Volume xx (200y), Number z, pp. 1–16

Category-specific Salient View Selection via Deep Convolutional Neural Networks

Seong-heum Kim[1], Yu-Wing Tai[†2], Joon-Young Lee[3], Jaesik Park[4], In So Kweon[1]

[1]Korea Advanced Institute of Science & Technology, Daejeon, Korea
[2]SenseTime Group Ltd., Hong Kong, China
[3]Adobe Research, CA, USA
[4]Intel Visual Computing Lab., CA, USA

Abstract

In this paper, we present a new framework to determine up front orientations and detect salient views of 3D models. The salient viewpoint to human preferences is the most informative projection with correct upright orientation. Our method utilizes two Convolutional Neural Network (CNN) architectures to encode category-specific information learnt from a large number of 3D shapes and 2D images on the web. Using the first CNN model with 3D voxel data, we generate a CNN shape feature to decide natural upright orientation of 3D objects. Once a 3D model is upright-aligned, the front projection and salient views are scored by category recognition using the second CNN model. The second CNN is trained over popular photo collections from internet users. In order to model comfortable viewing angles of 3D models, a category dependent prior is also learnt from the users. Our approach effectively combines category-specific scores and classical evaluations to produce a data-driven viewpoint saliency map. The best viewpoints from the method are quantitatively and qualitatively validated with more than 100 objects from 20 categories. Our thumbnail images of 3D models are the most favored among those from different approaches.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms, Viewing algorithms I.5.1 [Pattern Recognition]: Models—Neural Nets

1. Introduction

Over the past decade, the increasing number of 3D models have been organized and classified mainly through human endeavors [SMKF04, WSK*15]. Based on human annotations of pre-organized data collections, this paper focuses on the upfront orientation and salient view estimation of virtual 3D objects. In handling 3D objects, the automatic detection of upright, front orientations is useful for efficiently aligning, browsing, and arranging 3D models [CTSO03, SSB13, FRS*12]. Better selection of viewpoints also helps us to understand many 3D models without actually downloading the whole data or repeatedly scanning their projections in a 2D display device.

What is a natural pose of a 3D object? How do we define a good view? This problem is known as a key topic in selecting salient views of 3D models. Researchers have proposed a variety of measures for defining a good viewpoint. However, most previous approaches [PPB*05, SLF*11, LST12, LSN*14] are limited to using low-level data attributes. Consequently, the selected salient views do not always correspond to our common knowledge of the objects. We assume the object appearing in a specific viewpoint should be easily recognized by individuals. The other approaches [FCODS08, MS09, LZH12, ZLJW15] study some high-level measurements in seeking meaningful information, but generalization of the hand-designed features is not always straightforward for more sophisticated cases.

To resolve these issues, we first define a salient view as relative angles to the absolute standard, such as upfront orientations. Thus, we decompose the task of finding the best viewpoint into two sub-problems. One is estimating the upright orientation of a 3D object. The other is selecting the front view or even more salient projections of the upright model in the 2D domain.

† Corresponding Author

(a) Random projection (b) Upfront orientation (c) Our salient views

Figure 1: *Finding up front orientations and salient views of 3D models. (a) Thumbnail images from random projections. (b) Front views of upright objects. Our method estimates the upfront angles using classifiers trained with internet data collections. (c) The salient views to human preferences. Viewpoints with maximal saliency are given by the presented data-driven approach.*

We assume that there are some high-level patterns for the particularly preferred poses of 3D shapes, and people tend to preserve their photos when the 2D viewpoints are easily recognizable and maximally informative. In order to solve the problems in a scalable way, we take a data-driven approach with internet data collections and consider the deep Convolutional Neural Network (CNN) architectures [KSH12, WSK*15]. The CNN scores are used for assessment functions in finding upright, front vectors as well as salient views of 3D models.

Most objects have a preferred upright orientation in shape recognition [PRC81, TP89]. Our main assumption when encoding 3D data is that its orientation and categorization are independently learnable, but the relation between them should be considered in the final decision. Therefore, the first CNN architectures are instantiated in parallel to independently train two semantic properties: *upright orientation* and *shape templates*. Since the base of an object is closely related to its shape category [ea92, FCODS08], we then combine each output from the CNN models into a single shape descriptor.

With the second CNN architecture, visually preferred 2D patterns for *object categories* are encoded for a semantic score in measuring *familiarity* with a projection. What is salient to human perception depends on one's own visual experiences. Hence, we take advantage of a well-known public image database and assume that a projection that has higher confidence in object recognition receives higher saliency. Moreover, as studied in [PRC81, BTBV99], individuals are comfortable with viewpoints where an object in a known category is most often seen. We have modeled the human preferences of natural views according to each object category, so these semantic measurements are category-specific. The final *viewpoint saliency* considers both the high-level attributes and classical evaluations so that our best viewpoints can be upfront projections or projections that are more salient to human eyes.

The key contribution of this work is automatically encoding natural bases and informative projections of 3D models by studying CNN architectures for the best view selection problem. We observe the solution becomes more reliable as the number of well-aligned 3D models and category-labeled 2D pictures increases. Secondly, the automatic detection of up-front orientation also facilitates modeling the familiar viewing angles of different categories of objects. It is important to note that economical representations, such as thumbnail icons and previews of 3D models, are directly created by the familiar viewpoints, which helps us to recognize the 3D data more effectively than those from other algorithms.

Our algorithm works well for both artificial objects and non-rigid animal models. To evaluate our results quantitatively, we conduct a user study that confirms the thumbnail images from the viewpoints with maximal saliency are the most preferable in comparisons with results using only low-level attributes or random projections. In the qualitative comparisons with our baseline algorithms [LST12, SLF*11], we show perceptual improvements of our method using CNN activations over the other two methods. We believe this ultimately gives benefits to any types of display devices for 2D projections or user interfaces between humans and virtual 3D models.

2. Related work

In selecting the best view of a 3D object, there have been some open problems, such as finding upfront orientations of non-rigid objects and defining salient projections to human perception. We refer to the most representative works regarding these aspects.

Best view selection Psychophysical studies have shown that different views of a 3D object are not equally preferable [PRC81, BTBV99]. One reason is that we are sensitive to certain types of stimuli, such as contrast/curvature, complexity, and visibility.

The majority of previous works on viewpoint selection were based on low-level attributes, such as silhouette length, projected area, surface visibility, and other geometric quantities of projections [PB96, HS97]. Early work by Vázquez *et al.* [VFSH01] also used Shannon entropy to find the best view that maximizes the amount of information. Besides evaluating the 2D projections, Lee *et al.* [LVJ05] considered Gaussian-weighted mean curvatures as mesh saliency. Leifman *et al.* [LST12] introduced the local and global distinctiveness of vertices for estimating 3D saliency, which was extended by Shtrom *et al.* [SLT13] to detect the saliency for large-scale point sets.

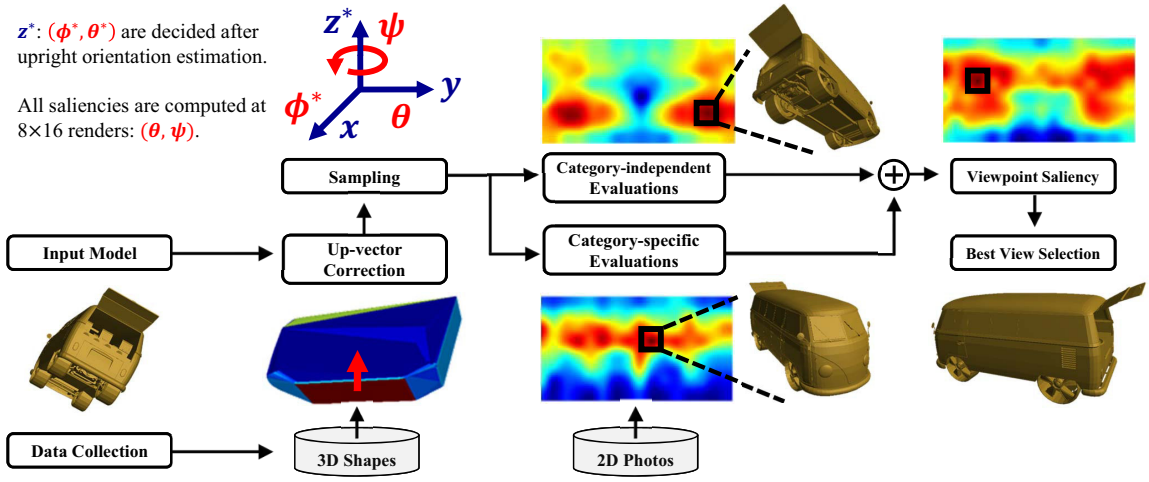

Figure 2: *System overview*

To develop general frameworks, Polonsky *et al.* [PPB*05] combined multiple measurements into a view descriptor, including surface area entropy, visibility ratio, curvature entropy, silhouette length, silhouette entropy, and topological complexity. Secord *et al.* [SLF*11] utilized the linear regression framework with multiple attributes in 2D and 3D to evaluate viewpoint saliency and they combined the evaluations using weights learnt from user preferences. In a recent work, Lienhard *et al.* [LSN*14] also obtained the view attributes taken by stochastically sampling a rule set of rendering parameters. The best thumbnails were recommended after grouping possible view attributes and sorting them by the user-defined number of clusters.

Compared with previous works, our approach chooses the best view by not only using low-level attributes, but also semantic scores. Focusing on natural orientations of objects, we select views that are effectively appealing to people.

Semantic views of 3D models Familiar features, such as the eyes of animals, ground planes, or natural orientations, make objects more readily recognizable [ea92, TK01]. In this sense, some researchers attempted to obtain semantics-driven viewpoints by identifying mesh segments [MS09, CGF09]. Meanwhile, Podolak*et al.* [PSG*06] introduced the planar-reflective symmetry transform and chose natural viewpoints by minimizing redundant symmetry. The concept of symmetry also provided a strong cue for upright orientation from the famous work [FCODS08]. By analyzing the characteristics of base planes, this approach extracted other geometric features, such as stability, parallelism, and visibility, from training examples. They encoded typical patterns of true ground planes, but the scope of this method was limited to man-made, rigid objects.

Recently, researchers have organized and utilized large 3D collections or databases of images for recovering semantic structures of data from low-level features [HCX*13]. For example, H. Laga [Lag11] treated the best-view selection problem as feature selection and classification tasks and provided category-dependent viewpoints in a data-driven manner. Liu *et al.* [LZH12] collected web images with a known category and processed them into feature vectors for checking the similarity to projections of 3D shapes. Another approach conducted a simple, statistical analysis of a high volume of internet photos in order to directly determine the canonical views [HO05, MW12]. In addition, [ZLJW15] demonstrated that the semantic viewpoints can be trained even by hand drawings.

However, these solutions are affected by not only their own data collections, but also all the steps of extracting hand-crafted features from high-dimensional inputs. Without proper prior knowledge, the specific methods often cause the loss of information when dealing with different datasets.

Deep Learning There is classical evidence of the relation between the favored viewpoint and the accuracy/speed of its category recognition [SM71, PRC81, TP89]. Based on this correlation, the shape recognition performance is greatly improved by finding the most informative viewpoints [CPCP15, WSK*15]. In contrast, our primary goal in this paper is not to increase the 2D/3D recognition accuracy itself, but to take upright orientations and viewpoints preferred by human perception with the aid of machine learning schemes. Also, our work differs from pose estimation, which typically refers to the problem of recovering poses of captured objects from input images [LPT13, SQLG15].

The Convolution Neural Network (CNN) has multiple layers of filter coefficients that produce compressive representations of input data with relatively little prior information. For instance, Krizhevsky*et al.* [LZH12] successfully captured low-dimensional semantic features from pixel observations by handling various kinds of variations in the image collections [KH11]. Followed by decision layers, the supervised learning methods showed the best performance in many labeling problems [KSH12, GDDM14, SZ15,

WSK*15, QSN*16, SMKLM15, HZRS15]. As CNN has begun a new trend, especially in extracting features, we take advantage of this mathematical tool for finding the upright orientation and salient views of 3D models.

This breakthrough is possible when the quality and quantity of human annotations are ever-increasing and millions of internet photos have become easily accessible [EVGW*, DDS*09, LMB*14]. In object recognition, it is now well-known that deep learning can benefit from a large amount of annotated data. In this paper, we have demonstrated how to apply deep learning to viewpoint selection. By using the already annotated image data together with the bounding boxes, object recognition becomes an important metric function for choosing the most salient renders.

Simultaneously, Yumer *et al.* [YAMK15] employed the deep learning framework for reducing a parametric space in high-dimensional procedural modeling. The shape variations and rendering parameters were encoded for a user-friendly interface, while we particularly focus on the learning viewpoint parameters of the renders.

3. Overview

Our system utilizes upfront 3D shapes and labeled 2D photos together with bounding box annotations. Based on the large-scale data collection, we correct the upright orientation of an input object. By rotating the upright model with pitch and yaw angles, we generate possible projections. We evaluate every sample projection with different geometric and semantic score functions. The upfront view and the most salient view are given by combining these evaluation scores. An overview of our approach is illustrated in Figure 2.

Given a 3D model, our first goal is to detect a natural base (Section 4), which determines the bottom side of a 3D model. This problem can be formulated as maximizing the saliency score of a base:

$$b^* = \arg\max_{\mathbf{b}} \mathcal{S}_b(M, \mathbf{b}|\mathcal{M}) \tag{1}$$

where M denotes the input 3D model, $\mathcal{M}$ is a shape database for learning natural orientations, $\mathbf{b}$ indicates a set of candidate bases of the model, and $\mathcal{S}_b$ is a function of the saliency score for upright orientation. We generate $\mathbf{b}$ with stable planes of a convex hull of the input object. From the 3D data with different bases, we train a deep CNN for extracting shape descriptors. The 3D shape is encoded in terms of the candidate bases $\mathbf{b}$, and a Random Forest (RF) classifier is trained and tested with the shape features. Having detected the ground plane with the maximum score $\mathcal{S}_b^*$, we then correct the upright orientation of the 3D model as the normal direction of the selected base plane b^*.

$$v^* = \arg\max_{\mathbf{v}} \mathcal{S}_v(M_{b^*}, \mathbf{v}|\mathcal{I}) \tag{2}$$

Given the upright 3D model M_{b^*}, the next goal is to detect the informative projection, which maximizes the viewpoint saliency (Section 5). We formulate the second problem as Eq. (2), where $\mathbf{v}$ indicates possible viewpoints of the upright model, and a set of projected images from the viewpoints is similarly computed as the light field representation in [CTSO03, CPCP15]. In other words, the normal orientation (ϕ^*, θ^*) were fixed for all sub-sections of Section 5. For front orientation estimation, we only search the front yaw angle ψ^* using classical, low-level information. By considering category-specific, high-level information, we finally predict salient viewpoints (θ, ψ) relative to the upfront angles $(\phi^*, \theta^*, \psi^*)$.

For each projected view, we compute the viewpoint saliency $\mathcal{S}_v$ using both low-level and high-level evaluations. The five most effective methods from existing works are employed for the category-independent measurements, and we propose novel measurements for the category-specific evaluations. The fine-tuned CNN model trained with the large-scale image collection is utilized in modeling the object-level saliency, and the category-dependent priors are statistically modeled by user preferences. These are all driven by the image dataset $\mathcal{I}$. The category-independent and category-specific evaluations are combined using the linear weights learnt from human subjects. The saliencies at non-sampled viewpoints are estimated through interpolation. On this viewpoint saliency map, our best viewpoint is obtained at the peak saliency $\mathcal{S}_v^*$.

We qualitatively evaluate the iconic/diagnostic projections with more than 100 objects, and quantitatively confirm them by a small user study. The details of the experimental setup and the results of the algorithm will be described in Section 6.

4. Learning Upright Orientation

In the first phase of our algorithm, we automatically estimate *upright orientation*, which means the normal direction of the natural base plane b^*. Specifically, the semantic properties of the upright objects are captured in the following procedures.

Pre-processing Since the mesh representation is not suitable for the convolutional operation, we can perform 3D convolution of the volume data after voxelizing the 3D shapes. For pre-processing, a 3D model is voxelized into a $32 \times 32 \times 32$ regular occupancy grid. We resize the voxel data to fit into a unit sphere and place its center at the origin of the Cartesian coordinate. Any object can be freely rotated around the x, y, and z-axes in order to align its up vector with the z-axis. In this manner, our oriented voxel model can be parameterized by two angles (ϕ, θ), and is assumed to be the same data with any rotation around the normal direction of the natural base.

Canonical Orientation Supposing an object is bounded with a cube, our *canonical orientations* are defined by the top, bottom, front, back, left, and right bases of the cube. In the case of an object that is aligned with the x, y, and z-axes, one of the six bases becomes a true ground plane of the ob-

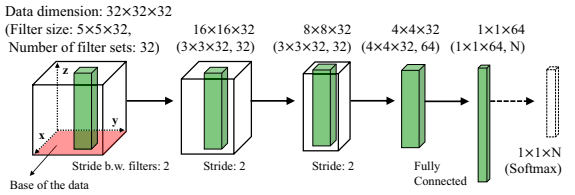

Figure 3: *Training CNNs. The CNN parameters are trained from learning canonical orientations (N=6) and shape templates (N=12).*

ject. Since they indicate specific features within the object, we learn the classical characteristics of canonical orientations through examples. Therefore, the semantic labels of training objects are the six orientations of the axis-aligned models by setting the different base planes, and all the possible rotations sharing one base plane are regarded as the same orientation. In this work, we select 720 objects from 48 different categories, and each of them is rotated around its up-vector by 10 degrees for data augmentation. In total, we have $720 \times 36 \times 6$ examples for training the CNN layers in Figure 3.

Shape Categorization We learn the characteristics of shape categories because certain prototypes of object shapes are related to their natural bases. Since there are some ambiguities between similar categories and there are also not enough examples for a few categories, we group them as 12 parent categories in terms of shape similarity and functionality. For the *shape templates*, we employ Aeroplane, Animal (four legs), Bed, Bottle, Chair, Cup, Human, Monitor, Plant, Table, Sofa, and Vehicle as the super-classes. In the voxel space, we define the broad categories to represent similar shape categories. Shape variations in each broad category are handled by increasing the number of training examples with different appearances. We collect 100 objects for each superclass and create $100 \times 36 \times 12$ sets of oriented voxels through the same manner of data augmentation. After that, we learn the second set of CNN parameters in order to classify all the training examples into 12 shape templates.

Training CNNs and RFs In order to capture the semantic information of the upright orientation, two CNN models are separately utilized for learning the canonical orientations $\mathbf{o}$ and shape templates $\mathbf{t}$, respectively, in the identical architectures. One instance inFigure 3 consists of three convolution layers for each, followed by a max-pooling layer, standard normalization [KH11, KSH12], and two fully connected layers. The first layer has a $5 \times 5 \times 32$ filter processing $32 \times 32 \times 32$ voxel data, while the candidate up vector is aligned with its z-axis. After training two sets of CNN parameters with different semantic labels, we replace the softmax layer of the CNN models with a Random Forest classifier. To achieve synergy between two independent semantic features, we concatenate each activation of the CNN models and train the new classifier. To be specific, two sets of

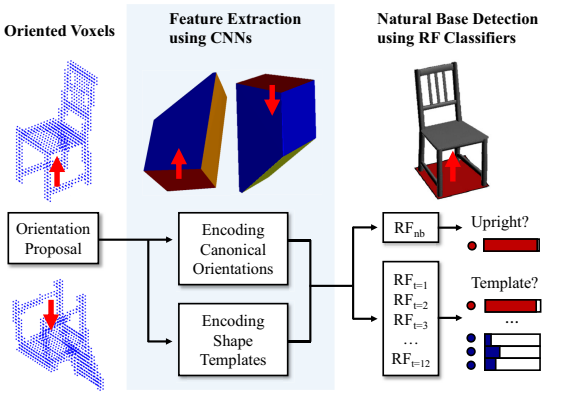

Figure 4: *Training RFs using shape features. All the candidate bases $\mathbf{b}$ from a convex hull of an object are scored by the output $\mathcal{S}_b$ of the classifier RF_{nb}. Note that high saliency is colored in red.*

64-dimensional activations from each of the last fully connected layers are combined into a shape feature vector as:

$$\mathcal{D}_{\mathcal{M}}(M,\mathbf{b}) = [\text{CNN}_o(\mathcal{X}(M,\mathbf{b})|\mathcal{M}) \;\; \text{CNN}_t(\mathcal{X}(M,\mathbf{b})|\mathcal{M})] \quad (3)$$

where $\mathcal{X}(M,\mathbf{b})$ denotes the rotated voxel data for aligning a candidate up vector with its z-axis. Note that the pose and category of this data is invariant to rotations around the z-axis. From the voxel data with different poses and categories, CNN_o is trained for six canonical orientations, and CNN_t is trained for learning the characteristics of 12 representative shape templates. Based on the CNN activations driven by the shape database $\mathcal{M}$, we define $\mathcal{D}_{\mathcal{M}}$ as a 128-dimensional shape descriptor for the given up vector.

Regarding learning the binary decision layers, we fix all the CNN parameters and train the RF classifiers with different guidances. For example, the classifier for a natural base RF_{nb} is trained with the 128-dimensional semantic vectors with a binary label, whether it is a true ground plane or not. Besides the natural base detection, we additionally train 12 class-specific RF classifiers RF_t $(t = 1,2,...,12)$ with the shape features and their memberships to the broad shape categories. In the stage illustrated in Figure 4, the shape features with respect to the candidate bases from orientation proposals can be collected as negative samples. Since the pose and category of this data is invariant to rotations around the up vectors, our data augmentation has been shown to be effective in deeply encoding upright orientation along the z-axis.

To the best of our knowledge, this is the first work to reformulate classical problems such as upright orientation estimation with a deep representation. In this paper, we utilize the same CNN structure for learning upright orientation as well as shape categorization, but the shape recognition based on upright 3D data is one of the active areas using deep learning frameworks. By replacing the relatively simple architecture with state-of-the-art CNNs such as [WSK*15, QSN*16, SMKLM15], we can expect to further improve our results with better CNN shape features.

Orientation Candidates Inspired by [FCODS08], our upright orientation proposal is based on the stable planes from a convex hull of a 3D model. Once we compute all the faces of the object convex hull, all coplanar polygons are merged to form one candidate base with the same up vector. The overall procedures are similar to those in the previous method, but we do not limit the candidate bases with the strict condition that the projected center of mass should lay on the supporting plane. Instead, we measure the Euclidean distance between the projected center of a 3D model and the barycenter of the supporting plane and sort all the candidate planes in terms of the distances. With this simplified convex mesh model, our system sets the maximum number of orientation candidates as 128, which allows it to sufficiently include the true ground plane of non-rigid objects even with unstable poses.

Natural Base Detection Given the trained CNN parameters and RF classifiers, we now query an input model with an unknown category and a random pose. For the first step, we generate the candidate up orientations from the stable bases of its convex hull. At every possible orientation, the model is rotated to ensure the current up vector becomes the z-axis in the voxel coordinate. The shape features are then extracted from two CNN models, and the CNN features from different annotation guidances are cascaded to form a single shape feature. Then, the combined feature vector is shared by the RF classifier for finding a natural base as well as 12 class-specific RF classifiers, so that the upright score is more reliable when the maximum value from the shape recognition scores is also high. As in Eq. (4), the 128-dimensional shape features for each candidate base are finally scored in the following:

$$\mathcal{S}_b(M,\mathbf{b}|\mathcal{M}) = \mathrm{RF}_{nb}\Big(\mathcal{D}_{\mathcal{M}}(M,\mathbf{b})\Big) \tag{4}$$

After testing all the candidate orientations, we pick the most salient base as the natural base of the input model. By setting the up vector with the maximal saliency, the pose of the model is corrected by the estimated orientation. On the upright orientation, we predict its broad category with the most confidence out of all class-specific RF classifiers, as displayed in Figure 4. The category prediction, $\mathcal{S}_t^* = \mathrm{RF}_t\Big(\mathcal{D}_{\mathcal{M}}(M,b^*)\Big)$, computed similarly to Eq. (4), is utilized to select the natural renders in the next phase of our algorithm.

5. Modeling Viewpoint Saliency

After upright orientation estimation, the up vector of a 3D model is fixed with the z^*-axis. Now, the second phase of our algorithm finds a *salient viewpoint* among the candidate viewpoints $\mathbf{v}$ ranged from $\theta : [-\pi/2, \pi/2]$, $\psi : [-\pi, \pi]$. θ and ψ stand for rotation angles about the y-axis and z-axis, respectively. We generate possible renders of the 3D model by sampling viewpoints at every 22.5° pitch angle θ and 22.5° yaw angle ψ. The saliency at the 8×16 viewpoints are computed using eight different methods. To evaluate the uniform size of the projected views, a virtual camera is directed at an object in the fixed distance, and its focal length is long enough to ensure there is little projective distortion in the projection.

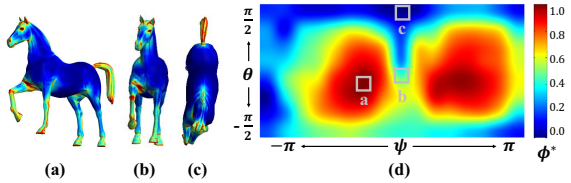

Figure 5: *Surface curvature. (a)-(c) are different projected views of* HORSE. *(a) is the view with maximum saliency, (b) is the front view of the model, and (c) is the view with minimum saliency. The color coded on the mesh model are the estimated curvature saliency.*

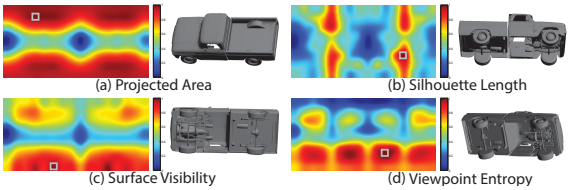

Figure 6: *Effect of classical saliency evaluations. The corresponding views with the maximum saliency are shown.*

We will first describe our five low-level and three high-level evaluations for each sampled projection. After that, we will explain the optimal weights for combining all the evaluation measurements. Our final viewpoint saliency map indicates all the iconic, discriminative information for understanding the input 3D data.

5.1. Category-independent Evaluations

We use five low-level saliency measurements, which are introduced in the previous literature [LVJ05, LST12, PB96, HS97, VFSH01]. The low-level attributes are solely determined from the geometry of the 3D model and its projection without any prior information about the object category of a model. In this sense, they are category-independent. Figure 5 and Figure 6 illustrate the effects of different low-level evaluations. The 2D representation is an unwarped saliency map on a unit sphere, and all of these saliencies are normalized to $[0,1]$ by the maximum value of the viewpoint saliency map.

Surface Curvature (f_{sc}) For this evaluation, we compute Gaussian-weighted mean curvatures of every vertex by [LVJ05], and the curvature values are smoothed over the surface in the manner of [LST12]. The higher the curvature of a vertex, the higher the saliency it represents. This saliency is propagated to mesh surfaces according to geodesic distances to all the other vertices. To compute the saliency of a projected viewpoint, the 3D saliencies are pro-

jected onto 2D images according to the viewpoint parameters. We aggregate and normalize the saliencies of the visible surfaces to form the saliency evaluation of a particular view.

Projected Area (f_{pa}) This attribute evaluates the saliency by maximizing the area of projection of a 3D model. This is proportional to the size of the observed silhouette. The larger the area in the image domain that it is projected, the higher the chance the object's projection will give enough information.

Silhouette Length (f_{sl}) This attribute evaluates the saliency by measuring the length of the silhouette boundaries projected on a 3D model. If the silhouette length is longer, it is believed that the projected viewpoint is more complicated. Thus, the silhouette length receives higher saliency when more complex boundaries appear in the projected view.

Surface Visibility (f_{sv}) This attribute is similar to the evaluation of the projected area. The difference is that it maximizes the area of visible 3D surfaces instead of maximizing the area of a 2D projection. This estimation can be achieved by measuring the area of visible surfaces in comparison to the total area of the surfaces of a 3D model.

Viewpoint Entropy (f_{ve}) In addition to the surface visibility, we consider the distribution of fractional visibility. The fractional visibility is defined as the area of a projected mesh face divided by its original area in the 3D domain. The distribution of the visibility is related to the diversity of the surface normal directions observed at a certain viewpoint. By computing the Shannon entropy of the distribution of fractional visibility, the best viewpoint favors surface variations in the presented projection.

5.2. Category-specific Evaluations

Our high-level evaluations are motivated by a psychophysical experiment by Palmer *et al.* [PRC81]. In the experiment, they asked human subjects to select the best canonical views of different objects. Unsurprisingly, different categories of objects have different preferred viewpoints. In the same manner, we hire a deep CNN model trained on ImageNet to recognize the category of tested 3D objects. For the object-level evaluation, the easily recognizable renders are preferred to other viewpoints with low classification scores. With a category identified in the recognition step, the comfortable angles are guided by a simple statistical model.

Category Recognition (f_{cr}) We use the category information both in upright estimation and salient view detection. In upright estimation, however, the initial voxel representation does not give any preference to viewing angles. Therefore, the category of 2D projections is finally confirmed using the initial shape recognition scores.

In this paper, the category recognition score is based on AlexNet trained with the large scale image collection from [EVGW*, DDS*09, LMB*14]. This is because AlexNet is one of the standard CNN architectures [KSH12], which can also be replaced by deeper architectures such as [SZ14, SLJ*15, HZRS15] that achieve state-of-the-art object recognition performance. The pre-trained CNN model uses 1,000 categories, which are too specific in our task. For simplicity, we assume the 3D objects in our experiments belong to one of the standard *PASCAL (Pattern Analysis, Statistical Modelling and Computational Learning)* categories 12 man-made subcategories and 8 non-rigid animal subcategories. After collecting additional images from [EVGW*, LMB*14] for each category and converting them in the gray scale, we fine-tune AlexNet with a new soft-max layer. Using this scheme, the 4096-dimension activations in the last layer are now linearly mapped to the 20 dimensions of a category score vector. For each projected viewpoint, the fine-tuned CNN model outputs a vector of classification scores.

Figure 7: *Category recognition. Higher saliency is given to viewpoints which are easier to recognize.*

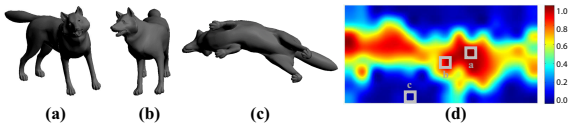

Figure 8: *Recognition saliency map. (a)-(c) are different projections of* DOG. *(a) is the most easily recognized view. (c) is the worst view for recognizing the model.*

$$s_c(\mathbf{v}) = \mathrm{CNN}_i(m_v|\mathcal{I}), \tag{5}$$

where s_c denotes the recognition score for the category $\mathbf{c}$, $\mathrm{CNN}_i(\cdot)$ is our second CNN architecture, and the training image set $\mathcal{I}$ has 90,000 gray images from the 20 categories. m_v is the rendered image of an upright 3D model M_{b^*} at viewpoint $\mathbf{v}$. Figure 7 illustrates the recognition score of a chair object with different viewpoints. The viewpoints that are easier to recognize received higher scores.

Since individual classification with different viewpoints can be noisy due to the model properties and rendering conditions, we apply Markov Random Field (MRF) to the sampled viewpoints to correct errors and enforce the smoothness of the estimated recognition saliency. For this step, prior knowledge from its 3D shape is recollected for improving the recognition accuracy. By mapping the 12 prototypes of the 3D shapes into 20 PASCAL categories, we set the bias values s_t^* for 2D classification from the previous 3D classification scores $\mathcal{S}_t^*$. For some parent categories in 3D, all the

related PASCAL classes in the similar structure are believed to have the same bias values. For example, a four-legged Animal prototype has the following five child categories in the projection domain: CAT, COW, DOG, HORSE, and SHEEP.

$$\arg\min_{\mathbf{c}} \sum_{\mathbf{v}} E_d\left(\lambda_1 s_t^* + s_c(\mathbf{v})\right) + \lambda_2 \sum_{\mathbf{v}} \sum_{\mathbf{v}' \in \mathcal{N}_v} E_n\left(s_c(\mathbf{v}), s_c(\mathbf{v}')\right) \quad (6)$$

where $s_c(\mathbf{v})$ is the recognition saliency of category $\mathbf{c}$ at viewpoint $\mathbf{v}$, $\mathcal{N}_v$ is the first-order neighboring viewpoints of $\mathbf{v}$ defined over the sampled viewpoints in 2D, and s_t^* is the prior from the previous shape recognition. With this bias term, the data term energy E_d has an offset value that is uniformly distributed and added to the saliency of the 20 categories. E_n is the smoothness term energy defined by a diagonal matrix, where each diagonal entry is the $l2$-norm distance between $s_c(\mathbf{v})$ and $s_c(\mathbf{v}')$. In this work, we use $\lambda_1 = 0.2$ for the biasing effect and $\lambda_2 = 0.1$ for the regularization.

To recognize the category of an object, we sum up the recognition saliency of all sampled viewpoints after MRF smoothing. The category that receives the highest total sum of saliency will be regarded as the estimated category of the tested object ($c^* = \arg\max_{\mathbf{c}} \sum_{\mathbf{v}} s_c(\mathbf{v})$). As a result, the recognition saliency map of that category will be regarded as the category recognition viewpoint saliency map ($f_{cr} = s_{c^*}$). Figure 8 shows an example of our recognition saliency map. This figure shows an interpolated map with category recognition scores from 8×16 viewpoints. The recognition saliency map indicates how important one viewpoint is over other viewpoints, while each category recognition score comes from an object recognition metric for a specific viewpoint.

Category Dependent Prior (f_θ, f_ψ) The recognition saliency evaluates the viewpoints according to ease of recognition, but the easiest recognizable view may not generally be the most typical view for the category to people. As studied in [BTBV99, SLF*11], some objects are more comfortably viewed from the most frequently observed viewpoints. We model this prior by manually annotating the presented orientations of objects in the internet photos. We count how often people capture the object at certain viewing angles in the 8×16 discrete space of the (θ, ψ) coordinate. In this voting space, every count has three levels of a comfortableness score for one sample image. Based on the collected user preferences from the randomly selected 5% of our training examples (225 for each category), we statistically model this prior for each category.

$$f_\theta(\mathbf{v}) = \frac{1}{Z} \sum_k \frac{4a \cdot \exp\left(b \cdot (\theta - \theta_c(k))\right)}{\left\{1 + \exp\left(b \cdot (\theta - \theta_c(k))\right)\right\}^2}, \quad (7)$$

$$f_\psi(\mathbf{v}) = \frac{1}{Z} \sum_k \frac{4a \cdot \exp\left(b \cdot |\psi - \psi_c(k)|\right)}{\left\{1 + \exp\left(b \cdot (\psi - \psi_c(k))\right)\right\}^2} \quad (8)$$

where $\theta_c(i)$ is the tilt angle of a selected viewpoint, a is the comfortableness score $\{0,1,2\}$, $b = 2$ is a parameter for controlling the decay rate, and Z is a normalization constant. Eq. (7) is the first-order derivative of the sigmoid function, which has the peak value at the weighted averages of pitch angles θ_c. This equation is a smooth function that gives higher saliency to the angles that are closer to the most commonly selected pitch angle. Figure 9 illustrates the effect of this prior. The prior for yaw angle f_ψ is defined similarly, but we take the absolute operation in order to hold a symmetric property of the side views. Additionally, ψ_c is not defined in some objects with rotational symmetry. This means each angle is category-dependent, so we separately model the priors for tilting and rotating the natural base, respectively.

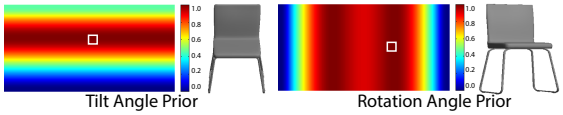

Figure 9: *Category dependent priors. Our category dependent priors of* CHIAR *on the tilt (pitch) angle and rotation (yaw) angle are shown.*

5.3. Front Orientation

Learning front orientation requires more 3D objects because data augmentation by rotating around an up-vector is no longer available. In practice, we recover the front orientation of a given upright object with 2D cues rather than direct 3D information. We assume correct roll and pitch angles and only search for the front yaw angle ψ. The front view is defined as the most complex, semantic part with two symmetric sides of upright objects.

In our front orientation proposal, we utilize the following three low-level saliencies: surface curvature, silhouette length, and viewpoint entropy. The candidate front angles are collected when the projection measurements of its side views are similar to each other. We pick several peak positions for the candidate angles in terms of the similarity of these attributes and select the front orientation, which has the maximal category recognition score among those of other candidate positions. The center position of our viewpoint saliency map is then set to the estimated front angle ψ^*. Once we mark the front view on the map, other canonical orientations, such as back, top, bottom, and two side views, are determined. This is considered our intermediate output.

5.4. Linear Regression

Our final step is to merge the individually estimated saliency maps, $\{f_{se}, f_{pa}, f_{sl}, f_{sv}, f_{ve}, f_{cr}, f_\theta, f_\psi\}$, to form the final saliency map $\mathcal{S}_v$. We use a linear regression approach:

$$\mathcal{S}_v = [f_{se}, f_{pa}, f_{sl}, f_{sv}, f_{ve}, f_{cr}, f_\theta, f_\psi]\, \mathbf{w}^\mathsf{T} \quad (9)$$

where $\mathbf{w} \in R^{8\times 1}$.

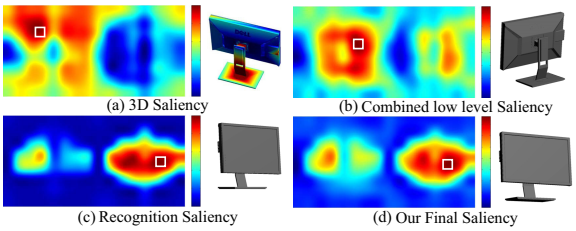

Figure 10: *Linear regression. (a) 3D saliency estimated by surface curvature. (b) Saliency map which combined all low-level evaluations. (c) Recognition saliency map. (d) Saliency map which combined both category-specific/independent evaluations. The combined weights are estimated from collected user preferences.*

To estimate the optimal weight **w**, we collect user preferences for our projection images. We render five upright objects from each category with 8×16 viewpoints. We then ask 20 users to select 6-8 best views that they feel the most comfortable with to represent the rendered model. Similar to the category-dependent prior, we learn a different w for each category c. This is because different categories of objects express different behaviors in terms of user comfortableness. We then solve the equation, which minimizes:

$$w_c^* = \arg\min_{\mathbf{w}} \sum_{\mathbf{v}\in\mathcal{V}_c} ||f^*(\mathbf{v}){w_c}^\mathsf{T} - 1||^2 + \sum_{\mathbf{v}\notin\mathcal{V}_c} ||f^*(\mathbf{v}){w_c}^\mathsf{T}||^2 \quad (10)$$

subject to $\sum_{k=1}^{8} w_c(k) = 1$, and $w_c(k) \in [0,1]$, where $\mathcal{V}_c$ is the collection of user-selected viewpoints for category **c**, and $f^* = [f_{se}, f_{pa}, f_{sl}, f_{sv}, f_{lc}, f_{cr}, f_\theta, f_\psi]$. Note that $\mathcal{V}_c$ allows multiple instances for the same viewpoint. Thus, if certain viewpoints are selected multiple times by users, this has higher inference for the estimation of w_c. Eq. (10) estimates the user-dependent weight by maximizing the saliency in $\mathcal{S}_v$ for viewpoints selected by users, and it minimizes the saliency for viewpoints that were never selected by users. Due to our limited number of human subjects, we fix the weights for low-level measurements $[f_{se}, f_{pa}, f_{sl}, f_{sv}, f_{ve}]$ for all categories as the same in [SLF*11] during the optimization.

Figure 10 shows our combined saliency map. For comparisons, we have also shown the 3D saliency map, category-independent saliency map, and category recognition saliency map in Figure 10. The category-independent saliency maps are combined with weights defined over $[f_{se}, f_{pa}, f_{sl}, f_{sv}, f_{ve}]$, and their sum is normalized as 1. Since we do not utilize any category-specific information in this low-level evaluation, the same weights to those used in [SLF*11] are applied for all categories. As illustrated in Figure 10, the category recognition saliency has stronger inferences than other evaluations for the final saliency map. The detected salient views also agree with human preferences. This is because the frontal-view images of a monitor appear more frequent than other views in the training data of the fine-tuned CNN model. Therefore, the CNN activation gives higher confidence to frontal views in the recognition.

48 Shape Categories	12 Shape Templates	20 PASCAL Categories
AIRCRAFT, BIPLANE	AEROPLANE	AEROPLANE
CAMEL, CAT, COW, DOG, ELEPHANT, HORSE, LION, SHEEP	ANIMAL(4-LEG)	CAT, COW, DOG, HORSE, SHEEP
BIRD	BIRD	BIRD
BOTTLE, VASE	BOTTLE	BOTTLE
BENCH, CHAIR, TOILET	CHAIR	CHAIR
BATHTUB, BOWL, BOAT, CUP, SINK	BOAT	BOAT
HUMAN	HUMAN	HUMAN
LAPTOP, MONITOR, TV	MONITOR	MONITOR
PLANT, FLOWERPOT	PLANT	PLANT
SOFA, BED	SOFA	SOFA
DESK, STOOL, TABLE	TABLE	TABLE
BIKE, MOTORBIKE, CAR, BUS, TRAIN	VEHICLE	BIKE, MOTORBIKE, CAR, BUS, TRAIN
BOOKSHELF, BUILDING, CONE, CURTAIN, GUITAR, KEYBOARD, LAMP, PIANO, RADIO,TENT, TV STAND	OTHERS	

Table 1: *Grouping categories. Our 48 categories of 3D inputs are defined into the 12 shape templates. The mappings from them to the 20 PASCAL categories in the projection domain are also shown.*

In contrast, using only the category-independent evaluations detects the view that has the most complex structures.

6. Experiments

In this section, we outline the procedures and examine the results of our experiments and discuss the benefits and limitations of this approach.

Data Collection Our method determines human preferences for viewpoints from a large-scale data collection. For learning 3D shapes of general categories, we downloaded 1,440 models of 48 categories from different datasets [XMS14, CTSO03, SP04, BRLB14, WSK*15]. With the rotation enlargement, we took 1440×36 samples with annotations and selected some of them for training and test tasks in the paper. Since the availability of 3D models for each category was different, we also queried some labeled objects of a few categories in need from 3D Warehouse, Turbo Squid, and Yobi3D. During this process, we excluded 3D models with only very few meshes or incorrect normal directions. More importantly, we noticed that some similar objects had few variations between their specific categories, and they could easily be grouped as a common prototype at a higher level. Table 1 summarizes the mapping from 48 specific categories to 12 representative shape templates.

In order to determine comfortable viewing angles for rendered images, we collected 2D photos captured by internet users. Following the convention in the classical image recognition task, we set our final categories as the standard 20 PASCAL classes [EVGW*]. In the projection domain, several PASCAL categories could have a common 3D shape template according to Table 1. Currently, we do not have enough texture information for 3D mesh models, so we used gray-scale images for determining viewpoint preferences. We collected 90,000 internet photos of 20 standard categories with the bounding box annotations

Methods	Orientation Proposal	Upright orientation Estimation	Front orientation Estimation
OURS (RIGID OBJECTS)	100%	87.7%	83.3%
OURS (ALL CATEGORIES)	98.4%	75.2%	71.8%
[FCODS08] (RIGID OBJECTS)	93.4%	83.1%	N/A
[FCODS08] (ALL CATEGORIES)	74.6%	63.9%	N/A

Table 2: *Up front orientation estimation. The algorithm performances of three steps (orientation proposal, natural base detection, front orientation estimation) are analyzed.*

from [DDS*09, EVGW*, LMB*14], and each category had 4,500 training images. From the shape collection, we randomly chose five 3D models for each class, and the render images of these 100 objects became the test images.

Running Time For the experimental setup, we used one computer with Intel i7-4790 CPU, Samsung 850 PRO 256GB SSD, and NVIDIA GeForce GTX 980 GPU. The first CNN model in our upright estimation was a multimodal architecture taking two independent human annotations in parallel, followed by fully connected layers leading into task-specific RF classifiers. Since it was a relatively small network, the total training time only required less than one day. The second CNN model was based on deeper representations with the large scale image collection. It took nearly three days straight to fine-tune the pre-trained AlexNet with our own dataset. However, it is noted that the testing time for one viewpoint of a 3D model was less than three seconds including the rendering pipeline, while the classical evaluations used in this paper took 15-20 seconds in average, for each viewpoint. Since we used 8×16 samples for the final saliency map of a test object, there was a great advantage on processing time in using CNN features compared to computing traditional features.

Upfront Orientaion Estimation For the upright orientation, instead of designing specific feature extraction methods, our approach increases the volume of training datasets and learns two sets of flexible parameters with different semantic properties. In our experiments, the classification accuracy in independent tasks reached 70% for canonical orientations and 88% for shape templates. After replacing the decision layers with a RF classifier and concatenating each activation from the CNN models, we achieved 4.6% and 11.3% improvements compared with the classical method, as shown in Table 2, in detecting the upright orientation of rigid and non-rigid objects.

One of the reasons that we can handle non-rigid objects better is the method of generating candidate bases. The assumption for a base that the center of mass should be projected inside the base plane is true in the real world, but it often does not strictly hold in non-rigid mesh models due to some physical conditions. In contrast to the previous approach, we utilized the stability assumption by simplifying the convex hull of an object to have a fixed number of faces (128 in our case). Considering all the faces as possible bases, we did not miss a true ground plane in the proposal step.

Recently, most 3D models on the web have already been aligned with canonical axes; thus, we only needed to reverse top-bottom bases, front-back sides due to the different convention. Thus, our method was especially robust for fixing the canonical orientations, and it required much less human intervention in finding the up-front orientation for such axis-aligned objects.

In the evaluation for front orientation estimation, we manually prepared the front orientations of 3D models according to our assumptions. For example, the front yaw angle of a cup with a handle was defined as the view showing that handle. Bottles without handles are rotational symmetric, so any yaw angles were acceptable.

Best View Selection In the paper, our ultimate goal is to find salient viewpoints for human preferences. Figure 11 and Figure 12 show some of the results for qualitative evaluations. The objects in Fig. 11 are examples where the category of the tested 3D models was correctly classified, and the objects in Fig. 12 are examples where the category of the tested 3D models was misclassified. Additionally, we tested non-learned objects in Figure 13, and other failure examples from our algorithm are discussed in Figure 14.

In the first columns of all the figures, the results using only the hand-crafted features in [SLF*11, LST12] are presented, and the results using only the CNN features, without relying on any hand-crafted features, are presented in the second columns. The final results using the combined low-level saliency, recognition saliency, and category dependent priors are presented in the third columns.

As shown in Fig. 11, some categories, such as BIKE, have balanced weights between category-independent and category recognition evaluations. We observe that both algorithms give reasonable viewpoints, and our algorithm explains why people liked such viewpoints in the context of object recognition. In contrast, for other categories, such as CAR and TABLE, the category recognition evaluations are more dominant. Additionally, some objects, such as AIRPLANE, BIKE, PLANT, and TABLE, have multiple peaks in their viewpoint saliency map. This is due to the symmetry structures of the tested object. For non-rigid animal objects, such as CAT, DOG, and HUMAN, there is usually a strong peak near their eyes and faces.

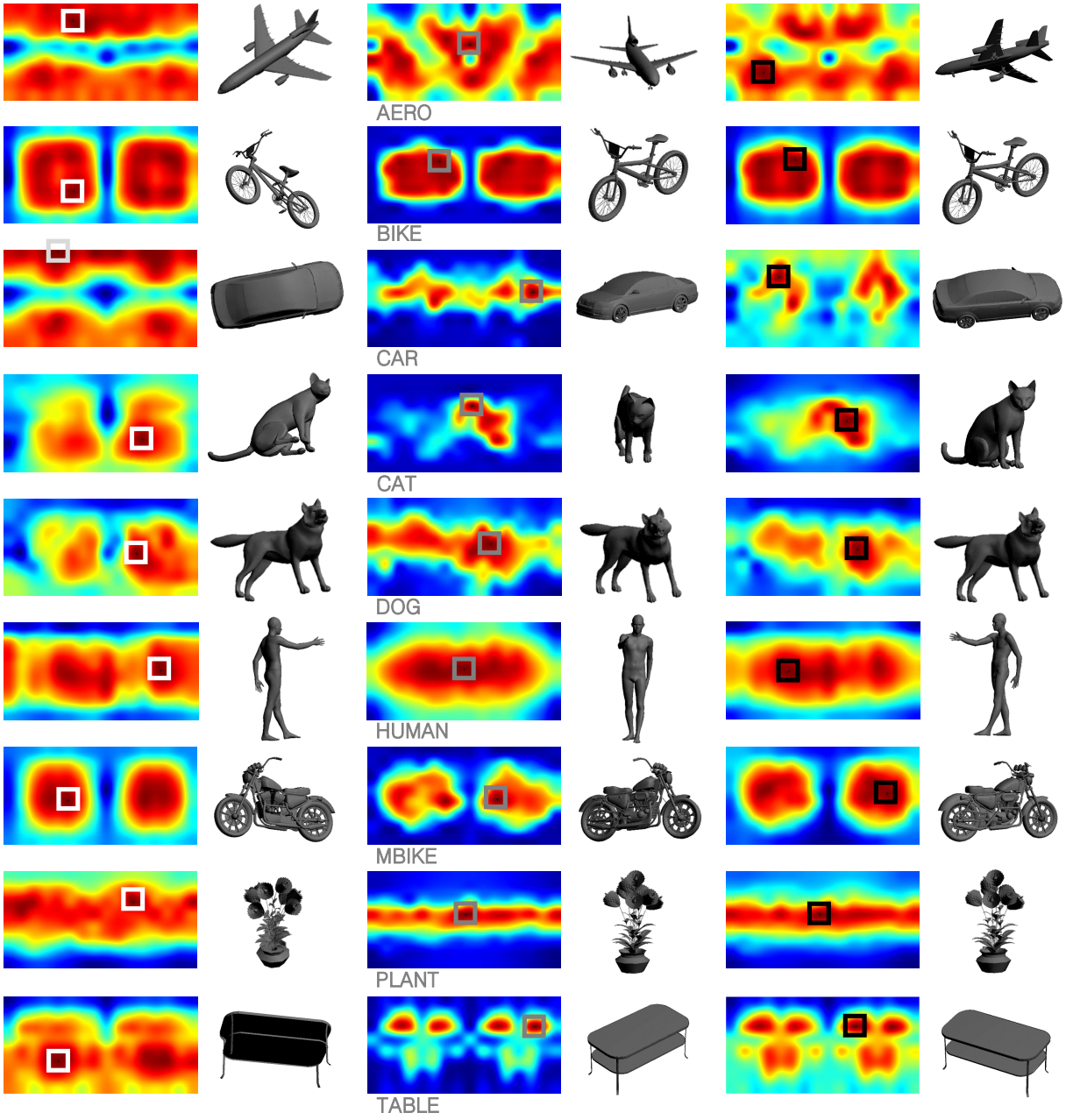

Figure 11: *Qualitative results on our estimated saliency maps and selected best views. Left: Low-level saliency. Middle: Recognition saliency. The recognized object categories are also shown. Right: Our final saliency. The most salient views are shown to the right of each saliency map. These examples are results where the category classification of tested objects is successful.*

In Fig. 12, although the category of tested objects was misclassified, the detected salient views are still reasonable. There are two reasons that account for the results. Above all, the low-level evaluation mostly resolves the ambiguities in the high-level evaluation to select preferable views, especially for categories that have balanced weights between high-level and low-level evaluations. In addition, although the recognized category may not be perfect, the recognition saliency map provides meaningful preferences for certain viewpoints over others. These viewpoints have richer information for classification; thus, they are preferable to human perception. Because of the similarity between categories, although an object was wrongly classified, it was defined as a category with a similar appearance. For example, a sheep

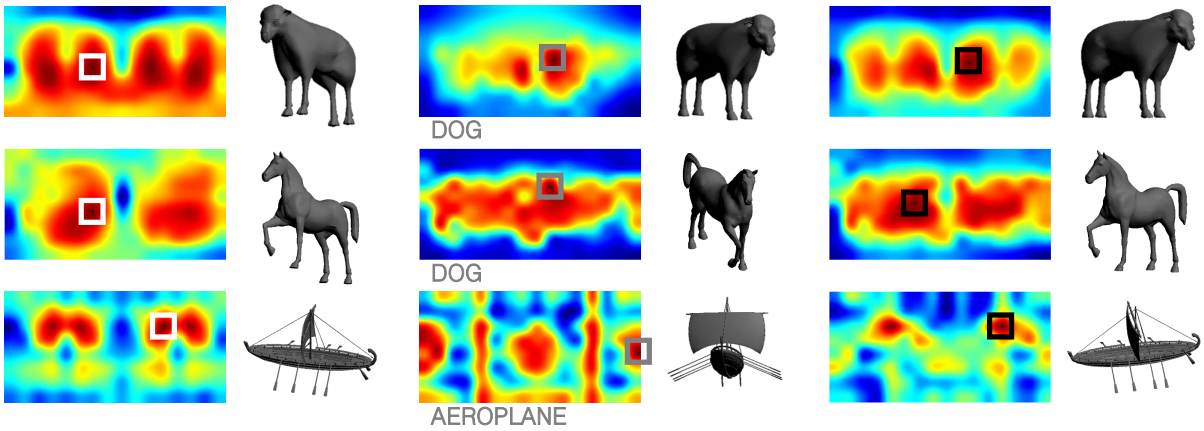

Figure 12: *Qualitative results on our estimated saliency maps and selected best views. Left: Category-independent saliency. Middle: Category-recognition saliency. The recognized object categories are shown. Right: Our final saliency. The most salient views are shown to the right of each saliency map. These are results where categories of tested objects were misclassified.*

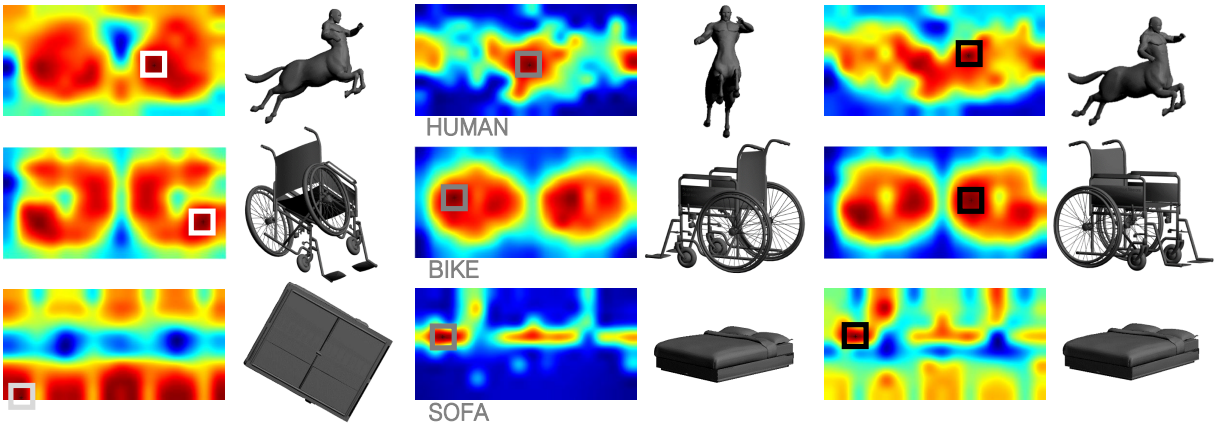

Figure 13: *Qualitative results on our estimated saliency maps and selected best views. Left: Low level saliency. Middle: Recognition saliency. The recognized object categories are shown. Right: Our final saliency. The most salient views are shown to the right of each saliency map. These are results where categories of tested objects do not belong to the 20 PASCAL categories.*

model was classified to the DOG category due to a similarity in appearance, and the selected salient view is acceptable, as illustrated in the second row of Fig. 12.

Although it was challenging for this data-driven framework, we also tested our algorithm to see how it would handle non-learned categories. Figure 13 shows the three unseen objects. The Kentauros in the first row shows the appearance of a person and horse simultaneously. In this case, the CNN evaluation scores were strong for the HUMAN and HORSE categories. After considering spatial smoothness using MRF, a more consistent category over various viewpoints was determined as HUMAN. Thus, the saliency of the HUMAN category influenced the final result, while it preserved low-level information as well. Even when multiple metaphors from different categories are implied in the projection, our algorithm seeks the most recognizable part of a known category from the given image database.

For another example, in the next row we observed that a wheelchair was classified as BIKE. In this case, the category-specific saliency of BIKE, which tends to emphasize its big wheels, influenced the final viewpoint. For the same reason, if the wheelchair model was classified as CHAIR, it would have shown familiar parts of chairs while hiding the wheels instead. We expect that either view does not precisely represent our typical experience of wheelchairs, but the other class-dependent terms and the low-level saliencies more or less prevent the biased results of the recognized category.

The bed shown in the third row of Figure 13 also does not belong to the PASCAL 20 categories, but it resembles SOFA over a certain range of views. Therefore, the best view of SOFA was determined in this case. Likewise, even though a test object was not included in our 20 categories, any appear-

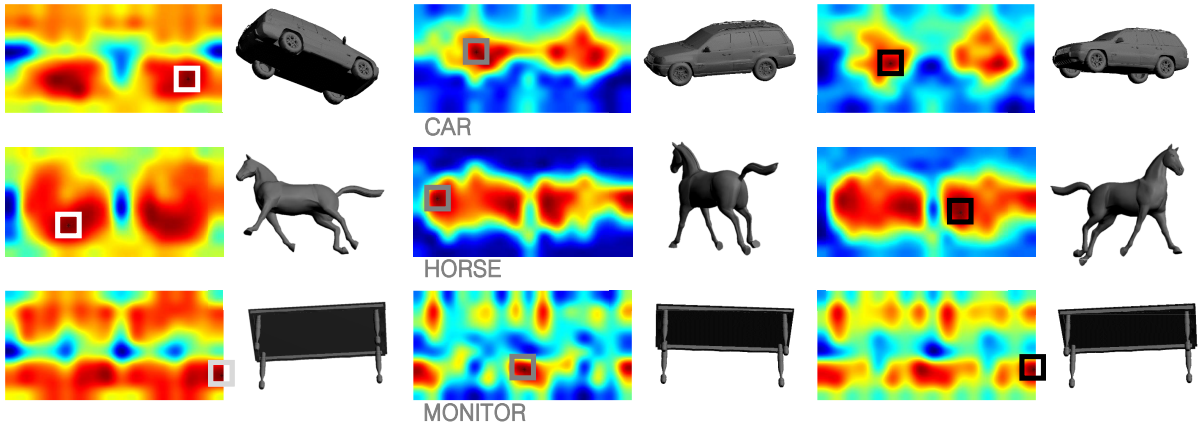

Figure 14: *Qualitative results on our estimated saliency maps and selected best views. Left: Low level saliency. Middle: Recognition saliency. The recognized object categories are shown. Right: Our final saliency. The most salient views are shown to the right of each saliency map. These are some failure examples from our algorithm.*

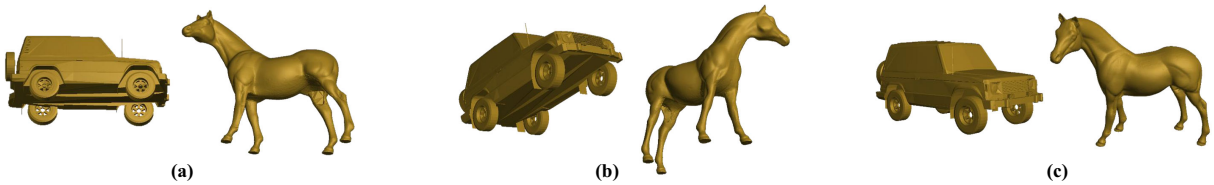

Figure 15: *Qualitative comparison with [LST12, SLF*11]. (a) 3D surface saliency computed by [LST12]. (b) Viewpoint descriptor with the linear weight as in [SLF*11]. (c) Our data-driven approach. Two models from [LST12] are shown for evaluating our recognition saliency.*

ance that resembled one of the PASCAL categories tended to be selected as a best view. We think this is a safe scheme because it at least prevents the algorithm from selecting a totally unfamiliar view.

Meanwhile, Figure 14 discusses three of our weaknesses more explicitly. Firstly, our algorithm failed when the adaptive weights for different cues were not working in an appropriate way. In the first row of Figure 14, the final result was worse than the result using only the CNN activations, while checking the bottom of a car's complex structure. Secondly, CNN tended to show the most frequent view in the training database, which perhaps not all people favor. In the current database, the HORSE category does not have many examples compared to DOG or CAT, and most photos do not show the face or eyes of a horse. Based on subjective characters of experience, it may seem that some results of HORSE mis-classified as DOG actually looked better than the correctly classified one, as shown in the second row. Lastly, the rendering pipeline can be improved by using material information. This is because shape information alone was not enough to photo-realistically render the objects, as they were captured in the real-world. For example, we saw the worst results when the rendered images had some ambiguities in appearance, as shown in the third row of Figure 14.

Qualitative Comparison with [LST12, SLF*11] Our category-independent saliencies consist of the most effective measurements in [SLF*11] and 3D surface saliency inspired by [LST12]. On top of this, we took advantage of the key observation that people keep photos of familiar objects captured at maximally informative viewpoints. Here, the effect of incorporating the category-specific saliencies with the classical measurements was compared with two state-of-the-art techniques [LST12, SLF*11] individually in Figure 15.

Intuitively, we found the perceptual improvements of our data-driven approach over the classical evaluations. For example, the bottom views of CAR and HORSE involved low-level information, such as strong curvatures of surfaces, high viewpoint entropy and the size of the projected area. As seen in the previous figures, both methods in [LST12, SLF*11] tended to emphasize low-level details, while they were not perceptually important. In contrast, the view with a higher category recognition score showed more meaningful parts than the other views that did not have such semantic information. Based this qualitative comparison with two different viewpoint selection algorithms respectively, we confirmed that the category learning process helps to generate iconic viewpoints of familiar objects.

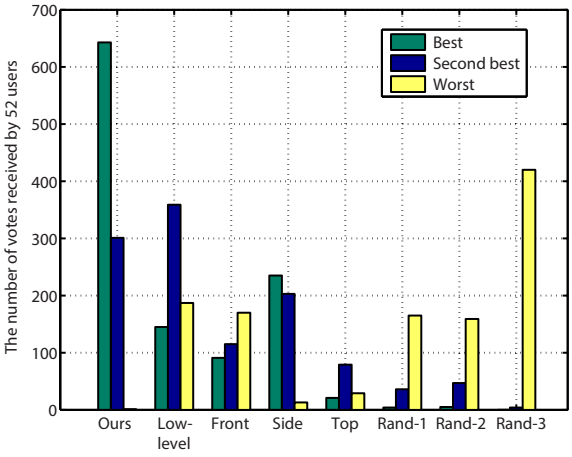

Figure 16: *User study. The 52 human subjects select the best, the second best, and the worst viewpoints among 8 images of the same object. The 8 images are generated using our method, our method with only low level evaluations, top view, front view, side view, and random projections. The bar graph shows the percentage of the received votes.*

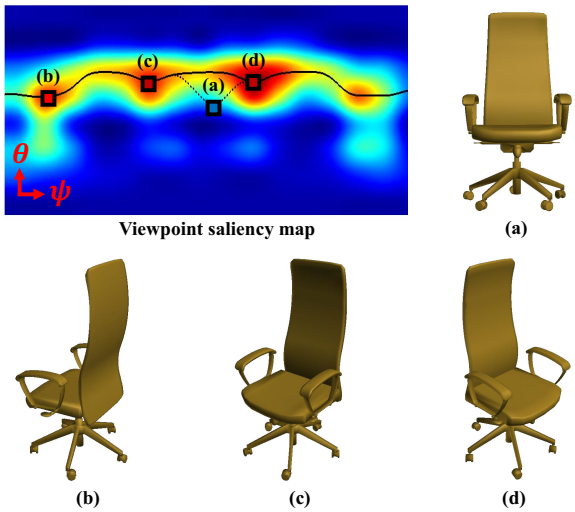

Figure 17: *Connecting salient views. (a) is a front view of a Chair. (b-d) are salient views with peak saliencies. The optimal path on the saliency map using dynamic programming (DP) is shown. Note that we can visit an important view of the model with a hard constraint (dotted line).*

User Study In order to evaluate our results quantitatively, we conducted a user study. In this experiment, 52 human subjects were asked to select the best view, the second best view, and the worst view among the 8 images of the same object. The images are from: our approach (1 image), the method using only the combined low-level saliencies (1 image), front, side, and top views (3 images), random projections with upright orientation (2 images), another random projection with no prior (1 image). For the low-level saliency, we implemented the existing methods in [SLF*11, LST12] and used the same weights of low-level saliencies as written in [SLF*11].

In the user study, they received, in total, 20 objects from each of PASCAL categories selected from our 100 test objects, and two non-learned objects were additionally given. For each object, the candidate views from eight different methods were shown without its category name. The views were listed in the same order to all subjects. All participants performed the subject test independently without communication. Figure 16 shows the bar graph of the voted plots.

For the best view selection, our approach got the largest number of votes while the side view got the second largest number of votes. For the second best view selection, the results from only low-level evaluations received the most votes. Interestingly, other viewpoints such as front, side, top and even random projections were often chosen. Based on this study, we observed human subjects first select the iconic views which helps to easily recognize its object category. Once identifying the category, people tend to pick discriminative viewpoints which gives more details about an object.

For example, the side view of BIKE or HORSE got strong supports for the first selection from users. The top, front and other projections got the positive votes in the second selection because those views are also informative for better understanding its unique appearances. The random projections received most votes for the worst view. This is obvious especially for some projected views in upside-down because they are difficult to recognize an object category in a short time.

Showcasing 3D Models Based on our saliency map (θ, ψ), we defined good viewpoints as the positions in the local maximums. The peak points can be estimated using expectation-maximization (EM) approaches. To connect these good viewpoints for briefly showcasing the object around its up-direction axis, we applied dynamic programming for its continuous path. Along the optimal path as shown in Figure 17, the category of the object is easily recognized from these familiar viewpoints by users. For post-processing, we took a simple smoothing filter for a more stable preview.

In the cost map of this optimization, we can also give additional positive and negative constraints for forcing it to visit particular positions or vice versa. For example, we can add a hard constraint to one of the classical layout views, such as a front view, so that our preview must include it.

Limitations The focus of our work is to utilize category learning in deciding the most representative, discriminative renderings of 3D models. Hence, one limitation of this approach involves the category-specific information. Although the CNN architecture we used for image feature extraction was originally designed to handle 1,000 object categories, the recognition saliency map for one specific category re-

quired a rich variety of annotated 2D photos. At the moment, it was difficult to collect an adequate number of well-cropped, labelled photos. Also, there were some ambiguities between similar categories due to the overlapping object appearances.

In this situation, it was relatively easier to collect thousands of photos together with ground-truth bounding boxes of 20 classical categories and to show the effectiveness of this approach using the PASCAL categories. Since a large number of labelled datasets are accumulating annually, we believe more discriminative sub-categorization will become available as well.

For estimating upright orientation, we can gain benefits from already up-corrected 3D models. Our 3D data collection covers various shape instances; however, the lack of quantity and quality of training data, such as non-rigid models with rare or special poses, might lead to ambiguities in finding upfront orientation. This suggests a future work where the performances of these tasks can be improved by increasing the size of the databases even more. One may expect to overcome some heuristics involved in front-view estimation because there are noticeable errors in generalizing our method for natural objects. This is also due to the lack of upfront 3D models of good quality.

The scope of this paper is limited to finding the best viewing angles, so the rest of the rendering variables are not related to the object recognition. For this reason, our current system does not seriously consider the texture information of 3D models. However, the photo-realistic renderings are currently overcoming the domain differences between virtual projections and real-world photos, and will possibly improve our recognition accuracy in the image domain; therefore, not just viewing parameters in determining human preferences, but we also suggest all the rendering parameters contributing a good view should be considered together in the context of category recognition. We believe this is another area that has room for improvement in future work.

7. Conclusion

We presented a novel framework to determine natural orientations and salient views of 3D objects. Unlike conventional approaches, our method learns the high-level semantic features and utilizes them to reflect human preferences of salient viewpoints. To pursue a scalable solution, we used a data-driven approach with two different CNN architectures for the semantic feature learning with 2D and 3D data.

Using these methods, we reached at the reasonable performance for the upright correction in the first phase and the front vector estimation in the second phase of our algorithm. In this paper, it was also shown that the high-level saliency favors natural appearances, frequently observed in our visual experiences, of real-world objects. Based on the key findings, we proposed class dependent terms and the optimal weight balance between low-level and high-level saliencies to prevent the biased view decision.

We qualitatively validated the presented algorithm using 100 objects in PASCAL categories, and quantitatively confirmed the benefits in the user study. In addition, We demonstrated possible applications such as thumbnail icons and attractive previews of 3D models.

Acknowledgement

We are grateful to anonymous reviewers for their constructive comments. This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) and Korea Creative Content Agency(KOCCA) grant funded by the Korea government(MCST) (R0132-15-1006, Developing the technology of open composable content editors for realistic media).

References

[BRLB14] BOGO F., ROMERO J., LOPER M., BLACK M.: Faust: Dataset and evaluation for 3d mesh registration. In *Computer Vision and Pattern Recognition (CVPR) 2014 IEEE Conference on* (2014), IEEE, pp. 3794–3801. 9

[BTBV99] BLANZ V., TARR M., BÜLTHOFF H., VETTER T.: What object attributes determine canonical views? *Perception-London 28*, 5 (1999), 575–600. 2, 8

[CGF09] CHEN X., GOLOVINSKIY A., FUNKHOUSER T.: A benchmark for 3d mesh segmentation. *ACM Transactions on Graphics 28*, 3 (July 2009), 73:1–73:12. 3

[CPCP15] CHEN Y.-C., PATEL V., CHELLAPPA R., PHILLIPS P.: Salient views and view-dependent dictionaries for object recognition. *Pattern Recognition* (2015). 3, 4

[CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3d model retrieval. In *Computer graphics forum* (2003), vol. 22, Wiley Online Library, pp. 223–232. 1, 4, 9

[DDS*09] DENG J., DONG W., SOCHER R., LI L.-J., LI K., FEI-FEI L.: Imagenet: A large-scale hierarchical image database. In *CVPR* (2009), pp. 248–255. 4, 7, 10

[ea92] ET AL S. E.: Orientation dependence in the recognition of familiar and novel views of three-dimensional objects. *Vision research 32*, 12 (1992), 2385–2400. 2, 3

[EVGW*] EVERINGHAM M., VAN GOOL L., WILLIAMS C. K. I., WINN J., ZISSERMAN A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. 4, 7, 9, 10

[FCODS08] FU H., COHEN-OR D., DROR G., SHEFFER A.: Upright orientation of man-made objects. In *ACM Transactions on Graphics* (2008), vol. 27, p. 42. 1, 2, 3, 6, 10

[FRS*12] FISHER M., RITCHIE D., SAVVA M., FUNKHOUSER T., HANRAHAN P.: Example-based synthesis of 3d object arrangements. *ACM Transactions on Graphics (TOG) 31*, 6 (2012), 135. 1

[GDDM14] GIRSHICK R., DONAHUE J., DARRELL T., MALIK J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR* (2014), pp. 580–587. 3

[HCX*13] HU S.-M., CHEN T., XU K., CHENG M.-M., MARTIN R. R.: Internet visual media processing: a survey with graphics and vision applications. *The Visual Computer 29*, 5 (2013), 393–405. 3

[HO05] HALL P., OWEN M.: Simple canonical views. In *BMVC* (2005). 3

[HS97] HOFFMAN D., SINGH M.: Salience of visual parts. *Cognition 63*, 1 (1997), 29–78. 2, 6

[HZRS15] HE K., ZHANG X., REN S., SUN J.: Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385* (2015). 3, 7

[KH11] KRIZHEVSKY A., HINTON G. E.: Using very deep autoencoders for content-based image retrieval. In *ESANN* (2011), Citeseer. 3, 5

[KSH12] KRIZHEVSKY A., SUTSKEVER I., HINTON G. E.: Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105. 2, 3, 5, 7

[Lag11] LAGA H.: Data-driven approach for automatic orientation of 3d shapes. *The Visual Computer 27*, 11 (2011), 977–989. 3

[LMB*14] LIN T.-Y., MAIRE M., BELONGIE S., HAYS J., PERONA P., RAMANAN D., DOLLÁR P., ZITNICK C. L.: Microsoft coco: Common objects in context. In *ECCV*. Springer, 2014, pp. 740–755. 4, 7, 10

[LPT13] LIM J. J., PIRSIAVASH H., TORRALBA A.: Parsing ikea objects: Fine pose estimation. In *Computer Vision (ICCV), 2013 IEEE International Conference on* (2013), IEEE, pp. 2992–2999. 3

[LSN*14] LIENHARD S., SPECHT M., NEUBERT B., PAULY M., MÜLLER P.: Thumbnail galleries for procedural models. In *Computer Graphics Forum* (2014), vol. 33, Wiley Online Library, pp. 361–370. 1, 3

[LST12] LEIFMAN G., SHTROM E., TAL A.: Surface regions of interest for viewpoint selection. In *CVPR* (2012), pp. 414–421. 1, 2, 6, 10, 13, 14

[LVJ05] LEE C., VARSHNEY A., JACOBS D.: Mesh saliency. In *ACM Transactions on Graphics* (2005), vol. 24, pp. 659–666. 2, 6

[LZH12] LIU H., ZHANG L., HUANG H.: Web-image driven best views of 3d shapes. *The Visual Computer 28*, 3 (2012), 279–287. 1, 3

[MS09] MORTARA M., SPAGNUOLO M.: Semantics-driven best view of 3d shapes. *Computers & Graphics 33*, 3 (2009), 280–290. 1, 3

[MW12] MEZUMAN E., WEISS Y.: Learning about canonical views from internet image collections. In *Advances in Neural Information Processing Systems* (2012), pp. 719–727. 3

[PB96] PLEMENOS D., BENAYADA M.: Intelligent display in scene modeling: New techniques to automatically compute good views. In *Proceedings of GraphiCon* (July, 1996). 2, 6

[PPB*05] POLONSKY O., PATANÉ G., BIASOTTI S., GOTSMAN C., SPAGNUOLO M.: What's in an image? *The Visual Computer 21*, 8-10 (2005), 840–847. 1, 3

[PRC81] PALMER S., ROSCH E., CHASE P.: Canonical perspective and the perception of objects. *Attention and performance IX 1*, 4 (1981). 2, 3, 7

[PSG*06] PODOLAK J., SHILANE P., GOLOVINSKIY A., RUSINKIEWICZ S., FUNKHOUSER T.: A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics 25*, 3 (2006). 3

[QSN*16] QI C. R., SU H., NIESSNER M., DAI A., YAN M., GUIBAS L. J.: Volumetric and multi-view cnns for object classification on 3d data. *arXiv preprint arXiv:1604.03265* (2016). 3, 5

[SLF*11] SECORD A., LU J., FINKELSTEIN A., SINGH M., NEALEN A.: Perceptual models of viewpoint preference. *ACM Transactions on Graphics 30*, 5 (Oct. 2011). 1, 2, 3, 8, 9, 10, 13, 14

[SLJ*15] SZEGEDY C., LIU W., JIA Y., SERMANET P., REED S., ANGUELOV D., ERHAN D., VANHOUCKE V., RABINOVICH A.: Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9. 7

[SLT13] SHTROM E., LEIFMAN G., TAL A.: Saliency detection in large point sets. In *ICCV* (2013), pp. 3591–3598. 2

[SM71] SHEPARD R., METZLER J.: Mental rotation of three-dimensional objects. *Science 171*, 972 (1971), 701–703. 3

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape modeling applications, 2004. Proceedings* (2004), IEEE, pp. 167–178. 1

[SMKLM15] SU H., MAJI S., KALOGERAKIS E., LEARNED-MILLER E.: Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 945–953. 3, 5

[SP04] SUMNER R., POPOVIC J.: Deformation transfer for triangle meshes. *ACM Transactions on Graphics (TOG) 23*, 3 (2004), 399–405. 9

[SQLG15] SU H., QI C. R., LI Y., GUIBAS L.: Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. *arXiv preprint arXiv:1505.05641* (2015). 3

[SSB13] SERIN E., SUMENGEN S., BALCISOY S.: Representational image generation for 3d objects. *The Visual Computer 29*, 6-8 (2013), 675–684. 1

[SZ14] SIMONYAN K., ZISSERMAN A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014). 7

[SZ15] SIMONYAN K., ZISSERMAN A.: Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations* (2015). 3

[TK01] TARR M., KRIEGMAN D.: What defines a view? *Vision Research 41*, 15 (2001), 1981–2004. 3

[TP89] TARR M. J., PINKER S.: Mental rotation and orientation-dependence in shape recognition. *Cognitive psychology 21*, 2 (1989), 233–282. 2, 3

[VFSH01] VÁZQUEZ P.-P., FEIXAS M., SBERT M., HEIDRICH W.: Viewpoint selection using viewpoint entropy. In *Proceedings of the Vision Modeling and Visualization Conference* (2001), vol. 1, pp. 273–280. 2, 6

[WSK*15] WU Z., SONG S., KHOSLA A., YU F., ZHANG L., TANG X., XIAO J.: 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1912–1920. 1, 2, 3, 5, 9

[XMS14] XIANG Y., MOTTAGHI R., SAVARESE S.: Beyond pascal: A benchmark for 3d object detection in the wild. In *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on* (2014), IEEE, pp. 75–82. 9

[YAMK15] YUMER M. E., ASENTE P., MECH R., KARA L. B.: Procedural modeling using autoencoder networks. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)* (2015), ACM. 4

[ZLJW15] ZHAO L., LIANG S., JIA J., WEI Y.: Learning best views of 3d shapes from sketch contour. *The Visual Computer* (2015), 1–10. 1, 3