Composing Good Shots by Exploiting Mutual Relations

Debang Li[1,2], Junge Zhang[1,2], Kaiqi Huang[1,2,3], Ming-Hsuan Yang[4,5]
[1] CRISE, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology, Beijing, China
[4] University of California, Merced [5] Google Research
{debang.li,jgzhang, kaiqi.huang}@nlpr.ia.ac.cn, mhyang@ucmerced.edu

Abstract

Finding views with a good composition from an input image is a common but challenging problem. There are usually at least dozens of candidates (regions) in an image, and how to evaluate these candidates is subjective. Most existing methods only use the feature corresponding to each candidate to evaluate the quality. However, the mutual relations between the candidates from an image play an essential role in composing a good shot due to the comparative nature of this problem. Motivated by this, we propose a graph-based module with a gated feature update to model the relations between different candidates. The candidate region features are propagated on a graph that models mutual relations between different regions for mining the useful information such that the relation features and region features are adaptively fused. We design a multi-task loss to train the model, especially, a regularization term is adopted to incorporate the prior knowledge about the relations into the graph. A data augmentation method is also developed by mixing nodes from different graphs to improve the model generalization ability. Experimental results show that the proposed model performs favorably against state-of-the-art methods, and comprehensive ablation studies demonstrate the contribution of each module and graph-based inference of the proposed method.

1. Introduction

Image composition plays a critical role in generating visually appealing shots that involves finding views from an image. In addition, finding a view is the key to numerous tasks, *e.g.*, image cropping [40, 48]. Automatically finding good views can help save a lot of time and effort for users, photographers, and designers, especially when dealing with a large number of images.

In the past decades, numerous methods [1, 4, 5, 9, 10, 18, 19, 23, 33, 34, 40, 41, 44, 45, 48, 24] have been developed for the automatic image cropping or good view recommendation. Existing methods mainly generate candidate regions at the first stage and then score these generated candidates based on the results of the saliency detection [1, 10, 2, 37, 31] or aesthetics assessment [45, 53, 33, 7, 26, 52, 51]. Recently, numerous data-driven methods [4, 5, 40, 41, 44] train the CNN models directly with the annotated data [4, 44] for this problem. The aforementioned methods mainly consider the region features of candidates when scoring them, ignoring the mutual relations between different regions (views) of an image. In contrast, we show that mining the relations between different regions can significantly help to find good views from an image.

In this work, we propose a graph-based model with the gated feature update to model these relations and update the region features with the mined relation features for finding good views (see Figure 1). The features of different regions are propagated on a graph that models the mutual relations through the graph convolution [21]. During the feature propagation, the relation features of the candidate

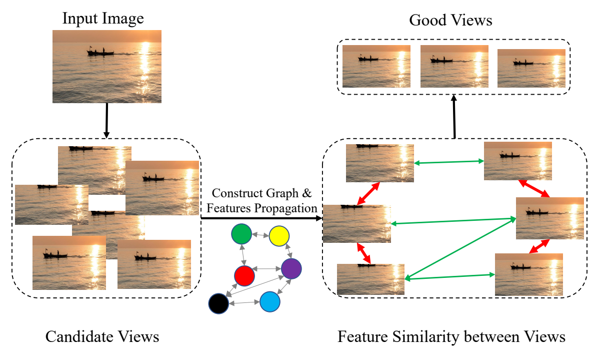


Figure 1. **Illustration of the proposed relation mining process for finding good views.** For a group of views from an input image, we construct a graph to model their relations and propagate the features of these views on the graph. As such, the similarities between good (well-composed) views and bad views become much smaller, thereby facilitating the task of finding good views.

regions are mined by taking account of the influence of the adjacent nodes in the graph, which can help collect more comparative information for predicting the scores of the regions. The mined relation features are fused with the region features through a gate that controls the influence of different features. To equip the graph with the stronger and more robust reasoning ability, we propose a data augmentation method that randomly selects the nodes from different graphs and constructs a new graph with the selected nodes for the prediction. Experiments show that the model can obtain better generalization ability by mixing different graphs up.

We design a multi-task loss to train the model, in which a weighted regression loss is used to predict the score of each region, especially pays more attention to those regions with high annotated scores due to the nature of returning the best regions in this problem. In addition, a ranking loss is applied to model the score gaps between different candidates explicitly. To incorporate more prior knowledge into the graph, we propose a regularization item to enhance the correlation between the constructed graph and annotations.

We make the following four contributions in this work:

- We propose a graph-based model with the gated feature update to find good views from images. To the best of our knowledge, this work is the first one that explicitly models the relations between different candidate regions for finding good views.
- We introduce a new data augmentation method to mix graphs for enhancing the generalization ability of the proposed model.
- We design a multi-task loss to train the model, which enforces both the predicted scores and sorting order to be close to the annotations and incorporates prior knowledge into the graph simultaneously.
- We demonstrate that the proposed algorithm performs favorably against state-of-the-art methods through extensive experiments and comprehensive ablation studies to analyze the contribution of each component of the proposed model and demonstrate why the graph-based module helps to find good views.

2. Related Work

2.1. Composing Good Views

Finding good composed views from an image draws much attention in the past decades [5, 44, 7, 26, 34, 10], which finds various applications such as image cropping [1, 4, 19, 23, 33, 40, 41, 45, 48, 51, 52, 53]. The typical pipeline for this problem is generating candidates at the first stage and ranking them according to some criteria. According to the scoring criteria, the existing methods can be broadly categorized as attention-based, aesthetics-based, or data-driven.

Attention-based methods [1, 10, 2, 37, 31] assume that the best views should draw more attention from people. Therefore, these approaches usually use the results of saliency detection methods [38, 50] to evaluate candidate views. Usually, the view with the highest average saliency score is selected as the best result.

Unlike the attention-based methods that only consider the saliency, the aesthetics-based algorithms [45, 53, 33, 7, 26, 52, 51] are more concerned with the overall aesthetic quality of different regions. Some approaches [53, 33, 7, 26] design the handcrafted features based on the photographic rules or image aesthetic characteristics to evaluate the candidates, whereas other methods [4, 19, 40, 41] adopt classification or ranking models trained on the aesthetics assessment datasets [32, 28] to predict the region scores.

In recent years, data-driven models [4, 5, 40, 41, 44, 48] are developed to find good views based on the recent datasets for image cropping [45, 10, 4, 48]. Different from existing methods that only use the region features to score views, we propose to model the relations between different regions for the prediction. Empirically, we show that exploiting mutual relations can significantly help to find good views.

2.2. Model Relations using Graphs

Learning relations between image pixels or regions is an essential task in computer vision, and graph structures can be used naturally to describe the properties. In recent years, graph-based relation learning and reasoning methods have been developed with the help of the graph convolution networks (GCNs) [21]. Modeling the relations between different classes with the graphs achieves great success in zero-shot learning [43] and few-shot learning [12] for object recognition. The relations among different attributes are also mined with the graph-based reasoning for attribute recognition [25]. Wang and Gupta [42] use the GCNs to learn the relations between the detected objects for video classification. Other tasks also benefit from the graph-based relation reasoning, *e.g.*, object recognition [3], video understanding [29], scene graph generation [47, 14], RGBD semantic segmentation [35], action recognition [46], multi-label image recognition [6], and object tracking [11]. Recently, Wang *et al.* [39] address relation mining among frames or images by gated GNNs for video segmentation and image co-segmentation.

3. Proposed Algorithm

3.1. Overview

In this paper, we propose a graph-based model that captures relations between different regions to find good views from images. Given an input image, the feature map is obtained from a backbone network (*e.g.*, VGG16 [36]). The

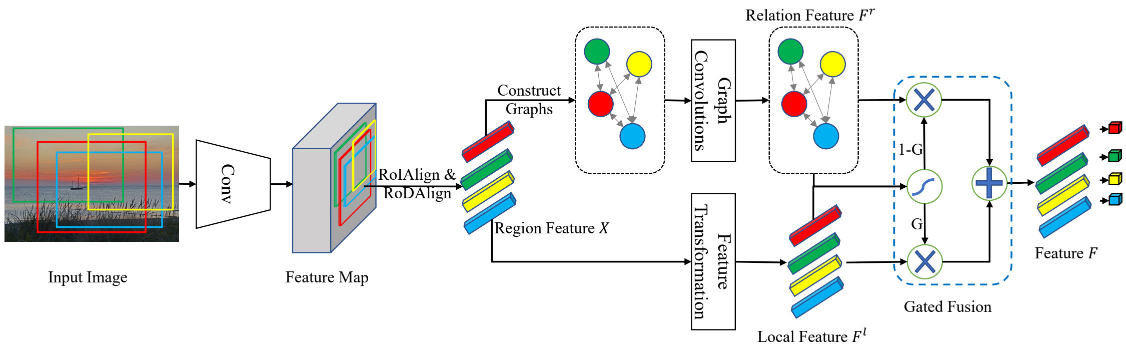


Figure 2. **Overview of the proposed model.** The proposed model uses a convolution block to obtain the feature map of the input image and employs the RoIAlign [15] and RoDAlign [48] schemes to extract the region feature $X \in \mathbb{R}^{N \times D^{in}}$, where N is the number of regions and D^{in} is the channel dimension of the region features. Then a graph is constructed according to the feature similarity between different regions, and the information propagation is performed on the graph using the graph convolution operation [21]. The relation feature $F^r \in \mathbb{R}^{N \times D^{out}}$ is captured through the information propagation and then fused with the transformed local feature $F^l \in \mathbb{R}^{N \times D^{out}}$ using a gated connection for the prediction, where D^{out} is the channel dimension of the output features.

feature vector of each predefined region is extracted from the feature map. We then construct a graph according to the similarity between different regions. During the training process, we use a regularization term to force the correlation between the adjacency matrix of the graph and the annotated score similarity matrix as strong as possible, so as to incorporate prior human knowledge in the constructed graph. The model propagates the region features on the graph using the graph convolution operation [21] to obtain the relation features, which provide more clues for the final prediction due to the comparative nature of this problem. We update the region features with the relation features adaptively through a gate connection. Finally, the score of each region is predicted based on the fused feature. The proposed model is illustrated in Figure 2.

3.2. Graph-based Relation Mining

Given the feature map of the input image, we extract the features of different regions in a way similar to [48]. First, we reduce the channel dimension of the feature map to 8 using a 1×1 convolution layer. Then, the RoIAlign [15] and RoDAlign [48] schemes are used to extract the RoI (region of interest) feature and RoD (region of discard) feature for each region using a pooling size of 9×9. The RoI and RoD features are contacted and passed through a fully-connected layer as the region features. We denote the region features extracted from an image as $X = [x_1, \ldots, x_i, \ldots, x_N] \in \mathbb{R}^{N \times D^{in}}$, where x_i is the feature of the i-th region, N is the number of regions, and D^{in} is the channel dimension of the feature for each region.

3.2.1 Reasoning Mutual Relations

With the region features, we first construct a graph to describe their mutual relations. We regard each region as a node (i.e., N nodes in the graph). Let $A \in \mathbb{R}^{N \times N}$ denote the adjacency matrix of the graph. The element $a_{m,n} \in A$, which represents the similarity (affinity) between the region x_m and region x_n, is computed by:

$$a_{m,n} = \begin{cases} e^{-\|W_m x_m - W_n x_n\|_2^2 / 2\sigma^2} & m \neq n, \\ 1 & m = n, \end{cases} \tag{1}$$

where $W_m \in \mathbb{R}^{D^{in} \times D^{out}}$ and $W_n \in \mathbb{R}^{D^{in} \times D^{out}}$ are two trainable matrices used to transform the region features, $\|\cdot\|$ denotes the Euclidean norm, and σ is set to 1 empirically. In Eq. 1, the diagonal elements of A are all set to be 1, which means each node in the graph has a self-loop.

After constructing the graph, the features of different regions are propagated on the graph using the graph convolution [21] to infer the relations between different regions. Given the adjacency matrix A and the region features X, the information propagation across different nodes can be formulated as:

$$F^r = AXW^r, \tag{2}$$

where $W^r \in \mathbb{R}^{D^{in} \times D^{out}}$ is the trainable weight that transforms the feature dimension from D^{in} to D^{out}, and $F^r = [f_1^r, \ldots, f_i^r, \ldots, f_N^r] \in \mathbb{R}^{N \times D^{out}}$ denotes the relation features for the N regions. The captured relation feature for the i-th region f_i^r aggregates the information propagated from other nodes of the graph, which is vital for the final prediction. As scoring these regions is an implicit sorting process by the model, capturing the relative relations between different regions can help take the influence of other regions into account when predicting the score for one region.

3.2.2 Incorporating Prior Knowledge into Graph

As the relation feature F^r is obtained by propagating region features on the graph, how the graph reflects the relations between different regions is essential for the proposed model. In Eq. 1, we use the features of two regions (x_m

and x_n) to compute the corresponding element (a_{mn}) of the adjacency matrix A, such that the elements of A are learned with the gradients back-propagated from the final loss function during the training process, which is an implicit learning process.

In addition to the above implicit learning process, we also incorporate prior knowledge into the constructed graph. To this end, we propose a regularization term that makes the elements of A have high correlations with the similarities between annotated scores of different regions. In particular, if the difference in annotated scores between the two regions is small, then their corresponding weight (affinity) in A should be large, and vice versa. To implement the regularization term, we also construct a matrix to evaluate the similarities between the annotated scores of different regions. Let $A^s \in \mathbb{R}^{N\times N}$ denote the matrix, and each element $a^s_{m,n} \in A^s$ is computed by:

$$a^s_{m,n} = e^{-(s_m - s_n)^2/2\sigma^2}, \tag{3}$$

where s_m and s_n are the annotated scores of the m-th and n-th regions, respectively, and σ is also set to 1 as in Eq. 1. Here, A^s reflects the similarities between the human-annotated scores of different regions, and we want to incorporate such prior knowledge into the graph by making the adjacency matrix A and A^s have a strong correlation. We compute the cosine similarity as the correlation between A and A^s:

$$Corr(A, A^s) = \frac{\sum\limits_{m,n}(a_{m,n} - \overline{a})(a^s_{m,n} - \overline{a^s})}{\| \sum\limits_{m,n}(a_{m,n} - \overline{a})^2 \sum\limits_{m,n}(a^s_{m,n} - \overline{a^s})^2 \|^{1/2}}, \tag{4}$$

where $\overline{a}$ and $\overline{a^s}$ are the average values of the matrices A and A^s, respectively. We use Eq. 4 as a regularization term in the loss function, which forces the constructed graph to have a strong correlation with the annotated prior knowledge.

3.2.3 Gated Region Feature Update

After obtaining the relation features F^r through the information propagation on the graph, we use F^r to update the region features for the prediction. Similar to the LSTM [17] and GRU [8] models, we employ a gate to control the feature fusion process rather than directly adding them together. Before the feature fusion, we first transform the dimension of the region feature X to fit the dimension of F^r:

$$F^l = XW^l, \tag{5}$$

where $W^l \in \mathbb{R}^{D^{in}\times D^{out}}$ is the trainable weight that transforms the feature dimension of X from D^{in} to D^{out}, and the transformed region (local) feature is denoted as $F^l = [f^l_1, \ldots, f^l_i, \ldots, f^l_N] \in \mathbb{R}^{N\times D^{out}}$. The gated feature fusion process is computed by:

$$\begin{aligned} G &= s(F^r W^{rg} + b^{rg} + F^l W^{lg} + b^{lg}), \\ F &= (1 - G) \odot F^r + G \odot F^l, \end{aligned} \tag{6}$$

where $G \in \mathbb{R}^{N\times D^{out}}$ is the gate that controls the influence of F^r and F^l, $s(\cdot)$ is the sigmoid activation function, $W^{rg} \in \mathbb{R}^{D^{out}\times D^{out}}$ and $W^{ig} \in \mathbb{R}^{D^{out}\times D^{out}}$ are the trainable weights, $b^{rg} \in \mathbb{R}^{D^{out}}$ and $b^{lg} \in \mathbb{R}^{D^{out}}$ are the trainable biases, $\odot$ denotes the Hadamard product, and $F = [f_1, \ldots, f_i, \ldots, f_N] \in \mathbb{R}^{N\times D^{out}}$ is the fused feature. Because the sigmoid function tends to push the output to approximately 0 or 1, the information in some channels of F comes from F^r and the information in other channels is from F^l, which helps the model adaptively select the useful information for the prediction rather than mixing all the information together. Experimental results in Section 4.4 demonstrate that directly combining F^r and F^l leads to worse results than the proposed gated feature fusion.

After obtaining F, we compute the scores of the N regions $P = [p_1, \ldots, p_i, \ldots, p_N] \in \mathbb{R}^{N\times 1}$ with

$$P = FW^p + b^p, \tag{7}$$

where $W^p \in \mathbb{R}^{D^{out}\times 1}$ and $b^p \in \mathbb{R}$ are the weight and bias of the last FC layer.

3.2.4 Mixing Graphs up for Data Augmentation

Data augmentation plays an essential role in the success of deep learning models. In this paper, we propose a data augmentation method to mix graphs for improving the generalization ability of the proposed model. Similar to the image data augmentation method [49] that generates images for training as the linear combinations of other training images, we propose to randomly mix different graphs up to construct a new graph for training. In particular, given two graphs, we randomly select nodes from different graphs and use Eq. 1 to construct a new graph for training while the label of each node remains unchanged. An illustration of the proposed data augmentation method is shown in Figure 3.

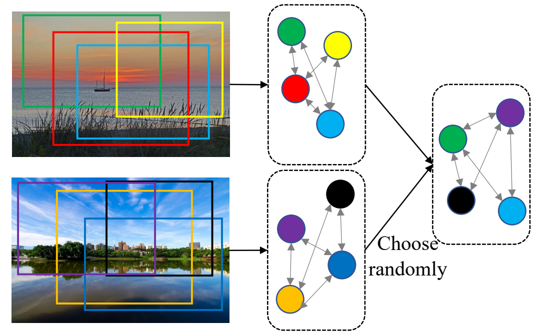


Figure 3. **Illustration of the proposed data augmentation method.** We randomly choose nodes from different images to construct a new graph for the data augmentation.

The mixed graphs containing nodes form different graphs provide more complex relations for the graph reasoning. Training the model in such complex environments can enable it to obtain more powerful and robust reasoning abilities under different conditions, thereby improving the generalization ability of the model. In practice, this process is randomly applied to the data with a 40% probability. Experimental results in Section 4.4 show that the proposed graph data augmentation method can help improve the performance a lot.

3.3. Loss Function

The whole model is end-to-end trainable using a multi-task loss function, which is the sum of three losses. Given an image containing N regions, the annotated and predicted scores of the i-th region are denoted as g_i and p_i, respectively. First, we use a weighted smooth L1 loss for the score regression, which is

$$L_{reg} = \frac{1}{N}\sum_{i=1}^{N} e^{\frac{\max(0, g_i - \overline{g})}{\sigma'}} L_1^s(p_i - g_i), \tag{8}$$

in which, $\overline{g}$ is the average score of all regions in the training set, σ' is set to 1, and L_1^s is the smooth L1 loss [13] computed by

$$L_1^s(x) = \begin{cases} 0.5x^2 & \text{if } x < 1, \\ |x| - 0.5 & \text{otherwise.} \end{cases} \tag{9}$$

The smooth L1 loss is widely used for the regression problem because of its robustness to outliers. As the nature of this problem is to find the best regions, the regions with high annotated scores are more important than those with low scores. Thus, we add a weight to the loss function according to the ground truth score g_i in Eq. 8.

Although the regression loss in Eq. 8 has implicitly modeled the sorting orders of different regions, we also use a ranking loss to model the score gaps between different regions explicitly. The importance of this ranking loss is validated in the ablation studies (see Section 4.4). The ranking loss is computed by

$$L_{rank} = \frac{\sum_{i,j}\max(0, -\varphi(g_i - g_j)((p_i - p_j) - (g_i - g_j))}{N(N-1)/2}, \tag{10}$$

where $\varphi(\cdot)$ is the sign function. L_{rank} forces the absolute value of the predicted score gap between two regions to be no less than the gap between the annotated scores to model the sorting relations explicitly.

In addition to L_{reg} and L_{rank}, there is also a regularization term $Corr(A, A^s)$ (in Section 3.2.2) that forces the constructed graph to have a strong correlation with the human-annotated prior knowledge. So the whole loss function is computed as

$$Loss = L_{reg} + \alpha L_{rank} - \beta Corr(A, A^s), \tag{11}$$

where α and β are the trade-off weights, and we set $\alpha = \beta = 1$ empirically in all experiments.

4. Experimental Results

4.1. Datasets and Evaluation Metrics

We perform experiments on the latest proposed GAICD dataset [48], which contains 1036 images with 89,519 annotated regions (crops) for training and 200 images for the test. We also use the metrics employed in the GAICD dataset [48] to evaluate different methods, including the average Spearman's rank-order correlation coefficient ($\overline{SRCC}$) and $Acc_{K/N}$. The $\overline{SRCC}$ is used to evaluate the rank correlation between the predicted and annotated scores of regions from each image. The $Acc_{K/N}$ (short for "return K of top-N accuracy") is used to compute how many of the top-K results predicted by the model belong to the top-N annotated regions. We set N to either 5 or 10 as the original settings [48], and evaluate $K = 1, 2, 3, 4$ for both $N = 5$ and $N = 10$, leading to 8 metrics ($Acc_{1/5}$, $Acc_{2/5}$, $Acc_{3/5}$, $Acc_{4/5}$, $Acc_{1/10}$, $Acc_{2/10}$, $Acc_{3/10}$, $Acc_{4/10}$). The average $Acc_{K/N}$ over K for each N is also computed as metrics: $\overline{Acc_N} = \frac{1}{4}\sum_{K=1}^{4} Acc_{K/N}$. The $\overline{SRCC}$ focuses on whether all candidates are ranked accurately, while the $Acc_{K/N}$ mainly considers whether the returned top-K results are acceptable. Please refer to [48] for more details about the above metrics.

4.2. Implementation Details

As most previous models [40, 44, 48] are based on the VGG16 model [36], we also use the convolution blocks from the VGG16 model (truncated at Conv4) as the backbone network for the fair comparison. The dimensions of the region features before and after the proposed relation reasoning module (D^{in} and D^{out} in Section 3.2) are set to 512 and 256, respectively. We employ the anchors defined in the GAICD [48] dataset as the candidates to search for the good views, because of the properties of finding good views (*i.e.*, the local redundancy, content preservation, and aspect ratio limitation), the number of candidates in an image is less than 90. When training the model, we randomly apply the proposed data augmentation method with a 40% probability. When the method is applied, the input is two images. Otherwise, the input is one single image. Similar to [48], we use 64 randomly selected regions from the input image(s) to construct the graph ($N = 64$ in Section 3.2) in the training stage, and N is equal to the number of all candidate regions in an image in the test stage. The short side of images is resized to 256, and the aspect ratios keep unchanged. The network is optimized in an end-to-end manner with the Adam optimizer [20] for 50 epochs with a weight decay of $1e^{-4}$. The warmup [16] is used in the first 5 epochs to increase the learning rate from 0 to $1e^{-4}$, then the cosine learning

Table 1. **Comparison with the state-of-the-art methods on the GAICD [48] dataset.** The results of other methods are from [48].

Model	Backbone	$Acc_{1/5}$	$Acc_{2/5}$	$Acc_{3/5}$	$Acc_{4/5}$	$\overline{Acc_5}$	$Acc_{1/10}$	$Acc_{2/10}$	$Acc_{3/10}$	$Acc_{4/10}$	$\overline{Acc_{10}}$	$\overline{SRCC}$	Runtime	Parameters
A2RL [23]	Alexnet [22]	23.0	-	-	-	-	38.5	-	-	-	-	-	274 ms	24.11M
VPN [44]	VGG16 [36]	40.0	-	-	-	-	49.5	-	-	-	-	-	11 ms	65.31M
VFN [5]	Alexnet [22]	27.0	28.0.0	27.2	24.6	26.7	39.0	39.3	39.0	37.3	38.7	0.450	1092 ms	11.55M
VEN [44]	VGG16 [36]	40.5	36.5	36.7	36.8	37.6	54.0	51.0	50.4	48.4	50.9	0.621	623 ms	40.93M
GAIC [48]	VGG16 [36]	53.5	51.5	49.3	46.5	50.2	71.5	70.0	67.0	65.5	68.5	0.735	8 ms	13.54M
Ours	VGG16 [36]	**63.0**	**62.3**	**58.8**	**54.9**	**59.7**	**81.5**	**79.5**	**77.0**	**73.3**	**77.8**	**0.795**	10 ms	13.68M

Table 2. **Comparison with the state-of-the-art methods on the HCDB [10] dataset.**

Model	IoU ↑	BDE ↓
Fang *et al.* [10]	0.740	-
Chen *et al.* [1]	0.640	0.075
Wang *et al.* [40]	0.810	0.057
A2RL [23]	0.820	-
VPN [44]	0.835	0.044
VEN [44]	**0.837**	0.041
GAIC [48]	0.834	0.041
Ours	0.836	**0.039**

Table 3. **Comparison with the state-of-the-art methods on the ICDB [45] dataset.**

Model	IoU ↑		
	Set 1	Set 2	Set 3
Yan *et al.* [45]	0.749	0.729	0.732
VFN [5]	0.764	0.753	0.733
Wang *et al.* [40]	0.813	**0.806**	**0.816**
A2RL [23]	0.802	0.796	0.790
VPN [44]	0.802	0.791	0.778
VEN [44]	0.781	0.770	0.753
GAIC [48]	0.799	0.781	0.779
Ours	**0.817**	0.805	0.795

rate decay [27] is used in the following 45 epochs. In addition to the proposed data augmentation method, we also randomly flip images and change the brightness, contrast, and saturation of images for the data augmentation.

4.3. Comparison with the State-of-the-arts

Quantitative comparison. First, we compare the performance of the proposed model with the state-of-the-art methods on the GACID dataset [48] in Table 1. The results show that the proposed model performs favorably against state-of-the-art methods. In particular, the proposed method uses the same backbone network and region feature extraction method (RoI+RoD) as the most competitive method GAIC [48], demonstrating the capabilities of the proposed modules of this paper. The contribution of each module is analyzed in Section 4.4.

The GACID dataset [48] is the latest one for this task, which shows that the IoU (Intersection-over-Union) based metrics used in the previous datasets [10, 45] cannot reliably evaluate the performance of the model. Despite the unreliable metrics used in the ICDB [45] and HCDB [10] datasets, we still show the results of the proposed model on these two datasets in Table 2 and 3, where the proposed model achieves similar IoU score compared to the state-of-the-art methods.

Table 4. **User study results.** We report the percentage of the results generated by different methods that are selected in the user study. The compared methods include the GAIC [48], VEN [44], VPN [44], VFN [5], and A2RL [23] models.

Model	Ours	GAIC	VEN	VPN	VFN	A2RL
Percentage	**25.9%**	20.7%	16.1%	17.8%	10.2%	9.3%

Runtime and model complexity. We also compare the running speed and model complexity of different models in Table 1. All models are run on the same PC with a single GPU. The proposed model runs faster than most state-of-the-art methods except the GAIC [48] method. Note that the VEN [44] method runs much faster than the speed reported in their original paper because there are much fewer candidates from images in the GAICD dataset [48].

Qualitative comparison. To further demonstrate the capabilities of the proposed model, we also conduct the qualitative comparison between the proposed method and the state-of-the-art methods [23, 5, 44, 48] in Figure 4. Compared to those methods, the proposed model can remove the unpleasant outer area of the source images more robustly. For example, in the second row of Figure 4, most compared methods cannot altogether remove the tree on the right side that hurts the image composition. However, the proposed method can remove it without any trace. More qualitative results are shown in the supplementary material.

User study. Evaluating the qualities of views from an image is subjective. Although our method achieves good results on the densely labeled GAICD [48] dataset, we still compare the proposed method with other methods through the user study. We randomly select 200 images from the GAICD [48], HCDB [10], and ICDB [45] datasets at a ratio of 67:67:66, and generate the results using different methods. Then we invite five experts to select the best view from these generated results for each image. Table 4 show that our method also achieves the best result on the user study.

4.4. Ablation Study

To better understand the proposed model, especially the contribution of each proposed module, we conduct a series of ablation studies using the GAICD [48] dataset.

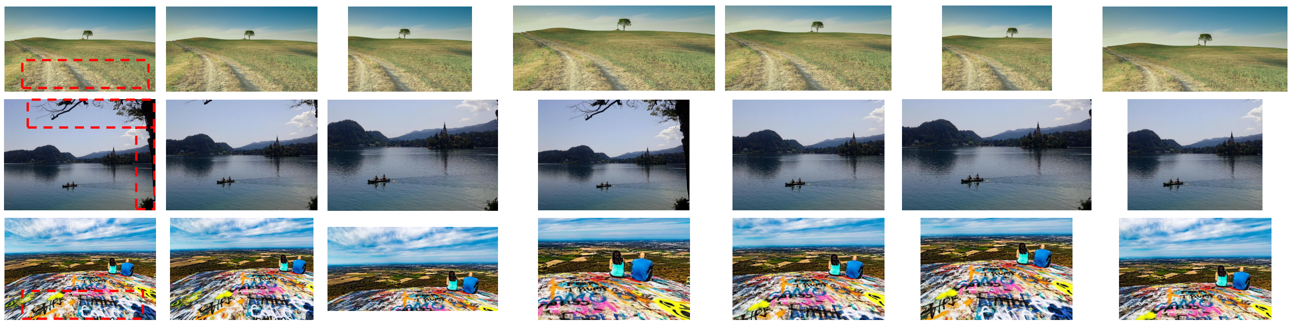


Figure 4. **Qualitative comparison of the returned top-1 view.** Compared to the existing methods (A2RL [23], VFN [5], VPN [44], VEN [44], GAIC [48]), the proposed method can remove the unpleasant outer area (the red dashed box) more robustly.

Table 5. **Ablation study on the model architecture.**

Gated Fusion	F^r	F^l	$\overline{Acc_5}$	$\overline{Acc_{10}}$	$\overline{SRCC}$
✓	✓	✓	**59.7**	**77.8**	**0.795**
	✓	✓	57.9	75.8	0.783
	✓		57.4	75.2	0.779
		✓	55.9	73.1	0.778

Table 6. **Ablation study on the loss function.** The $Corr$ is short for $Corr(A, A^s)$.

$Corr$	L_{rank}	L_{reg}	L_1^s	$\overline{Acc_5}$	$\overline{Acc_{10}}$	$\overline{SRCC}$
✓	✓	✓		**59.7**	**77.8**	**0.795**
	✓	✓		57.4	75.4	0.781
	✓			55.4	73.7	0.780
		✓		56.4	74.6	0.777
			✓	55.9	73.3	0.777

Model architecture. First, we analyze the contribution of each module in the model architecture. Since the feature used for the prediction is the gated fusion of the relation feature F^r from the graph and the local feature F^l from the FC layer (see Eq. 6), we eliminate the gated fusion, F^r, and F^l from the model respectively, then train and evaluate the model on the GAICD [48] dataset. When removing the gated fusion, we add the two features up for the prediction, which is $F = F^r + F^l$. The results are shown in Table 5. Only using the F^r for the prediction gets better results than only using the F^l, because the F^r contains more information propagated from other nodes in the graph. Adding the F^r and F^l directly only obtains a marginal boost. However, the gated fusion of these two features gets a significant performance improvement, demonstrating the importance of the gated feature fusion.

Loss function. Second, we study the influence of each component in the multi-task loss function and show the results in Table 6. The results demonstrate that the regularization term $Corr(A, A^s)$, which incorporates the prior knowledge into the graph, improves the performance of the model by a large margin. The reason is that the information from the final loss function is not good enough for the model to learn how to construct the graph, explicitly guiding the graph construction with the annotated information can help improve the relation modeling ability of the graphs. Only using the L_{reg} to train the model generally gets better results than only using the L_{rank} when returning the top-K regions, but enforcing both the predicted scores and score gaps to be close to the annotations simultaneously using the $L_{reg} + L_{rank}$ achieves better performance. In Section 3.3, we design the L_{reg} as a weighted Smooth L1 loss function that pays more attention to the regions with high annotated scores motivated by the characteristics of this problem. An interesting observation is that the weighted L_{reg} achieves much better results than the Smooth L1 loss when returning the top-K regions ($\overline{Acc_5}$ and $\overline{Acc_{10}}$). However, the overall sorting accuracy ($\overline{SRCC}$) has not changed significantly. The reason is that the L_{reg} pays more attention to the top-K regions, but performs similar to the Smooth L1 loss for most other regions, so the overall sorting results keep similar.

Table 7. **Ablation study on the probability of mixing graphs for the data augmentation.**

Probability of mixing graphs	$\overline{Acc_5}$	$\overline{Acc_{10}}$	$\overline{SRCC}$
0% (w/o augmentation)	57.5	75.5	0.780
20%	**60.2**	76.0	0.789
40%	59.7	**77.8**	**0.795**
60%	58.8	76.6	0.788
80%	58.8	76.6	0.784
100%	58.9	75.8	0.783

Data augmentation by mixing graphs. Third, we validate the proposed graph data augmentation method in Section 3.2.4. We randomly mix the graphs up with different probabilities and show the results in Table 7. The proposed data augmentation method can enhance the generalization ability of the model by a large margin when the probability of mixing graphs is from 20% to 40%, demonstrating the capabilities of the proposed method. However, the performance gain gets smaller when increasing the proportion of mixed graphs beyond 40%, indicating both source images and mixed ones are essential for training a highly generalized model.

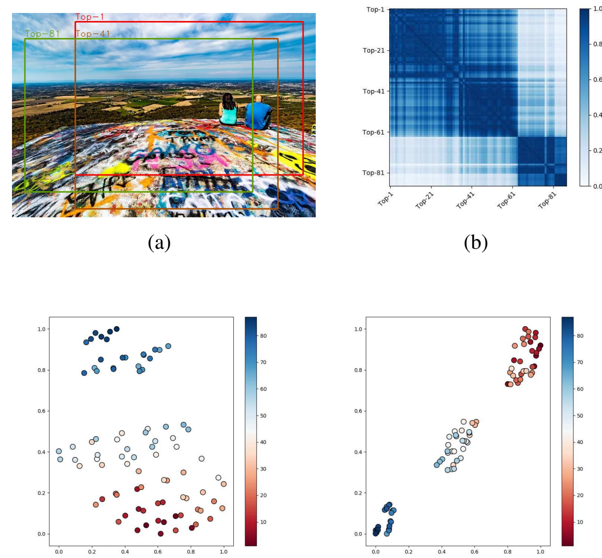
Table 8. **Ablation study on the number of graph nodes for training.** Because the number of candidates in an image is less than 90 in the GAICD dataset [48], for $N = 128$, we use all candidates to construct the graph when the input is a single image, but when the input is two images (for the proposed data augmentation), we randomly choose 128 candidates to construct the graph.

Graph Nodes	$\overline{Acc_5}$	$\overline{Acc_{10}}$	$\overline{SRCC}$
$N = 16$	**60.5**	76.6	0.781
$N = 32$	**60.5**	77.0	0.792
$N = 64$	59.7	**77.8**	**0.795**
$N = 128$	58.2	76.1	0.781

Number of graph nodes during training. Last, we study the influence of the number of graph nodes for training. With more graph nodes ($16 \rightarrow 64$), the performance of $\overline{Acc_{10}}$ and $\overline{SRCC}$ raises accordingly, but the performance of $\overline{Acc_5}$ keeps stable or gets even worse. When more graph nodes are considered in the training stage ($64 \rightarrow 128$), the performance drops accordingly. We think the reason is that the number of candidates in a single image is less than 90, we can randomly choose 128 candidates when the input is two images (for the proposed data augmentation), but we have to use all candidates to construct the graph when the input is a single image (a chance of 60%). Using all candidates will reduce the number of node combinations (randomness) during the training process, thereby hurting the generalization ability of the model.

4.5. Analysis of the Graph

The ablation study results in Section 4.4 show that the graph-based relation mining can enhance the capacity of the model. In this section, we want to have a more in-depth perspective to reveal why the graph-based reasoning (comparison) can help obtain better results. A visual example is shown in Figure 5. In Figure 5(b), the edge weights between different nodes are highly correlated with the similarities between the annotation scores. The closer the annotation scores are, the higher the weight is. By comparing Figure 5(c) and Figure 5(d), we find that the distances between the features of the good views and bad views get much larger after the graph-based feature propagation. The reason for the above observation is that, in the graph, the good views are connected to each other with large weights, and the bad views are also connected with substantial weights. However, the weights between the good views and bad views are much smaller, after the feature propagation, the distance between the aggregated features of good views and bad views gets much more substantial, which can help the model find the good views more easily.



(a) (b) (c) (d)

Figure 5. **A visual example of how graph-based reasoning performs.** (a) The source image, (b) the adjacency matrix of the constructed graph for the source image, (c) the t-SNE [30] visualization of the feature distribution for different candidates before the graph-based reasoning, (d) the feature distribution after the graph-based reasoning. In (b), the *Top-K* indicates that the region has the K-th highest annotated score among all regions. In (c) and (d), the number K in the color bar also indicates the region with the K-th highest annotated score among all regions, so red nodes represent regions with high annotated scores, and blue nodes represent regions with low annotated scores. Zoom in for the best view.

5. Conclusion

In this work, we propose a relation-aware model to find good views from images, which explicitly mines the mutual relations between different views. We introduce a graph-based module with the gated feature fusion to update the local feature with the mined relation feature. Furthermore, we also explore to incorporate the prior human knowledge into the graph and develop a new data augmentation method for the proposed model. In addition, we carefully design a multi-task loss for this problem, which considers the predicted scores and score gaps simultaneously. Extensive quantitative and qualitative evaluations demonstrate that the proposed method achieves state-of-the-art performance and enables robust searching of good views.

Acknowledgement

This work is funded by the National Natural Science Foundation of China (Grant 61876181, Grant 61673375, and Grant 61721004), the Projects of Chinese Academy of Sciences (Grant QYZDB-SSW-JSC006), and the NSF Career Grant (1149783). Debang is also supported by China Scholarship Council (CSC).

References

[1] Jiansheng Chen, Gaocheng Bai, Shaoheng Liang, and Zhengqin Li. Automatic image cropping: A computational complexity study. In *CVPR*, 2016.

[2] Li-Qun Chen, Xing Xie, Xin Fan, Wei-Ying Ma, Hong-Jiang Zhang, and He-Qin Zhou. A visual attention model for adapting images on small displays. *Multimedia systems*, 2003.

[3] Xinlei Chen, Li-Jia Li, Li Fei-Fei, and Abhinav Gupta. Iterative visual reasoning beyond convolutions. In *CVPR*, 2018.

[4] Yi-Ling Chen, Tzu-Wei Huang, Kai-Han Chang, Yu-Chen Tsai, Hwann-Tzong Chen, and Bing-Yu Chen. Quantitative analysis of automatic image cropping algorithms: A dataset and comparative study. In *WACV*, 2017.

[5] Yi-Ling Chen, Jan Klopp, Min Sun, Shao-Yi Chien, and Kwan-Liu Ma. Learning to compose with professional photographs on the web. In *ACM Multimedia*, 2017.

[6] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *CVPR*, 2019.

[7] Bin Cheng, Bingbing Ni, Shuicheng Yan, and Qi Tian. Learning to photograph. In *ACM Multimedia*, 2010.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv*, 2014.

[9] Seyed A Esmaeili, Bharat Singh, and Larry S Davis. Fast-at: Fast automatic thumbnail generation using deep neural networks. In *CVPR*, 2017.

[10] Chen Fang, Zhe Lin, Radomir Mech, and Xiaohui Shen. Automatic image cropping using visual composition, boundary simplicity and content preservation models. In *ACM Multimedia*, 2014.

[11] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. Graph convolutional tracking. In *CVPR*, 2019.

[12] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. *arXiv*, 2019.

[13] Ross Girshick. Fast r-cnn. In *ICCV*, 2015.

[14] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *CVPR*, 2019.

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[18] Jingwei Huang, Huarong Chen, Bin Wang, and Stephen Lin. Automatic thumbnail generation based on visual representativeness and foreground recognizability. In *ICCV*, 2015.

[19] Yueying Kao, Ran He, and Kaiqi Huang. Automatic image cropping with aesthetic map and gradient energy map. In *ICASSP*, 2017.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014.

[21] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv*, 2016.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[23] Debang Li, Huikai Wu, Junge Zhang, and Kaiqi Huang. A2-rl: Aesthetics aware reinforcement learning for image cropping. In *CVPR*, 2018.

[24] Debang Li, Huikai Wu, Junge Zhang, and Kaiqi Huang. Fast a3rl: Aesthetics-aware adversarial reinforcement learning for image cropping. *TIP*, 2019.

[25] Qiaozhe Li, Xin Zhao, Ran He, and Kaiqi Huang. Visual-semantic graph reasoning for pedestrian attribute recognition. In *AAAI*, 2019.

[26] Ligang Liu, Renjie Chen, Lior Wolf, and Daniel Cohen-Or. Optimizing photo composition. In *Computer Graphics Forum*, 2010.

[27] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv*, 2016.

[28] Wei Luo, Xiaogang Wang, and Xiaoou Tang. Content-based photo quality assessment. In *ICCV*, 2011.

[29] Chih-Yao Ma, Asim Kadav, Iain Melvin, Zsolt Kira, Ghassan AlRegib, and Hans Peter Graf. Attend and interact: Higher-order object interactions for video understanding. In *CVPR*, 2018.

[30] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

[31] Luca Marchesotti, Claudio Cifarelli, and Gabriela Csurka. A framework for visual saliency detection with applications to image thumbnailing. In *ICCV*, 2009.

[32] Naila Murray, Luca Marchesotti, and Florent Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *CVPR*, 2012.

[33] Masashi Nishiyama, Takahiro Okabe, Yoichi Sato, and Imari Sato. Sensation-based photo cropping. In *ACM Multimedia*, 2009.

[34] Jaesik Park, Joon-Young Lee, Yu-Wing Tai, and In So Kweon. Modeling photo composition and its application to photo re-arrangement. In *ICIP*, 2012.

[35] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3d graph neural networks for rgbd semantic segmentation. In *ICCV*, 2017.

[36] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014.

[37] Bongwon Suh, Haibin Ling, Benjamin B Bederson, and David W Jacobs. Automatic thumbnail cropping and its effectiveness. In *ACM UIST*, 2003.

[38] Eleonora Vig, Michael Dorr, and David Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. In *CVPR*, 2014.

[39] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *ICCV*, 2019.

[40] Wenguan Wang and Jianbing Shen. Deep cropping via attention box prediction and aesthetics assessment. In *ICCV*, 2017.

[41] Wenguan Wang, Jianbing Shen, and Haibin Ling. A deep network solution for attention and aesthetics aware photo cropping. *TPAMI*, 2018.

[42] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, 2018.

[43] Xiaolong Wang, Yufei Ye, and Abhinav Gupta. Zero-shot recognition via semantic embeddings and knowledge graphs. In *CVPR*, 2018.

[44] Zijun Wei, Jianming Zhang, Xiaohui Shen, Zhe Lin, Radomír Mech, Minh Hoai, and Dimitris Samaras. Good view hunting: Learning photo composition from dense view pairs. In *CVPR*, 2018.

[45] Jianzhou Yan, Stephen Lin, Sing Bing Kang, and Xiaoou Tang. Learning the change for automatic image cropping. In *CVPR*, 2013.

[46] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

[47] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018.

[48] Hui Zeng, Lida Li, Zisheng Cao, and Lei Zhang. Reliable and efficient image cropping: A grid anchor based approach. In *CVPR*, 2019.

[49] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv*, 2017.

[50] Jianming Zhang and Stan Sclaroff. Saliency detection: A boolean map approach. In *ICCV*, 2013.

[51] Luming Zhang, Mingli Song, Yi Yang, Qi Zhao, Chen Zhao, and Nicu Sebe. Weakly supervised photo cropping. *TMM*, 2013.

[52] Luming Zhang, Mingli Song, Qi Zhao, Xiao Liu, Jiajun Bu, and Chun Chen. Probabilistic graphlet transfer for photo cropping. *TIP*, 2012.

[53] Mingju Zhang, Lei Zhang, Yanfeng Sun, Lin Feng, and Weiying Ma. Auto cropping for digital photographs. In *ICME*, 2005.