RAPID: Rating Pictorial Aesthetics using Deep Learning*

Xin Lu[1] Zhe Lin[2] Hailin Jin[2] Jianchao Yang[2] James Z. Wang[1]
[1]The Pennsylvania State University
[2]Adobe Research
{xinlu, jwang}@psu.edu, {zlin, hljin, jiayang}@adobe.com

ABSTRACT

Effective visual features are essential for computational aesthetic quality rating systems. Existing methods used machine learning and statistical modeling techniques on handcrafted features or generic image descriptors. A recently-published large-scale dataset, the AVA dataset, has further empowered machine learning based approaches. We present the RAPID (RAting PIctorial aesthetics using Deep learning) system, which adopts a novel deep neural network approach to enable automatic feature learning. The central idea is to incorporate heterogeneous inputs generated from the image, which include a global view and a local view, and to unify the feature learning and classifier training using a double-column deep convolutional neural network. In addition, we utilize the style attributes of images to help improve the aesthetic quality categorization accuracy. Experimental results show that our approach significantly outperforms the state of the art on the AVA dataset.

Categories and Subject Descriptors

I.4.7 [**Image Processing and Computer Vision**]: Feature measurement; I.4.10 [**Image Processing and Computer Vision**]: Image Representation; I.5 [**Pattern Recognition**]: Classifier design and evaluation

General Terms

Algorithms, Experimentation

Keywords

Deep Learning; Image Aesthetics; Multi-Column Deep Neural Networks

*The research has been primarily supported by Penn State's College of Information Sciences and Technology and Adobe Research. The authors would like to thank the anonymous reviewers.

MM'14, November 03-07, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2647868.2654927.

1. INTRODUCTION

Automated assessment or rating of pictorial aesthetics has many applications. In an image retrieval system, the ranking algorithm can incorporate aesthetic quality as one of the factors. In picture editing software, aesthetics can be used in producing appealing polished photographs. Datta *et al.* [6] and Ke *et al.* [13] formulated the problem as a classification or regression problem where a given image is mapped to an aesthetic rating, which is normally quantized with discrete values. Under this framework, the effectiveness of the image representation, or the extracted features, can often be the accuracy bottleneck. Various handcrafted aesthetics-relevant features have been proposed [6, 13, 21, 3, 20, 7, 26, 27], including low-level image statistics such as distributions of edges and color histograms, and high-level photographic rules such as the rule of thirds.

While these handcrafted aesthetics features are often inspired from the photography or psychology literature, they share some known limitations. First, the aesthetics-sensitive attributes are manually designed, hence have limited scope. It is possible that some effective attributes have not yet been discovered through this process. Second, because of the vagueness of certain photographic or psychologic rules and the difficulty in implementing them computationally, these handcrafted features are often merely approximations of such rules. There is often a lack of principled approach to improve the effectiveness of such features.

Generic image features [23, 24, 22] are proposed to address the limitations of the handcrafted aesthetics features. They used well-designed common image features such as SIFT and Fisher Vector [18, 23], which have been successfully used for object classification tasks. The generic image features have been shown to outperform the handcrafted aesthetics features [23]. However, because these features are meant to be generic, they may be unable to attain the upper performance limits in aesthetics-related problems.

In this work, we intend to explore beyond generic image features by learning effective aesthetics features from images directly. We are motivated by the recent work in large scale image classification using deep convolutional neural networks [15] where the features are automatically learned from RGB images. The deep convolutional neural network takes pixels as inputs and learns a suitable representation through multiple convolutional and fully connected layers. However, the originally proposed architecture cannot be directly applied to our task. Image aesthetics relies on a combination of local and global visual cues. For example, the rule of thirds is a global image cue while sharpness and noise

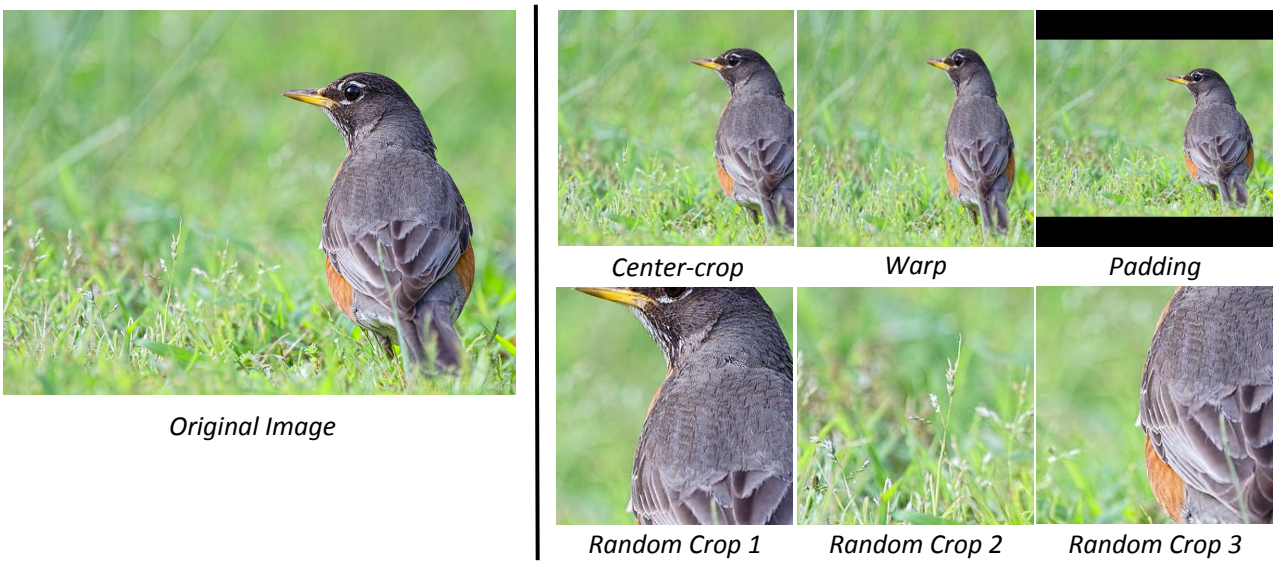

Figure 1: Global views and local views of an image. Global views are represented by normalized inputs: center-crop, warp, and padding (shown in the top row). Local views are represented by randomly-cropped inputs from the original high-resolution image (examples shown).

levels are local visual characteristics. Given an image, we generate two heterogeneous inputs to represent its global cues and local cues respectively. Figure 1 illustrates global vs. local views. To support network training on heterogeneous inputs, we extend the method in [15] by developing a double-column neural network structure which takes parallel inputs from the two columns. One column takes a global view of the image and the other column takes a local view of the image. We integrate the two columns after some layers of transformations to form the final classifier. We further improve the aesthetic quality categorization by exploring style attributes associated with images. We named our system RAPID, which stands for RAting PIctorial aesthetics using Deep learning. We used a recently-released large dataset to show the advantages of our approach.

1.1 Related Work

Earlier visual aesthetics assessment research focused on examining handcrafted visual features based on common cues such as color [6, 26, 27], texture [6, 13], composition [21, 20, 7], and content [20, 7], as well as generic image descriptors [23, 31, 24]. Commonly investigated color features include lightness, colorfulness, color harmony, and color distribution [6, 26, 27]. Texture descriptors vary from wavelet-based texture features [6], distribution of edges, to blur descriptors and shallow depth-of-field descriptors [13]. Composition features typically include the rule of thirds, size and aspect ratio [20], and foreground and background composition [21, 20, 7]. There have been attempts to represent the content of images using people and portrait descriptors [20, 7], scene descriptors [7], and generic image features such as SIFT [18], GIST [28], and Fisher Vector [23, 24, 22].

Despite the success of handcrafted and generic visual features, the usefulness of automatically learned features have been demonstrated in many vision applications [15, 4, 32, 30]. Recently, trained deep neural networks are used to build and associate mid-level features with class labels. Convolutional neural network (CNN) [16] is one of the most powerful learning architectures among the various types of neural networks (e.g., Deep Belief Net [10] and Restricted Boltzmann Machine [9]). Krizhevsky *et al.* [15] significantly advanced the 1000-class classification task in ImageNet challenge with a deep architecture of CNN in conjunction with dropout and normalization techniques, Sermanet *et al.* [30] achieved the-state-of-the-art performance on all major pedestrian detection datasets, and Ciresan *et al.* [4] reached a near-human performance on the MNIST[1] dataset.

The effectiveness of CNN features has also been demonstrated in image style classification [12]. Without training deep neural network, Karayev *et al.* extracted existing Decaf features [8] and used those features as input for style classification. There are key differences between that work [12] and ours. First, they mainly targeted style classification whereas we focus on aesthetic categorization, which is a different problem. Second, they used existing features as input to classification and did not train specific neural networks for style or aesthetics categorization. In contrast, we train deep neural networks directly from RGB inputs, which are optimized for the given task. Third, they relied on features from global views, while we leverage heterogeneous input sources, i.e., global and local views, and propose double-column neural networks to learn features jointly from both sources. Finally, we propose a regularized neural network based on related attributes to further boost aesthetics categorization.

As designing handcrafted features has been widely considered an appropriate approach in assessing image aesthetics, insufficient effort has been devoted to automatic feature learning on a large collection of labeled ground-truth data. The recently-developed AVA dataset [24] contains $250,000$ images with aesthetic ratings and a $14,000$ subset with style labels (e.g., rule of thirds, motion blur, and complementary colors), making automatic feature learning using deep learning approaches possible.

[1] http://yann.lecun.com/exdb/mnist/

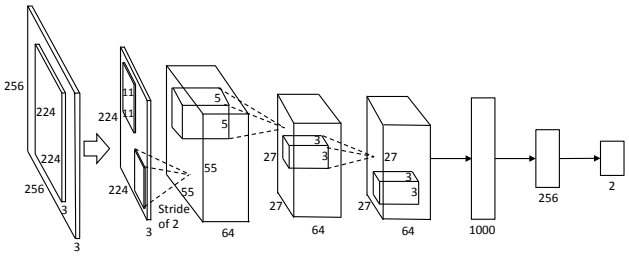

Figure 2: Single-column convolutional neural network for aesthetic quality rating and categorization. We have four convolutional layers and two fully-connected layers. The first and second convolutional layers are followed by max-pooling layers and normalization layers. The input patch of the size $224 \times 224 \times 3$ is randomly cropped from the normalized input of the size $256 \times 256 \times 3$ as done in [15].

In this work, we train deep neural networks on the AVA dataset to categorize image aesthetic quality. Specifically, we propose a double-column CNN architecture to automatically discover effective features that capture image aesthetics from two heterogeneous input sources. The proposed architecture is different from the recent work in multi-column neural networks [4, 1]. Agostinelli *et al.* [1] extended stacked sparse autoencoder to a multi-column version by computing the optimal column weights and applied the model to image denoising. Ciresan *et al.* [4] averaged the output of several columns trained on inputs with different standard preprocessing methods. Our architecture is different from that work because the two columns in our architecture are jointly trained using two different inputs: The first column of the network takes global image representation as the input, while the second column takes local image representations as the input. This allows us to leverage both compositional and local visual information.

The problem of assessing image aesthetics is also relevant to recent work of image popularity estimation [14]. Aesthetic value is connected with the notion of popularity, while there is a fundamental difference between the two concepts. Aesthetics concerns primarily with the nature and appreciation of beauty, while in the measurement of popularity both aesthetics and how interesting the visual stimulus is to the viewer population are important. For instance, a photograph of some thought-provoking subject may not be considered of high aesthetic value, but can be appreciated by many people based on the subject alone. On the other hand, a beautiful picture of flowers may not be able to reach the state of popularity if the viewers don't consider the subject of sufficient interestingness.

1.2 Contributions

Our main contributions are as follows.

- We conducted systematic evaluation of the single-column deep convolutional neural network approach with different types of input modalities for aesthetic quality categorization;
- We developed a double-column deep convolutional neural network architecture to jointly learn features from heterogeneous inputs;
- We developed a regularized double-column deep convolutional neural network to further improve aesthetic categorization using style attributes.

2. THE ALGORITHM

Patterns in aesthetically-pleasing photographs often indicate photographers' visual preferences. Among those patterns, composition [17] and visual balance [25] are important factors [2]. They are reflected in the global view (e.g., top row in Figure 1) and the local view (e.g., bottom row in the Figure). Popular composition principles include the rule of thirds, diagonal lines, and golden ratio [11], while visual balance is affected by position, form, size, tone, color, brightness, contrast, and proximity to the fulcrum [25]. Some of these patterns are not well-defined or even abstract, making it difficult to calculate those features for assessing image aesthetic quality. Motivated by this, we aim to leverage the power of CNN to automatically identify useful patterns and employ learned visual features to rate or to categorize the aesthetic quality of images.

However, applying CNN to the aesthetic quality categorization task is not straightforward. The different aspect ratios and resolutions in photographs and the importance of image details in aesthetics make it difficult to directly train CNN where inputs are typically normalized to the same size and aspect ratio. A challenging question, therefore, is to perform automatic feature learning with regard to both the global and the local views of the input images. To address this challenge, we take several different representations of an image, i.e., the global and the local views of the image, which can be encoded by jointly considering those heterogeneous representations. We first use each of the representations to train a single-column CNN (SCNN) to assess image aesthetics. We further developed a double-column CNN (DCNN) to allow our model to use the heterogeneous inputs from one image, aiming at identifying visual features in terms of both global and local views. Finally, we investigate how the style of images can be leveraged to boost aesthetic classification accuracy [29]. We present an aesthetic quality categorization approach with style attributes by learning a regularized double-column network (RDCNN), a three-column network.

2.1 Single-column Convolutional Neural Network

Deep convolutional neural network [15] takes inputs of fixed aspect ratio and size. However, an input image can be of arbitrary size and aspect ratio. To normalize image sizes, we propose three different transformations: center-crop (g_c), warp (g_w), and padding (g_p), which reflect the global view (I_g) of an image I. g_c isotropically resizes original images by normalizing their shorter sides to a fixed length s. Center-crop normalizes the input to generate a $s \times s \times 3$ input. g_c was adopted in a recent image classification work [15]. g_w anisotropically resizes (or warps) the original image into a normalized input with a fixed size $s \times s \times 3$. g_p resizes the original image by normalizing the longer side of the image to a fixed length s and padding border pixels with zeros to generate a normalized input of a fixed size $s \times s \times 3$. For each image I and each type of transformation, we generate an $s \times s \times 3$ input I_g^j with the transformation g_j, where $j \in \{c, w, p\}$. As resizing inputs can cause harmful information loss (i.e., the high-resolution local views) for aesthetic assessment, we also use randomly sampled fixed size

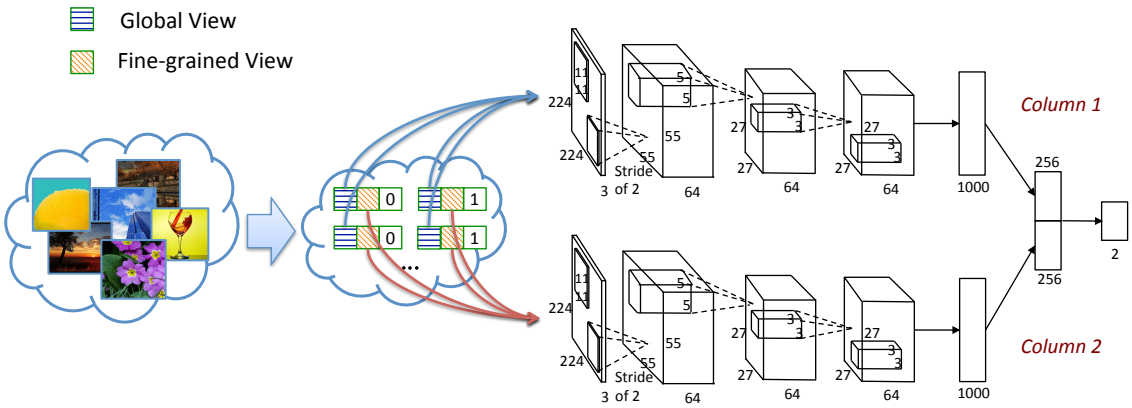

Figure 3: Double-column convolutional neural network. Each training image is represented by its global and local views, and is associated with its aesthetic quality label: 0 refers to a low quality image and 1 refers to a high quality image. Networks in different columns are independent in convolutional layers and the first two fully-connected layers. The final fully-connected layer are jointly trained.

(at $s \times s \times 3$) crops with the transformation l_r. Here we use g to denote global transformations and l to denote local transformations. This results in normalized inputs $\{I_l^r\}$ (r is an index of normalized inputs for each random cropping), which preserve the local views of an image with details from the original high-resolution image. We used these normalized inputs $I_t \in \{I_g^c, I_g^w, I_g^p, I_l^r\}$ for CNN training. In this work, we set s to 256, thus the size of I_t is $256 \times 256 \times 3$. To alleviate overfitting in network training, for each normalized input I_t, we extracted a random $224 \times 224 \times 3$ patch I_p or its horizontal reflection to be the input patch to our network.

We present an example for the four transformations, g_w, g_c, g_p, and l_r, in Figure 1. As shown, the global view of an image is maintained via the transformations of g_c, g_w, and g_p. Among the three global views, I_g^w and I_g^p maintain the relative spatial layout among elements in the original image. I_g^w and I_g^p follow rule of thirds whereas the I_g^c does not. In the bottom row of the figure, the local views of an original image are represented by randomly-cropped patches $\{I_l^r\}$. These patches depict the local details in the original resolution of the image.

The architecture of the SCNN used for aesthetic quality assessment is shown in Figure 2. It has a total of four convolutional layers. The first and the second convolutional layers are followed by max-pooling layers and normalization layers. The first convolutional layer filters the $224 \times 224 \times 3$ patch with 64 kernels of the size $11 \times 11 \times 3$ with a stride of 2 pixels. The second convolutional layer filters the output of the first convolutional layer with 64 kernels of the size $5 \times 5 \times 64$. Each of the third and forth convolutional layers has 64 kernels of the size $3 \times 3 \times 64$, and the two fully-connected layers have 1000 and 256 neurons respectively.

Suppose for the input patch I_p of the i-th image, we have the feature representation $\mathbf{x}_i$ extracted from layer fc256 (the outcome of the convolutional layers and the fc1000 layers), and the label $y_i \in \mathcal{C}$. The training of the last layer is done by maximizing the following log likelihood function:

$$l(\mathbf{W}) = \sum_{i=1}^{N} \sum_{c \in \mathcal{C}} \mathbb{I}(y_i = c) \log p(y_i = c \mid \mathbf{x}_i, \mathbf{w}_c) \quad , \qquad (1)$$

where N is the number of images, $\mathbf{W} = \{\mathbf{w}_c\}_{c \in \mathcal{C}}$ is the set of model parameters, and $\mathbb{I}(x) = 1$ iff x is true and vice versa. The probability $p(y_i = c \mid \mathbf{x}_i, \mathbf{w}_c)$ is expressed as

$$p(y_i = c \mid \mathbf{x}_i, \mathbf{w}_c) = \frac{\exp\left(\mathbf{w}_c^T \mathbf{x}_i\right)}{\sum_{c' \in \mathcal{C}} \exp\left(\mathbf{w}_{c'}^T \mathbf{x}_i\right)} \quad . \qquad (2)$$

The aesthetic quality categorization task can be defined as a binary classification problem where each input patch is associated with an aesthetic label $c \in \mathcal{C} = \{0, 1\}$. In Section 2.3, we explain a SCNN for image style categorization, which can be considered a multi-class classification task.

As indicated by the previous study [15], the architecture of the deep neural network may critically affect the performance. Our experiments suggest that the general guideline for training a good-performing network is to first allow sufficient learning power of the network by using sufficient number of neurons. Meanwhile, we adjust the number of convolutional layers and the fully-connected layers to support the feature learning and classifier training. In particular, we extensively evaluate the network trained with different numbers of convolutional layers and fully-connected layers, and with or without normalization layers. Candidate architectures are shown in Table 1. To determine the optimal architecture for our task, we conduct experiments on candidate architectures and pick the one with the highest performance, as shown in Figure 2.

With the selected architecture, we train SCNN with four different types of inputs (I_g^c, I_g^w, I_g^p, I_l^r) using the AVA dataset [24]. During training, we handle the overfitting problem by adopting dropout and shuffling the training data in each epoch. Specifically, we found that l_r serves as an effective data augmentation approach which alleviates overfitting. Because I_l^r is generated by random cropping, an image contributes to the network training with different inputs when a different patch is used.

We experimentally evaluate the performance of these inputs with SCNN. Results will be presented in Section 3. I_g^w performs the best among the three global input variations (I_g^c, I_g^w, I_g^p). I_l^r yields an even better results compared with I_g^w. Hence, we use I_l^r and I_g^w as the two inputs to train the proposed double-column network. In our experiments, we fix the dropout rate as 0.5 and initiate the learning rate with 0.001. Given a test image, we compute its normalized input and followed by generating the input patch, with

which we calculate the probability of the input patch being assigned to each aesthetic category. We repeat this process for 50 times, average those results, and pick the class with the highest probability.

2.2 Double-column Convolutional Neural Network

For each image, its global or local information may be lost when transformed to a normalized input using g_c, g_w, g_p, or l_r. Representing an image through multiple inputs can somewhat alleviate the problem. As a first attempt, we generate one input to depict the global view of an image and another to represent its local view.

We propose a novel double-column convolutional neural network (DCNN) to support automatic feature learning with heterogeneous inputs, i.e., a global-view input and a local-view input. We present the architecture of the DCNN in Figure 3. As shown in the figure, networks in different columns are independent in convolutional layers and the first two fully-connected layers. The inputs of the two columns are I_g^w and I_l^r. We take the two 256×1 vectors from each of the fc256 layer and jointly train the weights of the final fully-connected layer. We avoid the interaction between two columns in convolutional layers because they are in different spatial scales. During training, the error is back propagated to the networks in each column respectively with stochastic gradient descent. With the proposed architecture, we can also automatically discover both the global and the local features of an image from the fc1000 layers and fc256 layers.

The proposed network architecture could easily be expanded to multi-column convolutional networks by incorporating more types of normalized inputs. DCNN allows different architectures in individual networks, which may facilitate the parameter learning for networks in different columns.

In our work, network architectures are the same for both columns. Given a test image, we perform a similar procedure as we do with SCNN to evaluate the aesthetic quality of an image.

2.3 Learning and Categorization with Style Attributes

The discrete aesthetic labels, i.e., high quality and low quality, provided weak supervision to make the network converge properly due to the large intra-class variation. This motivates us to exploit extra labels from the training images to help identify their aesthetic characteristics. We propose to leverage style attributes, such as complementary colors, macro, motion blur, rule of thirds, shallow depth-of-field (DOF), to help determine the aesthetic quality of images because they are regarded as highly relevant attributes [24].

There are two natural ways to formulate the problem. The first is to leverage the idea of multi-task learning [5], which jointly construct feature representation and minimize the classification error for both labels. Assuming we have aesthetic quality labels $\{y_{ai}\}$ and style labels $\{y_{si}\}$ for all training images, the problem becomes an optimization prblem:

$$\max_{\mathbf{X},\mathbf{W}_a,\mathbf{W}_s} \sum_{i=1}^{N} \Big(\sum_{c\in\mathcal{C}_A} \mathbb{I}(y_{ai}=c) \log p(y_{ai} \mid \mathbf{x}_i, \mathbf{w}_{ac}) + \sum_{c\in\mathcal{C}_S} \mathbb{I}(y_{si}=c) \log p(y_{si} \mid \mathbf{x}_i, \mathbf{w}_{sc}) \Big) \quad , \tag{3}$$

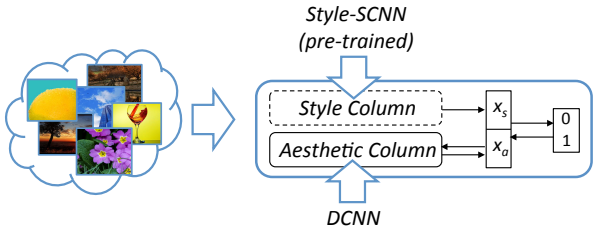

Figure 4: Regularized double-column convolutional neural network (RDCNN). The style attributes $\mathbf{x}_s$ are generated through pre-trained Style-SCNN and we leveraged the style attributes to regularize the training process of RDCNN. The dashed line indicates that the parameters of the style column is fixed during RDCNN training. While training the RDCNN, we only fine-tuned the parameters in the aesthetic column and the learning process is supervised by the aesthetic label.

where $\mathbf{X}$ is the features of all training images, $\mathcal{C}_A$ is the label set for aesthetic quality, $\mathcal{C}_S$ is the label set for style, and $\mathbf{W}_a = \{\mathbf{w}_{ac}\}_{c\in\mathcal{C}_A}$ and $\mathbf{W}_s = \{\mathbf{w}_{sc}\}_{c\in\mathcal{C}_S}$ are the model parameters. It is more difficult to obtain images with style attributes. In the AVA benchmark, among $230,000$ image with aesthetic labels only $14,000$ of them have style labels. As a result, we cannot jointly perform aesthetics categorization and style classification with a single neural network due to many missing labels.

Alternatively, we can use ideas from inductive transfer learning [29], where we target minimizing the classification error with one label, whereas we construct feature representations with both labels. As we only have a subset of images with style labels, we first train a style classifier with them. We then extract style attributes for all training images, and applied those attributes to regularize the feature learning and classifier training for aesthetic quality categorization.

To learn style attributes for $230,000$ training images, we first train a style classifier by performing the training procedure discussed in Section 2.1 on $11,000$ labeled training images (Style-SCNN). We adopted the same architecture as shown in Figure 2. The only difference is that we reduced the number of filters in the the first and fourth convolutional layers to a half due to the reduced number of training images. With Style-SCNN, we are maximizing the log likelihood function in Equation 1 where $\mathcal{C}$ is the set of style labels in the AVA dataset. We experimentally select the best architectures (to be shown in Table 4) and inputs (I_g^c, I_g^w, I_g^p, I_l^r). The details are described in Section 3. Given an image, we apply the learned weights and extract the features from the fc256 layer as its style attribute.

To facilitate the network training with style attributes of images, we propose a regularized double-column convolutional neural network (RDCNN) with the architecture shown in Figure 4. Two normalized inputs of the aesthetic column are I_g^w and I_l^r, same as in DCNN (Section 2.2). The input of the style column is I_l^r. The training of RDCNN is done by solving the following optimization problem:

$$\max_{\mathbf{x}_a,\mathbf{W}_a} \sum_{i=1}^{N} \sum_{c=1\in\mathcal{C}_a} \mathbb{I}(y_{ai}=c) \log p(y_{ai} \mid \mathbf{x}_{ai}, \mathbf{x}_{si}, \mathbf{w}_{ac}) \quad , \tag{4}$$

where $\mathbf{x}_{si}$ are the style attributes of the i-th training image, $\mathbf{x}_{ai}$ are the features to be learned. Note that the maximiza-

Table 1: Accuracy for Different SCNN Architectures

	conv1 (64)	pool1	rnorm1	conv2 (64)	pool2	rnorm2	conv3 (64)	conv4 (64)	conv5 (64)	conv6 (64)	fc1K	fc256	fc2	Accuracy
Arch 1	√	√	√	√	√	√	√	√			√	√	√	**71.20%**
Arch 2	√	√	√	√	√	√					√	√	√	60.25%
Arch 3	√	√	√	√	√	√	√				√	√	√	62.68%
Arch 4	√	√		√	√		√	√	√	√	√	√	√	65.14%
Arch 5	√	√	√	√	√	√	√	√				√	√	70.52%
Arch 6	√	√	√	√	√	√	√	√	√		√	√	√	62.49%
Arch 7	√	√	√	√	√	√	√	√	√	√	√	√	√	70.93%

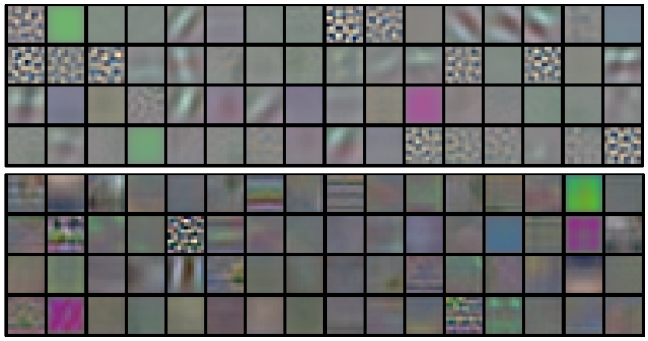

Figure 5: 128 convolutional kernels of the size $11 \times 11 \times 3$ learned by the first convolutional layer of DCNN for aesthetic quality categorization. The first 64 are from the local view column (with the input I_l^r) and the last 64 are from the global view column (with the input I_g^w).

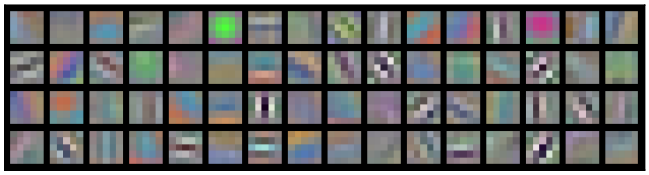

Figure 6: 64 convolutional kernels of the size $5 \times 5 \times 3$ learned by the first convolutional layer of CNN for object classification on the CIFAR dataset.

tion does not involve style attributes $\mathbf{x}_s$. In each learning iteration, we only fine-tuned the parameters in the aesthetic column and the learning process is supervised by the aesthetic label. The parameters of the style column are fixed and the style attributes $\mathbf{x_{is}}$ essentially serve as a regularizer for training the aesthetic column.

3. EXPERIMENTAL RESULTS

We evaluated the proposed method for aesthetics quality categorization on the AVA dataset [24]. We first introduce the dataset. Then we report the performance of SCNN with different network architectures and normalized inputs. Next, we present aesthetic quality categorization results with DCNN and qualitatively analyze the benefits of the double-column architecture over a single-column one. We also demonstrate the performance of RDCNN with the accuracy of trained style classifier and aesthetic categorization results with style attributes incorporated. Finally, we summarize the computational efficiency of SCNN, DCNN, and RDCNN in training and testing.

Table 2: Accuracy of Aesthetic Quality Categorization with Different Inputs

δ	I_l^r	I_g^w	I_g^c	I_g^p
0	71.20%	67.79%	65.48%	60.43%
1	68.63%	68.11%	69.67%	70.50%

Table 3: Accuracy of Aesthetic Quality Categorization for Different Methods

δ	[24]	SCNN	AVG_SCNN	DCNN	RDCNN
0	66.7%	71.20%	69.91%	73.25%	74.46%
1	67%	68.63%	71.26%	73.05%	73.70%

3.1 The Dataset

The AVA dataset contains a total of $250,000$ images, each of which has about 200 aesthetic ratings ranging from one to ten. We followed the experimental settings in [24], and used the same collection of training data and testing data: $230,000$ images for training and $20,000$ images for testing. Training images are divided into two categories, i.e., low-quality images and high-quality images, based on the same criteria as [24]. Images with mean ratings smaller than $5-\delta$ are referred to as low-quality images, those with mean ratings larger than or equal to $5+\delta$ are high-quality images. We set δ to 0 and 1 respectively to generate the binary ground truth labels for the training images. Images with ratings between $5-\delta$ and $5+\delta$ are discarded. With $\delta = 0$, there are $68,000$ low-quality images and $167,000$ high-quality images. With $\delta = 1$, there are $7,500$ low-quality images and $45,000$ high-quality images. For the testing images, we fix δ to 0, regardless what δ is used for training. This results in $5,700$ low-quality images and $14,000$ high-quality images for testing.

To learn style attributes, we use the subset of images with style labels from the AVA dataset as the training set. The 14 style classes include complementary colors, duotones, HDR, image grain, light on white, long exposure, macro, motion blur, negative images, rule of thirds, shallow DOF, silhouettes, soft focus, and vanishing point. The subset contains $11,000$ images for training and $2,500$ images for testing.

3.2 SCNN Results

We compare the performance of SCNN with different layer combinations and normalized inputs on aesthetic quality categorization task. Table 1 presents seven different architectures and their overall accuracy. As shown, the selected layer for each architecture is labeled with a check mark. In all seven architectures, we use I_l^r as the input with $\delta = 0$. The results show that the architecture Arch 1 performs the best, which partially indicates the importance of choosing a proper number of convolutional layers and fully connected layers, and having normalization layers.

(a) Images ranked the highest in aesthetics by DCNN

(b) Images ranked the lowest in aesthetics by DCNN

Figure 7: Images ranked the highest and the lowest in aesthetics generated by DCNN. Differences between low-aesthetic images and high-aesthetic images heavily lie in the amount of textures and complexity of the whole image.

With the network architecture fixed to Arch 1, we compare the performance of SCNN with different inputs, i.e., I_g^c, I_g^w, I_g^p, I_l^r. We train classifiers with both $\delta = 0$ and $\delta = 1$ for each input type. The overall accuracy is presented in Table 2. The results show that I_l^r yields the highest accuracy among four types of inputs, which indicates that l_r serves as an effective data augmentation approach to capture the local aesthetic details of images. I_g^w performs much better than I_g^c and I_g^p, which is the best among the three inputs for capturing the global view of images.

Based on the above observation, we choose Arch 1 as the architecture of our model, with I_l^r as input. As shown in

Figure 8: Test images correctly classified by DCNN but misclassified by SCNN. The first row shows the images that are misclassified by SCNN with the input I_l^r. The second row shows the images that are misclassified by SCNN with the input I_g^w. The label on each image indicates the ground-truth aesthetic quality.

Table 4: Accuracy for Different Network Architectures for Style Classification

	conv1 (32)	pool1	rnorm1	conv2 (64)	pool2	rnorm2	conv3 (64)	conv4 (32)	conv5 (32)	conv6 (32)	fc1K	fc256	fc14	MAP	Accuracy
Arch 1	√	√	√	√	√	√	√	√			√	√	√	**56.81%**	**59.89%**
Arch 2	√	√	√	√	√	√					√	√	√	52.39%	54.33%
Arch 3	√	√	√	√	√	√	√				√	√	√	53.19%	55.19%
Arch 4	√	√		√	√		√	√	√	√	√	√	√	54.13%	55.77%
Arch 5	√	√	√	√	√	√	√	√				√	√	53.94%	56.00%
Arch 6	√	√	√	√	√	√	√	√	√		√	√	√	53.22%	57.25%
Arch 7	√	√	√	√	√	√	√	√	√	√	√	√	√	47.44%	52.16%

Table 3, the performance of this setting is better than the state of the art on the AVA dataset for both $\delta = 0$ and $\delta = 1$.

3.3 DCNN Results

We adopt the SCNN architecture Arch 1 for both columns in DCNN. Figure 5 illustrates the filters of the first convolutional layer for trained DCNN. The first 64 are from the local column (with the input I_l^r), while the last 64 are from the global column (with the input I_g^w). Compared with the filters trained in the object recognition task on CIFAR dataset[2] (shown in Figure 6), the filters learned with image aesthetic labels are smoother and cleaner without radical intensity changes. This indicates that differences between low-aesthetic and high-aesthetic image cues mainly lie in the amount of texture and the complexity of the whole image. The difference can be observed from typical test images presented in Figure 7. The images ranked the highest in aesthetics are generally smoother than those ranked the lowest. This finding substantiates the importance of simplicity and complexity features recently proposed for analyzing visual emotions [19].

To quantitatively demonstrate the effectiveness of trained DCNN, we compare its performance with that of the SCNN as well as [24]. As shown in Table 3, DCNN outperforms SCNN for both $\delta = 0$ and $\delta = 1$, and significantly outperforms the earlier work. To further demonstrate the effectiveness of joint training of DCNN, we compare DCNN with AVG_SCNN, which averaged the two SCNN results with I_g^w and I_l^r as inputs. As shown in Table 3, DCNN outperforms the AVG_SCNN for both δ values.

[2] `http://www.cs.toronto.edu/~kriz/cifar.html`

Figure 9: 32 convolutional kernels of the size $11 \times 11 \times 3$ learned by the first convolutional layer of Style-SCNN for style classification.

To qualitatively analyze the benefits of the double-column architecture, we visualize ten test images correctly classified by DCNN but incorrectly by SCNN. We present the examples in Figure 8. Images in the first row are misclassified by SCNN with the input I_l^r. Images in the second row are misclassified with the input I_g^w. The label on each image indicates the ground-truth aesthetic quality. As shown, images misclassified by SCNN with the input I_l^r usually contain a dominant object, which is because I_l^r does not consider the global information in an image. Images misclassified by SCNN with the input I_g^w often have detailed information in their local views that can improve the classifier if can be properly leveraged.

3.4 Categorization with Style Attributes

To demonstrate the effectiveness of the style attributes for aesthetic quality categorization, we first evaluate the style classification accuracy with SCNN. We then compare the performance of RDCNN with DCNN.

3.4.1 Style Classification

We train the style classifier with SCNN, and visualize the filters learned by the first convolutional layer of SCNN in Figure 9. We test the trained model on 2,573 images.

Figure 10: Test images correctly classified by RDCNN and misclassified by DCNN. The label on each image indicates the ground truth aesthetic quality of images.

Table 5: Accuracy of Style Classification with Different Inputs

	I_l^r	I_g^w	I_g^c	I_g^p
AP	56.93%	44.52%	45.74%	41.78%
MAP	56.81%	47.01%	48.14%	44.07%
Accuracy	59.89%	48.08%	48.85%	46.79%

For each image, we randomly sample 50 patches of the size $224 \times 224 \times 3$, and average the prediction results. To compare our results with the results reported in [24], we use the same experimental setting. We perform similar experiments as discussed in Section 3.2 by comparing different architectures and normalized inputs. The comparison results for different architectures are shown in Table 4. The selected layer for each architecture is labeled with a check mark. We achieve the best accuracy for style classification with Arch 1 and I_l^r as input (Table 5). It indicates the importance of local view in determining the style of an image. It shows the effectiveness of l_r as a data augmentation strategy in case of limited training data. We did not compare our style classification results with Karayev *et al.* [12] as their evaluations were done on a randomly selected subset of test images.

The Average Precision(AP) and Mean Average Precision (MAP) are also calculated. The best MAP we achieved is 56.81% which outperforms the accuracy of 53.85% reported in [24].

3.4.2 *Aesthetic Quality Categorization with Style Attributes*

We demonstrate the effectiveness of style attributes by comparing the best aesthetic quality categorization accuracy we have achieved with and without style attributes. As shown in Table 3, RDCNN outperforms DCNN for both δ values.

To qualitatively analyze the benefits brought with the regularized double-column architecture, we show typical test images that have been correctly classified by RDCNN but misclassified by DCNN in Figure 10. Those examples correctly classified by RDCNN are mostly with the following styles: rule-of-thirds, HDR, black and white, long exposure, complementary colors, vanishing point, and soft focus. This indicates that styles attributes help aesthetic quality categorization.

3.5 Computational Efficiency

Training SCNN for a specific input type takes about two days. Training DCNN takes about three days. For RDCNN, style attribute training takes roughly a day, and RDCNN training three to four days. Classifying 2,000 images (each with 50 views) takes about 50 minutes, 80 minutes, and 100 minutes for SCNN, DCNN, and RDCNN, respectively, with Nvidia Tesla M2070/M2090 GPU.

4. CONCLUSIONS

We present a double-column deep convolutional neural network approach for aesthetic quality rating and categorization. Rather than designing handcrafted features or adopting generic image descriptors, aesthetic-related features are learned automatically. Feature learning and classifier training are unified with the proposed deep neural network approach. The double-column architecture takes into account both the global view and local view of an image for judging its aesthetic quality. Besides, image style attributes are leveraged to improve the accuracy. Evaluating with the AVA dataset, which is the largest benchmark with rich aesthetic ratings, our approach shows significant better results than earlier-reported results on the same dataset.

5. REFERENCES

[1] F. Agostinelli, M. Anderson, and H. Lee. Adaptive multi-column deep neural networks with application to robust image denoising. In *Advances in Neural*

Information Processing Systems (NIPS), pages 1493–1501. 2013.

[2] R. Arnheim. In *Art and visual Perception: A psychology of the creative eye.* Los Angeles. CA: University of California Press., 1974.

[3] S. Bhattacharya, R. Sukthankar, and M. Shah. A framework for photo-quality assessment and enhancement based on visual aesthetics. In *ACM International Conference on Multimedia (MM)*, pages 271–280, 2010.

[4] D. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3642–3649, 2012.

[5] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *International Conference on Machine Learning (ICML)*, pages 160–167, 2008.

[6] R. Datta, D. Joshi, J. Li, and J. Wang. Studying aesthetics in photographic images using a computational approach. In *European Conference on Computer Vision (ECCV)*, pages 288–301, 2006.

[7] S. Dhar, V. Ordonez, and T. Berg. High level describable attributes for predicting aesthetics and interestingness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1657–1664, June 2011.

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. In *Technical report, 2013. arXiv:1310.1531v1*, 2013.

[9] G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[10] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[11] D. Joshi, R. Datta, E. Fedorovskaya, Q. T. Luong, J. Z. Wang, J. Li, and J. B. Luo. Aesthetics and emotions in images. In *IEEE Signal Processing Magazine*, 2011.

[12] S. Karayev, A. Hertzmann, H. Winnermoller, A. Agarwala, and T. Darrel. Recognizing image style. In *British Machine Vision Conference (BMVC)*, 2014.

[13] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 419–426, 2006.

[14] A. Khosla, A. Das Sarma, and R. Hamid. What makes an image popular? In *International World Wide Web Conference (WWW)*, pages 867–876, 2014.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1106–1114, 2012.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[17] O. Litzel. In *On Photographic Composition.* New York: Amphoto Books, 1974.

[18] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[19] X. Lu, P. Suryanarayan, R. B. Adams Jr, J. Li, M. G. Newman, and J. Z. Wang. On shape and the computability of emotions. In *ACM International Conference on Multimedia (MM)*, pages 229–238. ACM, 2012.

[20] W. Luo, X. Wang, and X. Tang. Content-based photo quality assessment. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2206–2213, 2011.

[21] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *European Conference on Computer Vision (ECCV)*, pages 386–399, 2008.

[22] L. Marchesotti and F. Perronnin. Learning beautiful (and ugly) attributes. In *British Machine Vision Conference (BMVC)*, 2013.

[23] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka. Assessing the aesthetic quality of photographs using generic image descriptors. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1784–1791, 2011.

[24] N. Murray, L. Marchesotti, and F. Perronnin. AVA: A large-scale database for aesthetic visual analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2408–2415, 2012.

[25] W. Niekamp. An exploratory investigation into factors affecting visual balance. In *Educational Communication and Technology: A Journal of Theory, Research, and Development*, volume 29, pages 37–48, 1981.

[26] M. Nishiyama, T. Okabe, I. Sato, and Y. Sato. Aesthetic quality classification of photographs based on color harmony. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 33–40, 2011.

[27] P. O'Donovan, A. Agarwala, and A. Hertzmann. Color compatibility from large datasets. *ACM Transactions on Graphics (TOG)*, 30(4):63:1–12, 2011.

[28] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision (IJCV)*, 42(3):145–175, 2001.

[29] J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 22(10):1345–1359, 2010.

[30] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage features learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3626–3633, 2013.

[31] H.-H. Su, T.-W. Chen, C.-C. Kao, W. Hsu, and S.-Y. Chien. Scenic photo quality assessment with bag of aesthetics-preserving features. In *ACM International Conference on Multimedia (MM)*, pages 1213–1216, 2011.

[32] Y. Sun, X. Wang, and X. Tang. Hybrid deep learning for face verification. In *The IEEE International Conference on Computer Vision (ICCV)*, 2013.