Proceedings of the World Congress on Engineering and Computer Science 2010 Vol I
WCECS 2010, October 20-22, 2010, San Francisco, USA

# A Statistical Method for Selecting Pattern Descriptors of Textured 3D Models

Motofumi T. Suzuki * Yoshitomo Yaginuma * Haruo Kodama *

*Abstract*—**This paper describes a similarity retrieval technique for 3D models with solid textures. Three dimensionally extended fractal based descriptors have been extracted from each 3D model in a database. Various sizes of fractal filters were used for extracting descriptors of the 3D models, and the descriptors were compared as indices of the 3D models. Since the filter size used in the system affects retrieval performances, it is important to determine the correct size. In our experiment, portions of the database were marked as a learning data set, and the relationship between the descriptors and filter sizes was analyzed by multiple regression analysis using the learning data set. Our experimental system reflects the analysis results for choosing optimal filter size at each query for maximizing similarity retrieval performance. This preliminary experimental retrieval system has been implemented for searching for similar 3D models with solid texture patterns. The experimental results of our approach showed retrieval performance improvements in terms of recall-precision rates.**

***Keywords: 3D Model, Volumetric Data, Solid Texture, Similarity Retrieval,Pattern Descriptor***

## 1 Introduction

In recent years, computation powers of both software and hardware for computer graphics have increased rapidly. Computer graphics applications which use 3D models are available for not only typical desktop computers but also for laptop computers, PDAs, and even graphical cellular phones. Due to the increasing number of 3D models, search engines for 3D models are important applications today. Since 3D models are represented as sets of numerical data, retrieval indices are needed for the search engines. The indices could be keyword-based metadata or numerical descriptors. Since the indices play important roles in 3D model search engines, intensive research related to 3D model retrieval has been taking place since the late 1990s. Today over several hundred papers can be found related to research topics such as similarity retrieval, shape descriptor extraction, and shape descriptor comparisons. In general, descriptors of the primitive attributes for 3D models include (1) shapes, (2) colors and (3) textures. Most similarity retrieval research focuses on attributes related to the shapes of 3D models, however, there has been little research conducted on attributes related to colors or textures. Details of the various types of shape descriptors can be found in several important survey papers and books [1] [2] [3].

Although textures applied to 3D models influence the similarity evaluations of 3D models, descriptors for 3D models with textures have not been sufficiently investigated. One important pioneering work of a 3D model retrieval technique [4] demonstrated an intuitive solution to handle attributes including shapes, colors, and textures using their web-based system. The system allows users to control weight values applied to each attribute, and 3D models were retrieved by the system reflecting which attributes were used dominantly. Some papers [5] [6] deal with color descriptors including material color or light information of 3D models including transparency and reflection. There are several view based type 3D model retrieval techniques which use various 2D images for capturing a 3D model from multiple viewpoints. This kind of method converts and captures three dimensional visual information including shapes, colors and textures into 2D images, thus it is possible to integrate these attributes for similarity retrievals. A paper [7] introduced a similarity retrieval technique for 3D models with 3D solid textures. The 3D solid textures are different in presentation compared to other 2D UV mapping type textures as shown in Figure 1. The paper described a technique for synthesizing a database of textured 3D models from both a 2D image database and a 3D model database. Descriptors for each textured 3D model were extracted using fractal based analysis using multiple sizes of fractal filters.

In our research, the similarity retrieval technique for the solid textured 3D models was examined using the fractal-based descriptor extraction technique in conjunction with a multiple regression analysis. Our preliminary experimental system can choose an adequate fractal descriptor filter size reflecting multiple regression analysis.

*The Open University of Japan, 2-11 Wakaba, Mihama-ku, Chiba-shi 261-8586, Japan. WEB: http://www.code.ouj.ac.jp/ Email: motofumi@ouj.ac.jp; suzuki.motofumi@gmail.com

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

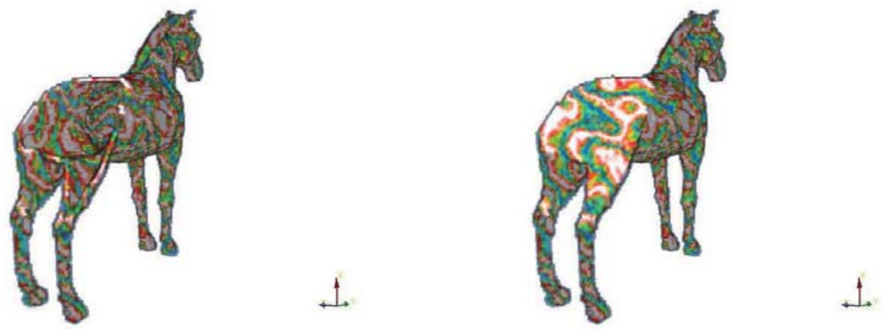

Figure 1: UV texture mapping (left) and solid texture mapping (right).

## 2 Similarity Retrievals of Textured 3D Models

This section describes (1) a 3D model with multiple attributes, (2) the process required for similarity retrieval and (3) fractal descriptors.

### 2.1 3D Model with Multiple Attributes

When 3D model descriptors with multiple attributes are used in retrieval systems, several methods are used to handle these descriptors depending on software applications. For instance, if shape descriptors are used for 0% and texture descriptors are used for 100%, the system can search for 3D models with similar textures regardless of the shapes of the 3D models. Oppositely, if shape descriptors are used for 100% and texture descriptors are used for 0%, the system can search for 3D models with similar shape regardless of the textures of the 3D models. Also, it is possible to apply arbitrary weights to descriptors for mixing multiple attributes, and the system can search for 3D models with both similar shapes and similar textures by reflecting the weights.

Another approach for mixing multiple attributes can be performed in two stages. For the first stage, the system can search for 3D models based on shape descriptors and can obtain a sorted similarity list. For the second stage, the system can use the list for identifying identical shapes of 3D models and can apply search based on texture descriptors. When the database contains an extremely large number of 3D models, the system has a higher possibility of returning a larger number of 3D models with identical shapes. In such a case, texture based descriptors can be used for retrieval of 3D models with these identical shapes for refining retrieval results.

In this research, the system is implemented for dealing with only texture descriptors. In other words, the system ignores shape descriptors of each 3D model, and the system retrieves 3D models with similar texture pattern regardless of the 3D model shapes. The system extracts descriptors from inside of the 3D models, and descriptors are not extracted from the surface of the 3D models. Since the number of sampling points is different in each 3D model, the descriptors are normalized in consideration of the number of sampling points.

### 2.2 Process Required for Similarity Retrieval

Basically, the procedure is quite similar to the similarity retrieval of 2D images and 3D polygonal models. In our approach, three dimensional descriptors are used rather than two dimensional descriptors. Our three dimensional descriptor extraction technique allows a system to directly obtain descriptors from the 3D models, and conversion of the 3D models into a series of 2D images is not necessary.

### 2.3 Fractal Descriptors

The fractal-based descriptors have been used in various 2D texture image analysis applications. These applications estimate the fractal dimensions of the 2D textures images, and the fractal dimensions can be used for evaluating patterns of 2D textures. Several methods are used for computing fractal dimension which include a box-counting method and Hurst analysis[8]. In our experiments, the Hurst analysis technique is used because the fractal dimension value is computed at arbitrary points in the 3D solid textures, which is a convenient method of handling various shapes of 3D models. Details of the Hurst analysis and the related applications can be found in many papers [8] [9] [10]. The Hurst analysis involves filter operations which are similar to the convolution computations. Various sizes of fractal dimension filters (FD3, FD5, FD7 and FD9) can be used for the computations, and unique descriptor values are obtained reflecting the filter sizes. For example, a larger filter can capture thick wood grain patterns, and a small filter can capture thin wood grain patterns as shown in Figure 2. Determination of adequate filter size, and the characteristics of textures including resolution, frequency, directionality and

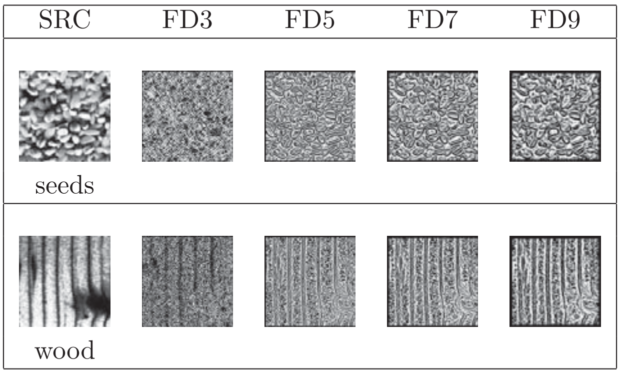


Figure 2: Images generated by applying various sizes of fractal filters. A larger filter (FD9) can capture thick wood grain patterns, and a small filter (FD3) can capture thin wood grain patterns.

anisotropy should be carefully considered. In our experiments, multiple regression analysis was used for determining the filter size. The analysis is used to learn about the relationship between several predictor variables and criterion variable. The portions of the database for the textured 3D models were marked as a learning data set. All the learning data sets were analyzed for extracting fractal descriptors (FD3, FD5, FD7, and FD9), and the similarity retrievals were conducted on the entire set of data for computing recall-precision rates. The averages of recall-precision rates and the averages of F-measure values for each texture type were computed. In the analysis, filter size was set to criterion variable (3, 5, 7 and 9), and texture descriptors were set to predictor variables (FD3, FD5, FD7, and FD9; 20 descriptors $\times$ 4).

## 3 Experiments and Results

This section describes (1) experimental databases of textured 3D models, (2) similarity retrieval performances and (3) similarity retrieval using our approach.

### 3.1 Experimental Databases of Textured 3D models

To test the method described in the previous section, a database of 3D solid textures was synthesized first, and then the synthesized data are applied to a database of 3D models. The solid textures can be synthesized in various ways. One of the most popular technique is a procedural texture approach [11], and noise functions are involved in the technique for synthesizing natural irregular patterns. In recent years, other types of techniques have been introduced for 3D solid texture synthesis [12] [13]. These new techniques have made it possible to synthesize solid textures from 2D texture images. Since the 2D images are not limited to images synthesized by noise functions, various patterns of 3D solid textures can be obtained. Although these new techniques can synthesize realistic 3D solid texture patterns, some techniques involve heavy computation of voxel searching and matching. Since our focus was synthesizing an experimental database that contains a large number of 3D solid textures with various patterns, a simple and fast technique was applied which was used in paper [7]. The method is also similar to an interpolated texture method described in paper [14]. A database of 3D models was obtained from the Princeton Shape Benchmark (PSB)[15]. A database of 2D image textures was obtained from OUTEX [16] which is a set of texture benchmark data. Although the OUTEX benchmark data contain a large number of various types of textures, only a portion of the data was used for our experiment. 3D model data were voxelized using the software program called 'binvox' [17] which implemented the currently available efficient voxelization algorithms [18]. In our experiments, voxel sizes of 128 $\times$ 128 $\times$ 128 were specified, and the program was used for converting the 3D polygonal models onto the 3D voxelized models. A simple Boolean operation was used for mapping the 3D solid texture onto the voxelized 3D model. Because of limitation of hard disk space, only 20% of the OUTEX 2D images were used for our experiments. The data were selected by skipping 5 data indices repeatedly, and thus 1276 solid textures were used. There are 319 classes of data and each class contains 4 data sets. The classes were grouped into 28 types, and the types of textures are shown in Figure 4 in the next subsection.

### 3.2 Similarity Retrieval Performances

Similarity retrieval performances were evaluated using descriptors computed based on various fractal filter sizes (3, 5, 7, and 9), and corresponding descriptors (Fractal Dimension) are denoted as FD3, FD5, FD7 and FD9. Figure 3 shows a graph of recall-precision rates for each retrieval performance, and the lines in the graph indicate the retrieval performance computed using FD3, FD5, FD7 and FD9. In addition, F-measure values were computed from the recall-precision rates (Figure 4). In general, a larger F-measure value indicates better retrieval performance, and thus the similarity performance using FD5 was most optimal in this case.

### 3.3 Similarity Retrieval Performances using our approach

In our approach, portions of experimental data were marked as learning data. In the experiments, 50% of the data items (638 items) were randomly marked as learning data. The learning data set was used for finding the relations between texture descriptors and filter size. The multiple regression analysis technique was applied, and the regression function was obtained which predicts the best filter size of descriptors to be used for each query. The R-square value for the analysis was 0.901, and it was an acceptable value for the use of the regression function. The R-square values which were an indicator of how the regression function prediction were correct. All of the 3D

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

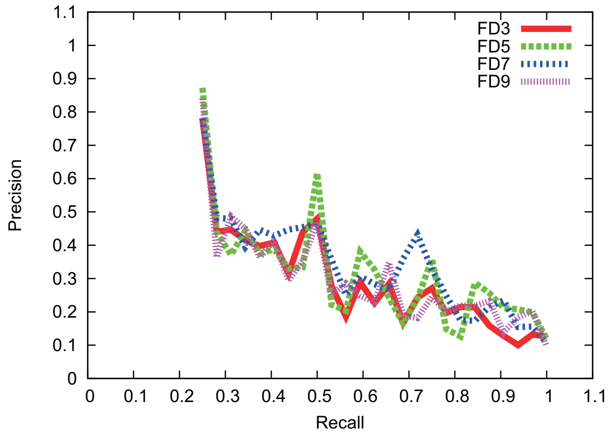


Figure 3: Recall-Precision graph (FD3, FD5, FD7, and FD9). Similarity retrieval performances were evaluated using descriptors computed based on various fractal filter sizes (3, 5, 7, and 9).

| | texture | FD3 | FD5 | FD7 | FD9 |
|---|---|---|---|---|---|
| 00 | bar.rice | 9.7(1) | 10.1(0) | 9.0(2) | 8.2(3) |
| 01 | canvas | 23.6(3) | 24.5(1) | 24.4(2) | 25.0(0) |
| 02 | c.board | 1.9(3) | 1.9(2) | 1.9(1) | 1.9(0) |
| 03 | carpet | 13.4(3) | 17.2(0) | 17.1(1) | 15.7(2) |
| 04 | chips | 14.6(0) | 12.7(2) | 14.1(1) | 12.0(3) |
| 05 | c.stone | 10.6(1) | 10.9(0) | 10.3(2) | 9.8(3) |
| 06 | flakes | 11.5(1) | 11.8(0) | 11.4(2) | 10.6(3) |
| 07 | flour | 13.9(0) | 13.6(1) | 12.8(3) | 12.9(2) |
| 08 | foam | 9.4(1) | 9.5(0) | 9.0(2) | 8.6(3) |
| 09 | fur | 13.1(3) | 14.0(0) | 13.3(2) | 13.7(1) |
| 10 | granite | 7.6(2) | 7.6(1) | 7.6(0) | 7.6(3) |
| 11 | granu. | 12.4(0) | 11.5(1) | 10.2(3) | 10.9(2) |
| 12 | gravel | 7.7(0) | 6.9(1) | 6.5(2) | 6.4(3) |
| 13 | groats | 7.6(3) | 9.2(0) | 8.9(1) | 8.7(2) |
| 14 | leather | 12.3(0) | 9.9(3) | 10.1(2) | 10.3(1) |
| 15 | mineral | 8.0(3) | 9.2(0) | 8.3(1) | 8.1(2) |
| 16 | paper | 11.4(0) | 8.9(2) | 8.9(1) | 8.6(3) |
| 17 | pasta | 15.1(3) | 16.3(0) | 15.7(1) | 15.7(2) |
| 18 | pellet | 7.4(3) | 11.3(0) | 10.2(2) | 10.5(1) |
| 19 | plastic | 6.4(1) | 5.3(2) | 5.1(3) | 7.0(0) |
| 20 | quartz | 18.5(3) | 23.6(2) | 23.9(0) | 23.6(1) |
| 21 | rubber | 10.8(0) | 10.8(1) | 10.1(2) | 9.0(3) |
| 22 | sand | 1.9(3) | 1.9(2) | 1.9(1) | 1.9(0) |
| 23 | spaper | 10.0(0) | 9.5(2) | 9.6(1) | 9.0(3) |
| 24 | seeds | 11.2(0) | 7.8(3) | 10.7(1) | 8.5(2) |
| 25 | tile | 12.1(3) | 16.7(0) | 13.6(2) | 14.6(1) |
| 26 | wpaper | 9.7(1) | 10.1(0) | 8.8(3) | 9.5(2) |
| 27 | wood | 16.1(3) | 17.7(1) | 17.2(2) | 17.9(0) |

Figure 4: F-measure values for each texture. F-measure values computed from recall-precision graphs using the following equation: Fm $= 2 \times$((precisionrecall) $\div$ (precision+recall))

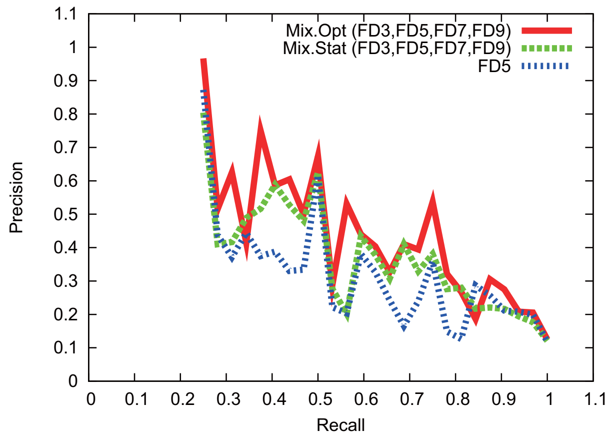


Figure 5: Recall-Precision graph (FD5, FD-V[Mix.Stat] and the system optimal[Mix.Opt] )

models (1276 items) in the database were analyzed to determine whether the regression function can successfully predict correct filter size for each queried 3D model. In the experiments, 90% (1149 out of 1267) of the filters were selected by the system correctly.

Figure 5 shows a graph of recall-precision rates for both FD5 and FD-V (FD with Various filter sizes) which is the method proposed in this research. In the figure, the FD-V shows good retrieval results compared to that of FD5. Figures 6 and 7 show examples of the similarity retrieval of textured 3D models. In each figure, the ten most similar 3D solid textures determined by the system are shown, and the left image shows the query key. In each textured 3D model, the corresponding 2D texture used to synthesize the 3D solid texture is shown. The retrieval examples marked with asterisks '*' indicate that the retrieval uses the filter size recommended by our experimental system.

## 4 Summary and Future Work

In our research, a similarity retrieval technique for textured 3D models is investigated. For our experiments, a database that contains a large number of 3D solid textures with various patterns was synthesized. Each textured 3D model was evaluated using fractal based descriptors which were computed based on various filter sizes (FD3, FD5, FD7, FD9 and FD-V). The use of an adequate filter size is important for similarity retrieval, especially for the texture descriptor extractions. The portions of the database were marked as a learning data set, and the relationship between the descriptors and filter sizes were analyzed using a multiple regression analysis. Our experimental system reflects the analysis results for choosing optimal filter size at each query. Preliminary experimental results show improvements in recall-precision rates using our approach.

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

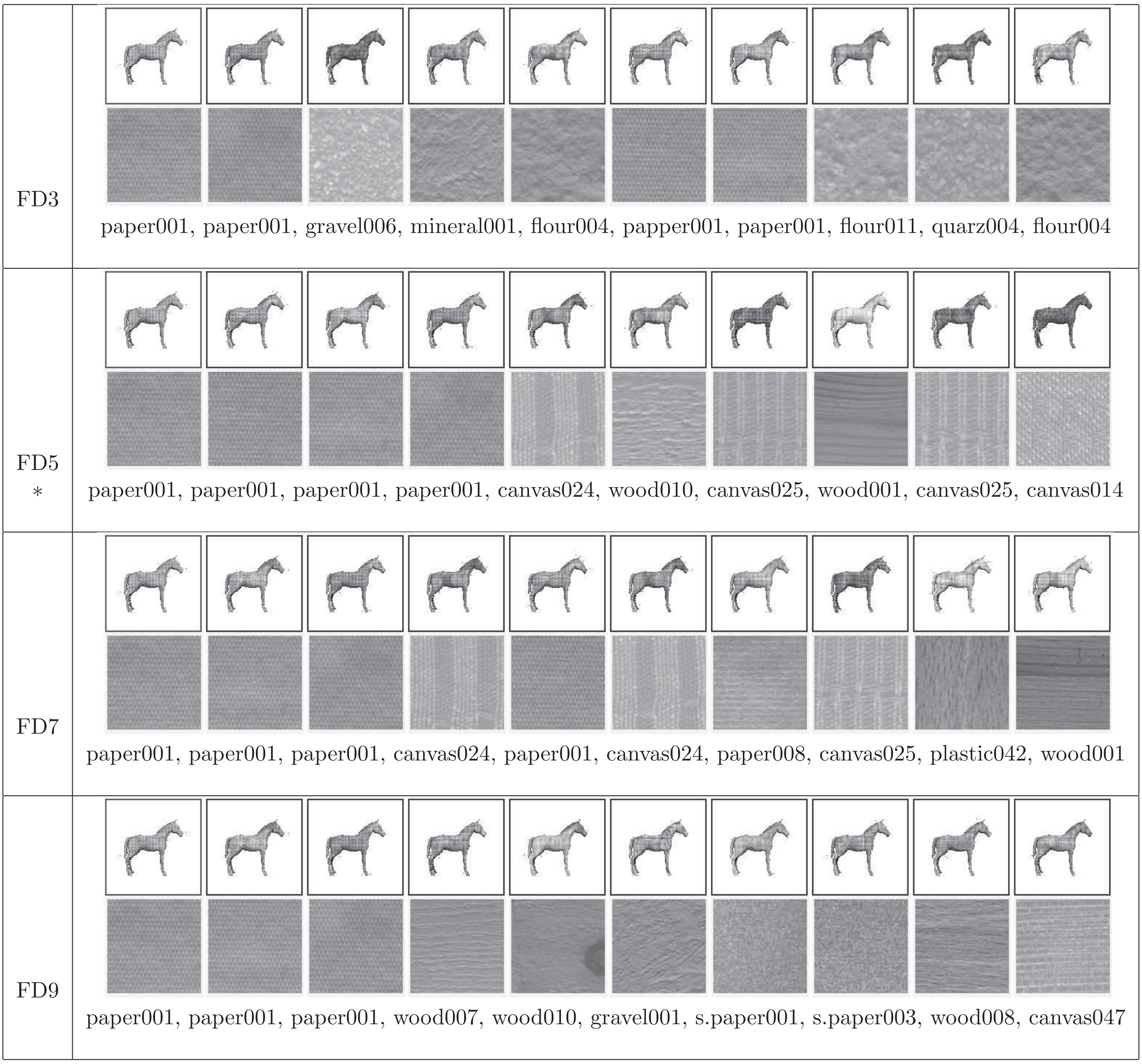


Figure 6: Examples of similarity retrievals. The ten most similar 3D solid textures determined by the system are shown, and the left image shows the query key. In each textured 3D model, the corresponding 2D texture used to synthesize the 3D solid texture is shown. The retrieval examples marked with asterisks $*$ indicate that the retrieval uses the filter size recommended by our experimental system. The preliminary system with a limited number of 3D model data can be accessed at the following web address: http://motosuzuki.air-nifty.com/blog/retrieval-system.html

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

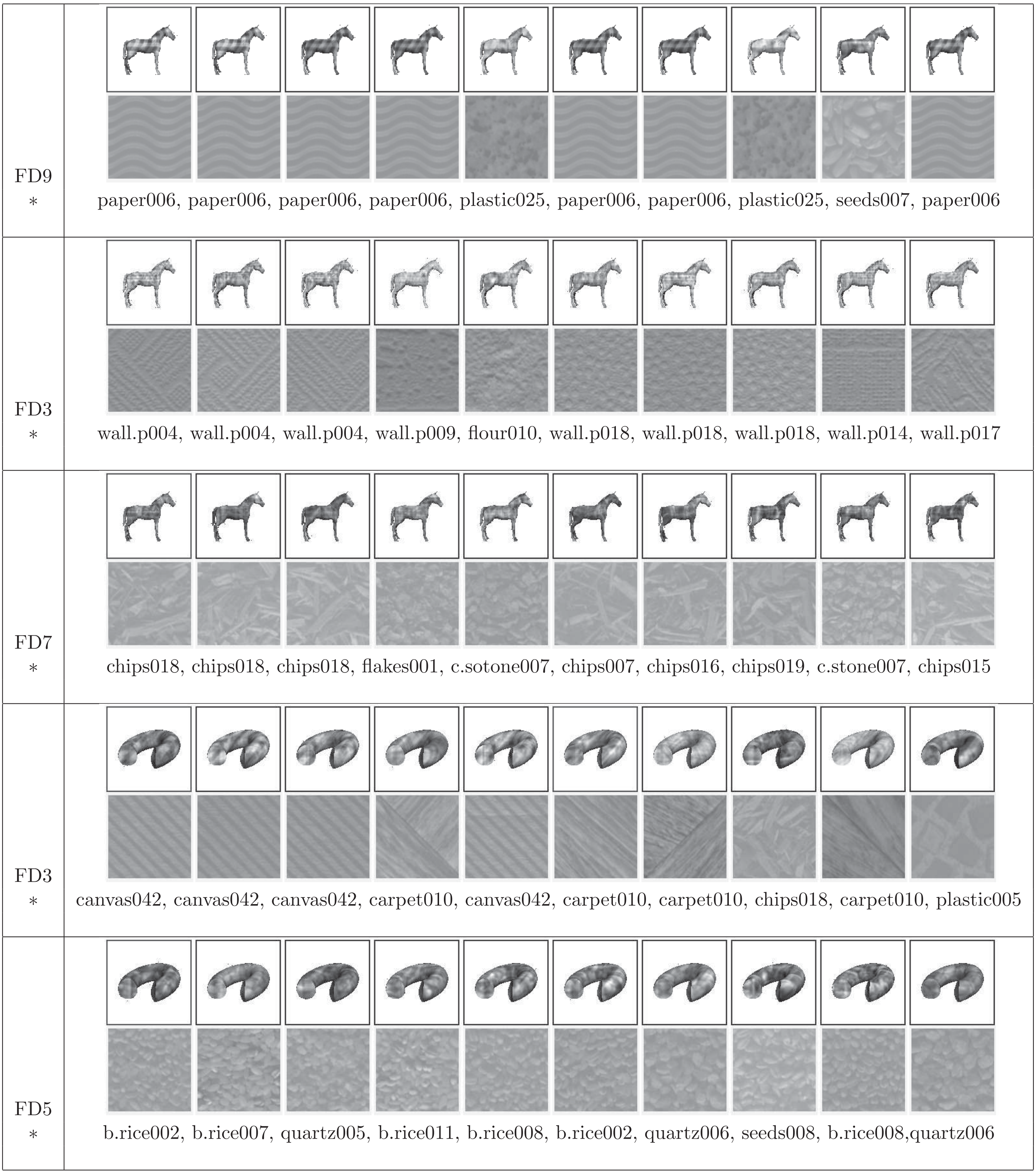


Figure 7: Examples of similarity retrievals. Similarity retrieval performances were evaluated using descriptors computed based on various fractal filter sizes (FD3, FD5, FD7, and FD9). The use of an adequate filter size is important for the similarity retrieval especially in the texture descriptor extractions. Our experimental system reflects the analysis results for choosing optimal filter size at each query.

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Currently, our system is designed for analyzing 3D solid textures. The system will be extended to handle not only 3D solid textures but also UV mapping type textures. Other types of descriptors which can extract texture patterns from the 3D models will be investigated, and the similarity retrieval of those descriptors will be compared as a future work.

**Acknowledgments**
This research was supported by research grants from a grant-in-aid from the Ministry of Education, Science, Sports and Culture, Japan (Wakate-B-21700127).

## References

[1] Yubin Yang, Hui Lin, Content-based 3D model retrieval: a Survey, *IEEE Transactions on Systems, Man and Cybernetics: Part C- Applications and Reviews*, Vol.37(6), pp.1081-1098, 2007.

[2] J. W. H. Tangelder, R. C. Veltkamp, A Survey of Content Based 3D Shape Retrieval Methods, *Proceedings Shape Modeling International*, pp.145-156, 2004.

[3] Jean-Luc Dugelay, Atilla Baskurt, Mohamed Daoudi, 3D Object Processing: Compression, Indexing and Watermarking, *Wiley*, ISBN: 978-0-470-06542-6, 2008.

[4] Eric Paquet, Marc Rioux: Nefertiti: a query by content system for three-dimensional model and image databases management, *Image Vision Computing*, Vol.17, No.2, pp157-166, 1999.

[5] Wei Wei, Yubin Yang, Jinjie Lin and Jiabin Ruan, Color-based 3D Model Classification using Hopfield Neural Network, *2008 International Conference on Computer Science and Software Engineering*, pp.883-886, 2008.

[6] Motofumi T. Suzuki and Yuji Y. Sugimoto, Personalized Retrieval from Material Color Databases for 3D Polygonal Models, *6th International Conference on Virtual Systems and Multimedia (VSMM2000)*, pp.577-584, 2000.

[7] Motofumi T. Suzuki, Yoshitomo Yaginuma, Haruo Kodama, A Similarity Retrieval Technique for Textured 3D Models, *International Conference on Information Retrieval and Knowledge Management*, (CAMP10), pp.131–137, 2010.

[8] J. C. Russ, Surface Characterization: Fractal Dimensions, Hurst Coefficients, and Frequency Transforms, Journal of Computer Assisted Microscopy, Vol.2, No.3, pp.161-183. 1990.

[9] J. C. Russ, Processing images with a local Hurst operator to reveal textural differences, Journal of Computer Assisted Microscopy, Vol.2, 4, pp.249–257, 1990.

[10] J. C. Russ, Fractal Surfaces, ISBN: 978-0-306-44702-0, 1994.

[11] Ken Perlin, An image synthesizer, Computer Graphics, (SIGGRAPH 1985 Proceedings), Vol.19, pp.287-296, 1985.

[12] Y. Dong, S. Lefebvre, X. Tong, and G. Drettakis, Lazy solid texture synthesis, in Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering), 2008.

[13] J. Kopf, C.-W. Fu, D. Cohen-Or, O. Deussen, D. Lischinski, and T.-T. Wong, Solid texture synthesis from 2d exemplars, ACM Transactions on Graphics (Proc. of SIGGRAPH 2007), vol. 26, no. 3, 2007.

[14] Ludovic Paulhac, Pascal Makris, Jean-Yves Ramel, A Solid Texture Database for Segmentation and Classification Experiments, VISSAPP (2), pp.135-141, 2009.

[15] P. Shilane, et al, The Princeton Shape Benchmark, *Shape Modeling International*, pp.167-178, 2004.

[16] T. Ojala, T. Menp, M. Pietikinen, J. Viertola, J. Kyllnen, and S. Huovinen, Outex - new framework for empirical evaluation of texture analysis algorithms, Int. Conf. on Pattern Rec., pp.701706, 2002.

[17] Patrick Min, binvox (3D mesh voxelizer) http://www.google.com/search?q=binvox, 2004.

[18] Fakir S. Nooruddin and Greg Turk, Simplification and Repair of Polygonal Models Using Volumetric Techniques, *IEEE Transactions on Visualization and Computer Graphics*, vol.9, pp.191-205, 2003.