A Large-Scale Shape Benchmark for 3D Object Retrieval: Toyohashi Shape Benchmark

Atsushi Tatsuma*, Hitoshi Koyanagi†, Masaki Aono‡
* Toyohashi University of Technology, Aichi, Japan
E-mail: atsushi@kde.cs.tut.ac.jp
† Toyohashi University of Technology, Aichi, Japan
E-mail: koyanagi@kde.cs.tut.ac.jp
‡ Toyohashi University of Technology, Aichi, Japan
E-mail: aono@tut.jp Tel/Fax: +81-532-44-6764/+81-532-44-6757

***Abstract*—In this paper, we describe the Toyohashi Shape Benchmark (TSB), a publicly available new database of polygonal models collected from the World Wide Web, consisting of 10,000 models, as the largest 3D shape models to our knowledge used for benchmark testing. TSB includes 352 categories with labels. It can be used for both 3D shape retrieval and 3D shape classification.**

Until now, the most well-known 3D shape benchmark has been the PSB, or the Princeton Shape Benchmark, consisting of 1,814 models, including the half as training data and the remaining half as testing. The TSB is approximately 6 times larger than the PSB. Unlike textual data such as TREC and NTCIR data collections, 3D shape repositories have been suffering from the shortage of data, and from the difficulty in testing the scalability of any algorithms that work on top of given benchmark data set.

In addition to the TSB, we propose a new shape descriptor which we call DB-VLAT (Depth-Buffered Vector of Locally Aggregated Tensors). During the comparison with the TSB, we will demonstrate that our new shape descriptor exhibits the best search performance among those known programs to which we have had access on the Internet, including the Spherical Harmonic Descriptor and Light-Field Descriptor.

We consider that the TSB can be a step toward the next generation 3D shape benchmark having massive 3D data collection, and hope it will serve for many purposes in both academia and industry.

I. Introduction

Recently, with the technological advancement of general-purpose computers, more and more 3D objects have been used in many fields such as manufacturing, entertainment, and medical simulation. The 3D models are thus explosively increasing, which makes it absolutely indispensable to manage and re-use those 3D models efficiently. This tendency has triggered the need to search more effectively for similar 3D objects [1], [2]. If we can retrieve a 3D object that perfectly meets our need from massive 3D databases and integrate the retrieved object into a composite and complex 3D object, it is expected that we can greatly cut the expense of creating a new 3D object from scratch [3].

The similarity search process of 3D shape objects is analogous to the content-based retrieval of documents and images. Fig. 1 is an example illustrating the 3D shape retrieval process in general. When a query 3D shape is given from a user, the system attempts to extract the *shape descriptor* that grasps the shape features such as intrinsic geometry and topology of the 3D shape object. Then, the system computes the dissimilarity between the query object and those stored in the system database. Finally, the system sorts in ascending order the retrieved results based on the dissimilarity and returns the ranked list to the user. The key to good 3D shape search system depends upon finding good shape descriptors that faithfully represent inherent "shapes."

To evaluate shape descriptors, good benchmark data are essential. Several 3D shape benchmark data have been proposed so far. The most popular among those proposed is the Princeton Shape Benchmark (PSB) [4], consisting of 1,814 models, with 907 training sets and the same number of test sets. In particular, the PSB test set has been most frequently used to evaluate search performance of 3D shape objects. It contains 907 objects with 92 classes including *doors*, *biplanes*, and *flowers*. PSB covers a wide range of different objects in a balanced way, and it thus gives us basic search performance in a variety of objects. There are other special-purpose 3D shape benchmarks, such as the Engineering Shape Benchmark [5] and the 3D Architecture Shape Benchmark [6], which are summarized in the next section.

Most of the previously known 3D shape benchmark data consist of less than 1,000 objects. Recently, the Large Scale Retrieval Track [7] of SHape REtrieval Contest (SHREC) is available in 10,000 objects. However, it has been revealed that meaningful objects are only 493, and the remaining objects are random generated content. Bronstein et al [8] refers to the part of SHREC10 data [9] as "large-scale shape retrieval benchmark," even though they used 1,184 objects from SHREC10.

In this paper, we describe the Toyohashi Shape Benchmark (TSB), consisting of 10,000 fully meaningful massive 3D objects. We have classified TSB into 352 categories manually, in the hope that it will be used as one of the truly large-scale 3D shape benchmarks. We also propose our new shape descriptor called DB-VLAT. Using TSB, we compare several different shape descriptors including Spherical Harmonic Descriptor and Light-Field Descriptor.

In the remainder of the paper, we first review previous 3D benchmarks in the section of Related Work. Then, we describe the Toyohashi Shape Benchmark (TSB) in section

III. In section IV, we describe our unique shape descriptor *DB-VLAT*. Section V describes experiments and evaluations using TSB, followed by concluding the paper in the last section.

II. Related Work

We have seen that several 3D shape benchmark data have been proposed and been available for almost a decade ever since the Princeton Shape Benchmark was published [4] in 2004. PSB contains 1,814 3D shape models in OFF format [10], where it is divided into 907 training and 907 test data sets, respectively.

The test set of the two has been primarily employed for evaluating search performance of various algorithms. PSB test set consists of 92 classes including *helicopter*, *horse*, and *hourglass* classes. The 3D shape objects in PSB has a wide variety of objects, which makes PSB attractive for evaluating search performance in general.

NIST Shape Benchmark (NSB) [11] is another 3D shape benchmark which has a variety of objects like PSB. NSB has 800 shape objects and is divided into 40 classes, and each class has 20 shape objects; NSB is thus designed to have no bias from class to class. There are other 3D databases which have less than PSB such as MPEG-7 database [12] and proprietary database *Digimation model bank* [13].

Engineering Shape Benchmark (ESB) [5] is a 3D shape benchmark primarily for mechanical parts. ESB consists of 801 3D shape objects with 42 classes. ESB is suited for evaluating 3D CAD model search.

The McGill 3D Shape Benchmark (MSB) [14] is for articulated objects (255 objects with 10 classes). MSB is suitable for evaluating robustness of articulated objects such as *humans*, *teddy bears* and *insects*.

3D Architecture Shape Benchmark (ASB) [6] has 2,257 architectural objects with 180 classes. ASB is suitable for evaluating architectural objects to simulate *houses*, *buildings*, and *furniture* including *chairs* and *windows*. It should be noted that Trimble's SketchUp Web site [15] has a large amount of architectural objects, but is not a benchmark for evaluation.

SHape REtrieval Contest (SHREC), well-known for 3D shape similarity search contest Web site [16], contains many specialized 3D shape objects including Generic Models, CAD Models, and 3D Face Models.

The Toyohashi Shape Benchmark (TSB) proposed in this paper is a newly constructed benchmark, having 10,000 fully meaningful 3D objects with 352 classes.

III. Toyohashi Shape Benchmark

Most of the previous 3D shape benchmarks have a total of less than 1,000 3D data. For small-scale comparison in terms of search performance, a less than 1,000 data might be beneficial. However, a small-scale data set permits those algorithms which are neither scalable nor efficient. Researchers doing 3D shape search have been awaiting a larger database to test their algorithms and greater feasibility of their way to maintain features (or descriptors), in terms of scalability and efficiency. This motivates us to develop a ten times larger 3D shape benchmark for research and development of the 3D shape search system.

The TSB consists of 10,000 3D shapes with 352 categories. The task of creating a 3D shape benchmark entails two sub-tasks; collecting 3D models and adding labels to classify them into categories. For the first sub-task, we employ publicly available data, which have not yet been labeled. Specifically, a part of our 3D shape models come from the NTU 3D Model Database (NTU) [17], which consists of 10,910 unclassified and unlabeled models. NTU keeps these data obtained from the Web and make them publicly available. The problem is that most of them are extremely difficult for humans to classify and label. Another part of our models come from 3,168 models of SHREC'10 Track: Generic 3D Warehouse [9]. These two parts total 14,078 unclassified and unlabeled models. We adopt Object File Format (OFF) [10] with no surface normals and textures, in line with the PSB.

For the sub-task of classifying 3D shape models, we first employed Passive Aggressive Algorithm [18] for tentative classification.

In defining the 3D shape descriptors, we employed our developed method *Dense Voxel Spectrum Descriptor* [19], while for the training data set, we used Princeton Shape Benchmark Training Sets [4]. Under these circumstances, we manufactured a special-purpose Web tool as illustrated in Fig. 2, and our laboratory members joined to work with this tool for manually classifying the 3D shape data collected. With our Web tool, it is possible to intuitively preview and confirm an arbitrary 3D shape model from our benchmark, based on WebGL [20] 3D viewer with the thumbnail images. We repeated the process of the category construction by carefully examining its validity, and ended up with 352 categories for 10,000 3D shape models. It took three months to complete the task.

Table I lists all the basic classes (or categories) we made. In TSB, we further subdivide basic classes into detailed sub-classes. For instance, *animal* classes have *dog*, *horse*, and *pig* sub-classes. As a whole, compared to PSB, the TSB has more classes and more detailed sub-classes, which makes it special in terms of the degree of difficulty to obtain good performance for both search and classification.

IV. DB-VLAT 3D Shape Descriptor

Although our major objective in to the paper is to provide a publicly available massive 3D shape benchmark for search and classification, we also propose a new 3D shape descriptor which we call DB-VLAT (Depth-Buffered Vector of Locally Aggregated Tensors).

DB-VLAT is basically categorized into a 3D shape feature composed of Bag-of-Visual Words (BoVW) [21], [22]. BoVW has been used in pattern recognition [23], [24]. To compute BoVW, we first extract local features from the collection of images, and then apply clustering such as k-Means to the local features to obtain the center of clusters (or *Visual Words*). Finally, we assign local features from each image to the nearest *Visual Words*, construct a histogram by taking *Visual Words*

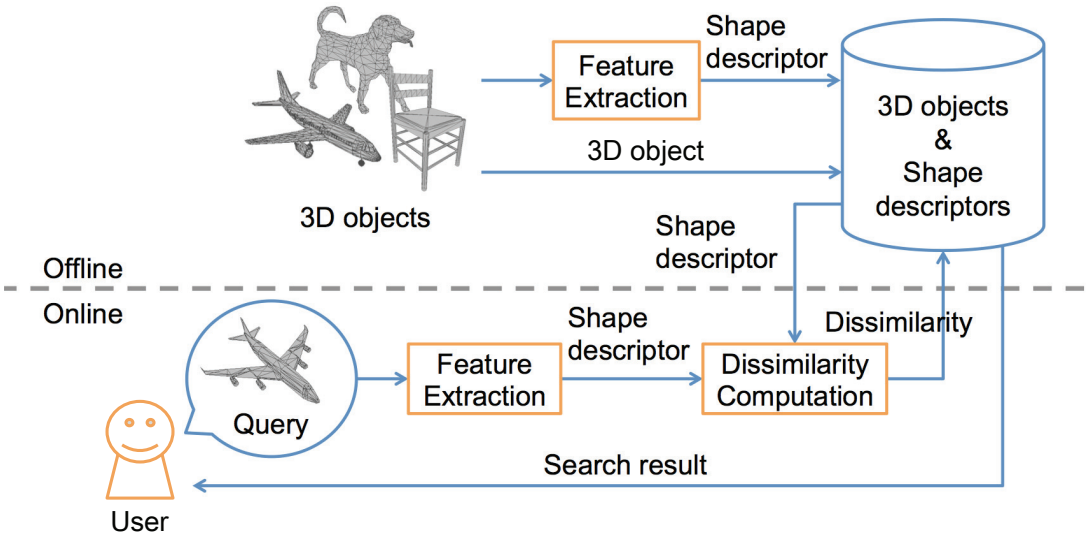


Fig. 1. Diagram of a 3D object retrieval system.

as a horizontal axis and the frequency of local features as vertical axis, and make the histogram as the feature vector of the image.

BoVW is regarded as a statistical feature based on the distribution of local features. In addition to the histogram of frequency from BoVW, it is known that, in the field of pattern recognition, the methods that consider high order statistics, such as means and distributions, have higher recognition accuracy [25], [26].

From these observations, we propose DB-VLAT, which consists of local features of Vector of Locally Aggregated Tensors (VLAT) [26] from depth-buffered images through rendering against 3D shape objects.

Fig. 3 illustrates, step-by-step, how our proposed DB-VLAT's feature vector is generated. This will be elaborated in the following. In DB-VLAT, we first apply pose normalization. 3D shape objects are usually defined by different authors with distinct authoring tools, which makes the position, size, and orientation of 3D shape objects quite different from each other. To resolve this problem, Tatsuma et al [27] invented *PointSVD* that aligns the centroid and principal axes by generating random points on the surface of 3D shape objects, and *NormalSVD* that aligns the surface normals with respect to principal axes. In DB-VLAT, we adopt the combination of *PointSVD* and *NormalSVD* for pose normalization.

Once pose normalization is done, we enclose the 3D shape object with a regular octahedron. From each vertex of the octahedron as well as from the midpoint of each edge, we perform depth-buffer image rendering with 256×256 resolution. Note that in a total of 18 viewpoints are defined. We extract a DIFT (Dense SIFT) [28], [29], [30] as local features from each depth-buffer image. DIFT is a feature vector similar to SIFT descriptor [31] extracted from the patch with an equal interval and size. In DB-VLAT, a patch has 4 pixel intervals of 76×76 size.

We extract DIFT from each 3D shape object in the training data set, and generate visual words from the collection of SIFT descriptors. The visual words in DB-VLAT is thus defined as the center of a cluster obtained by k-Means clustering. We generated a codebook of $k = 32$ visual words.

We calculate DB-VLAT with a codebook of k visual words $V = [\mathbf{v}_1, \ldots, \mathbf{v}_k] \in \mathbb{R}^D$. First, we assign local features $\mathbf{x} \in \mathbb{R}^D$ to the nearest visual word $\mathbf{v}_i$, compute the matrix $S^{(i)}$ by using visual word $\mathbf{v}_i$ and assign local features to $\mathcal{N}(\mathbf{v}_i)$.

$$S^{(i)} = \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{v}_i)} (\mathbf{x} - \mathbf{v}_i)(\mathbf{x} - \mathbf{v}_i)^T$$

The vector $\mathbf{s}_i$ consists of the components in the upper triangular part of a symmetric matrix $S^{(i)}$.

$$\begin{aligned} \mathbf{s}_i &= \text{Upper}(S^{(i)}) \\ &= [S^{(i)}_{1,1}, \ldots, S^{(i)}_{1,D}, S^{(i)}_{2,2}, \ldots, S^{(i)}_{2,D}, \ldots, S^{(i)}_{D,D}] \end{aligned}$$

Next, we generate the vector $\mathbf{f}$, consisting of vector $\mathbf{s}_i$ calculated from each visual word $\mathbf{v}_i$.

$$\mathbf{f} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k]$$

Since the vector $\mathbf{f}$ is a sparse vector, we avoid using Euclidean distance. Instead we employ the power normalization to reduce the sparseness of a vector [25].

$$\bar{\mathbf{f}} = \text{sign}(\mathbf{f})|\mathbf{f}|^\alpha,$$

where $\text{sign}(\cdot)$ denotes the sign function, and $|\cdot|$ denotes absolute value. α is a parameter of arbitrary real values and is set to 0.5 according to Perronnin et al [25].

Finally, we normalize the vector $\mathbf{f}$ with the L2-norm to obtain our proposed DB-VLAT shape descriptor.

$$\text{DB-VLAT} = \bar{\mathbf{f}}/||\bar{\mathbf{f}}||$$

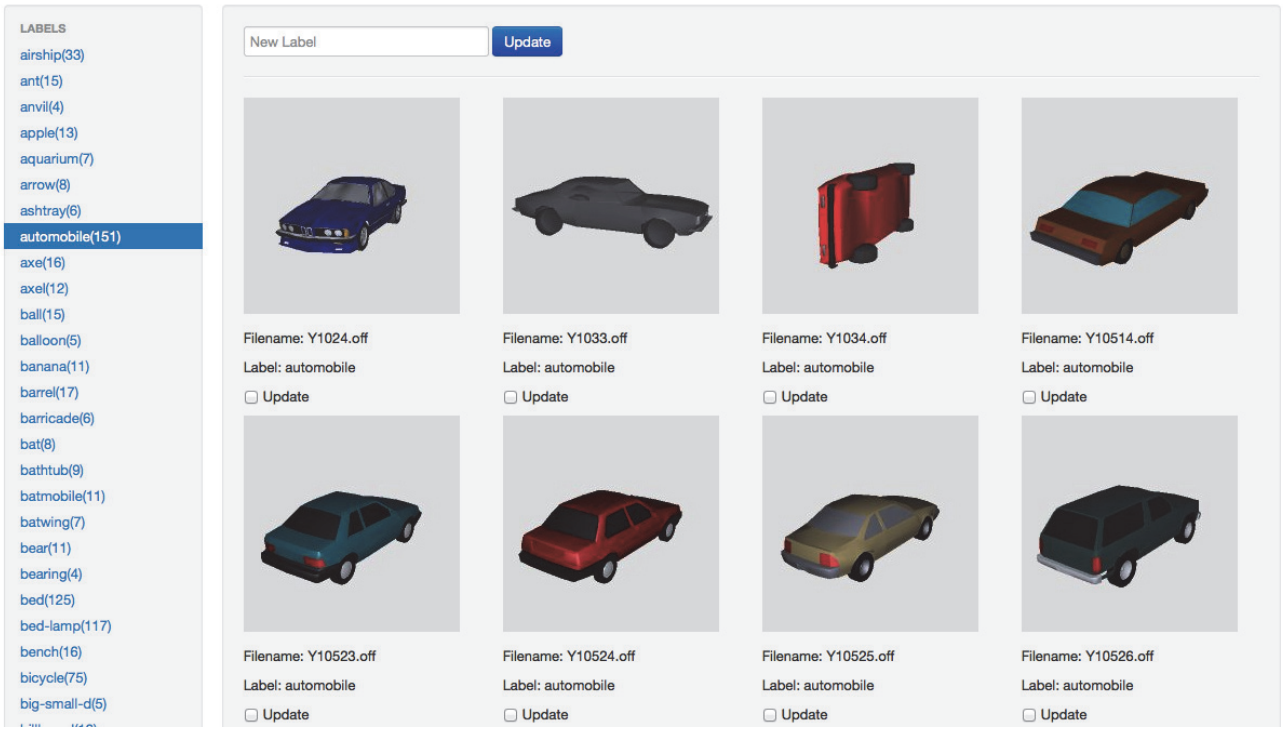


Fig. 2. Tools for classification of 3D object dataset.

To compare two DB-VLAT descriptors, we simply compute the Euclidean distance between them.

V. Experiment and Evaluation

We have carried out experiments to compare search performance of representative 3D search methods with shape descriptors using the TSB.

A. Shape Descriptors

For the 3D shape similarity search, extracting shape descriptors accurately from a given 3D shape object plays an important role.

The difficulty of extracting a shape descriptor lies in the fact that most 3D shape models available on the Internet are made up of a collection of disconnected polygons (called "polygon soup") in 3D space. Examples of file formats where polygon soup can be defined include OFF [10], VRML [32], [33], OBJ [34], COLLADA [35], and X3D [36]. It is thus usually very difficult, if not impossible, to extract a volumetric or topological descriptor from a given 3D object. Most of the feature extraction algorithms known to date have dealt with polygon-soup based models by transforming data into voxel or silhouette representation in advance.

Shape descriptors are classified into several groups, based on how shape representation is defined, given an arbitrary 3D shape. In our comparative experiments to follow, we chose shape descriptors based on point sets, voxels, silhouette images, and depth-buffer images. We also compared them with our proposed shape descriptor DB-VLAT.

1) D2 Shape Distribution (D2): Shape Descriptor D2 proposed by Osada et al [37] generates random point sets on the surface of 3D shape objects, collects the distance and frequency between the two points arbitrarily selected, and make them into a histogram. This histogram serves as the D2 Shape Descriptor.

2) Surflet-Pair Relation Histograms (SPRH): Wahl et al [38] proposed Surflet-Pair Relation Histograms (SPRH), which are a four-dimensional histogram shape descriptor accounting for the geometric information represented by not only the distance between two randomly selected points, but the orientation of the two points with respect to the normal vector to the surface to which they belong.

3) Spherical Harmonic Descriptor (SHD): Kazhdan et al [39] proposed Spherical Harmonic Descriptor (SHD), which is a voxel shape descriptor, a rotation invariant representation of spherical harmonic functions in terms of the energies at different frequencies.

4) Dense Voxel Spectrum Descriptor (DVD): Dense Voxel spectrum Descriptor (DVD) [19] is a shape descriptor accounting for the Fourier spectrum, based on the voxels and their block representation, defined by regularly spaced intervals and overlapping cells of blocks.

5) Light-Field Descriptor (LFD): Chen et al [17] proposed Light Field Descriptor (LFD), a shape descriptor composed

TABLE I
CLASSIFICATION OF MODELS IN TSB.

Class Name	#Objects	Class Name	#Objects	Class Name	#Objects	Class Name	#Objects	Class Name	#Objects
airship	33	ant	15	anvil	4	apple	13	aquarium	7
arrow	8	ashtray	6	automobile	151	axe	16	axel	12
ball	15	balloon	5	banana	11	barrel	17	barricade	6
bat	8	bathtub	9	batmobile	11	batwing	7	bear	11
bearing	4	bed	125	bed-lamp	117	bench	16	bicycle	75
big-small-d	5	billboard	12	biplane	69	block	7	boat	14
body	16	bone	18	book-set	55	bookend	4	boot	7
bottle	161	bowling	4	boxroom	10	brain	5	bridge	10
bucket	14	bug-feathered	10	bug-humanlike	13	building	35	bus	108
bus-shelter	4	bush	9	bust	18	butterfly	3	cage	4
camel	4	camera	7	can	10	candle	14	cannon	20
car-body	26	carousel	4	casket	4	cassette-tape	3	castle	18
cat-scan	3	cd-case	4	chain	4	character	248	chess	54
chess-set	5	chest	8	chip	29	church	10	cigarette	8
city	34	cleaver	17	clip-board	3	clock	43	closed-laptop	15
closed-piano	75	coffee-maker	6	column	27	commercial-plane	45	compass	12
component	14	computer-body	13	computer-keyboard	114	computer-monitor	12	container-truck	83
convertible	27	couch	40	covered-wagon	6	cow	8	crocodile	4
crypt	4	cube	6	cubes	10	cup	20	darts	10
deer	5	desk	27	desktop	16	dinosaur	35	diskette	14
dog	18	dolphin	22	door	45	door-knob	4	dragon	9
dragonfly	6	drawer	61	drill	7	driver	16	driver-head	9
drum-set	103	ear	3	earth	10	elephant	5	enterprise	80
extinguisher	7	eye-glasses	33	face	33	falcon	7	fan	37
felidae	26	fence	16	fighter-plane	234	fireplace	7	fish	143
flag	16	flip-phone	23	floor-lamp	101	flower	55	flying-balloon	36
flying-bird	42	flying-saucer	19	fork	7	four-legged-chair	186	frame	20
frog	5	game	6	gazebo	6	gear	16	geographic-map	55
glass-with-stem	40	glider	52	grape	5	grave	51	grid	9
griffin	3	guillotine	4	guitar	148	gun	120	hair	7
hammer	18	hand	36	hand-bell	4	hand-drill	4	hand-light	12
hang-glider	16	hat	15	head	116	heart	4	helicopter	98
helmet	30	hemisphere	5	horn	15	horse	78	horseshoe	3
hourglass	13	house	50	human	314	hydrant	18	iceberg	7
inkwell	4	iron	5	jellyfish	3	jewel	3	joystick	15
kangaroo	4	key	7	knife	120	ladder	7	ladybird	4
landscape	33	lathe	4	leaf	18	leg	4	lift-car	10
light	9	light-bulb	17	lighter	4	lighthouse	8	lock	5
long-chair	4	magnifier	6	mailbox	7	maze	20	meteor	10
microphone	10	microscope	7	mixer	4	modern-chair	13	modern-folly	4
molecule	10	monkey	4	motorbike	142	mushroom	9	nocontainer-truck	135
noleg-chair	17	normal-phone	120	nunchuck	3	one-legged-chair	21	opened-book	32
opened-laptop	181	opened-piano	90	organ	12	pear	11	pedal	3
piano-board	27	pig	3	pincerlike	8	pipe	5	planter	3
pod-racer	8	pole	64	pool	18	pool-table	12	pot	28
potted-plant	84	power-pole	5	printer	9	propeller-plane	257	pump	6
pumpkin	5	pushpin	3	pyramid	12	race-car	35	rack	4
rail	6	rake	4	range	8	recognizer	7	recorder	14
refrigerator	12	remote	7	ring	9	ring-toy	5	robot-animallike	11
robot-arm	10	robot-fourleg	7	robot-humanlike	35	robot-joint	30	robot-noleg	14
robot-pod	9	robot-twoleg	26	rocket	185	rocking-chair	5	rocking-horse	3
room	40	saddle	5	satellite	46	satellite-dish	10	scanner	4
scissors	3	scorpion	4	seashell	11	shade	10	shark	20
shelf	126	shield	15	ship	60	shoe	13	shovel	13
sign	24	single-book	79	single-drum	88	sink	9	skateboard	5
skull	15	slide	5	slider-phone	30	slot-machine	8	smoke	7
snake	5	spacefighter	4	spacepod	7	spaceship	43	spaceshuttle	23
spacestation	21	speaker	8	spear	7	sphere	21	spider	12
spoon	55	spray	4	spring	8	stadium	9	stake	4
standing-bird	76	stapler	3	stationery	31	stealth	47	steps	16
stool	12	store	6	stove	4	study-lamp	106	stuffed-specimen	7
submarine	32	suitcase	10	supercar	8	swing	7	switch	4
sword	166	syringe	5	table	115	tank	53	tape-deck	4
telephone	37	temple	3	three-wheeler	11	tiefighter	49	toilet	8
tooth	4	torch	3	tower	4	traffic-light	7	train	143
trash	11	tree	120	triangle	4	turntable	5	turtle	7
tv	16	typography	31	umbrella	8	unicycle	4	unicycle-fork	13
vase	55	video-camera	11	video-tape	4	violin	39	violin-case	3
warbird	13	watch	16	water-jug	6	weathercock	4	whale	4
wheel	78	wheeled-chair	97	wind-chime	6	windmill	4	window	18
wrench	10	x-wing	15						

of Fourier spectra and Zernike moment, computed by rendering a large amount of 2D silhouette images from multiple viewpoints. LFD is known to be superior to D2 and SHD from the comparative experiments with PSB [4].

6) DESIRE Descriptor (DESIRE): Bustos et al [40] have noticed that different shape descriptors, such as voxel representation and silhouette edges, have their own strength and weakness. Vranic [41] proposed a DESIRE descriptor to cope with the above situation. Specifically, DESIRE descriptor takes care of Fourier spectra from depth-buffer and silhouette images, and a "ray" vector, emanating from the origin of the silhouette image and traveling in equiangular radial directions.

7) Multi-Fourier Spectra Descriptor (MFSD): Tatsuma et al [27] proposed a Multi-Fourier Spectra Descriptor (MFSD), a composite shape descriptor consisting of 2D Fourier spectra extracted from depth-buffer, silhouette, and contour images, and 3D Fourier spectrum extracted from voxels.

8) DB-VLAT: DB-VLAT is a shape descriptor proposed in this paper. It is classified into Bag-of-Visual Words (BoVW) [21], [22], computed by a collection of depth-buffered images through rendering of a given 3D shape model.

B. Evaluation

Given a benchmark data set, we determine how well can 3D shape matching algorithms work be verified by several evaluation criteria mentioned below.

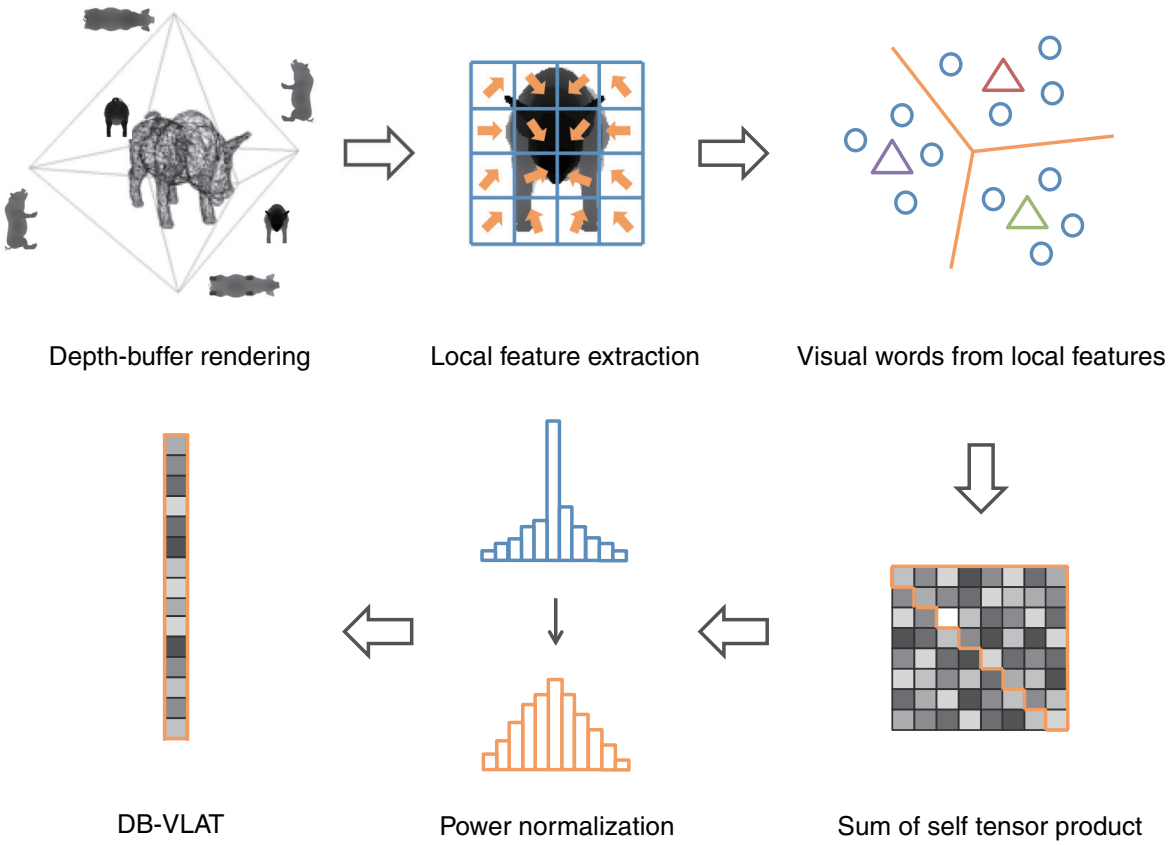


Fig. 3. DB-VLAT extraction process.

1) Precision-Recall Curve: *Recall* is the fraction of relevant 3D objects that are retrieved, and *precision* is the fraction of retrieved 3D objects that are relevant. For each query object in class C and any number of top T ranked list out of R relevant 3D objects, recall and precision can be represented by the following formulae:

$$Precision = R/T$$

$$Recall = R/C$$

Usually, a precision-recall curve is plotted by taking recall as the horizontal axis, and precision as the vertical axis. This curve represents a trade-off between recall and precision. When recall is smaller, it often happens that precision is larger, whereas when recall is larger, precision tends to be smaller. By plotting two or more precision-recall curves that correspond to the search algorithms with the associated shape descriptors, it is possible to compare the effectiveness of the search algorithms intuitively.

2) First Tier (FT) and Second Tier (ST): *First Tier* is a precision of top $T = C$ retrieved objects, and T becomes $C - 1$ if a query itself is included in the ranked result. *First Tier* represents how much similar 3D objects appear in the top ranked list. *Second Tier* is the precision of top $T = 2C$ retrieved objects. Analogous to *First Tier*, if a query is included in the ranked list, T becomes $2*(C-1)$.

3) Nearest Neighbor (NN): *Nearest Neighbor (NN)* in the search evaluation represents a measure if the top in the ranked list is relevant or not. In other words, NN is the percentage of the closest matches that belong to the same class as the query with $T = 1$.

4) F-Measure (F): *F-Measure* is a harmonic mean of precision and recall. For 3D shape search, F-measure has been historically taken for the top $T = 32$ search results [4].

$$F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

5) Discounted Cumulative Gain (DCG): Discounted Cumulative Gain (DCG) represents a measure that accounts for how many relevant objects are found around the top T ranked list. DCG is defined recursively as follows:

$$DCG(T) = \begin{cases} G_1, & T = 1 \\ DCG(T-1) + \frac{G_T}{log_2(T)}, & otherwise \end{cases}$$

where G_T is 1 if the T-th objects in the retrieved list belongs to the same class as the query, and is 0 otherwise. If the total

number of 3D objects in the benchmark is denoted by N, the DCG is defined as follows:

$$DCG = \frac{DCG(N)}{1 + \sum_{j=2}^{C} \frac{1}{log_2(j)}}$$

C. Results

To investigate the effectiveness, we have conducted experiments using our proposed massive 3D shape database called Toyohashi Shape Benchmark (TSB).

Fig. 4 shows the micro-average of Precision-Recall curve for all the 3D shape objects in the TSB. DB-VLAT exhibits the largest precision over all the recall. On the average, it is confirmed that DB-VLAT outperforms the other methods in terms of both precision and recall. The overall results with respect to search performance are similar to those when using Princeton Shape Benchmark (PSB), in the sense that relative search performance among the selected previously known methods has a similar tendency. Specifically, composite shape descriptors such as MFSD and DESIRE generally have good search results on average. However, since TSB is almost ten times larger than PSB, the search performance as a whole becomes lower than PSB. This tendency is easily guessed because TSB characteristically has a wide variety of objects as listed in Table I.

Table II summarizes the micro-average of *First Tier (FT)*, *Second Tier (ST)*, *Nearest Neighbor (NN)*, *Discounted Cumulative Gain (DCG)*, and *F-measure* with all the 3D shape data in TSB. The bold numeric values show the highest performance. Among all the evaluation criteria, DB-VLAT depicts the highest value, and the table thus demonstrates that search performance of DB-VLAT is superior to the previous methods.

Most of the evaluation results are 5 to 10 percent lower than those with PSB [4], which makes TSB special in the sense that TSB provides a not only larger but severer benchmark for any shape matching algorithm to get a good search result. We also note that TSB can be used for the purpose of supervised learning such as classification.

We have also investigated class-by-class evaluations for the selected shape matching algorithms. Fig. 5 demonstrates Precision-Recall curves for six typical classes chosen from TSB. DB-VLAT indicates high search performance on average in all classes. DB-VLAT shows high search accuracy not only in simple shapes like “bus” but in complex shapes like “trees.” We conjecture that DB-VLAT is effective even for complex shapes because it employs higher order statistics based on local features.

Here we give dtailed consideration by picking up selected classes and their search performance. For peculiar 3D shape classes such as “bicycle” in Fig. 5 (a), most of the shape matching algorithms exhibit high search performance, while for latently ambiguous shape classes such as “couch” class in Fig. 5 (b), most algorithms end up with low search performance. The “bus” class in Fig. 5 (c) is another example of favorable results with most of the algorithms, including

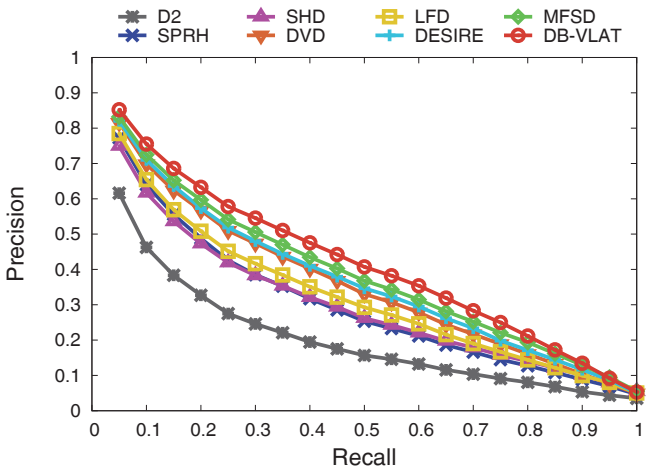


Fig. 4. Precision-Recall curve performance comparison of shape descriptors with TSB.

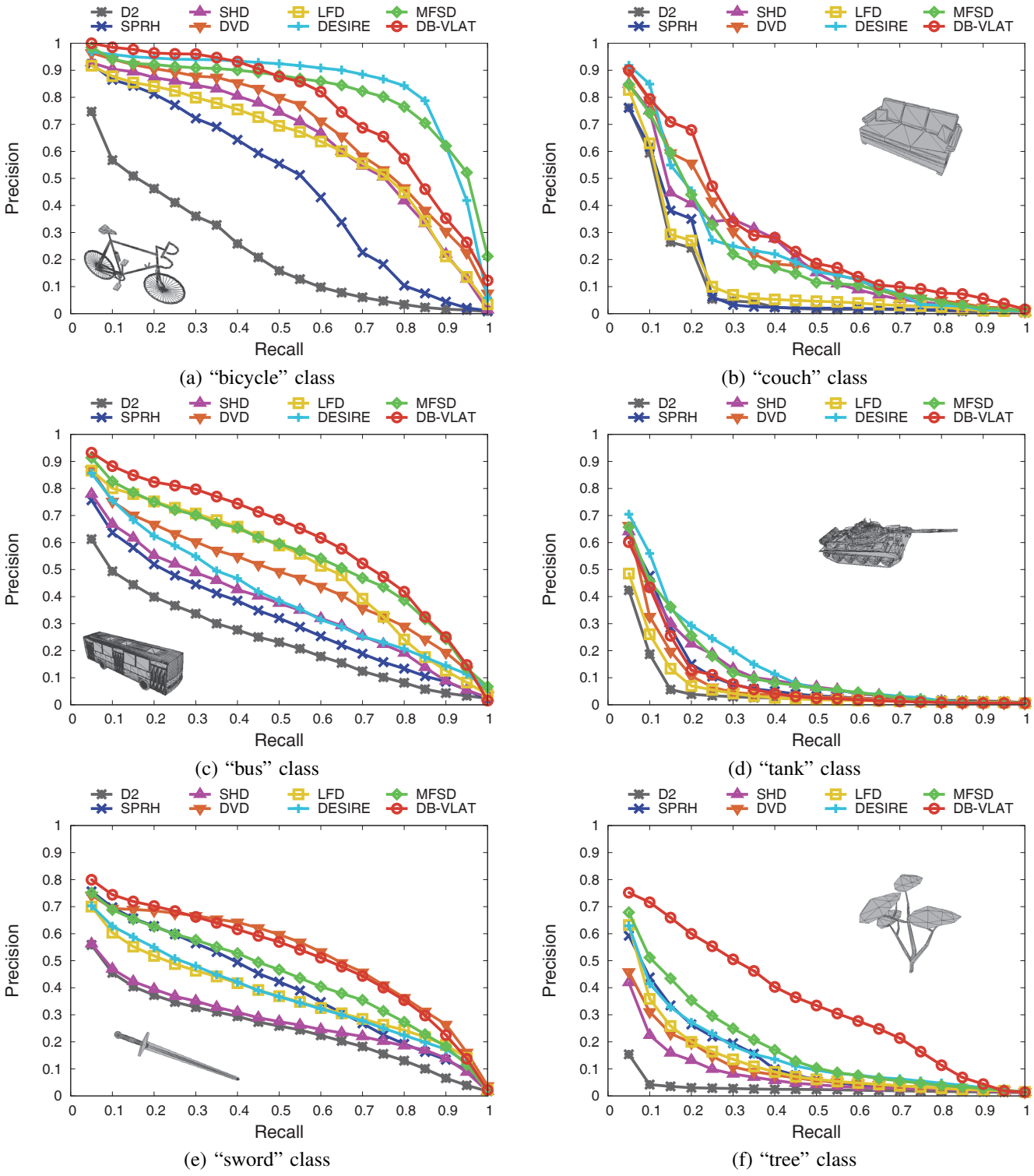
TABLE II
COMPARING SHAPE DESCRIPTORS USING TSB IN TERMS OF FT, AT, NN, DCG, AND F.

Method	FT	ST	NN	DCG	F
D2	0.1908	0.2491	0.6197	0.5593	0.1162
SPRH	0.2812	0.3539	0.7528	0.6376	0.1757
SHD	0.2833	0.3613	0.7477	0.6424	0.1816
DVD	0.3328	0.4163	0.7992	0.6822	0.2129
LFD	0.3035	0.3782	0.7670	0.6543	0.1964
DESIRE	0.3419	0.4287	0.7989	0.6872	0.2210
MFSD	0.3607	0.4508	0.8020	0.6989	0.2277
DB-VLAT	**0.3869**	**0.4723**	**0.8326**	**0.7135**	**0.2409**

DB-VLAT, LFD and MFSD in particular. DB-VLAT, LFD and MFSD have a common feature of shape descriptors originating from images such as silhouette and depth-buffer through 3D rendering. For the vehicle-like objects, shape descriptors of visual outlooks behave well. On the other hand, a special vehicle-like “tank” class in Fig. 5 (d), things are quite different. One obvious reason might be that the tank has an oblong cannon, which makes things more complicated than they appear to be. In “sword” class in Fig. 5 (e), DVD and DB-VLAT have relatively high search performance. This is partly because objects like sword are thin and elongated, and they might be grasped by voxels (DVD) and local features (DB-VLAT), more easily than image-based shape descriptors. Finally, in “tree” class in Fig. 5 (f), most of the shape matching algorithms suffer from bad search results except DB-VLAT. This is in part because there are a many classes such as the “potted-plant” class similar to trees.

VI. CONCLUSION

In this paper, we have proposed a massive 3D shape benchmark, Toyohashi Shape Benchmark (TSB), a publicly available framework for comparing shape matching algorithms. TSB includes 10,000 3D shape objects, 352 categorized classes. TSB can be accessed on the Web (http://www.kde.cs.tut.ac.jp/benchmark/tsb/). The TSB makes



(a) "bicycle" class

(b) "couch" class

(c) "bus" class

(d) "tank" class

(e) "sword" class

(f) "tree" class

Fig. 5. Precision-Recall curves for each class in TSB: (a)bicycle; (b)couch; (c)bus; (d)tank; (e)sword; (f)tree.

it possible to conduct research on scalable search and classification, which may not be possible with the ordinary approaches such as one that maintains all the information in $N \times N$ adjacent matrices of distances and similarity scores.

We also proposed a new shape descriptor called DB-VLAT (Depth-Buffered Vector of Locally Aggregated Tensors). During the comparison using the TSB, we demonstrated that DB-VLAT exhibited the best search performance among those known programs to which we have had access on the Internet.

It is well-known that there are benchmark datasets for documents and images, which number well beyond 100,000. In future, we plan to increase 3D objects much more than that of TSB, which might serve as really big data for 3D shape objects.

ACKNOWLEDGMENTS

This work was partially supported by the Ministry of Education, Culture, Sports, Science and Technology, for JSPS Fellows. The research was conducted by the Strategic Information and Communication R&D Promotion Programme (SCOPE 112306001) of Ministry of Internal Affairs and Communications (MIC) Japan.

REFERENCES

[1] N. Iyer, S. Jayanti, K. Lou, Y. Kalyanaraman, and K. Ramani, "Three-dimensional shape searching: state-of-the-art review and future trends," *Computer-Aided Design*, vol. 37, no. 5, pp. 509–530, 2005.

[2] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3d shape retrieval methods," *Multimedia Tools Applications*, vol. 39, no. 3, pp. 441–471, 2008.

[3] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, "Modeling by example," in *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pp. 652–663, ACM, 2004.

[4] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton Shape Benchmark," in *Proc. of Shape Modeling International*, pp. 167–178, 2004.

[5] S. Jayanti, Y. Kalyanaraman, N. Iyer, and K. Ramani, "Developing an engineering shape benchmark for CAD models," *Computer-Aided Design*, vol. 38, no. 9, pp. 939–953, 2006.

[6] R. Wessel, I. Blümel, and R. Klein, "A 3D Shape Benchmark for Retrieval and Automatic Classification of Architectural Data," in *Proc. of Eurographics Workshop on 3D Object Retrieval 2009*, pp. 53–56, Mar. 2009.

[7] R. C. Veltkamp, G.-J. Giezeman, H. Bast, T. Baumbach, T. Furuya, J. Giesen, A. Godil, Z. Lian, R. Ohbuchi, and W. Saleem, "SHREC'10 Track: Large Scale Retrieval," in *Proc. of Eurographics Workshop on 3D Object Retrieval 2010*, pp. 63–69, 2010.

[8] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov, "Shape Google: Geometric Words and Expressions for Invariant Shape Retrieval," *ACM Transations on Graphics*, vol. 30, no. 1, pp. 1–20, 2011.

[9] V. T. Porethi, A. Godil, H. Dutagaci, T. Furuya, Z. Lian, and R. Ohbuchi, "SHREC'10 Track: Generic 3D Warehouse," in *Proc. of Eurographics Workshop on 3D Object Retrieval 2010*, pp. 93–100, 2010.

[10] P. S. Retrieval and A. Group, "Object File Format." http://segeval.cs.princeton.edu/public/off_format.html.

[11] R. Fang, A. Godil, X. Li, and A. Wagan, "A New Shape Benchmark for 3D Object Retrieval," in *Proc. of the 4th International Symposium on Advances in Visual Computing*, ISVC '08, (Berlin, Heidelberg), pp. 381–392, Springer-Verlag, 2008.

[12] T. Zaharia and F. Prêteux, "3d shape-based retrieval within the MPEG-7 framework," in *SPIE Conf. on Nonlinear Image Processing and Pattern Analysis XII*, pp. 133–145, Jan. 2001.

[13] Digimation, "The Digimation Model Bank Library." http://www.digimation.com/home/Modelbank.aspx.

[14] K. Siddiqi, J. Zhang, D. Macrini, A. Shokoufandeh, S. Bouix, and S. Dickinson, "Retrieving articulated 3-d models using medial surfaces," *Machine Vision and Applications*, vol. 19, pp. 261–275, May 2008.

[15] Trimble, "SketchUp." http://sketchup.google.com/.

[16] AIMATSHAPE.NET, "SHREC Home Page." http://www.aimatshape.net/event/SHREC.

[17] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On Visual Similarity Based 3D Model Retrieval," *Computer Graphics Forum*, vol. 22, no. 3, pp. 223–232, 2003.

[18] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, " Online Passive-Aggressive Algorithms," *Journal of Machine Learning Research*, vol. 7, pp. 551–585, 2006.

[19] B. Li, A. Godil, M. Aono, X. Bai, T. Furuya, L. Li, R. López-Sastre, H. Johan, R. Ohbuchi, C. Redondo-Cabrera, A. Tatsuma, T. Yanagimachi, and S. Zhang, "SHREC'12 Track: Generic 3D Shape Retrieval," in *Proc. of Eurographics Workshop on 3D Object Retrieval 2012*, pp. 119–126, 2012.

[20] K. Group, "WebGL - OpenGL ES 2.0 for the Web." http://www.khronos.org/webgl/.

[21] T. Furuya and R. Ohbuchi, "Dense sampling and fast encoding for 3d model retrieval using bag-of-visual features," in *Proc. of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pp. 26:1–26:8, ACM, 2009.

[22] Z. Lian, A. Godil, and X. Sun, "Visual Similarity Based 3D Shape Retrieval Using Bag-of-Features," in *Proc. of the 2010 Shape Modeling International Conference*, (Washington, DC, USA), pp. 25–36, IEEE Computer Society, 2010.

[23] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. of Workshop on Statistical Learning in Computer Vision, ECCV*, pp. 1–22, 2004.

[24] J. Sivic and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos," in *Proc. of the Ninth IEEE International Conference on Computer Vision*, vol. 2 of *ICCV '03*, pp. 1470–1477, IEEE Computer Society, 2003.

[25] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. of European Conference on Computer Vision*, pp. 143–156, 2010.

[26] D. Picard and P. H. Gosselin, "Improving image similarity with vectors of locally aggregated tensors," in *Proc. of the 18th IEEE International Conference on Image Processing*, pp. 669–672, 2011.

[27] 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, 2008.

[28] F.-F. Li and P. Perona, "A Bayesian Hierarchical Model for Learning Natural Scene Categories," in *Proc. of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2 of *CVPR '05*, pp. 524–531, IEEE Computer Society, 2005.

[29] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha, "Real-Time Visual Concept Classification," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 665–681, 2010.

[30] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained Linear Coding for Image Classification," in *Proc. of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3360–3367, 2010.

[31] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[32] J. Hartman and J. Wernecke, *The VRML 2.0 Handbook*. Addison Wesley, 1996.

[33] W. C. (ISO/IEC JTC1), "The Virtual Reality Modeling Language, International Standard ISO/IEC 14772-1:1997 ." http://www.web3d.org/x3d/specifications/vrml/ISO-IEC-14772-VRML97/.

[34] K. Rule, *3D Graphics File Format: A Programmer's Reference*. Addison Wesley, 1996.

[35] K. Group, "COLLADA - 3D Asset Exchange Schema." http://www.khronos.org/collada/.

[36] W. C. (ISO/IEC JTC1), "Extensible 3D (X3D) encodings, International Standard, ISO/IEC 19776-1.2:2009 – Part 1: Extensible Markup Language (XML) encoding." http://www.web3d.org/files/specifications/19776-1/V3.2/index.html.

[37] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape Distributions," *ACM Transactions on Graphics*, vol. 21, pp. 807–832, 2002.

[38] E. Wahl, U. Hillenbrand, and G. Hirzinger, "Surflet-Pair-Relation Histograms: A Statistical 3D-Shape Representation for Rapid Classification," in *Proc. of International Conference on 3D Digital Imaging and Modeling*, pp. 474–482, 2003.

[39] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors," in *Proc. of the 2003 Eurographics Symposium on Geometry Processing*, SGP '03, pp. 156–164, 2003.

[40] B. Bustos, D. A. Keim, D. Saupe, T. Schreck, and D. V. Vranic, "Using entropy impurity for improved 3d object similarity search," in *IEEE International Conference on Multimedia and Expo*, pp. 1303–1306, 2004.

[41] D. V. Vranic, "DESIRE: a composite 3D-shape descriptor," in *Proc. of IEEE International Conference on Multimedia and Expo*, pp. 962–965, 2005.