New Composite Shape and Texture Descriptors for 3D Model Retrieval

Hero Yudo Martono and Masaki Aono
Toyohashi University of Technology, Toyohashi, Japan
Email: hero@kde.cs.tut.ac.jp, aono@tut.jp

***Abstract*—Nowadays, problem of shape and texture for 3D retrieval is still a challenge research. Although several methods exist, but we still have a space to improve the performance. In this paper, we aim to improve our previous 3D shape features and inserting texture features. We first do pose normalization as a process of adjusting the size, location, and orientation of a given object in a canonical space and generate three types of non-color image rendered such as silhouette image, depth buffer image and contour image and one binary voxel. We get shape features from four independent Fourier spectra with periphery enhancement, which called multi Fourier spectral descriptor. We second generate both color image rendered and color voxel. We build a color histogram as texture features by using both color images rendered and color voxel based on distance and color level. Shape and texture features are finally combined together by linear summation. We conduct experiments based on the SHREC 2013 and 2014 track retrieval on textured 3D models. The experiment results show on how our method outperform in NN while using dataset SHREC 2013 and in FT and ST while using SHREC 2014.**

***Index Terms*—3D texture, pose normalization, shape features, texture features, voxel**

I. Introduction

The increasing number of 3D models on the internet with a variety of information about shape and texture, coupled with the emergence of (SHape REtrieval Contest) SHREC track: Retrieval and Classification on Textured 3D models, in 2013 [1] and 2014 [2] has led competition to produce a more efficient algorithm. Although 3D model retrieval system mainly based on the shape and texture is relatively new, but there were some existing methods, some of them will be briefly described.

In SHREC 2014, several researchers proposed some methods. Abdelrahman *et al.* [2] described a method 3D shape textured by combining a shape and a photometric contribution and used scale invariant heat kernel signature [3]-[5]. V. Garro and A. Giachetti proposed Histogram of the Multiscale Area Projection Transform (MAPT) [6]. This method was based on a spatial map that encoded the likelihood of the points inside the shape of being centers of spherical symmetry. We also proposed our method in [2] by two runs using combination Histogram of Oriented Gradients (HoG), Local Binary Pattern, Local Ternary Pattern, Weber Local Descriptor, Multi Fourier Spectral Descriptor and Multiresolution Representation Local Binary Pattern Histogram (MRLBPH) which captured texture features of rendered image from 3D model by analyzing multi resolution representation using LBP. C. Li, A. Godil, A. Ben Hamza used the spectral geometry based framework in [7]-[9] for texture 3D shape representation and retrieval. This method was based on the eigen decomposition of the Lapclace-Beltrami Operator (LBO), which provided a rich set of Eigen bases that were invariant to isometric transformation. S. Velasco-Forero proposed a method that basically computed two features: a shape and a color descriptor [10]-[11]. A shape was represented by a Geodesic Distance Matrix (GDM) and a color was represented by a CIElab color histogram. The basic idea was to compute the average Earth Mover Distance (EMD) distance between RGB histogram for two given shapes. D. Girgi proposed Textured Shape Distribution (TSD) which was a color-aware variant on classical Shape Distribution. He also tried to combine with several other methods [12], [13]. The conclusion of the most common approach was to combine features of shape and texture.

Based on the above reasons, we propose a novel method for 3D texture retrieval. We extend our previous 3D shape features, called MFSD, by combining texture features. We produce texture features by building a color histogram. We build a color histogram by extracting both color image rendered and color voxel on each 3D model. Therefore generating a good quality both color image rendered and color voxel is very important. Because of every 3D object has different in shape, smoothness, position, and scale, then we first do pose normalization in advance to obtain a normal 3D object in terms of both size, position and pose. M. Church, A. V. Blondet [14] and Vranic [15] also proposed pose normalization method. In this case we use our pose normalization point SVD [16].

The main contribution of this paper includes: (1) Generating texture features by building a color histogram based on color image rendered and color voxel. As far as we know, this is the first time extracting color voxel to build a color histogram. Quality of color voxel is very important to generate good discriminant texture features. (2) Combining our previous shape features and new

Manuscript received January 20, 2015; revised August 25, 2015.

©2015 Journal of Image and Graphics

doi: 10.18178/joig.3.2.107-111

texture features as linear combination. After studying the issues related to the previous, we formulate the framework as illustrated in Fig. 1. Section 1 present introduction and related work. Section 2 presents a description of shape features. Section 3 presents a description of texture features. Section 4 presents how to measure the similarity between two 3D objects based on shape and texture. The experimental results are shown in Section 5, and finally Section 6 describes the conclusions of this study.

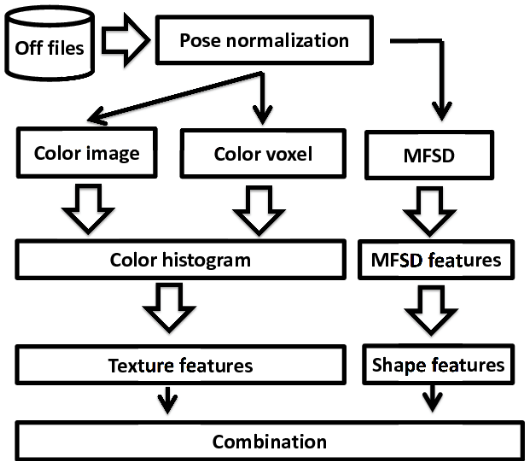

Figure 1. The general architecture of the proposed method, consisting of doing pose normalization, generating image rendered, conducting voxelization, and generating shape features and texture features.

II. 3D Shape Features

Many 3D shape retrieval methods have been proposed. Each method has its own advantages. One of them is our previous method called Multi-Fourier Spectra Descriptor (MFSD). It is composed of four independent Fourier spectra with periphery enhancement. It allows us to faithfully capture the inherent characteristics of an arbitrary 3D object shape regardless of the dimension, orientation, and original location of the object when it is first defined.

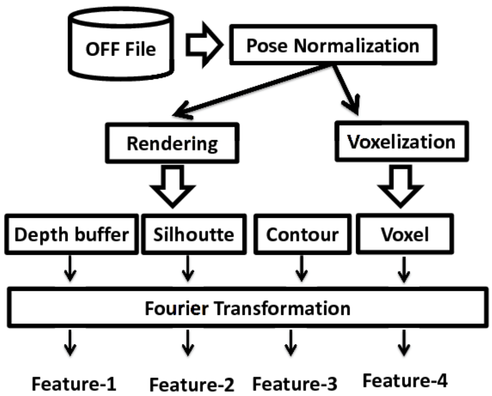

Figure 2. The process of shape features extraction (multi-Fourier spectral descriptor) is started by reading off files, doing pose normalization, rendering, computing Fourier transformation and producing 4 types independent shape features. They are combined by linear summation.

We use MFSD method illustrated in Fig. 2 to obtain a shape feature. The process as follows: 1) Reading off files, 2) Doing pose normalization, 3) Generating a depth buffer image rendered, silhouette image rendered and contour image rendered, 4) Conducting voxelization and producing a non-color voxel, 5) Performing fourier transformation to all images rendered and non-color voxel then generating final shape features, 6) Computing a distance matrix between 3D models, 7) Normalizing a distance matrix. For more detail, process of MFSD refers to the literature [16].

III. 3D Texture Features

Every 3D object has information about number vertices and faces. Each vertex has a coordinate information (x, y, z) and color information (red, green, blue). A Face is a triangular that consists of three vertices. Building color histogram as texture features, we perform two processes both generating a color image rendered and generating color voxel. Getting a good quality of color image rendered, we first do pose normalization as processing. The advantage of pose normalization is shown in Fig. 3. It illustrates the different pose of image result.

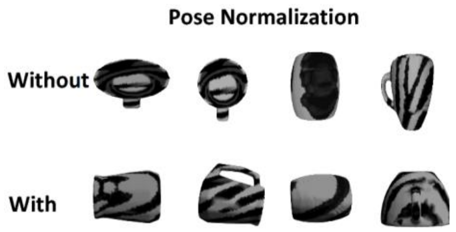

Figure 3. This picture shows the difference pose of images rendered, the upper without using pose normalization and lower using pose normalization.

A. Color Image Rendered

The result of images rendered will be in a different pose, if we render them not inappropriate viewpoint. This will make big problem and generate big different features between two similar objects as shown in Fig. 4. This proves that doing good pose normalization is very important, because it ensures correct viewpoint when we will generate an image rendered.

Figure 4. This picture shows the possibility of color image rendered which generated from several viewpoints.

After generating color image rendered, then we build a color histogram based on distance and color level. We sum color distribution on each pixel to know difference information between images. It means that we calculate distance on each pixel position to center of image, then we build a color histogram on its channels separately (red, green, blue). Fig. 5 presents an example illustration of generating a color image histogram.

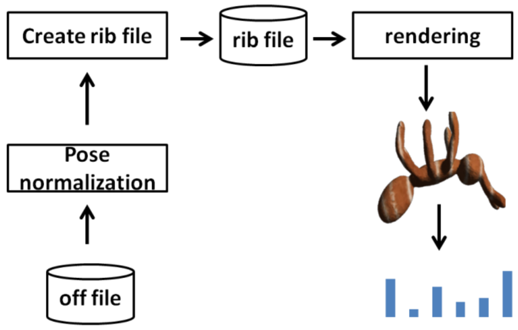

Figure 5. Rendering of a color image is started by reading off files, doing pose normalization, and generating a color image.

A color image rendered has pixel size 400x400. We quantize them based on color and distance. We use on each color channel (RGB) to represent color image information. Finally, we get color image histogram which has 300 bins. Table I shows how we quantize images based on color and distance.

TABLE I. QUANTIZATION INTERVALS FOR IMAGE COLOR

Quantization level	Color	Image Distance	Color Distance
1	0-25	0-20	0-7
2	26-50	21-40	8-15
3	51-75	41-60	16-23
4	76-100	61-80	24-31
5	101-125	81-100	32-39
6	126-150	101-120	40-47
7	151-175	121-140	48-55
8	176-200	141-160	56-64
9	201-225	161-180	-
10	226-255	181-200	-

B. *Color Voxel*

The second of texture features is obtained by building a color histogram based on a color voxel. Fig. 6 shows the process of building color voxel and generating color voxel histogram. A color voxel is a volume element representing a color value on a regular grid in 3D space. By understanding voxels, we can estimate any 3D solid object. We first need to normalize space for voxel representation, map an arbitrary polygonal into cubes. We quantize a cube into 64×64×64 voxels. These voxels features are not like the ones that has been already done by [15] and Tatsuma [16]. They build voxels without including color elements, while this includes color element. To calculate the final value on each voxel, we use interpolation. The proper color value in grid voxel is very important to build a discriminant color histogram.

We will explain how to build voxel. It first has to normalize space by reading the 3D object file to get information about number of vertices and faces. We then calculate mean value to predict center of 3D mass object by summing all vertices and dividing by number of vertices. Next, we seek the most distance point and the closest from the center point then calculate all the area of faces to determine the proper number of random points on each face. This will ensure that each face with different area will get the appropriate number of points. We choose point on each face in accordance with a predetermined number. We next determine point on each face by using the following formula.

$$p = \left(1 - \sqrt{r1}\right) \mathbf{a} + \sqrt{r1}\ (1 - r2)\, \mathbf{b} + \sqrt{r1}\, r2\, \mathbf{c}$$

This formula is very common on determining point in a face. Because every point has color information, we must use an interpolation formula to assign color information, rgb channel, on each new point. Next, we must map every new point to voxel in bounding cube. We then generate color voxel histogram which has 300 bins.

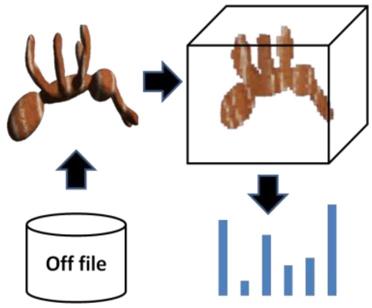

Figure 6. This image describes the voxelization process, it is started by reading off files, performing normalization according to the size specified in 3D space, and calculating color interpolation on each voxels.

IV. DISSIMILARITY MEASURE

The next step is calculating the dissimilarity between 3D objects based on their features. We use a standard 3D retrieval system where given a set of queries (Q) and the other set of candidate 3D models (M). For the experiment, we compute the dissimilarity between the queries with all existing 3D models using their features. The approachment is taken by comparing between shape features and texture features from the query to all existing 3D models separately. We calculate the dissimilarity using Manhattan distance formula. The most similar one is when the dissimilarity value closes to zero. Table II shows the result of each our experiment using dataset texture SHREC 2014. We get the best in experiment when we combine between MFSD and Color Image Histogram first, then we generate mean from them. Finally, we combine with Color Voxel Histogram and this is our final proposed method.

TABLE II. TEXTURE RELEVANT ON SHREC 2014

Method	NN	FT	ST	ADR
MFSD	0.342	0.290	0.431	0.239
Color Image Histogram (CIH)	**0.667**	0.403	0.529	0.239
Color Voxel Histogram (CVH)	0.646	0.392	0.516	0.334
MFSD +CIH	0.541	0.431	0.582	0.239
(MFSD+CIH) + CVH	0.621	**0.463**	**0.630**	**0.377**

V. EXPERIMENTS

To assess the performance of our proposed method, we use a standard SHREC 2013 and 2014 dataset. SHREC 2013 dataset consists of 240 mesh models grouped in 33

©2015 Journal of Image and Graphics

classes that have similar in shape and texture, as well as 10 classes that have in shape only. SHREC 2014 dataset consists of 572 mesh models grouped in 16 shape class and 13 texture class. For assessment, two models are said to be highly relevant if they share shape and texture class. If two models have only on once shape or texture class, they are concluded not highly relevant. Table III shows that our method has best performance on Nearest Neighbor evaluation when using dataset SHREC 2013, while Table IV shows our method outperforms on First Tier and Second Tier evaluation. Fig. 7 shows the precission-recall curve when using dataset SHREC 2014 and our method outperforms on some points.

TABLE III. SHREC 2013 ON TEXTURED 3D MODEL

Method	NN	FT	ST
proposed	**0.928**	0.676	0.835
A1	0.515	0.553	0.710
A2	0.508	0.561	0.730
Be	0.019	0.175	0.209
Zh	0.174	0.135	0.214
Gi	0.788	0.658	0.748
G1	0.417	0.526	0.799
G2	0.898	0.733	0.893
G3	0.519	0.579	0.772
V1	0.807	0.511	0.633
V2	0.879	**0.764**	**0.904**
V3	0.909	0.733	0.863

TABLE IV. SHREC 2014 ON TEXTURED 3D MODEL

Method	NN	FT	ST	ADR
Proposed	0.621	**0.463**	**0.630**	0.377
AEF1	0.098	0.226	0.350	0.205
AEF2	0.123	0.228	0.351	0.206
GG1	0.696	0.404	0.530	0.349
GG2	0.722	0.432	0.557	0.368
GG3	0.665	0.384	0.504	0.336
LBG2	0.676	0.412	0.565	0.353
LBG3	0.512	0.306	0.406	0.279
LBG4	0.394	0.325	0.437	0.284
TA	0.563	0.336	0.456	0.294
Ve1	0.735	0.396	0.540	0.342
Ve2	0.593	0.338	0.469	0.299
Ve3	0.336	0.275	0.369	0.248
XL	0.108	0.149	0.192	0.159
Gi1	0.890	0.324	0.401	0.313
Gi2	**0.894**	0.365	0.448	0.340
Gi3	0.813	0.455	0.590	**0.383**

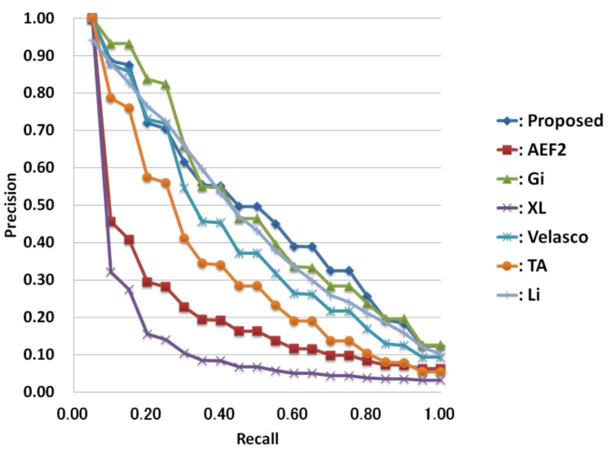

Figure 7. Performance of all participants on SHREC 2014 to the average precision-recall curve.

We try to show some results in Fig. 8 and Fig. 9. The left side is a query while the next right is the result. Numbers below the model are numbered model which randomized by the committee. We can see that on each query, we will get results with similar in shape and texture.

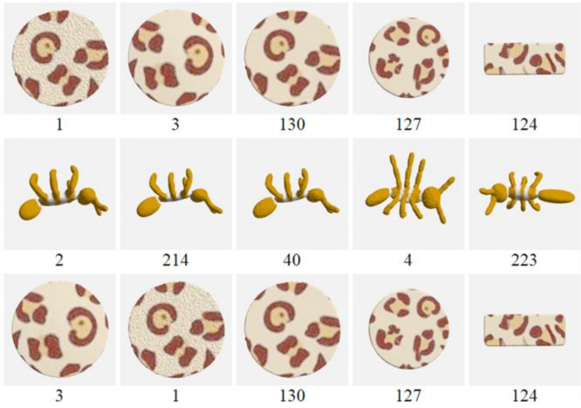

Figure 8. The screenshot of our 3D texture model retrieval system with query a horse on left side and answers on the right side.

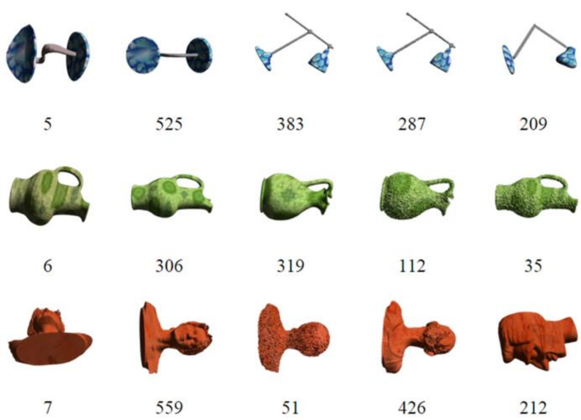

Figure 9. The screenshot of our 3D texture retrieval system with a vase as a query on left side and its answers on right side.

VI. CONCLUSION AND FUTURE WORK

3D model retrieval based on shape and texture is still an interesting research problem. Though there are some existing methods, but we still have a chance to improve the retrieval performance. In this paper, we propose a new approachment by combining texture features and shape features. For texture features, we build a color histogram based on both color image rendered and color voxel. For shape features, we use our previous methods called MFSD. Assessing performance, we use a standard dataset for 3D texture published in SHREC 2013 and 2014. Even though our approachment can not outperform in all evaluations, but our method is the best in the FT and ST evaluation. Finally, the opportunity improving performance in this area is still open in the future because it plays increasingly important roles in practical applications.

ACKNOWLEDGMENT

The authors would like to thank DIKTI (Indonesian Government for Higher Education) for the scholarship.

REFERENCES

[1] A. Cerri, S. Biasotti, *et al.*, "SHREC'13 track: Retrieval on textured 3D models," in *Proc. Sixth Eurographics Workshop on 3D Object Retrieval*, 2013, pp. 73-80.

[2] S. Biasotti, A. Cerri, *et al.*, "SHREC'14 Track: Retrieval and classification on textured 3D models," in *Proc. Seventh Eurographics Workshop on 3D Object Retrieval (3DOR 2014)*, Strasbourg, 2014, pp. 1-11.

[3] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," in *Computer Graphics Forum*, Blackwell Publishing Ltd, 2009, pp. 1383-1392.

[4] M. Reuter, *et al.*, "Laplace-Beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis," *Computer-Aided Design*, vol. 41, no. 10, pp. 739-755, 2009.

[5] A. M. Bronstein, *et al.*, "Shape google: Geometric words and expressions for invariant shape retrieval," *ACM Trans. on Graphics (TOG)*, vol. 30, no. 1, 2011.

[6] A. Giachetti and C. Lovato, "Radial symmetry detection and shape characterization with the multiscale area projection transform," in *Computer Graphics Forum*, Blackwell Publishing Ltd, 2012, pp. 1669-1678.

[7] C. Y. Li and A. B. Hamza, "Intrinsic spatial pyramid matching for deformable 3D shape retrieval," *International Journal of Multimedia Information Retrieval*, vol. 2, no. 4, pp. 261-271, 2013.

[8] C. Y. Li and A. B. Hamza, "A multiresolution descriptor for deformable 3D shape retrieval," *The Visual Computer*, vol. 29, no. 6-8, pp. 513-524, 2013.

[9] C. Y. Li and A. B. Hamza, "Spatially aggregating spectral descriptors for nonrigid 3D shape retrieval: A comparative survey," *Multimedia Systems*, vol. 20, no. 3, pp. 253-281, 2014.

[10] V. Surazhsky, *et al.*, "Fast exact and approximate geodesics on meshes," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, 2005.

[11] D. Smeets, *et al.*, "Isometric deformation modelling for object recognition," in *Computer Analysis of Images and Patterns*, Springer Berlin Heidelberg, 2009, pp. 757-765.

[12] S. Biasotti, *et al.*, "PHOG: Photometric and geometric functions for textured shape retrieval," in *Computer Graphics Forum*, Blackwell Publishing Ltd, 2013.

[13] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proc. Symposium on Geometry Processing*, 2003, pp. 156-164.

[14] M. Chaouch and A. Verroust-Blondet, "Alignment of 3D models," *Graphical Models*, vol. 71, no. 2, pp. 63-76, 2009.

[15] D. V. Vranic, D. Saupe, and J. Richter, "Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics," in *Proc. IEEE Fourth Workshop on Multimedia Signal Processing*, 2001, pp. 293-298.

[16] A. Tatsuma and M. Aono, "Multi-Fourier spectra descriptor and augmentation with spectral clustering for 3D shape retrieval," *The Visual Computer*, vol. 25, no. 8, pp. 785-804, 2009.

Hero Yudo Martono received Bachelor's degree in Electrical Department from Institut Teknologi Sepuluh Nopember and Master's degree in Information Science from Institut Teknologi Bandung, both of them are in Indonesia. Currently he is a Ph.D. student in Information and Computer Science Department of Toyohashi University of Technology. His current research interests include 3D retrieval system on shape and texture.

Masaki Aono received Bachelor's and Master's degrees in Information Science from University of Tokyo and a Ph.D. in Computer Science from Rensselaer Polytechnic Institute. She worked for IBM Research, Tokyo Research Laboratory from 1984 to 2003. He is currently a Professor in the Information and Computer Science Department of Toyohashi University of Technology, where he is teaching object-oriented programming computer architecture, and knowledge data engineering. His current research interests include text and data mining, information extraction, and information retrieval for multimedia. He has been a Japanese delegate of the ISO/IEC JTCI SC24 Standard Committee since 1996.