Pacific Graphics (2016)
E. Grinspun, B. Bickel, and Y. Dobashi (Editors)



# Viewpoint Selection for Taking a good Photograph of Architecture†

Jingwu He[1], Wenzhe Zhou[1], Linbo Wang[2], Hongjie Zhang[1], and Yanwen Guo[1‡]

[1]National Key Lab for Novel Software Technology, Nanjing University, NanJing, 210023, China
[2]MOE Key Laboratory of Intelligent Computing and Signal Processing, Institute of Media Computing, Anhui University, HeFei, 230039, China

**Abstract**

*This paper studies the problem of how to choose the viewpoint for taking good photographs for architecture. We achieve this by learning from professional photographs of world famous landmarks that are available in the Internet. Unlike the previous efforts devoted to photo quality assessment which mainly rely on visual features, we show in this paper combining visual features with geometric features computed on the 3D models can result in a more reliable evaluation of viewpoint quality. Specifically, we collect a set of photographs for each of 6 world famous architectures as well as their 3D models from Internet. Viewpoint recovery for images is carried out by an image-model registration process, after which a newly proposed viewpoint clustering strategy is exploited to validate users' viewpoint preference when photographing landmarks. Finally, we extract a number of 2D and 3D features for each image based on multiple visual and geometric cues, and perform viewpoint recommendation by learning from both 2D and 3D features, achieving superior performance over using solely 2D or 3D features. We show the effectiveness of the proposed approach through extensive experiments.*

Categories and Subject Descriptors (according to ACM CCS): I.3.m [Computer Graphics]: Computational photography—

## 1. Introduction

Modern digital cameras and smart-phones enable ordinary people to take and store photographs easily today. At the same time, with the rapid development of the Internet, the number of photos with architectures that can be accessed is growing explosively. For example, when people take a vacation trip, they usually want to take some photos with architectures in the background, especially some famous attractions. However, what makes for a visually-pleasing good landmark photo is probably a common question shared by novice photographers. Although people may give various answers to it, there should be no doubt that viewpoint selection plays a crucial role. However, choosing a good viewpoint when photographing is not an easy task, especially for an unsophisticated photographer. This quandary could be very much relieved if the picture-taker is provided with an useful viewpoint recommendation tool. With this in mind, we study here the relationship between viewpoint selection and the beauty of architecture photos.

Photo aesthetics assessment has been studied a lot in existing literatures. Many researchers devote their efforts on learning from visual features, including image compositional [LT08, DJLW06, GLGW12], content [DOB11, LWT11, Dic85], and simplicity [KTJ06]. Although considerable progress has been achieved, no works are specifically designed for assessing architecture photos. On the other hand, a number of works study the problem of viewpoint selection for 3D models [MPLC11, PSTV10, MS09] and are extended to various applications [LZH12, SGSS08]. Moreover, Bae et al. proposed computational rephotography that allows photographers to take modern photos that match a given historical photograph [BAD10]. Inspired by these studies, we set up a viewpoint recommendation framework by learning from both visual and geometric features of photos people rank high on the Internet.

We initiate our learning framework by collecting a set of photographs for each of 6 world famous architectures as well as their 3D models from Internet. Before stepping into the learning task, we need first recover geometric information of each image. To this end, we exploit Structure from Motion (SfM) to obtain a coarse 3D cloud model as well as a camera matrix of each photo under the model. A registration process is performed to match the coarse model with a pre-provided fine 3D model. Viewpoint of each input image regarding to the fine model is further estimated by transferring its camera matrix under the coarse one based on correspondence relationship.

To better justify the motivation behind our work of viewpoint assessment and recommendation, we further conduct studies on

† This work is supported by the National Natural Science Foundation of China under Grants 61373059, 61321491, 61672279, and 61333014, and the Natural Science Foundation of Jiangsu Province under Grant BK20150016.

‡ Corresponding author



DOI: 10.2312/pg.20161333

users' preference when photographing. To this end, we inspect the viewpoints of all the images and check whether people are accustomed to shoot at certain specific angles given a specific architecture. This is achieved by a viewpoint clustering procedure in the underlying geometric space. Considering that the viewpoint space is essentially a Riemannian manifold with the structure of Matrix Lie Group, we introduce a Riemannian metric to measure the proximity between two viewpoints. Once the metric is defined, people's viewpoint preferences are successfully verified by K-medoids clustering, which confirms our motivation.

Next, we focus our efforts on the task of viewpoint recommendation, which is resolved by learning from visual and geometric features as aforementioned. Specifically, for each of the landmark photo sets collected, we first extract multiple 2D features by quantifying various visual cues and knowledge of photo aesthetics, such as color, HOG [DT05] feature, Rule-of-Thirds, etc. A number of 3D features describing the geometry of viewpoints, e.g. project area, surface visibility, etc., are generated as well. To start the learning framework, we further conduct user study by asking the participants to rank the goodness of training photos. With all these prepared, we perform training using three learning algorithms with ten-fold cross validation, selecting the best by comparing their performance.

In addition, learning results on solely 2D or 3D features are reported. It shows less comparative performance than the learner using 2D and 3D features simultaneously, verifying the necessity of considering both visual and geometric knowledge when photographing. Finally, we conclude the learning framework by demonstrating rendered images of mostly recommended viewpoint of several 3D models.

In summary, the main contributions of this paper include:

- We propose a new framework for viewpoint analysis as well as recommendation when photographing architectures. Promising results are reported and high-quality viewpoint recommendation are demonstrated.
- Users' viewpoint preference is analyzed by a clustering process, in which an effective distance measure is introduced to describe the proximity of viewpoints. Results suggest that people tend to share consistent viewpoint preference when taking architecture photos, justifying the necessity of viewpoint recommendation.
- We extract multiple features based on various visual and geometric cues, and investigate their effectiveness in describing the quality of viewpoints. Most importantly, we propose to fuse 2D and 3D features for the task of good viewpoint learning, achieving considerable performance improvement over learning from solely 2D or 3D features.

A schematic overview of our system is given in Fig 1. The remainder of this paper is organized as follows. Viewpoint recovery and viewpoint preference analysis are introduced in Section 2. Section 3 describes the features we proposed in both visual and geometric aspects. We conduct experiments and compare with previous methods in Section 4. Section 5 shows our applications of photograph assessment and viewpoint recommendation for users, and Section 6 concludes the whole paper.

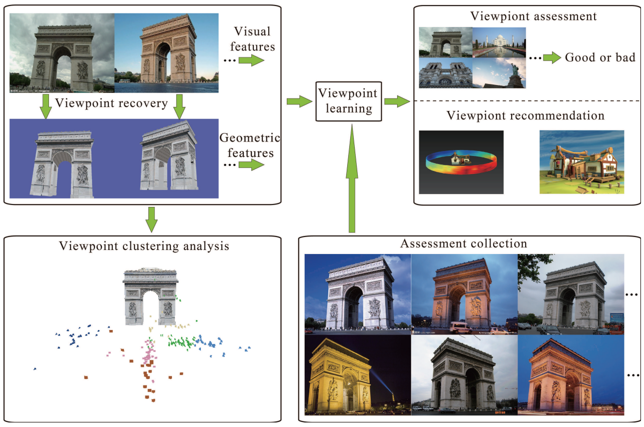


Figure 1: We first recover viewpoints for each photograph, then extract 2D and 3D features and collect preference for each image. Next, we train a classifier to make viewpoint assessment and recommendation for users.

## 2. Viewpoint Preference Analysis

In this section, we investigate people's preference of viewpoint selection when photographing. We seek to verify that people share consistent feeling of beauty if given images of specific angles of an architecture. To achieve the goal, we first perform viewpoint estimation for all the images collected. Thereafter, all the viewpoint parameters are stacked up together to form a geometric space. A robust clustering procedure is further exploited to discover dense clusters, which correspond to the viewpoints people used to shoot at most frequently. Next, we show details of viewpoint estimation and clustering in Section 2.1 and 2.2, respectively.

### 2.1. Viewpoint Estimation

To estimate the viewpoint of the input image set, we first apply Structure from Motion (SfM) [BL05, W*11] to construct a point cloud model as well as compute the viewpoint of each given photo. Although promising progress has been achieved in utilizing SfM [SSS06], the obtained model is generally so coarse that it is difficult to directly extract valid geometric features for the subsequent learning task. We thus match the coarse model with a provided fine model and transfer the viewpoint estimated to effectively obtain viewpoint knowledge under the fine model.

- **Model Registration.** After applying SfM algorithm on a set of photos of a specific architecture model, we will get a point cloud model as well as each photo's camera matrix under it. But the point cloud model is not suitable for geometry processing. It is necessary to recover camera matrices under the 3D surface mesh model.
  Ideally, the point cloud model and 3D surface mesh model can coincide after only scale, rotation and translation transformations. Considering $\boldsymbol{x}$ and $\boldsymbol{y}$ are the corresponding points on 3D mesh model and point cloud model, respectively. Denoting $c$, $\mathbf{R}$, $\mathbf{t}$ as the scale, rotation and translation parameters, we have:

$$c\mathbf{R}\boldsymbol{x} + \mathbf{t} = \boldsymbol{y}. \tag{1}$$

Figure 2: Registration process between the point cloud model and 3D surface mesh model. The first and second column represents the point cloud model and the 3D surface mesh model, respectively. In the third column, we render both of them after registration. For these two examples, the transformed 3D surface mesh model coincides with the point cloud model, which means that the registration algorithm is effective.

With several corresponding points $(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})$ picked up interactively, getting all transformation parameters by solving simultaneous equations is a feasible scheme. We solve it by using an optimization-based method. The parameters $c$, $\mathbf{R}$, $\mathbf{t}$ are selected by minimizing the loss function:

$$\underset{c,\mathbf{R},\mathbf{t}}{\operatorname{argmin}} \sum_{k=1}^{n} \|\boldsymbol{y}^{(k)} - (c\mathbf{R}\boldsymbol{x}^{(k)} + \mathbf{t})\|_2. \tag{2}$$

Figure 2 shows the registration process between point cloud and 3D mesh model, seeing that the model registration algorithm is simple and efficient.

- **Transformation Transferring.** So far we have known the transformations between these two models and every photo's camera matrix under the point cloud model. Supposing that $\mathbf{R}'$ and $\mathbf{t}'$ are one photo's external camera parameters under the point cloud model, $\boldsymbol{z}$ is a point $\boldsymbol{P}$'s position under camera coordinate system, try to transfer these relationships:

$$\boldsymbol{z} = \mathbf{R}'\boldsymbol{y} + \mathbf{t}', \tag{3}$$

where $\boldsymbol{y}$ is point $\boldsymbol{P}$'s position under point cloud model coordinate system. Substitute Eq.(1) in Eq.(3) to obtain:

$$\boldsymbol{z} = c\mathbf{R}'\mathbf{R}\boldsymbol{x} + (\mathbf{R}'\mathbf{t} + \mathbf{t}'). \tag{4}$$

The coordinate transformation from a point in 3D mesh model coordinate system ($\boldsymbol{x}$) to a point in camera coordinate system as Eq.(4), which means the photo's rotation and translation parameters are $c\mathbf{R}'\mathbf{R}$ and $(\mathbf{R}'\mathbf{t} + \mathbf{t}')$, respectively.

### 2.2. Viewpoint Clustering for Preference Analysis

Given a set of photos, we estimate their viewpoints regarding to the provided 3D model as described above. Now, we study whether people tend to shoot at some specific angles when standing in front of an architecture. We achieve this by performing a clustering procedure in the underlying geometric space.

A photo of an architecture may be regarded as a visual experience by looking at the model of the architecture from a specific angle and holding on at a certain point. More formally, it is to project the model of the architecture with a model-view matrix and a projection matrix. As the two matrices together determine the content of a photo, the model-view matrix records various parameters about camera, including position, orientation, etc., and plausibly dominate the viewers' sense of viewpoint for a model. The projection matrix, on the other hand, contains information about the projection plane, imposing more influence on the size of the architecture presented in the photo. Therefore, we focus on model-view matrices recovered from the given photo set for analysis of viewpoint preference.

We stack up all the model-view matrices together to form a viewpoint space. Notice that this is in essential not a vector space, in which Euclidean distance metric is embedded. Moreover, it is one kind of matrix Lie groups, equipped with the structure of analytic Riemannian manifold. Unlike existing approaches relying on heuristic metric, e.g. homography overlap distance [WL11, WL13], to describe the viewpoint similarity among photos, we introduce a Riemannian metric that specifically designed for matrix Lie groups [Ros02] to define the distance between two model-view matrices, and thus better measures the viewpoint proximity of two photos. More formally, given two model-view matrices $\mathbf{x}$ and $\mathbf{y}$, it is defined as

$$d(\mathbf{x}, \mathbf{y}) = \|log(\mathbf{x}^{-1}\mathbf{y})\|_F, \tag{5}$$

where $\|.\|_F$ denotes the Frobenius norm of a matrix.

With distance metric defined above, we perform K-medoids clustering for viewpoint preference analysis. More sophisticated clustering algorithm, such as Mean-shift clustering, can be applied here. We choose the classical K-medoids algorithm with $K = 15$, considering its simplicity and being easy to implement. Moreover, it suffices to show that people often share consistent viewpoint preferences when photographing architecture in our experiments, as illustrated by Figure 3. This is in compliance with our common knowledge and justifies the necessity for viewpoint recommendation.

## 3. Proposed Features

In this section, we introduce the 2D visual and 3D geometric features extracted for the learning task. We describe 2D features first and 3D features next.

**Visual features.** We extract various image features concerning image appearance, local structure, composition, etc. More specifically, it includes:

- $\mathbf{v}_{HOG}$: *HOG (Histogram of oriented gradient).* HOG is used to describe the shape context of architecture for different viewpoints.
- $\mathbf{v}_{color}$: *Color entropy and distribution.* Color features are defined by making a statistics of every pixel's RGB value in a photo. To reduce noise and minimize computation, we quantize the red,

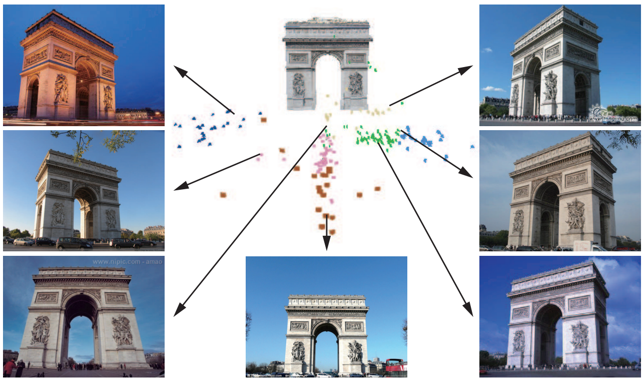

Figure 3: Result of our k-medoids clustering with the distance defined on Lie groups. Besides, some represented clustering photos were showed.

green and blue channels into 8 values. Then we can get a histogram with $512 = 8^3$ bins. With this histogram we can make two kinds of computation: Entropy of histogram and Distribution of histogram:

$$\begin{aligned} c_e &= \sum P_i \log P_i, \quad \text{where } P_i = \frac{\mathbf{H}(i)}{\sum \mathbf{H}(i)}, \\ c_d &= 1 - \sum P_i^2, \quad \text{where } P_i = \frac{\mathbf{H}(i)}{\sum \mathbf{H}(i)}, \end{aligned} \tag{6}$$

where $H$ is the histogram, and is normalized to unit length, $c_e$ and $c_d$ represent color entropy and distribution, respectively. These features try to identify the difference in the color palette between different viewpoints.

- $\mathbf{v}_{hue}$: *Hue count.* We calculate the hue count of an image as follows. Color images are converted to its HSV representation. We only consider pixels with brightness values in the range $[0.15, 0.95]$ and saturation $s > 0.2$. A histogram with 20 bins is computed on the good hue values. Let $m$ be the maximum value of the histogram:

$$\begin{aligned} m &= \max_i \mathbf{H}(i), \\ N &= \{i \mid \mathbf{H}(i) > \alpha m\}. \end{aligned} \tag{7}$$

In this equation, $\alpha = 0.05$ controls the noise sensitivity of the hue count:

$$\mathbf{v}_{hue} = 20 - \|N\|. \tag{8}$$

The smaller the value is, the more colorful the photo of viewpoint becomes.

- $\mathbf{v}_{rule}$: *Rule of Thirds.* This feature is applied to optimization of composition work [LCWCO10, ZWH13], and gets a pretty result. We refer the readers to [LT08] for details. Rule of Thirds is used as a guideline for salient regions' positions to take a good photo.
- $\mathbf{v}_{bright}, \mathbf{v}_{contrast}$: *Brightness and Contrast.* Brightness and Contrast are low level features [KTJ06]. Besides, professionals recognize that contrast and brightness could be quite different between different viewpoints.

**Geometric features.** There are lots of features about viewpoint selection for 3D model. We extract them including: area, silhouette, depth, surface curvature, angles and some empirical rules.

- $\mathbf{g}_{area}, \mathbf{g}_{surface}$: *Project area and Surface visibility.* Project area computes the projected area of the model in the image plane as a fraction of the overall image area. Surface visibility is defined as the ratio of visible surface area in a particular view to the total surface area of an object [PB96]. They are related to the area of the shape as seen from a particular viewpoint.
- $\mathbf{g}_{sl}, \mathbf{g}_{sc}$, and $\mathbf{g}_{sce}$: *Silhouette length, Silhouette curvature and Silhouette curvature extrema.* Silhouette length defines the overall length of the object's silhouette in the image plane. Silhouette curvature is introduced as an attribute and generate significant information to the viewer [Fel05, VBP*09]. Moreover, silhouette attributes are believed to be the first index into the human memory of shapes.
- $\mathbf{g}_{dm}, \mathbf{g}_{dd}$: *Depth max and Depth distribution.* $\mathbf{g}_{dm}$ is defined as the maximum depth value of any visible point of the shape. Depth distribution is introduced to encourage a broad, even distribution of depths in the scene [SLF*11]. Besides, depth features can help avoid degenerate viewpoints.
- $\mathbf{g}_{mc}$ *and* $\mathbf{g}_{gc}$: *Mean curvature and Gaussian curvature.* Surface curvature is assumed related to the shape's semantic features.
- $\mathbf{g}_{ap}$: *Above preference.* People tend to prefer views from slightly above the horizon [BTB99], thus this attribute is computed to value the preference of viewpoint.

$$\mathbf{g}_{ap} = \mathcal{G}(\phi; \frac{3\pi}{8}, \frac{\pi}{4}), \quad \mathcal{G}(x,\mu,\sigma) = e^{\frac{-(x-\mu)^2}{\sigma^2}}, \tag{9}$$

where $\phi$ is the latitude with 0 at the north pole and $\pi/2$ at the equator, and $\mathcal{G}$ is the non-normalized Gaussian function $exp(-(x-\mu)^2/\sigma^2)$.

- $\mathbf{g}_{outer}$: *Outer points.* When we take a photo of a famous scene, we always want to take a picture including the whole architecture.

$$\mathbf{g}_{outer} = \frac{\sum_{i \in \text{Outer}}}{\sum_{i \in \text{all}}}, \tag{10}$$

where $i$ is the point on 3D model. It computes the ratio of points out of photo and all points in 3D mesh model.

- $\mathbf{g}_{angles}$: *Axes angles.* Typical computer graphics models generated by CAD or acquisition processes do indeed have a stored model axis $x, y$ and $z$. For a given viewpoint, after applying Model-View transformation on these three axes, we can get $x_m, y_m$ and $z_m$ in camera coordinate system. $\mathbf{g}_{angles}$ is a 9-dimension vector defined as:

$$\mathbf{g}_{angles} = \angle(\mathbf{a}, \mathbf{b}), \tag{11}$$

where $\mathbf{a} \in \{x_m, y_m, z_m\}$ ,and $\mathbf{b} \in \{x_c, y_c, z_c\}$, $\{x_c, y_c, z_c\}$ are camera's axes under camera coordinate system.

- $\mathbf{g}_{ball}$: *Ball coordinate.* In a traditional 3D scene, camera's position will also affect user's preference. We take the camera's ball coordinate as camera's position in the architecture's model coordinate system described as $[r, \theta, \phi]$, and we take $\mathbf{g}_{ball} = [\theta, \phi]$ as features for a given viewpoint.

## 4. Experiments

We perform the comparative experiments on a set of photographs taken from each of 6 world famous architectures as well as their

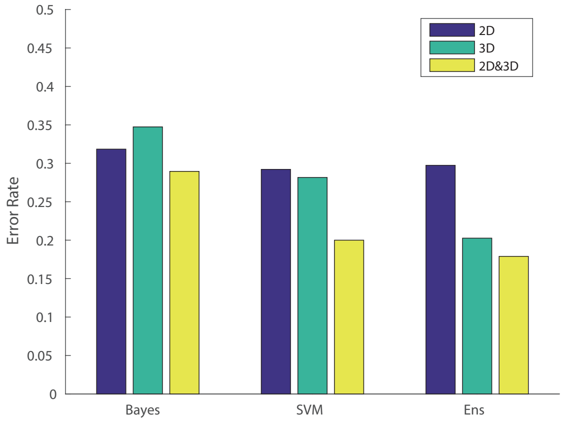


Figure 4: Performance of different classifiers on photos. As we can see, the result of combining 2D and 3D features performs better.

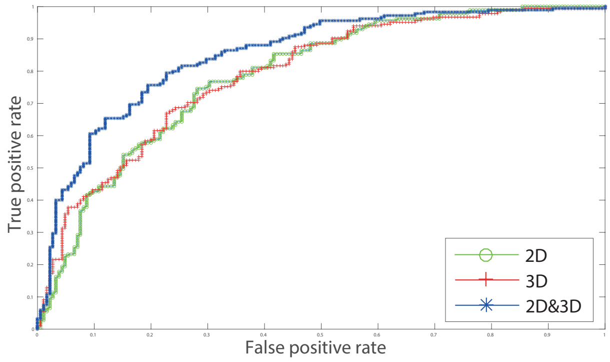


Figure 5: Comparison of SVM classifier performance with different type of features. Combining visual features and 3D geometry features under photo's viewpoint improves the classifier's performance efficiently.

3D models. Totally, we collect 2371 photos from Internet. We also perform user studies on the Amazon Mechanical Turk (AMT) by asking the users to score each photo from 1 to 5, where 1 and 5 indicates the worst and best viewpoint of corresponding landmarks respectively. Each photo is scored 20 times by different users to avoid scoring bias, with the average score defining the viewpoint goodness of a given photo. To conduct training more effectively, we further rule out some of the photos that have conflict scores (with most of the users' score 1 or 5 and average score around 2.5) and thus encourage the trainer to learn from photos with unanimous score.

We choose three different learners, namely Bayes classifier, SVM classifier with RBF kernel and Ensemble learning with random forest containing 80 trees, to train the learning models and compare their performance. All the learners are trained with ten-fold cross-validation. We train each learner on solely 2D visual features and 3D geometric features as benchmarks and compare them against the learners fed with 2D-3D mixed features, which is obtained by concatenating the visual and geometric feature vectors of each photo.

Overall performance comparison of all three learners is reported in Figure 4. Basically, all learners trained on 2D features achieved comparable performance while the SVM and Ensemble learner obtain smaller error rates against Bayes learner when dealing with 3D and 2D-3D mixed features. Between the former two learners, Ensemble learner outperforms SVM with a very small margin. More importantly, learners trained with both visual and geometric features achieve consistent superior performance over that trained with either 2D or 3D features, no matter what type of classifier is used. Figure 5 shows the performance of applying SVM on combined features. We can see that using features combined with 2D and 3D will surely improve the performance of the classifier. Figure 6 presents several exemplar photos with good and bad viewpoints evaluated by the Ensemble learner with mixed features, showing the effectiveness of the proposed approach.

## 5. Applications

Given an architecture model and its corresponding photographs, our system can utilize both visual features and geometric features to make viewpoint assessment and recommendation for users. Moreover, our system can also make viewpoint evaluation if given a single architecture model. To achieve this, we uniformly sample 1024 viewpoints in a limited range around the input model and render images of it from each viewpoint. The sample range starts from the ground to a certain height level, covering the viewpoints people likely to shoot at. We then evaluate the viewpoint goodness of each rendered image with the Ensemble learner. We plot the heat map of recommended viewpoints for an input model as well as two exemplar highly recommended viewpoint images in Figure 7. As illustrated, the right-front side dominates the best viewpoints of the little house model.

Figure 7 shows the application of automatic viewpoint recommendation for a virtual 3D scene.

## 6. Conclusion, Limitations and Future Work

In this paper, we propose to combine both visual features and geometric features to make viewpoint assessment and recommendation for users to take a good photograph of architecture, which leads to superior performance against learning from solely 2D or 3D features. Besides, we also conduct studies on users' preference when photographing to verify people's viewpoint preference when photographing.

**Limitations and Future Work.** In the future, we plan to improve our approach in the following aspects. First, our current system requires users to provide a fine 3D model for an input architecture, which may limit the application of the proposed framework. Second, since not all the handcrafted features are as effective as expected, we seek to try more sophisticated feature extraction strategy, e.g., deep learning. Finally, we may also exploit Label Distribution Learning (LDL) to give users more assistance for viewpoint recommendation.

Figure 6: Exemplar photos with good (top row) and bad (bottom row) viewpoint evaluated by our Ensemble learner with 2D-3D mixed features.

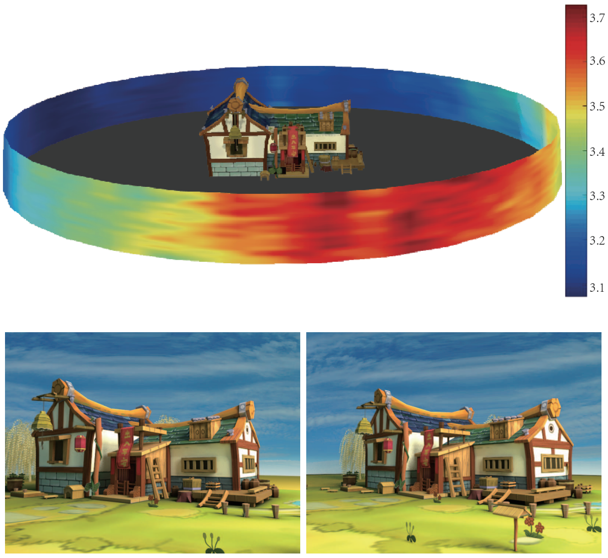


Figure 7: The heat map of our recommended viewpoints for a given model, as well as two recommended viewpoints in the second row.

## References

[BAD10] BAE S., AGARWALA A., DURAND F.: Computational rephotography. *ACM TOG 29*, 3 (2010), 24. 1

[BL05] BROWN M., LOWE D. G.: Unsupervised 3d object recognition and reconstruction in unordered datasets. In *Fifth International Conference on 3-D Digital Imaging and Modeling (3DIM'05)* (June 2005), pp. 56–63. 2

[BTB99] BLANZ V., TARR M. J., BÃIJLTHOFF H. H.: What object attributes determine canonical views? *Perception 28*, 5 (1999), 575–599. 4

[Dic85] DICHTER E.: What's in an image. *Journal of consumer marketing 2*, 1 (1985), 75–81. 1

[DJLW06] DATTA R., JOSHI D., LI J., WANG J. Z.: *Studying Aesthetics in Photographic Images Using a Computational Approach*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 288–301. 1

[DOB11] DHAR S., ORDONEZ V., BERG T. L.: High level describable attributes for predicting aesthetics and interestingness. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (June 2011), pp. 1657–1664. 1

[DT05] DALAL N., TRIGGS B.: Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (June 2005), vol. 1, pp. 886–893 vol. 1. 2

[Fel05] FELDMAN JACOB; SINGH M.: Information along contours and object boundaries. *Psychological Review 112*, 1 (2005), 243–252. 4

[GLGW12] GUO Y., LIU M., GU T., WANG W.: Improving photo composition elegantly: Considering image similarity during composition optimization. In *Computer Graphics Forum* (2012), vol. 31, Wiley Online Library, pp. 2193–2202. 1

[KTJ06] KE Y., TANG X., JING F.: The design of high-level features for photo quality assessment. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (June 2006), vol. 1, pp. 419–426. 1, 4

[LCWCO10] LIU L., CHEN R., WOLF L., COHEN-OR D.: Optimizing photo composition. In *Computer Graphics Forum* (2010), vol. 29, Wiley Online Library, pp. 469–478. 4

[LT08] LUO Y., TANG X.: *Photo and Video Quality Evaluation: Focusing on the Subject*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 386–399. 1, 4

[LWT11] LUO W., WANG X., TANG X.: Content-based photo quality assessment. In *2011 International Conference on Computer Vision* (Nov 2011), pp. 2206–2213. 1

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

[MPLC11] MARCHESOTTI L., PERRONNIN F., LARLUS D., CSURKA G.: Assessing the aesthetic quality of photographs using generic image descriptors. In *2011 International Conference on Computer Vision* (Nov 2011), pp. 1784–1791. 1

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

[PB96] PLEMENOS D., BENAYADA M.: Intelligent display in scene modeling. new techniques to automatically compute good views. In *Int. Conf. on Computer Graphics & Vision* (1996). 4

[PSTV10] PRATIKAKIS I., SPAGNUOLO M., THEOHARIS T., VELTKAMP R.: Semantics-driven approach for automatic selection of best views of 3d shapes. 1

[Ros02] ROSSMANN W.: *Lie groups: an introduction through linear groups*, vol. 5. Oxford University Press on Demand, 2002. 3

[SGSS08] SNAVELY N., GARG R., SEITZ S. M., SZELISKI R.: Finding paths through the world's photos. *ACM Trans. Graph. 27*, 3 (2008), 15:1–15:11. 1

[SLF*11] SECORD A., LU J., FINKELSTEIN A., SINGH M., NEALEN A.: Perceptual models of viewpoint preference. *ACM Trans. Graph. 30*, 5 (Oct. 2011), 109:1–109:12. 4

[SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo tourism: Exploring photo collections in 3d. In *ACM SIGGRAPH 2006 Papers* (New York, NY, USA, 2006), SIGGRAPH '06, ACM, pp. 835–846. 2

[VBP*09] VIEIRA T., BORDIGNON A., PEIXOTO A., TAVARES G., LOPES H., VELHO L., LEWINER T.: Learning good views through intelligent galleries. In *Computer Graphics Forum* (2009), vol. 28, Wiley Online Library, pp. 717–726. 4

[W*11] WU C., ET AL.: Visualsfm: A visual structure from motion system. 2

[WL11] WEYAND T., LEIBE B.: Discovering favorite views of popular places with iconoid shift. In *2011 International Conference on Computer Vision* (Nov 2011), pp. 1132–1139. 3

[WL13] WEYAND T., LEIBE B.: Discovering details and scene structure with hierarchical iconoid shift. In *The IEEE International Conference on Computer Vision (ICCV)* (December 2013). 3

[ZWH13] ZHANG F. L., WANG M., HU S. M.: Aesthetic image enhancement by dependence-aware object recomposition. *IEEE Transactions on Multimedia 15*, 7 (Nov 2013), 1480–1490. 4