarity order against the search keys.

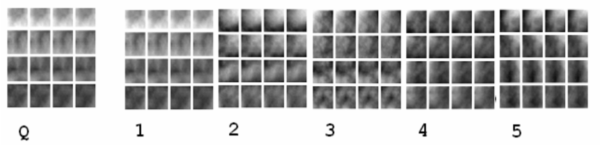

Figure 8a: Similarity Search Example 1.
Level 2 (16 x 16 x 16)

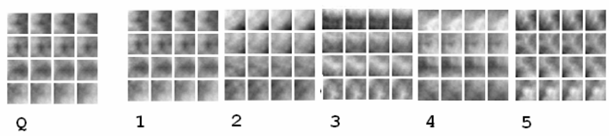

Figure 8b: Similarity Search Example 2.
Level 2 (16 x 16 x 16)

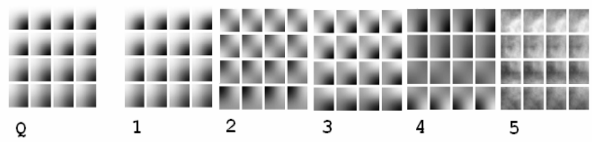

Figure 8c: Similarity Search Example 3.
Level 2 (16 x 16 x 16)

4.3 Recall Precision Graphs

In the experiment, a total of 100 3D solid textures are synthesized as a database which is composed of 50 textures of type (a) and 50 textures of type (b). 3D solid textures with multiple resolutions have been generated from each 3D solid texture in the database. Resolution levels 0 through 5 have been tested to find effective resolutions for similarity retrievals. The number of 3D solid texture data is limited to only 100 to calculate recall precision curves. This is because all relevant items in the database have to be determined by the judgment of the users which requires a significant amount of time. Therefore the number of the items is limited to a small number for the experiment. The items in the database are classified if they are relevant against search key items based on similarity evaluations by five users. Average values of recall rates and precision rates for five users are calculated.

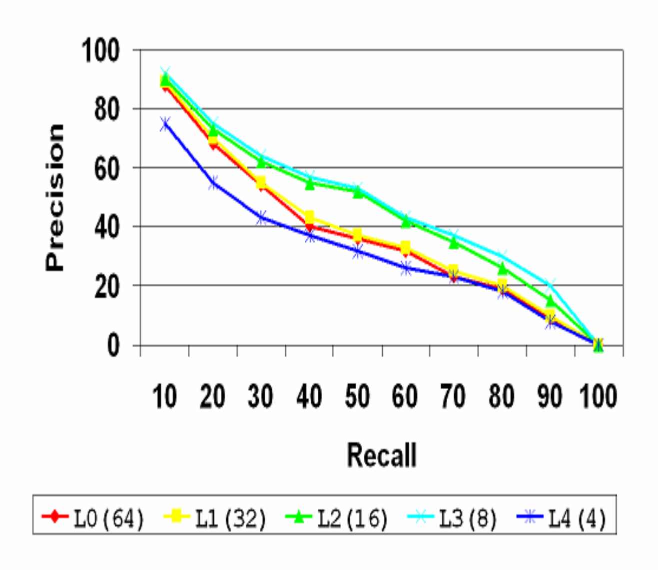

Figure 9: A recall precision graph for different resolutions (Level 0 through 4)

Figure 9 shows a recall precision graph for 3D solid textures with different resolutions. As shown in the Figure 9, resolution levels 2 (16 x 16 x 16) and 3 (8 x 8 x 8) show relatively high recall rates and precision rates. Since 3D HLAC shape descriptors capture local shape features, 3D solid textures with extremely high and low resolutions do not increase the classification rates. This result is quite similar to typical 2D image classification and pattern recognition applications in which extremely high descriptor resolutions do not improve classification rates. Effective resolutions vary and depend on the types of data in the databases. The system should utilize a software program which can generate multiple resolutions for each item in the databases, so various types of 3D solid textures can be handled more accurately.

5 Conclusion and Future Work

Multi-resolution pyramids are used as a data structure for 3D HLAC mask patterns to handle various resolutions of shape descriptors. The multi resolution pyramid data structures are implemented to a search system to evaluate similarity retrieval and pattern classification rates of 3D HLAC. Artificially generated 3D solid texture databases are used for the experiment, and we have found that extremely high or low resolutions are not suitable for similarity searches of 3D solid textures. The use of multi-resolution pyramids helps pattern classification systems handle various types of 3D solid textures, and it also improves classification rates.

In the experiment, only 3D solid textures based on Perlin's noise functions are used. To identify the efficiencies of using 3D HLAC, other types of 3D solid textures will be tested as a future work.

Acknowledgements:

This research was supported by research grants from the National Institute of Information and Communications Technology and a grant-in-aid from the Ministry of Education, Science, Sports and Culture, Japan (#EYS-B(2)-15700115).

References:

[1] T. Kurita, N. Otsu, and T. Sato, Proceedings of 11th International Conference on Pattern Recognition, Vol.II, pp.213-216, 1992.

[2] N. Otsu and T. Kurita, A new scheme for practical flexible and intelligent vision systems, Proceedings of IAPR Workshop on Computer Vision, Tokyo, pp.431-4352, 1998.

[3] K. Hotta, T. Kurita and T. Mishima, Scale Invariant Face Detection Method using Higher-Order Local Autocorrelation Features extracted from Log-Polar Image, Third IEEE International Conference on Automatic Face and Gesture Recognition, pp.70-75, 1998.

[4] F. Goudail, E. Lange, T. Iwamoto, K. Kyuma and N. Otsu, Face Recognition System using Local Autocorrelations and Multi-scale Integration, IEEE Trans. on Pattern Analysis and

Machine Intelligence, Vol. 18, No. 10, pp.1024-1028, 1996.
[5] V. Popvici and J.P. Thiran, Higher Order Autocorrelations for Pattern Classification, Proceedings of the International Conference on Image Processing, 2001.
[6] J.A. MacLaughlin and J. Raviv, Nth-Order Autocorrelations in Pattern Recognition, Information and Control, Vol.12, pp.121-142, 1968.
[7] T. Kurita and S Hayamizu, Gesture Recognition Using higher order local autocorrelation features of PARCOR, IEICE Trans. Inf. & Syst., Vol.E86-D, No.4, April, 2003.
[8] D. Chetverikov and Z. Foldvari, Affine-Invariant Texture Analysis in Machine Vision, World Scientific, 2000, Series in Machine Perception and Artificial Intelligence.
[9] M. Kubo, Z. Aghabari, K.S Oh, and A. Makinouchi, Image Retrieval by Edge Features Using Higher Order Autocorrelation in a SOM Environment. IEICE Trans. Info and Sys., Vol. E86-D, No.8, Aug. 2003.
[10] J. Huang, R. Yagel,V. Filippov, Y. Kurzion, An Accurate Method for Voxelizing Polygon Meshes, ACM 1998 Symposium on Volume Visualization, 1998, 119 – 126.
[11] Tomas Akenine-Moller, Fast 3D Triangle-Box Overlap Testing, Journal of graphics tools 6(1):29-31, 2001
[12] Ken Perlin, An image synthesizer, Computer Graphics (SIGGRAPH 1985 Proceedings), Volume 19, pp.287—296, 1985.
[13] D. S. Ebert, F. K. Musgrave, D. Peachey, K. Perlin, and S. Worley, Texturing and Modeling, A Procedural Approach, ISBN 0-12-228760-6, Academic Press, 1994.
[14] K. S. Fu and S. Y. Lu. Computer generation of texture using a syntactic approach, Computer Graphics (SIGGRAPH 1978), Vol.12, pp.147-152, 1978.
[15] S. Haruyama and B. A. Barsky, Using stochastic modeling for texture generation, IEEE Computer Graphics and Applications, 4(3) pp.7-19, 1984.
[16] B. J. Schacter and N. Ahuja, Random Pattern Generation Processes, Computer Graphics and Image Processing, Vol.10, pp.95-114, 1979.
[17] A. Fournier, D. Fussell, and L. Carpenter, Computer rendering of stochastic models, Communications of the ACM, 25 (6), pp.371-384, 1982.
[18] M. T. Suzuki, Y. Yaginuma, and N. Osawa, A HLAC Shape Descriptor Extraction Method for 3D Solid Textures, The World Scientific Engineering Academy and Society Transaction on Computers, Issue 3, Volume 3, pp.768--773, ISSN:1109-2750, 2004.
[19] M. T. Suzuki, Y. Yaginuma, and N. Osawa, A Similarity Evaluation Method for 3D Models by Using HLAC Mask Patterns, The World Scientific Engineering Academy and Society Transaction on Computers, Issue 3, Volume 3, pp.713--718, ISSN:1109-2750, 2004.

[04/2004m]