https://doi.org/10.2352/ISSN.2470-1173.2017.14.HVEI-142

Defining Self-Similarity of Images Using Features Learned by Convolutional Neural Networks

Anselm Brachmann and Christoph Redies, Experimental Aesthetics Group, Jena University Hospital, Jena, Germany

Abstract

In recent years, Convolutional Neural Networks (CNNs) have gained huge popularity among computer vision researchers. In this paper, we investigate how features learned by these networks in a supervised manner can be used to define a measure of self-similarity, an image feature that characterizes many images of natural scenes and patterns, and is also associated with images of artworks. Compared to a previously proposed method for measuring self-similarity based on oriented luminance gradients, our approach has two advantages. Firstly, we fully take color into account, an image feature which is crucial for vision. Secondly, by using higher-layer CNN features, we define a measure of self-similarity that relies more on image content than on basic local image features, such as luminance gradients.

Introduction

Self-similarity is a prominent feature of images that depict natural scenes, patterns or objects. In a broad sense, self-similarity measures to which degree features of smaller subregions of an image are also present in the image as a whole. In natural images, self-similarity has been calculated based on different statistical properties. For example, images of natural scenes exhibit a scale-invariant (self-similar) Fourier power spectrum [17]. Moreover, they possess a fractal structure, which has been studied with the box-counting method [14] and was previously linked to naturalness and preference of scenes [8]. Amirshahi et al. [2] proposed a method to calculate self-similarity based on a pyramid of histograms of oriented luminance gradients (PHOG, [3]).

Interestingly, large subsets of visual artworks share self-similar properties with images of natural scenes ([16, 6]). Recent progress in the field of experimental aesthetics demonstrates that self-similarity, in combination with other features, allows distinguishing artworks from non-artistic images like photographs of everyday objects or human faces ([4, 15, 7]).

Most algorithms that were developed to measure self-similarity have the disadvantage that they focus on specific image features (for example, luminance gradients) while partially or completely neglecting other features (for example, color), which are as relevant for human perception. In the present work, we therefore propose a method that uses diverse features learned by Convolutional Neural Networks (CNNs); these features are strikingly similar to those found in the human visual system [19]. In particular, besides edge information, they take into account color features and spatial frequency information as well, and thereby mirror visual system function more closely than the other methods mentioned above.

We ask the following questions:

- Are features learned by CNNs suitable to define a measure of self-similarity?
- How does a CNN-based measure of self-similarity compare to previously used measures, in particular the PHOG measure?
- Can one make use of the increasingly abstract representations that are obtained by consecutive filtering in CNNs to define a measure of self-similarity that focuses on image content rather than on basic structural image features?

The remainder of this article is organized as follows: The next section introduces the previously proposed method for measuring self-similarity using PHOG. After we briefly introduce CNNs, we will describe our approach, which is based on CNN features. Next, we will introduce the image datasets that we use in our experiment and will present our experimental results. An answer to the above questions will be given in the last section of this article.

Previous Work: PHOG Self-Similarity

Self-similarity as proposed by [2] is based on a pyramid representation of histograms of oriented gradients (PHOG, [3]). Histograms of oriented gradients (HOG, [5]) were originally designed for object recognition, whereas their pyramid representation was used to estimate the similarity between two images by calculating the distance of their respective PHOG descriptors. Amirshahi et al. [2] proposed the following algorithm for calculating the self-similarity of an image: First, the HOG is calculated for the entire image, referred to as layer 0. Subsequently, the image is divided into four equally sized subregions and the HOG is calculated for each subregion, respectively. The resulting histograms are referred to as histograms on layer 1. This procedure can be repeated up to layer l, on which the image is divided into 4^l equally sized subregions. The gradient orientations are then binned into k equal intervals and each histogram is normalized.

To assign a measure of self-similarity, the histogram of each subregion is compared to the histogram at a lower level by using the Histogram Intersection Kernel (HIK), which provides a measure of similarity of two normalized histograms $\mathbf{h}$ and $\mathbf{h}'$ with k bins, and is defined as follows:

$$\mathrm{HIK}(\mathbf{h},\mathbf{h}') = \sum_{i=1}^{k} \min(h_i, h_i') \tag{1}$$

In the first approach published by the authors, the measure of self-similarity is calculated as the median of the HIK of each subregion and its respective parent region:

$$M_{\mathrm{Se}}(I,L) = \mathrm{median}(\mathrm{HIK}(h_s, h_{\mathrm{Pr}(s)}) | s \in \mathrm{Sections}(I,L)) \tag{2}$$

where I is the image, L is the level, on which the self-similarity is computed, h_s is the HOG value for a sub-image s in the Sections(I,L) and Pr(s) corresponds to the parent of sub-image s. In more recent publications of the PHOG method, different choices were made with regard to which levels were used for comparison. Here, we will adopt the method by [4], where all subregions on level 3 are compared to the ground level. Further improvements were suggested by [15], who extended the measure of self-similarity to be used on color images by first converting the image to Lab color space and computing the gradient image as follows:

$$G_{\max}(x,y) = \max(\|\nabla I_L(x,y)\|, \|\nabla I_a(x,y)\|, \|\nabla I_b(x,y)\|). \quad (3)$$

Although this method detects color gradients in the opponent (red-green and blue-yellow) color space, color gradients do not contribute substantially to the resulting gradient image and, furthermore, any information on color differences is lost in the ensuing analysis. In the present work, we will overcome this restriction by using a different approach.

Defining Self-Similarity Based on CNN Filters

CNNs were first proposed by [12] and have gained huge popularity in recent years because they yielded record-breaking results in various image classification challenges (for example, see [11], [18]). CNNs learn a hierarchy of different filters that are applied to an input image, allowing them to extract meaningful information that is needed for detecting or classifying objects within an image. Recent progress in computing technology and the availability of huge amounts of data for training have helped to increase their popularity even further. Various architectures that consist of different building parts have been investigated. We will not describe these models in full detail here, but instead give a short overview of how CNNs process images.

Typically, CNNs include some of the following layer types: (i) *Convolutional layers*, in which the channels of the previous layer are convolved with a set of different filters and then transformed by a nonlinear activation function. (ii) Another widely used layer type is a *pooling layer*, which performs a subsampling operation, either by averaging or taking the maximum response over image regions of specified sizes. (iii) The model we use in our experiments introduces an additional *normalization layer*, which performs local brightness normalization of filter responses. Different layer types can be stacked on top of each other, so that consecutively applied filtering of the input image extracts more and more abstract features. (iv) Finally, *fully connected layers* on top are used to assign a class label to each input image, based on the features extracted by the previous layer. This list is not comprehensive and other layer types have been proposed in ongoing research.The filters are learned in a supervised manner with the backpropagation algorithm, which compares a class label assigned by the network with the true label and changes the network parameters according to their contribution to the current error.

Here, we adopt and modify the method proposed by [2]. Instead of using edge detectors, we use filters learned by CNNs. In doing so, we gain two advantages over the original method. Firstly, color is taken into account as an independent property and not by detecting opponent color edges only. Moreover, CNN filters respond to a range of spatial frequencies. Secondly, CNN structure is hierarchic and encodes more and more abstract features at higher layers, finally representing objects and shapes. We can make use of this structure to develop a self-similarity measure that reflects image content and goes beyond focusing on low-level features, such as edges.

In our experiments, we use the architecture proposed by [11] as provided in the Caffe Library [9], but drop the fully connected layers on top. This modification allows us to resize the input to have a dimension of 600×800 pixels.

Our measure of self-similarity is obtained as follows: First, every image is processed by the network, which yields 96 filter response maps after the first convolutional layer, 256 after the second, 384 after the third and fourth, and 256 after the last convolutional layer in the network. Using these filter responses, we then build a histogram of the maximum responses in subregions of the image, i. e. we perform a max-pooling operation over a grid of equally sized areas in the image. Our measure of self-similarity is calculated as follows:

$$M_{\mathrm{Se}}(I,L) = \mathrm{median}(\mathrm{HIK}(h_s, h_G) | s \in \mathrm{Sections}(I,L)) \quad (4)$$

Similar to the originally proposed method [2], a histogram h_G of filter responses obtained from image I at the ground level is compared to all histograms at level L. Following the approach by [15], we chose to use level 3 for our calculations, which results in 8×8 equally sized subregions. According to the authors, calculations on level 3 are suited best to distinguish image categories in their experiments; beyond that level, results are more unstable. The median of all calculated values yields the final measure of the self-similarity of an image. A value of 1 indicates high self-similarity, whereas a value approaching 0 is obtained for an image of low self-similarity. Synthetic example images that illustrate the advantage of using CNN features over PHOG features are shown in Figure 1.

Image Database

In this study, we used a total of eight different categories of images, namely images of man-made things like objects and urban scenes, as well as several sets of more or less self-similar natural patterns like clouds, plant patterns, large vistas, branches and lichen (Table 1) [15]. In addition, we compare these categories to a set of artworks of Western provenance (JenAesthetics dataset, [1]), which consists of 1629 different images and covers different artistic styles like Renaissance, Baroque, Romanticism, Realism, Impressionism, Modern Art, etc., ranging from the 15th to the 21st century. All other categories were taken with a Canon camera (EOS 500D) by one of the authors. Example images can be found in Fig. 2. All images were scaled to a resolution of 600×800 pixels, because a fixed resolution is required by the defined input resolution of the CNN.

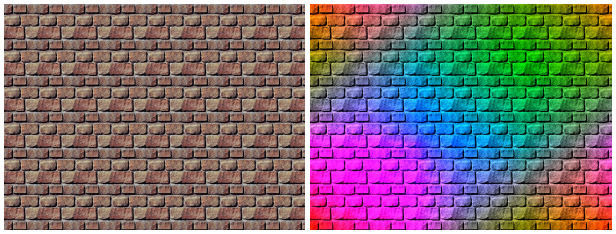
Experimental Results

CNN Features on the First Layer

We compared our method to calculate self-similarity by using CNN features to the PHOG-based method, which solely relies on edge information. Hence, we calculated the two measures for

synthetic images and for different image categories. Synthetic images are shown in Figure 1 and illustrate the advantage of using CNN features over PHOG features. As expected for CNN features, differences in color and spatial frequency are reflected in a change of the value of self-similarity, whereas these differences have almost no effect when using PHOG features.

Figure 3 shows a plot of the results for all image categories analyzed; mean values and standard deviations are given in Table 1. Natural patterns like lichen, branches or plant patterns, which are known to be highly self-similar, yield the highest self-similarity values with our CNN-based measure. Images of objects, which show low self-similarity in general, yield relatively low values with both methods. These results suggest that our novel measure is well suited to calculate self-similarity. The average self-similarity increases slightly from 0.812 using PHOG to 0.835 using CNNs. A likely explanation for this increase is that the CNN-based measure reflects color also, which is more homogeneous across most images, especially for natural images, whereas the PHOG method relies on edges exclusively. Branches, for example, are often foreground to a blue sky (Fig. 4b), which yields higher self-similarity with the CNN method; the PHOG method ignores this homogeneous background coloration. Similarly, urban scenes often feature large patches of sky, and, in addition, the facades of depicted buildings show rather uniform coloring, which make the images more self-similar with the CNN method. Interestingly, clouds slightly decrease in self-similarity and show a higher variance for the CNN-based values compared to the PHOG-based values.



(a) PHOG: 0.95, CNN: 0.97 (b) PHOG: 0.94, CNN: 0.81

(c) PHOG: 0.98, CNN: 0.94 (d) PHOG: 0.97, CNN: 0.75

Figure 1: Synthetic images that illustrate the advantage of using CNN features over PHOG features. The images on the top (1a, 1b) show a prominent change in color, which is reflected in a change of self-similarity only when using CNN features. On the bottom row (1c, 1d), a similar effect can be observed for a change in spatial frequency: While the PHOG-derived measure does not reflect this change, the CNN-based measure drops notably.

This variance may be due to a heterogeneity of the images in the dataset: multiple fleecy clouds are highly self-similar (Fig. 4c), whereas single large clouds (Fig. 5c) yield low values of self-similarity. Further examples of how color contributes to the two measure can be found in Figures 4 and 5.

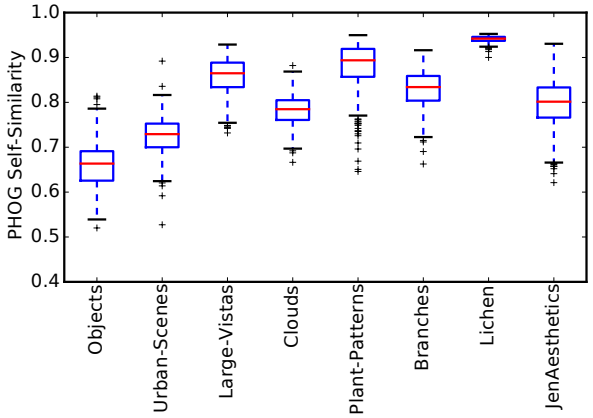
CNN Features on Higher Layers

Besides developing features akin to human vision, another intriguing property of CNNs is the increasingly abstract nature of features on higher layers. While on the first layer, edges and colored areas of an image are detected, features are grouped on higher layers, building blob detectors or even more abstract features that respond to distinct objects [20]. Here, we ask whether we can make use of this property in order to measure not only self-similarity of edges and color, but also self-similarity of more abstract (higher-level) features which may reflect content. Hence, besides using features of the first convolutional layer as presented

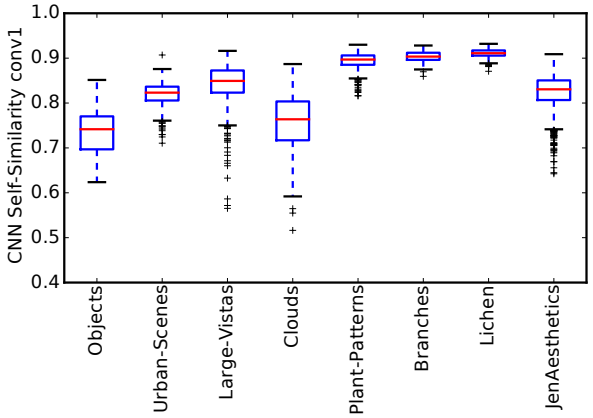
Figure 2: Example images of the categories we use in our experiment. From left to right, top row to bottom: objects, urban scenes, large vistas, clouds, plant patterns, branches, lichen and an image from the JenAesthetics dataset. The artwork shows *Isaac's Servant Tying the Bracelet on Rebecca's Arm*, painted by Benjamin West in 1775.

above, we also chose to use features of convolutional layers $2-5$ to calculate self-similarity. We used the same method as described above, but replaced the histograms that were computed on the first layer responses by histograms from higher convolutional layers. Results are given in Tables 2 and 3.

The mean self-similarity decreases from 0.835 on convolutional layer 1 (Table 1) to around 0.660 on convolutional layer 2, being relatively stable on the following layers until dropping again drastically to 0.33 on convolutional layer 5 (Tables 2 and 3). Interpreting these results is not straightforward and would be highly speculative, especially for higher layers, since it is not certain what specific image features that layers capture exactly.



(a)



(b)

Figure 3: Comparison between self-similarity defined on simple edges (PHOG, a) and on the first-layer features of a CNN (b). The overall results tend to be comparable, while subtle changes can be found for individual categories: branches and urban scenes become more self-similar because images of this category typically share large patches of similar color, such as sky or facades. Plant patterns show less variance because these images are mostly single-colored.

Table 1: Comparison between PHOG self-similarity and CNN self-similarity calculated with features on convolutional layer 1.

Category	Images	PHOG self-sim.	CNN self-sim.
Objects	207	0.660 ± 0.055	0.732 ± 0.049
Urban scenes	219	0.724 ± 0.044	0.818 ± 0.029
Large vistas	473	0.859 ± 0.041	0.840 ± 0.048
Clouds	268	0.783 ± 0.035	0.757 ± 0.064
Plant patterns	331	0.880 ± 0.053	0.893 ± 0.020
Branches	301	0.828 ± 0.043	0.903 ± 0.011
Lichen	244	0.941 ± 0.007	0.910 ± 0.010
JenAesthetics	1629	0.798 ± 0.048	0.826 ± 0.037
All	3672	0.812 ± 0.076	0.835 ± 0.060

Comparing the different image categories (Tables 1 and 2), one can observe a change when going from first-layer features to second-layer features. For example, lichen growth patterns are the most self-similar images in our experiment on layer 1, but fall behind on higher layers, being less self-similar than plant patterns

(a) P: 0.65, C: 0.87 (b) P: 0.66, C: 0.87 (c) P: 0.69, C: 0.80

Figure 4: Example images that are more self-similar when computed with CNN features (C) of the first convolutional layer than when computed with PHOG features (P). The edges in the subregions of the image differ, i. e. they are not self-similar, but the homogeneous colorations (green for plant patterns and blue for skies) result in higher self-similarity values for the CNN-based method.

(a) P: 0.87, C: 0.57 (b) P: 0.69, C: 0.51 (c) P: 0.87, C: 0.56

Figure 5: Example images that are less self-similar when computed with CNN features of the first convolutional layer than when computed with PHOG. (a) When color is taken into account, our measure of self-similarity differentiates between the green field and the blue sky. (b) The color difference of the red candle, yellow matchboxes and the shadow in the upper part of the image is neglected by the PHOG method but covered by the CNN method. (c) The cloud features many different shades from white to gray and blue, so that the image is less self-similar overall.

Table 2: CNN-based self-similarity computed on features of convolutional layers 2 and 3.

Category	conv2	conv3
Objects	0.630 ± 0.054	0.594 ± 0.037
Urban scenes	0.674 ± 0.042	0.631 ± 0.035
Large vistas	0.632 ± 0.042	0.651 ± 0.038
Clouds	0.599 ± 0.071	0.613 ± 0.044
Plant patterns	0.708 ± 0.049	0.713 ± 0.024
Branches	0.698 ± 0.029	0.698 ± 0.012
Lichen	0.693 ± 0.019	0.728 ± 0.012
JenAesthetics	0.658 ± 0.050	0.669 ± 0.037
All	0.660 ± 0.056	0.667 ± 0.048

Table 3: CNN-based self-similarity computed on features of convolutional layers 4 and 5.

Category	conv4	conv5
Objects	0.605 ± 0.046	0.319 ± 0.051
Urban scenes	0.630 ± 0.054	0.347 ± 0.056
Large vistas	0.592 ± 0.041	0.270 ± 0.053
Clouds	0.622 ± 0.038	0.262 ± 0.031
Plant patterns	0.689 ± 0.048	0.400 ± 0.078
Branches	0.666 ± 0.022	0.369 ± 0.036
Lichen	0.667 ± 0.021	0.363 ± 0.055
JenAesthetics	0.637 ± 0.054	0.341 ± 0.076
All	0.637 ± 0.054	0.334 ± 0.076

and branches when using second-layer features.

Figure 6 shows the three most self-similar images for the first layer (Fig. 6a–6c) and the second layer (Fig. 6d–6f). While a self-similar pattern on the first layer depends on edges and color only, our measure provides a more abstract detection of self-similarity on the second layer where repeating patterns like twigs or leaves make an image more self-similar. The same can be observed for the category of artworks. Example images are shown in Figure 7. On the first layer, the three most self-similar images are all impressionist paintings, which are composed of many colored brush strokes that are distributed across the entire image. In these images, high self-similarity probably results from the many repetitive edges between the adjacent brush strokes. One the next layer, repeating pictorial elements, such as color and luminance transitions that delineate faces and body parts, may contribute to their high self-similarity. Therefore, we conclude that second-layer features are suitable for defining a measure of self-similarity that focuses more on abstract features in an image.

The above results prompt the question whether self-similarity can be calculated also with even more abstract features above layer 2. In our experiment, we did not obtain an added value to detect generic image features when going beyond layer 2 (see example images in Figures 6g–6i and 7g–7i). Rather, on convolutional layer 3, the pictorial elements that constitute the high self-similarity seem to consist of particular color combinations and more complex forms that are seen in depictions of leaves (Figure 6h,6i), fruits and poppy flowers (Fig. 7h–7i), some of which the CNN may have seen during training. These results are compatible with a recent study by [19] who investigated the generality of CNN features by transferring layers between networks that had been trained on different tasks. The authors then retrained the networks on other tasks and found the first two layers to be universal in the sense that they were interchangeable after training between different tasks. Above the second layer, namely on layers 3 to 5, generality was no longer observable as performance dropped. This finding indicates that the features learned on the higher layers may not perform well simply due to the fact that they are trained on other categories of images. Any self-similarity detected at higher levels may thus be limited to objects, which the network has seen before. This limitation makes higher-layer features unsuitable for the case of detecting self-similarity in arbitrary images.

(a) conv1: 0.93 (b) conv1: 0.93 (c) conv1: 0.93

(d) conv2: 0.80 (e) conv2: 0.81 (f) conv2: 0.80

(g) conv3: 0.76 (h) conv3: 0.76 (i) conv3: 0.77

Figure 6: (a)-(c) The images in the dataset that are most self-similar when using CNN features of the first convolutional layer. (d)-(f) The images that are most self-similar when using the features of the second convolutional layer. While on the first layer, self-similarity is defined by fine detail like small color blobs, the self-similarity of the images in the second row seems to reflect more abstract image features like leaves or twigs. (g)-(i) Using features above the second layer results in no observable advantages over layers 1 and 2.

Conclusion and Outlook

In this work, we investigated how suitable features learned by CNNs are to define self-similarity. Features of early layers in a CNN are known to be generic for object recognition and resemble edge, color and spatial frequency detectors in the human vision system. Therefore, we propose a novel measure of self-similarity based on CNN filters, which goes beyond the simple edge detectors that are used in the PHOG method [2]. By analyzing exemplary image sets, we demonstrate the advantage of

using color and spatial frequency information in addition to edge information. When we used second-layer features to define our self-similarity measure, more complex features are detected and, consequently, the focus shifts from edge and color information to the content of the image.

In future studies, the measure of self-similarity proposed by us could be applied to the field of computational aesthetics, a subfield of computer vision. In this field, CNN models with their possibility to apply end-to-end feature learning have been adapted to assess image quality [13, 10]. While CNN models show good performance in this task, their interpretability lacks behind, i. e. understanding what makes some images more appealing than others is difficult with these models. In experimental aesthetics, another (psychological) approach to study image preference, specific image statistics are studied in images, such as artworks, which are aesthetically preferred by human observers [4, 6, 7, 8, 15, 16]. In the present study, we combined these two approaches by using psychologically inspired deep features for the explicit design of a novel measure for self-similarity, an image property that has been related to artworks previously [15, 2]. How useful this measure is for studying human preferences for particular visual stimuli or for distinguishing artworks from other types of image categories, remains to be established in future experiments.

(a) conv1: 0.90 (b) conv1: 0.91 (c) conv1: 0.91

(d) conv2: 0.75 (e) conv2: 0.75 (f) conv2: 0.76

(g) conv3: 0.73 (h) conv3: 0.73 (i) conv3: 0.74

Figure 7: The top rows shows the most self-similar artworks when using features of convolutional layer 1, the middle row when using features of convolutional layer 2 and the bottom row when using features of convolutional layer 3. While repetitive, fine colored brush strokes make images of impressionist paintings highly self-similar on the first layer, images tend to display related content and repeating patterns on the second and third level.

References

[1] Seyed Ali Amirshahi, Joachim Denzler, and Christoph Redies. JenAesthetics-a Public Dataset of Paintings for Aesthetic Research. Technical report, 2013.

[2] Seyed Ali Amirshahi, Michael Koch, Joachim Denzler, and Christoph Redies. PHOG analysis of self-similarity in esthetic images. *Proc. SPIE (Human Vision and Electronic Imaging XVII)*, 8291:82911J, 2012.

[3] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Representing shape with a spatial pyramid kernel. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, CIVR '07, pages 401–408, New York, NY, USA, 2007. ACM.

[4] Julia Braun, Seyed Ali Amirshahi, Joachim Denzler, and Christoph Redies. Statistical image properties of print advertisements, visual artworks and images of architecture. *Frontiers in Psychology*, 4(808), 2013.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *In CVPR*, pages 886–893, 2005.

[6] Daniel J Graham and David J Field. Statistical regularities of art images and natural scenes: Spectra, sparseness and nonlinearities. *Spatial Vision*, 21(1-2):149–164, 2007.

[7] Daniel J. Graham and Christoph Redies. Statistical regularities in art: Relations with visual coding and perception. *Vision Research*, 50(16):1503 – 1509, 2010.

[8] Caroline M. Hagerhall, Terry Purcell, and Richard Taylor. Fractal dimension of landscape silhouette outlines as a predictor of landscape preference. *Journal of Environmental Psychology*, 24(2):247–255, 2004.

[9] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[10] Yueying Kao, Ran He, Kaiqi Huang, and Tieniu Tan. Deep aesthetic quality assessment with semantic information. *arXiv preprint arXiv:1604.04970*, 2016.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[12] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[13] Xin Lu, Zhe Lin, Hailin Jin, Jianchao Yang, and James Z Wang. Rating image aesthetics using deep learning. *IEEE Transactions on Multimedia*, 17(11):2021–2034, 2015.

[14] Heinz-Otto Peitgen, Hartmut Jürgens, and Dietmar Saupe. *Chaos and Fractals*. Springer-Verlag, New York, 1992.

[15] Christoph Redies, Seyed Ali Amirshahi, Michael Koch, and Joachim Denzler. Phog-derived aesthetic measures applied to color photographs of artworks, natural scenes and objects. In Andrea Fusiello, Vittorio Murino, and Rita Cuc-

chiara, editors, *Computer Vision – ECCV 2012. Workshops and Demonstrations: Florence, Italy, October 7-13, 2012, Proceedings, Part I*, pages 522–531. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[16] Christoph Redies, Jens Hasenstein, and Joachim Denzler. Fractal-like image statistics in visual art: similarity to natural scenes. *Spatial Vision*, 21(1):137–148, 2007.

[17] Eero P. Simoncelli and Bruno A. Olshausen. Natural image statistics and neural representation. *Annual Review of Neuroscience*, 24:1193–1216, 2001.

[18] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[19] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *CoRR*, abs/1411.1792, 2014.

[20] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *Deep Learning Workshop, International Conference on Machine Learning (ICML)*, 2015.

Author Biography

Anselm Brachmann received his M.Sc. in the field of Computer Science in 2013 from Freie Universität Berlin. From 2014 to 2015, he worked at the Neurorobotics Research Laboratory at Humboldt Universität Berlin. He joined the Experimental Aesthetics Group in Jena in late 2015.

Christoph Redies received his MD from the University of Göttingen (1977) and his PhD in neurosciences from McGill University (1983). He carried out postgraduate research at M.I.T., Kyoto University, the Max Planck Institute in Tübingen and the University of Freiburg. Since 2003, he is a full professor at the University of Jena. During the last decade, he performed research in the emerging field of experimental aesthetics, focusing on statistical image properties.