

# Semantics-Driven Approach for Automatic Selection of Best Views of 3D Shapes

Hamid Laga <sup>† 1</sup>

<sup>1</sup>Global Edge Institute  
Tokyo Institute of Technology, Japan

---

## Abstract

We introduce a new framework for the automatic selection of the best views of 3D models. The approach is based on the assumption that models belonging to the same class of shapes share the same salient features that discriminate them from the models of other classes. The main issue is learning these features. We propose a data-driven approach where the best view selection problem is formulated as a classification and feature selection problem; First a 3D model is described with a set of view-based descriptors, each one computed from a different viewpoint. Then a classifier is trained, in a supervised manner, on a collection of 3D models belonging to several shape categories. The classifier learns the set of 2D views that maximize the similarity between shapes of the same class and also the views that discriminate shapes of different classes. Our experiments using the LightField (LFD) descriptors and the Princeton Shape Benchmark demonstrate the performance of the approach and its suitability for classification and online visual browsing of 3D data collections.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling —

---

## 1. Introduction

Recent technological advances made 3D acquisition, modeling, and visualization technologies widely accessible to several domains including Computer-Aided Design (CAD), molecular biology, medicine, digital archiving, and entertainment. This has resulted in large-scale collections of 3D models that are available from different sources. Efficient extraction and reuse of this data requires efficient tools for browsing the large collections. While 3D retrieval tools can be used for this purpose, they are rather well suited for situations where the user is able to formulate a query. In many cases, however, the user would want to browse visually the content of a database or the results of a search query in order to get a broad overview of the available 3D models. In such situations, the 3D models should be displayed in the form of few representative views, called also best or salient 2D views. Each one should carry the information that allow understanding the structure of the shape and distinguishing

it from other shape classes. Manual selection of salient 2D views of 3D models is particularly not feasible for large collections.

The saliency of a 2D view of a 3D object can be defined as a function of some view-dependent shape properties. The best view is then the one that maximizes this function [PPB\*05]. Previous work, such as view entropy [VFSH03] and mesh saliency [LVJ05], assume that the best view of an object is the one that carries the largest amount of information about that object independently of the context in which it is defined. In this paper, we define the best views of a 3D object as the views that allow to discriminate the object from the other objects in the database. The solution we propose is based on the assumption that 3D models belonging to the same class of shapes share the same salient features that discriminate them from other classes of shapes. Finding the best views of a 3D model can then be formulated as a feature selection problem.

The major difference of our approach compared to previous work is that we consider a 3D model within a *context*. The context, as defined in this paper, is the data collec-

---

<sup>†</sup> hamid@img.cs.titech.ac.jp

tion to which the 3D model belongs to. Previous works on best view selection ignore this context; the best views are selected as the ones that maximize the visible information and minimize redundancy. This definition is suitable, as a preprocessing step, for retrieval by reducing the number of features to speedup the matching, and for visualizing a single object. Our formulation is data-dependent, and therefore the best views vary according to the database content and its classification. This definition is particularly suited for visual browsing of 3D data collections where the user would like to distinguish quickly the differences between the models in the database. Other applications include automatic summarization of the contents of a large collection and automatic thumbnail generation.

### 1.1. Related work

There have been extensive work on pose normalization of 3D models. This includes shape alignment with Principal Component Analysis (PCA), principal symmetry axis [PSG\*06], and upright orientation estimation [FCODS08]. Although these approaches do not estimate the best view of a 3D shape, they can be used as a preprocessing step to reduce the search space.

Best view selection approaches can be classified into two main categories [FCODS08]; (1) approaches that minimize visible redundant information such as symmetry [PSG\*06] or similarity [YSY\*06], and (2) approaches that maximize the visibility of interesting contents using metrics like viewpoint entropy [VFSH03], view saliency [LVJ05], or shape distinction [SF07]. In the first category, Yamauchi et al. [YSY\*06] and Denton et al. [DDA\*04] studied the similarity and stability relationship between different 2D views of a 3D model; similar views are clustered together and centers of different clusters form a subset of representative 2D views that minimize redundant visible information. These approaches are based on k-means clustering where the number of salient views is manually set by the user. Ansari et al. [ADV07] proposed an adaptation of the X-mean algorithm where the number of characteristic views is automatically adapted to the complexity of the 3D object. Podolak et al. [PSG\*06] automatically selects good viewpoints that minimize the symmetry seen from the viewpoint.

Approaches in the second category aim at maximizing the amount of geometric features visible from the good viewpoints. The main stream is as follows;

- A set of features are extracted from the 3D model,
- An importance value is assigned to each feature,
- The goodness of each view is defined as a function of the importance of the features that are visible from a given viewpoint, and
- The set of views that maximize this quantity are selected as the good views of the 3D model.

Lee et al. [LVJ05] define the best view of a 3D object as the

one that maximizes the visible saliency from the corresponding viewpoint. The saliency measure is strictly related to the mean curvature. Polonsky et al. [PPB\*05] describe a number of metrics for measuring the goodness of a view. This includes surface area entropy, visibility ratio, curvature entropy, silhouette length, silhouette entropy, topological complexity, and surface entropy of semantic parts. These measures are computed for a set of candidate views. The view with the highest score is considered to be the most informative. Takahashi et al. [TFTN05] focus on solid objects such as volumes. In this work, the global optimal viewpoint is estimated by finding a compromise between locally optimal viewpoints for the feature components obtained after decomposing the entire volume. To take into account the high-level semantics in the best view selection, Mortara and Spagnuolo [MS09] uses semantic oriented segmentation to automatically extract meaningful features of a 3D shape. These methods find views that carry maximum information about the shape, they do not find ones that allow to visually discriminate the shape from others of different class. Since shapes exhibit high inter-class variation, these methods do not guarantee that shapes of the same class will have the same best views. Hence, these approaches are suitable for visualizing single objects out of context, rather than visually exploring the contents of a 3D data collection.

In order to capture the high-level semantic concepts of the 3D shapes, which are very important for visualization and exploration, we consider the problem in the context of 3D shape repositories where the data are clustered into semantic classes. The models within each class share common semantic concepts. Best-view selection can then be formulated as a problem of learning these features by the mean of feature selection and feature importance measurement. In this line of research, Shilane and Funkhouser [SF06, SF07] proposed an approach for automatically selecting the distinctive regions of 3D shapes. 3D models are represented with a large set of features computed at different locations on the shape. The retrieval performance of each feature is automatically predicted, and the most effective ones are selected to be used during the retrieval. Given that the descriptors are computed locally, the approach allows to select the most important regions of the surface of the 3D shape. As reported by the authors, by using view-based descriptors, instead of region descriptors, this approach can be extended in a straightforward manner for best view selection.

The main issue in this approach is that when a feature is not distinctive (i.e., present in all objects) then the distance of this feature to all objects in the database will be zero. If the returned ranked list is ordered in such a way that objects belonging to the same class of this shape are in the top then the feature will be assigned the highest distinction value and therefore it will be selected as highly salient. Hence, this approach finds common features of a class of shapes. In our case, we are interested in finding the features that discriminate the class of shapes from the other classes.

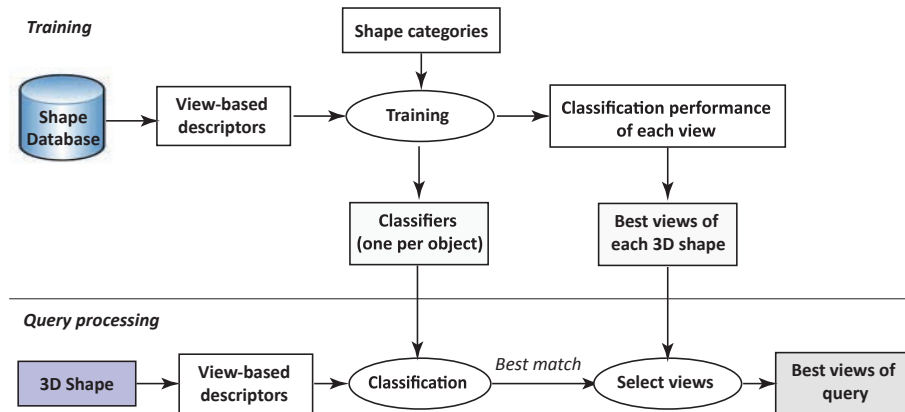


Figure 1: Overview of the framework.

Laga and Nakajima [LN08] use boosting to learn, in a supervised manner, the subset of views that discriminate a class of shapes from other classes. The approach requires prior pose normalization of all 3D models, converges to local optima, and performs poorly on classes with few training data. Furthermore, adding new objects to the database requires training again the classifiers.

The approach we propose in this paper can be seen also as a new measure of distinction. The main deviation from [SF06, SF07] is that distinction measure is based on the classification performance of each view of the 3D model. The algorithm we propose describes each shape with a set of view-dependent features. For each feature, we train a classifier that learns to discriminate the shape from other classes of shapes if it is described with that feature. The measured classification error is then considered as a measure of distinction of this feature. Specifically, we make the following contributions:

- An algorithm for learning the discriminative 2D views of a class of shapes from a training set.
- A measure for the discrimination ability of 2D views with respect to the semantic classes defined by the database classification. The measure is based on the classification performance of the feature.

The selected views are database-dependent which is an important feature for efficient visual browsing of 3D data collections.

Best view selection has many applications in Computer Graphics and online browsing of digital media contents. We are particularly motivated by the automatic generation of thumbnails of 3D models, automatic summarization of the database contents, and 2D-based 3D model search.

## 1.2. Overview

Figure 1 gives an overview of the proposed best-view selection approach. It performs as follows; First, every model in the database is described with a set of features describing the shape when viewed from a specific viewpoint. In our implementation we used 100 Light Field Descriptors (LFD) [CTSO03]. During the training stage, for each view  $X_j^i$  of a 3D model  $S_i$ , we train a classifier using Gentle AdaBoost which returns the likelihood that the model  $S_i$  belongs to a class of shapes  $C$  when described with the feature  $X_j^i$ . This likelihood measure formulated as the classification error is then the distinction measure of the feature  $X_j^i$ . The best views are then selected as the ones that maximize this measure.

At run-time, given the user-specified 3D model  $Q$ , a ranked list of  $k$ -best views is produced in a two-stage process. First, a large set of features are computed from the query model  $Q$ , in the same manner as for the database models. Then a retrieval list of the highly relevant objects to  $Q$  is found. The best views of  $Q$  are selected as the ones that are most similar to the best views of the object on the top of the retrieval list.

The key step is the way we predict the saliency of each feature with respect to a class of shapes in the training set. More formally, the saliency of a feature  $X$  with respect to a class of shapes  $C$  is the ability of this feature to discriminate the shapes of class  $C$  from the shapes of other classes in the database. Mathematically, the saliency can be directly related to the overall classification error of the feature. We use  $k$ -fold cross validation in order to estimate the classification error using training data. In the following sections, we detail each step of the proposed framework.

## 2. Approach

In this paper we use the following notation:

- We are given a collection  $\mathcal{S}$  of  $m$  polygon soup models,  $\mathcal{S} = \{S_i, i = 1, \dots, m\}$ .
- A partition  $\mathcal{C}$  of  $\mathcal{S}$ , where  $\mathcal{C} = \{C_i, i = 1, \dots, m_i\}, C_i \cap C_j = \emptyset$  and  $\cup C_i = \mathcal{S}$ .  $\mathcal{C}$  is referred in the remaining parts of the paper as a classification.
- $X_k^i$  refers to the descriptor of the  $k$ -th view of the model  $S_i$ .

## 2.1. View descriptors

The first step of the process is to represent a 3D object with a set of features describing its properties when viewed from different viewpoints.

Formally, we sample a set of  $N$  points from the surface of the unit sphere bounding the shape  $S_i$ . These points will be used as camera's viewpoints from which  $N$  two-dimensional views of the object are rendered. There are several sampling approaches including random sampling following the uniform distribution, or using the vertices of a spherical graph constructed by successive loop subdivisions of an initial icosahedron. We adopt the second approach in order to keep the adjacency structure between the viewpoints.

In a second step, each view is described with a Light-Field descriptor which is a combination of 35 Zernike moments and 10 Fourier coefficients [CTSO03]. These descriptors will be used in an initial step to cluster together adjacent views that are very similar reducing the set of 2D views into a small subset of views that are dissimilar. To do so we follow the same idea as in Yamauchi et al. [YSY\*06]. The set of viewpoints forms a spherical graph; the weight of an edge connecting two viewpoints is set as the dissimilarity between the 2D views. We use  $k$ -means clustering to segment the spherical graph into  $n$  clusters, instead of graph partitioning using MeTis [Kar10] as in [YSY\*06] since the balancing property is not required. The centroid of the clusters are used as candidates for representing the shape  $S_i$ .

This procedure is applied to every shape  $S_i$  in the database. At the end,  $S_i$  will be represented with a set of  $n$  views,  $n \leq N$ , each of which is represented with a feature vector  $X_k^i, k = 1, \dots, n$ , of dimension 45 (35 Zernike moments and 10 Fourier coefficients). We write  $S_i = \{X_k^i, k = 1, \dots, n\}$ .

## 2.2. Learning best views

In this paper we assume that a 2D view  $X_k^i$  of a 3D model  $S_i$  is good if it is able to discriminate the shape  $S_i$  from the shapes of other classes in the database. Based on this assumption, we define a measure of *importance* of a view as its classification performance.

First, we define the dissimilarity between a view  $X_k^i$  of a mesh  $S_i$  to every other mesh  $S_j$  in the database as

$$\text{dist}(X_k^i, S_j) = \min_{k'} \|X_k^i - X_{k'}^j\|. \quad (1)$$

where  $\|\cdot\|$  is the Euclidean norm. We then compute the importance of the view by evaluating its classification performance. Shilane et al. [SF07] defines the distinction of a feature as its retrieval performance. They used the Discounted Cumulative Gain (DCG) as a performance metric which requires the full retrieval list. While other measures, such as the nearest neighbor and precision for fixed-length retrieval list, can be used, their performance depends in many situations on the ordering of the retrieval list when many features have the same dissimilarity value. Classification performance-based metrics have the advantage that they do not require the entire retrieval list; for each model, we use the entire positive examples and only a subset of the negative examples randomly sampled from the database. It can be efficiently evaluated on the training set using  $k$ -fold cross validation.

In our implementation, we compute offline for every view of every object the distances given by Equation 2. Then for every view  $X_k^i$  of the object  $S_i$ , we learn a binary classifier  $\Phi_k^i$  defined as:

$$\Phi_k^i(X_k^i, S_j) = \begin{cases} 1 & \text{if } S_i \text{ is similar to } S_j, \\ -1, & \text{otherwise} \end{cases} \quad (2)$$

Two shapes are assumed similar if they belong to the same class of shapes. The classification error is then given by:

$$E_k^i = \frac{1}{2} \sum_{j=1}^{m_i} (t_j - \Phi_k^i(X_k^i, S_j))^2. \quad (3)$$

where  $m_i$  is the number of samples used for training, and  $t_j$  is the desired output for  $\Phi_k^i(X_k^i, S_j)$ , that is,  $t_j = 1$  if  $S_j$  is in the same class as  $S_i$ , and  $t_j = -1$  otherwise.

To select the best view of a 3D model  $S_i$ , we sort its views in ascending order according to their classification errors. The view with minimum error, i.e., the top of the list, is selected as the best one. Note that the classification error as defined in Equation 3 is a training error and therefore is not informative on the behavior of the feature at the test phase. For this reason we use the  $k$ -fold cross validation algorithm, with  $k = 5$ , to estimate efficiently the classification error. The algorithm performs as follows:

- The training set is randomly split into  $k$  subsets of same size.
- One subset is left out and the classifier is trained on the remaining data.
- The left-out subset is used as test data to measure the classification error.
- Repeat the above procedure  $k$ -times, every time leaving out a different subset.
- The classification error is the average classification error from the  $k$  errors obtained in the previous steps.

This algorithm requires setting the free parameter  $k$  of the  $k$ -fold cross validation. In all our experiments we set it manually to 5. Although there is no theoretical justification for

**Algorithm 1:** Learning the best views - training phase**Input:**

- A collection  $\mathcal{S}$  of 3D models.
- A classification  $\mathcal{C}$  of  $\mathcal{S}$ .

**Output:** A ranked list of best views for every model
 $S_i \in \mathcal{S}$ .
**for** every object  $S_i \in \mathcal{S}$  **do**
 Compute  $n$  view-based descriptors (LFD in our case).
**end for****for** every object  $S_i \in \mathcal{S}$  **do****for** every view descriptor  $X_k^i, k = 1, \dots, n$  of  $S_i$  **do**

- Compute the dissimilarity vector:

 $D_k^i = \{dist(X_k^i, S_j), j = 1, \dots, m_t\}$  and record the closest feature  $X_k^j$  to  $X_k^i$ .

 - Train with Gentle AdaBoost a binary classifier  $\mathcal{C}_{||}^j$  for the feature  $X_k^i$ .

 - Record the classification error computed using  $k$ -fold cross validation.
**end for**
 - Sort the views  $\{X_k^i\}$  in ascending order according to their estimated classification error.

 - Top views in the sorted list are the best views of the model  $S_i$ .
**end for**

this choice, we found experimentally, that this value provides a good compromise between computation time and generalization performance of the trained classifiers.

The last component to define is the type of classifier to use for measuring the classification error. In our early implementation we explored Boosting-like learning with dissimilarity functions. By doing so, the data samples are represented not by feature vectors but in terms of their pairwise dissimilarities computed using Equation 2. Learning with dissimilarity functions has the computational advantage of operating on a 1D space rather than a high-dimensional feature space. However, the classification error on the training data is very high. We justify this low performance by the fact that classes are not well separated in the product space, which is a necessary condition for efficient training with dissimilarity measures [WSY\*09]. Consequently, we choose to train the classifiers in the feature space. We use Gentle AdaBoost provided in the GML AdaBoost Matlab Toolbox [Vez10].

An important point to be considered is the choice of training data. In our implementation, we divide the data set  $\mathcal{S}$  into  $k$  parts, we do it in such a way that the number of positive examples in each subset is the same. We do that by selecting in each subset  $\frac{100}{k}$ % of positive examples and  $\frac{100}{k}$ % of negative examples. By doing so, we avoid the case where some of the subsets will contain no positive examples, since the number of positive examples is often much smaller than the number of negative examples, as it is the case for our data.

The training algorithm is summarized in Algorithm 1. Figure 2 shows examples of the selected best views 3D models from the Princeton Shape Benchmark [SMKF04].

### 3. Results

We implemented the proposed framework and tested it with the Princeton Shape Benchmark (PSB) [SMKF04]. The PSB comes with four levels of classification. The coarsest classification contains two classes: man-made objects and natural objects. Each of these classes exhibit very high-intra class variability and finding common features is very hard even for human. The finest classification however suffers from the lack of training data on some classes and therefore is not reliable for supervised learning. We choose to use coarse1 classification.

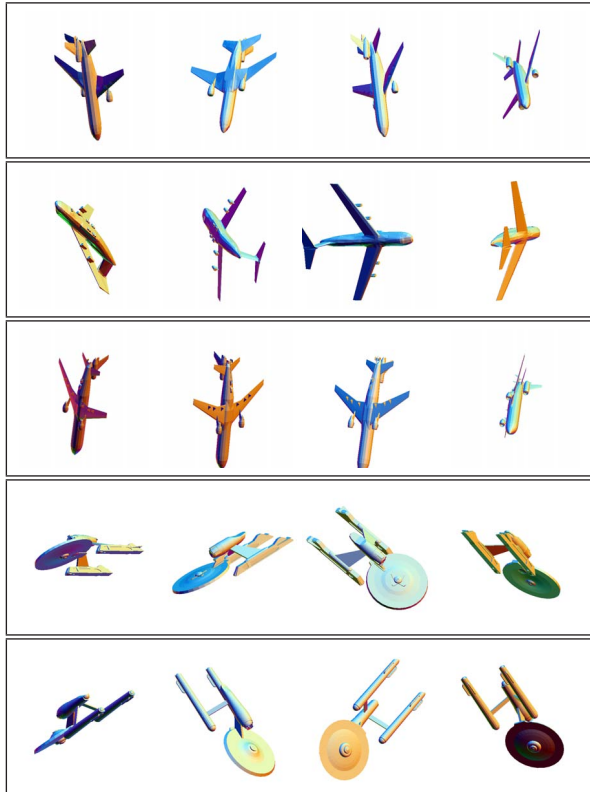
Our algorithm requires setting manually two parameters: the number of viewpoints  $N$  to sample on the bounding sphere and the number of clusters  $n$ . In our implementation, we choose  $N = 252$  uniformly distributed viewpoints which is sufficient compared to previous work in best-view selection [YSY\*06] and object recognition [MA00]. We set the number of clusters to  $n = 10$  which we found to be a good compromise between computation time during training and performance. Automatically adapting the number of characteristic views  $n$  to the complexity of the shape is an important issue that we plan to explore in the future.

Figure 2 shows the top-four best views selected for winged aircraft models. Other examples like human models and quadruped animals are shown in Figures 3 and 4. These examples show that the selected views exhibit the important features of their corresponding 3D models. It shows that several 3D models belonging to the same class have the same best-views (such as row 1 and 3 of Figure 2).

One of the major challenges in best view selection is finding automatically the proper orientation of the model in the image plane once the view point is found. To the best of our knowledge this is an open problem for any type of models, although some solutions have been previously proposed for man-made models [FCODS08]. Since the descriptors we use for view description are rotation invariant in the image plane we implemented an additional processing step which aligns the models to their principal axis in the image plane. This simple procedure is particularly efficient for elongated shapes such as human body models. Figure 5 shows the same example as in Figure 3 after orientation alignment. Various other examples are shown in Figure 6.

**Effect of the database.** As our approach is fully data-driven the selected best views depend on the classification of the models in the training data set. In this experiment we consider the coarse2 classification and show in Figure 7 the selected best views for two winged aircraft models which belong to the vehicle class. We can see that the selected views are different from the example in Figure 2 where we used the





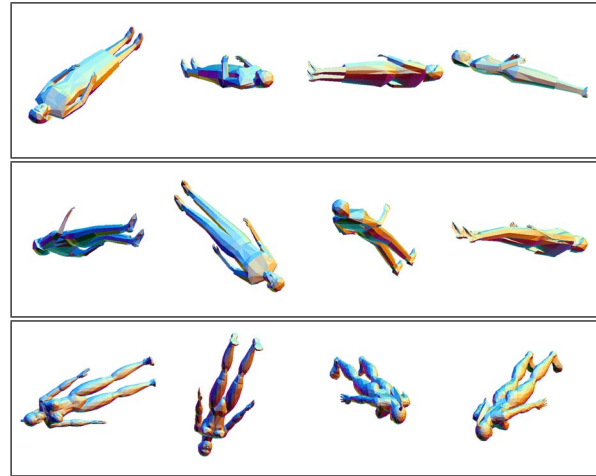
**Figure 2:** Examples of the selected best views using the coarse1 classification of the Princeton Shape Benchmark (class aircraft\_winged\_vehicle). Each row corresponds to one 3D model. The saliency is decreasing from left to right.

coarse1 classification for training. This example shows that our framework is very sensitive to the degree of intra-class variability in the training data. As one may expect, objects belonging to the same class of shapes should naturally have the same salient features. Our framework is robust on data with low intra-class variability. One way to improve the performance on high intra-class variability classifications such as the coarse2, and coarse3 of the PSB is to use different types of view descriptors, such as depth maps, that capture the surface properties.

#### 4. Conclusion

We proposed in this paper a new approach for the automatic selection of the best views of 3D models. Our definition of the goodness of a view is motivated by the need for efficient visual exploration of collections of 3D data. We proposed a framework for learning these views in a supervised manner. The approach is data-driven and therefore captures the semantics of the collection.

There are several improvements that can be made to our



**Figure 3:** Examples of the selected best views using the coarse1 classification of the Princeton Shape Benchmark (class animal\_human). Each row correspond to one 3D model. The saliency is decreasing from left to right.

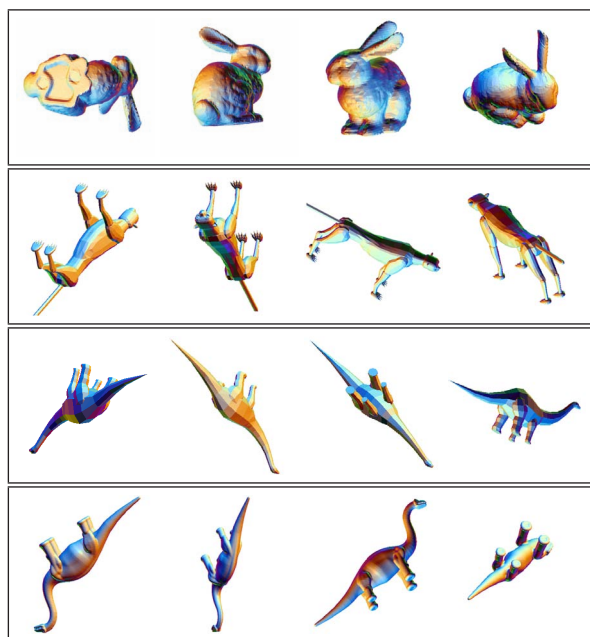
implementation. Our framework is robust on data with low intra-class variability and the performance decreases when the intra-class variability is high. This is the case for example with the coarse3 classification of the PSB. Finding distinctive features that are robust to high intra-class variability is a challenging problem to investigate in the future.

The Light Fields descriptors (LFD) we used for describing 2D views of 3D models capture only the silhouette of the shape when viewed from different viewpoints. By doing so, it cannot for example distinguish between the front and back views of a 3D model. We plan to experiment in the future with depth-based descriptors that can capture the surface properties of the shape.

A 3D model may have multiple classifications which can be hierarchical, such as the Princeton Shape Benchmark, as well as fuzzy. Exploiting such structures at the training stage may reveal interesting features of 3D models. Finally, best-view selection can be seen as a particular case of the general problem of 3D shape normalization which includes finding the upright and frontal orientations which are particularly challenging in the presence of symmetries. Combining our framework with recent results in upright orientation of shapes [FCODS08] is a promising direction to explore.

#### Acknowledgments

The 3D models used in this paper are from the Princeton Shape Benchmark, courtesy of the Shape Analysis Group, Princeton University. The author would like to thank the reviewers for their valuable comments and suggestions. This research is carried out by the support of the Japan Society

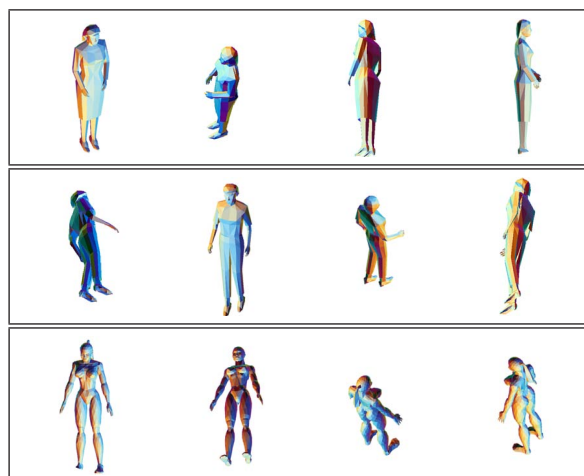


**Figure 4:** Examples of the selected best views using the coarse1 classification of the Princeton Shape Benchmark (class animal\_quadruped). Each row correspond to one 3D model. The saliency is decreasing from left to right.

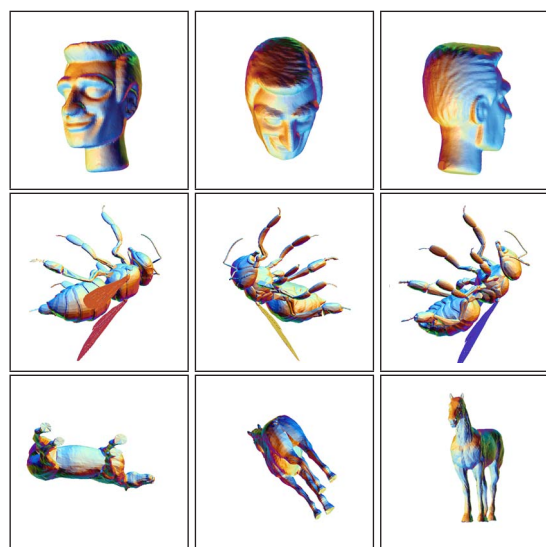
for the Promotion of Science (JSPS), Grant-in-Aid for Scientific Research Wakate-B Number 21700096.

## References

- [ADV07] ANSARY T. F., DAUDI M., VANDEBORRE J.-P.: A bayesian 3-d search engine using adaptive views clustering. *IEEE Transactions on Multimedia* 9, 1 (2007), 78–88. 2
- [CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUYOUNG M.: On visual similarity based 3D model retrieval. *Computer Graphics Forum* 22, 3 (2003), 223–232. 3, 4
- [DDA\*04] DENTON T., DEMIRCI M. F., ABRAHAMSON J., SHOKOUFANDEH A., DICKINSON S.: Selecting canonical views for view-based 3-D object recognition. *International Conference on Pattern Recognition* 2 (2004), 273–276. 2
- [FCODS08] FU H., COHEN-OR D., DROR G., SHEFFER A.: Upright orientation of man-made objects. In *SIGGRAPH '08: ACM SIGGRAPH 2008 papers* (New York, NY, USA, 2008), ACM, pp. 1–7. 2, 5, 6
- [Kar10] KARYPIS G.: Metis - family of multilevel partitioning algorithms. <http://glaros.dtc.umn.edu/gkhome/views/metis>, 1 2010. 4
- [LN08] LAGA H., NAKAJIMA M.: Supervised Learning of Salient 2D Views of 3D Models. *The Journal of the Society for Art and Science* 7, 4 (December 2008), 124–131. 3
- [LVJ05] LEE C. H., VARSHNEY A., JACOBS D. W.: Mesh saliency. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), ACM Press, pp. 659–666. 1, 2
- [MA00] MOKHTARIAN F., ABBASI S.: Automatic selection of

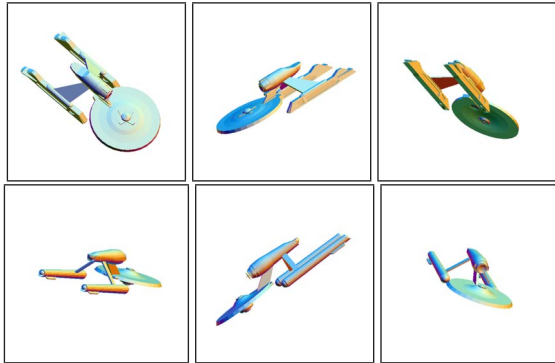


**Figure 5:** Examples of the selected best views using the coarse1 classification of the Princeton Shape Benchmark (class animal\_human) after aligning the models to their principal axis in the image plane.



**Figure 6:** Various 3D models and their top three best views selected by our algorithm.

- optimal views in multi-view object recognition. In *The British Machine Vision Conf. (BMVCA'00)*, IEEE Computer Society, 272–281. *NDR. National* (2000), pp. 272–281. 5
- [MS09] MORTARA M., SPAGNUOLO M.: Semantics-driven best view of 3D shapes. *Computers and Graphics* 33, 3 (2009), 280–290. 2
- [PPB\*05] POLONSKY O., PATANÈ G., BIASOTTI S., GOTSMAN C., SPAGNUOLO M.: What's in an image? Towards the Computation of the "Best" View of an Object. *The Visual Computer (Proceedings of Pacific Graphics)* 21, 8–10 (2005), 840–847. 1, 2



**Figure 7:** Effect of the classification on the best view selection. This example uses coarse2 classification of the PSB: the two winged aircraft models are classified in the vehicle category and therefore the selected views differ from the results when using coarse1 classification as shown in the last two rows of Figure 2.

- [PSG\*06] PODOLAK J., SHILANE P., GOLOVINSKIY A., RUSINKIEWICZ S., FUNKHOUSER T.: A planar-reflective symmetry transform for 3D shapes. 549–559. 2
- [SF06] SHILANE P., FUNKHOUSER T.: Selecting Distinctive 3D Shape Descriptors for Similarity Retrieval. *IEEE International Conference on Shape Modeling and Applications (SMI2006) 0* (2006), 18. 2, 3
- [SF07] SHILANE P., FUNKHOUSER T.: Distinctive regions of 3D surfaces. *ACM Transactions on Graphics* 26, 2 (e 07), 7. 2, 3, 4
- [SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *SMI'04: Proceedings of the Shape Modeling International 2004 (SMI'04)* (june 2004), pp. 167–178. 5
- [TFTN05] TAKAHASHI S., FUJISHIRO I., TAKESHIMA Y., NISHITA T.: A feature-driven approach to locating optimal viewpoints for volume visualization. In *IEEE Visualization Conference (VIS2005)* (2005), IEEE Computer Society, p. 63. 2
- [Vez10] VEZHNEVET A.: Gml adaboost matlab toolbox. <http://graphics.cs.msu.ru/science/research/machinelearning/adaboosttoolbox>, 1 2010. 5
- [VFSH03] VÁZQUEZ P.-P., FEIXAS M., SBERT M., HEIDRICH W.: Automatic view selection using viewpoint entropy and its applications to image-based modelling. *Computer Graphics Forum* 22, 4 (2003), 689–700. 1, 2
- [WSY\*09] WANG L., SUGIYAMA M., YANG C., HATANO K., FENG J.: Theory and algorithm for learning with dissimilarity functions. *Neural Computation* 21, 5 (2009), 1459–1484. 5
- [YSY\*06] YAMAUCHI H., SALEEM W., YOSHIZAWA S., KARNI Z., BELYAEV A., SEIDEL H.-P.: Towards stable and salient multi-view representation of 3D shapes. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)* (2006), p. 40. 2, 4, 5