

## Semantically Interpretable Activation Maps: what-where-how explanations within CNNs

2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW 2019)

Marcos Gonzalez, D.; Lobry, Sylvain; Tuia, D.

<https://doi.org/10.1109/ICCVW.2019.00518>

This publication is made publicly available in the institutional repository of Wageningen University and Research, under the terms of article 25fa of the Dutch Copyright Act, also known as the Amendment Taverne. This has been done with explicit consent by the author.

Article 25fa states that the author of a short scientific work funded either wholly or partially by Dutch public funds is entitled to make that work publicly available for no consideration following a reasonable period of time after the work was first published, provided that clear reference is made to the source of the first publication of the work.

This publication is distributed under The Association of Universities in the Netherlands (VSNU) 'Article 25fa implementation' project. In this project research outputs of researchers employed by Dutch Universities that comply with the legal requirements of Article 25fa of the Dutch Copyright Act are distributed online and free of cost or other barriers in institutional repositories. Research outputs are distributed six months after their first online publication in the original published version and with proper attribution to the source of the original publication.

You are permitted to download and use the publication for personal purposes. All rights remain with the author(s) and / or copyright owner(s) of this work. Any use of the publication or parts of it other than authorised under article 25fa of the Dutch Copyright act is prohibited. Wageningen University & Research and the author(s) of this publication shall not be held responsible or liable for any damages resulting from your (re)use of this publication.

For questions regarding the public availability of this publication please contact [openscience.library@wur.nl](mailto:openscience.library@wur.nl)

# Semantically Interpretable Activation Maps: what-where-how explanations within CNNs

Diego Marcos  
Wageningen University  
The Netherlands  
diego.marcos@wur.nl

Sylvain Lobry  
Wageningen University  
The Netherlands  
sylvain.lobry@wur.nl

Devis Tuia  
Wageningen University  
The Netherlands  
devis.tuia@wur.nl

## Abstract

A main issue preventing the use of Convolutional Neural Networks (CNN) in end user applications is the low level of transparency in the decision process. Previous work on CNN interpretability has mostly focused either on localizing the regions of the image that contribute to the result or on building an external model that generates plausible explanations. However, the former does not provide any semantic information and the latter does not guarantee the faithfulness of the explanation. We propose an intermediate representation composed of multiple Semantically Interpretable Activation Maps (SIAM) indicating the presence of predefined attributes at different locations of the image. These attribute maps are then linearly combined to produce the final output. This gives the user insight into what the model has seen, where, and a final output directly linked to this information in a comprehensive and interpretable way. We test the method on the task of landscape scenicness (aesthetic value) estimation, using an intermediate representation of 33 attributes from the SUN Attributes database. The results confirm that SIAM makes it possible to understand what attributes in the image are contributing to the final score and where they are located. Since it is based on learning from multiple tasks and datasets, SIAM improve the explainability of the prediction without additional annotation efforts or computational overhead at inference time, while keeping good performances on both the final and intermediate tasks.

## 1. Introduction

Deep learning (DL) models are nowadays entering many fields of application due to their clear advantages in terms of prediction accuracy. Among the different DL models, deep Convolutional Neural Networks (CNN) dominate the landscape of Computer Vision tasks and keep expectations aloft by promises of superhuman autonomous driving or health

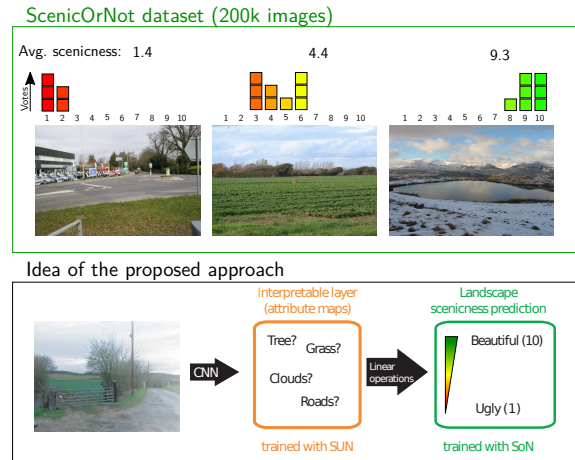


Figure 1. Examples of images from the ScenicOrNot dataset and corresponding crowdsourced scenianness scores (top). We propose to make use of two distinct datasets such that the final task, scenianness prediction, is solved by linearly combining the results on a more interpretable intermediate task, attribute prediction using the images from the SUN Attributes database (bottom).

diagnosis. At the same time, a drawback of DL is increasingly being put forward: the inscrutable nature of their decision making process. Often referred to as black boxes, CNNs don't allow to easily understand what elements of the input contributed to the output and in which way [16].

The end user might require an explanation that is simple enough to be easily interpretable, while the CNN needs to perform a highly complex set of operations to solve the task [7]. This creates a trade-off between how faithful the explanation is to the inner workings of the CNN and how interpretable it is [12]. In this paper, we argue that the explanation should actually be part of the model. This is achieved by an interpretable bottleneck ensuring that the explanation contains all the information being used to produce the result. This effectively eases the tradeoff between the interpretability of the prediction and the faithfulness of such interpretation, since the explanation will include both aspects

by design. However, this risks to create another trade-off, this time between interpretability and performance, due to the limiting capacity of the interpretable bottleneck.

We explore the possibility to gain interpretability by learning (and predicting) an intermediate semantic representation from auxiliary datasets on related tasks. We do so by constraining the bottleneck of a CNN to predict class-specific maps, which are useful to interpret the final decision of the model on an harder to interpret final task. We rely on the idea of interpretable decomposition [37], where we assume that the final task can be explained as a linear combination of a series of semantic contributions.

As the final task, we focus on the highly subjective visual problem of estimating landscape scenicness (i.e. aesthetic value) [27, 31]. We use a crowdsourced dataset from the ScenicOrNot<sup>1</sup> project (SoN, Fig. 1, top). The model needs to capture the average perception of a large amount of annotators. This subjectiveness makes it hard for a user to evaluate the faithfulness of the prediction of such model making it important to understand which visual elements led to the final decision. To provide evidence on the model’s inner decision process, we force it to use a combination of objective elements (a subset of 33 relevant SUN Attributes [23], Fig. 1, bottom) in its last intermediate representation layer, just before providing the scenicness score. By doing so, the user receives both the score and the relative contribution of each interpretable element as a set of attention maps: both can be further used to assess confidence and/or generate new knowledge about landscape preferences.

Our results suggest that it is possible to make a CNN predicting scenicness interpretable in terms of semantic landscape elements, and this without increasing the annotation or computational efforts and with a minimal decrease in terms of performance on the final task.

## 2. Related work

Interpretable deep learning for solving visual tasks is becoming a major research field. This section provides a review of the different strategies that have been explored in this direction.

**Attribute based zero-shot learning.** The relationship between attributes and classes is exploited for zero-shot learning, a learning setting in which some classes have no training samples, but can be related to known classes via a shared set of attributes. Given a set of known classes  $\mathcal{Y}$  with at least one training sample and a disjoint set of hidden classes  $\mathcal{Z}$  (no training samples), zero-shot learning [18] aims at assigning a new element from the input feature space  $\mathcal{X}$  to one class in  $\mathcal{Y} \cup \mathcal{Z}$ . This can be done by leveraging a set of attributes  $\mathcal{A}$ , also referred to as Semantic Output Codes [22],

common to  $\mathcal{Y}$  and  $\mathcal{Z}$ , that can be used to uniquely describe each class [6]. Direct Attribute Prediction (DAP) [17] is a family of methods for zero-shot learning in which two functions,  $f : \mathcal{X} \rightarrow \mathcal{A}$  and  $g : \mathcal{A} \rightarrow \mathcal{Y} \cup \mathcal{Z}$  are composed to perform classification.

Although not initially devised to improve interpretability, we propose to use an architecture inspired in DAP to make sure that the final result depends only on the learned attributes.

**Attributes without supervision.** Citing relevant visual attributes is an intuitive way of explaining an image-based decision. CNNs have been shown to automatically learn representations that are well correlated to visual attributes [5, 2] that can be leveraged to get an intuitive idea of what elements are used for the output [9, 21]. This behaviour can be improved further by adding a loss during training that makes the activation maps of each filter more attribute-like, such as by encouraging them to be class-specific and localized [34]. Nevertheless most individual filters in current approaches can not readily be assigned a semantic label [8].

**Interpretability by localization.** A direction to improve the interpretability of a CNN’s result is to point out which parts of the input contribute the most to the output. Researches considered occlusions of parts of the image as ways to assess the region’s importance [33, 25, 24]. Alternatively, Class Activation Maps (CAM [36]), use an average pooling operation on the last feature tensor, right before the last fully connected layer, to assess region importance. This allows to see which locations are being used the most to generate the output. Grad-CAM [26] and LRP [20] use the gradient information to backtrack an output to the input elements it is most sensitive to. Such localization methods have been shown to improve the perceived trustworthiness of DL models [25]

**Interpretability by generating explanations.** Interpretability by localization lacks the expressivity that is expected from explanations in human communication. This has been addressed by building an external model that is trained to generate a plausible explanation to the output of the visual model [10], which can then also be combined with localization [13, 11]. Another approach to present post-hoc semantic explanations is to decompose the activation map provided by a localization method, such as CAM, using an interpretable basis of maps [37], such that the final map is reconstructed using a combination of maps that are semantically interpretable.

**Compositionality.** Generally referred to the semantics of natural language, compositionality implies that the mean-

<sup>1</sup>[urlhttp://scenicornot.datasciencelab.co.uk/](http://scenicornot.datasciencelab.co.uk/)

ing of an expression is formed by the combination of the meanings of its parts. This principle can also be found in applications on images, such as for the task of extracting high-level information by combining low-level cues [30], and on videos, in order to use the presence of concepts, for instance individual actions and objects, to classify video sequences as belonging to an event category (*e.g.* wedding, sport event, *etc.*) [32]. Exploring the high-level representation space of CNNs using sets of images containing the same concept has been proposed as a way to hint these concept’s presence in the image [15]. Methods for making the flow of information in CNNs as local as possible are also researched to make the models more compositional [29, 28], since they encourage each individual region of the input to contribute to the output independently from its context.

**Joint learning of semantic hierarchies.** Tasks such as action recognition are well suited to a hierarchical representation, in which objects [14] and object sub-actions [35] can be learned jointly and combined to obtain the final result. Following the same logic, images can be represented by a semantic bottleneck that describes them and that can then be used for some downstream task. The bottleneck can adopt different forms, such as pieces of text that describe the image [3], objects [4, 19] or object parts and their attributes [1]. Such representations are intelligible for humans and can thus be easily interpreted.

In this work, we design an interpretable layer performing localization of objects and attributes in the image. Similarly to CAM, we use activation maps before the fully connected layers, but we force those maps to correspond to fixed concepts. We exploit the idea of compositionality by assuming that our main uncertain task (scenicness prediction) can be predicted by a linear combination of semantic interpretable concepts, which we learn in a supervised fashion. To do so, we use a dataset (SUN Attributes) disjoint from the one employed to train for the main task (SoN). In this way we exploit the semantic information contained in the auxiliary dataset and provide interpretable intermediate maps, as well as a transparent look at their importance in the final decision.

### 3. Semantically Interpretable Activation Maps (SIAM).

SIAM consists of an end-to-end trainable CNN based on a two-level hierarchical output with a DAP [17] structure. As in [34], we want to obtain interpretable feature maps without the need for any additional annotation. But instead of relying on an unsupervised loss, we make use of an already existing dataset that contains relevant attributes (or concepts). This removes the requirement of having to inspect a substantial part of the dataset to understand the

correct interpretation of each feature map, since the attributes are predefined. Our approach reassembles the interpretable basis representation method of [37], with the main difference that our system is trained end-to-end and does not allow a residual; all the high-level information used to solve the final task must be contained in the interpretable maps. Our model also uses the average pooling technique of CAM [36] to also provide the approximate location on the image of each attribute without the need for any positional ground truth and without any additional overhead at inference time.

Figure 2 summarizes the proposed Semantically Interpretable Activation Maps (SIAM) architecture using as example a subset of the SUN Attributes as the semantic bottleneck  $\mathcal{A}$  and landscape scenicness as final output  $\mathcal{Y}$ . The first block of the model,  $f : \mathcal{X} \rightarrow \mathcal{A}$ , outputs as many feature maps as there are attribute classes (Section 3.1). This first level output is used as input to the second block,  $g : \mathcal{A} \rightarrow \mathcal{Y}$ , which multiplies each map with a learned spatial template (see Fig. 4 for a visualization of templates after training), and linearly combines the resulting activations to obtain the final scenicness score (Section 3.2). This direct dependence between the attribute maps  $\mathcal{A}$  and the final output  $\mathcal{Y}$  allows to understand what elements are being detected and how they are contributing to the output.

The two blocks are trained jointly using the corresponding datasets providing labels for attributes and scenicness, respectively (Section 3.3).

#### 3.1. Predict attributes with $f$

As  $f$  we use a standard CNN architecture, ResNet-50. Given an input image  $\mathbf{x} \in \mathcal{X} = \mathbb{R}^{M \times M \times 3}$ , the output  $f(\mathbf{x})$  is a tensor  $\mathbf{A} \in \mathcal{A} = \mathbb{R}^{m \times m \times A}$  of activation maps (one map  $\mathbf{A}_i \in \mathbb{R}^{m \times m}$  for each attribute  $i \in \{1, \dots, A\}$ ). In practice we have used  $M = 500$  and  $m = 15$ , but only a center crop of the maps of size  $11 \times 11$  is used, to reduce potential border effects. An average pooling is then applied to the maps to return a vector  $\mathbf{a} \in \mathbb{R}^A$  of length  $A$ , the number of attribute classes. The elements  $a_i \in \mathbf{a}$  are subject to a sigmoid non-linearity before being compared to the ground truth attribute annotation  $\bar{a}_i \in \bar{\mathbf{a}}$  via a multi-class binary cross entropy:

$$L_{SUN} = \frac{1}{A} \sum_i [\bar{a}_i \log(a_i) + (1 - \bar{a}_i) \log(1 - a_i)], \quad (1)$$

#### 3.2. Combine attributes for the final result with $g$

We choose the function  $g$ , with the output  $y = g(\mathbf{A})$  being a scalar, to be formed by a concatenation of linear operators, making the mapping between  $\mathcal{A}$  and  $\mathcal{Y}$  easily interpretable as a single linear mapping that provides a template for each attribute. This operation can be thought of as a

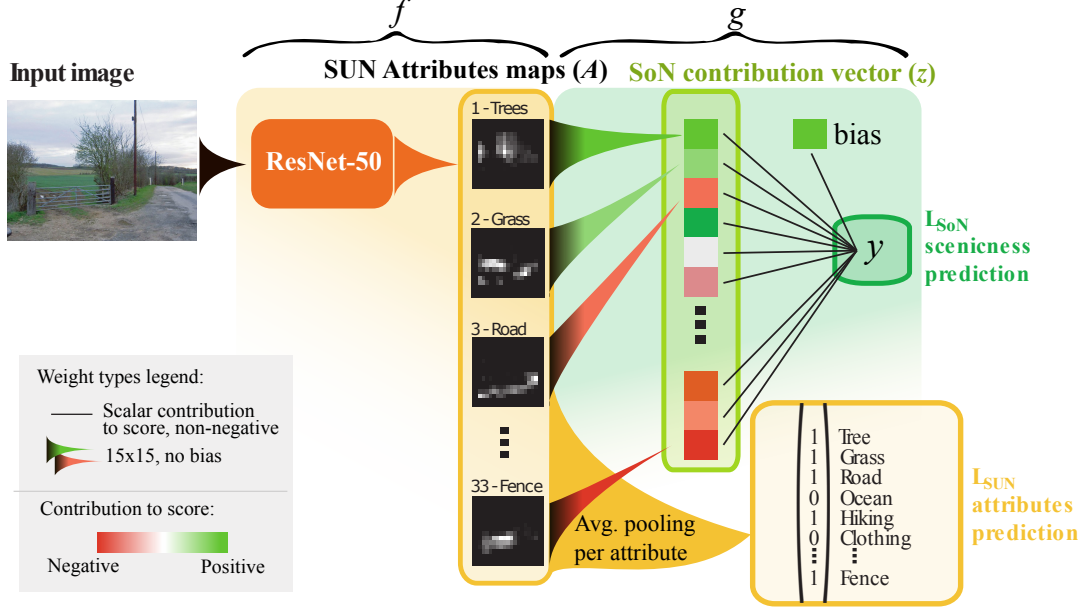


Figure 2. Flowchart of the proposed model. Each attribute map is multiplied with a learned template (see Fig. 4 for a visualization), and the resulting activations are linearly combined to obtain the scenicsness score. If the input image belongs to the SUN database, only the  $L_{SUN}$  loss is computed and  $f$  is updated. If it belongs to the SoN database, only the  $L_{SoN}$ , which allows to update  $f \circ g$ .

weighted average pooling, where each element of the template indicates the weight of the attribute, positive or negative, towards the final output at every spatial location (see Figure 4 for the templates obtained for the SUN attribute classes).

A single fully connected template  $\mathbf{T}^i$  is multiplied element-wise with each individual attribute map, yielding a single scalar per attribute  $z^i = \mathbf{T}^i \cdot \mathbf{A}^i$ , without using a bias term. The absence of a bias ensures that the output is non-zero exclusively if the attribute has been detected. To initialize the templates we chose not to use any randomization, since that could inject biases in the how attributes affect the final output. Instead, we learn two non-negative templates for each attribute map. Both are then combined, one with a positive weight, the other with a negative weight. This allows to initialize both templates with a constant positive value, reducing the bias and leading to a spatially smoother result. The linear combination makes it algebraically equivalent to using a single template. The resulting vector  $\mathbf{z} \in \mathbb{R}^A$  is then linearly projected onto the output scalar  $y = \mathbf{w} \cdot \mathbf{z} + b$ , this time with a learnable bias, which can capture the average value of the output over the dataset. This output is then compared to the crowdsourced scenicsness value  $\bar{y}$  using a Square Error loss,

$$L_{SoN} = (y - \bar{y})^2. \quad (2)$$

### 3.3. Joint training

SIAM solves two inter-dependent tasks: the prediction of attributes,  $\mathbf{A} = f(\mathbf{x})$ , and the prediction of scenicsness

based on these attributes,  $y = g(\mathbf{A})$ . These two tasks involve two separate losses, needing annotations of attributes (Eq. (1)) or scenicsness scores (Eq. (2)), respectively. We use two different and non-overlapping datasets for these two tasks: the SUN Attributes database [23] is used to learn the attribute maps and the ScenicOrNot (SoN<sup>2</sup>) is used as a reference for the main task of predicting the scenicsness. In practice, we first fine-tune the first part of the model ( $f$ ) on the sub-task of predicting the attributes by minimizing  $L_{SUN}$ . Then, the network is finetuned again using both tasks,

$$L = L_{SUN} + 0.1L_{SoN}, \quad (3)$$

in order to learn  $f \circ g$  using samples from both the SUN and the SoN databases. Samples from either database are used alternately, and only one of the two losses propagates gradients at each time: when a SUN sample is considered, only  $L_{SUN}$  generates a learning signal; when the sample is from SoN, only  $L_{SoN}$  does. The contribution of  $L_{SUN}$  is set to be an order of magnitude larger than that of  $L_{SoN}$  to prevent the model from improving on scenicsness prediction at the expense of its performance on the attributes.

## 4. Results and discussion

To investigate the ‘performance vs. interpretability’ trade-off, we test the proposed approach on the problem of automatic landscape scenicsness estimation. We use the dataset provided by the SoN project, from which we ob-

<sup>2</sup><http://scenicornot.datasciencelab.co.uk/>

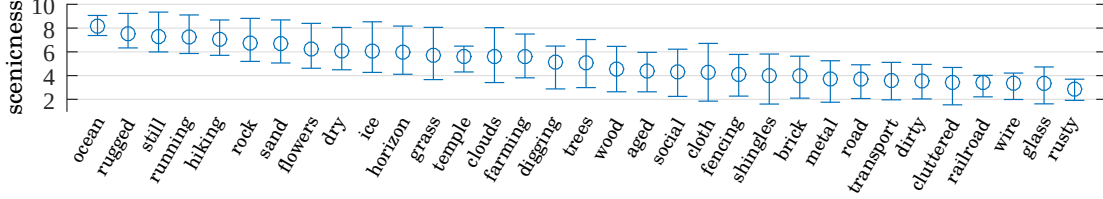


Figure 3. Co-occurrence between class presence in an image and its scenicness value on our dataset of annotated attributes over SoN images. Running and still refer to water bodies, and ice includes snow.

Data	Label	# images		
		Training	Validation	Test
SoN	$\mathcal{Y}$	150'000	20'000	30'000
SUN	$\mathcal{A}$	11'414	1'497	1'429
Son + SUN	$\mathcal{A}, \mathcal{Y}$	-	-	90

Table 1. Number of samples per dataset

tained the first 200'000 listed outdoor images from the UK with at least 3 crowd-sourced scenicness scores. Ordered by ID, we took the first 150'000 images for training, followed by 20'000 images for validation purposes and 30'000 for testing (Table 1).

Regarding the landscape attributes, they are learned using the SUN Attributes database [23], from which we have chosen 33 attributes relevant for our task (Figs. 3 and 4).

In addition to checking the scenicness estimation performances of our model, we need to verify that the attributes are being correctly predicted on the images from the SoN database. To this end, an additional set of 90 SoN images was labeled by 4 different annotators with the same 33 attributes from SUN. Note that this set of SoN images with SUN attributes is used for validation purposes only and it is never used during training of either block of SIAM.

#### 4.1. Performance on the original datasets

Table 2 shows the impact of using a constrained-but-interpretable representation bottleneck on the performance in both tasks. We report both the Root Mean Square Error (RMSE), as well as Kendall's  $\tau$  rank correlation coefficient, as in [27], to assess the performances on the scenicness estimation task. Average precision is reported for the attribute detection task [23].

As a baseline for SoN, we use a finetuned ResNet-50, pretrained on ImageNet, on the task of regressing scenicness values without making use of attributes. This results in a performance comparable to the one reported in [27], where the authors obtained values for Kendall's  $\tau$  ranging from 0.62 to 0.65 on their test set using finetuned DL models.

We observe that training our model SIAM in two separate steps,  $f$  only on SUN and then  $g$  only on SoN with  $f$  frozen, results in a substantial drop in accuracy, with a 25% increase in RMSE with respect to the baseline. However, finetuning the whole model,  $g \circ f$ , jointly on SoN and

	SoN RMSE	SoN $\tau$	SUN
Baseline	0.987	0.640	-
SIAM (ours)	1.01	0.607	0.418
SIAM (no finetuning)	1.24	0.496	0.331

Table 2. Numerical results. Scenicness prediction on the SoN test set (RMSE and Kendall's  $\tau$ ) and SUN attribute prediction on the SUN dataset (average precision). The last row corresponds to SIAM with only  $g$  trained on SoN, instead of the full model  $f \circ g$ .

SUN, not only reduces substantially this gap (by an order of magnitude, to 2.3% RMSE increase), but also significantly improves the prediction on the subset of SUN attributes: the average precision on the SUN test set increases from 0.33 to 0.41. This suggests that both tasks are correlated and confirms that optimizing jointly over the two losses does not penalize the attribute detection. This is of high importance for the final interpretability of the model, since a substantial drop in the attribute detection performance would risk making the interpretation of each attribute map meaningless.

#### 4.2. Attribute detection on ScenicOrNot images

The small set of 90 SoN images, selected to represent the whole range of scenicness values and annotated with SUN attributes, with 10 images randomly selected for each bracket between integer scenicness scores. This allows us to get an idea of the co-occurrence between the presence of these classes and the scenicness values, as shown in Fig. 3. We can see how a few classes, such as those related to water (*ocean*, *still water* and *running water*), *rugged* and *hiking* are very correlated with high scenicness values, while most man-made classes tend to co-occur with below average scenicness. A few other classes, such as *trees*, *grass*, *clouds* or *farming* are much less polarized in terms of their average associated scenicness.

Table 3 shows the numerical results on these 90 images. The models used here, including the baseline, are the same described in the previous section and were trained on the original training sets of SoN and SUN (Table 1). The over-representation of images with very low and very high scores favours both higher RMSE and Kendall's  $\tau$  values. On scenicness prediction, SIAM matches the performance of the unconstrained model in terms of RMSE, although still lags behind in terms of Kendall's  $\tau$ . We observe a sub-



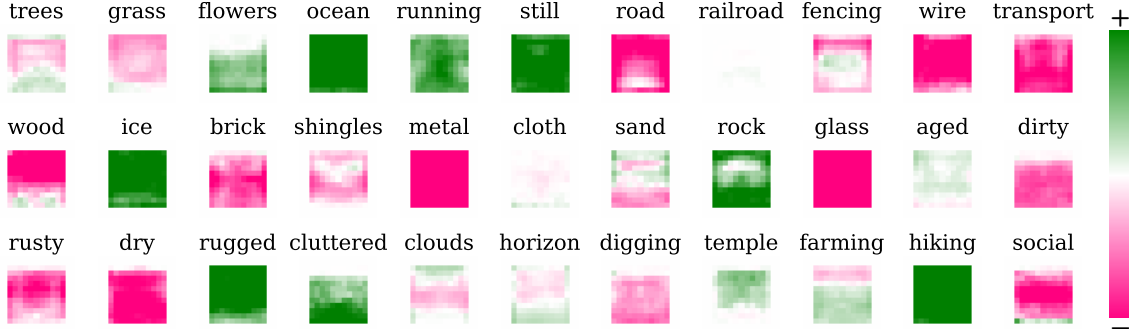


Figure 4. Learned templates on each attribute class that are used by the model for scenicity prediction. They allow to visualize how the presence of each attribute in different locations of the image influences the scenicity score.

	SoN RMSE	SoN $\tau$	SUN
Baseline	1.23	0.747	-
SIAM (ours)	1.22	0.700	0.501
SIAM (no finetuning)	1.68	0.526	0.449

Table 3. Numerical results on the 90 images SoN subset (SoN+SUN in Table 1). Scenicity prediction on the SoN test set (RMSE and Kendall’s  $\tau$ ) and SUN attribute prediction on the SUN dataset (average precision). The last row corresponds to SIAM with only  $g$  trained on SoN, instead of the full model  $f \circ g$ . The average agreement between the four annotators on the attributes is 0.496.

stantial improvement on the SUN attributes detection task, matching the average agreement between annotators, which is of 0.496. This suggests that the improvement on SUN attributes prediction observed in the SUN database (Table 2) generalizes well to SoN images, which do not contain SUN attribute labels at training time. This indicates that the semantic bottleneck can be indeed trusted when used to interpret the contribution of each attribute to the scenicity on the SoN images.

### 4.3. Visual analysis

**Attribute templates.** Figure 4 shows the per-attribute SIAM templates  $T^i$  learned for predicting the scenicity. Dark green represents a large positive impact on the scenicity when the attribute is present in the corresponding location, while dark magenta means a large negative contribution. White represents a null contribution. The templates in Fig. 4 show that the attributes that contribute most consistently are, on the positive side, *hiking*, *rugged*, water (*ocean* and *still*) and *ice* (which includes snow), and on the negative side *metal*, *glass*, *wire*, *transport* and *dry*. The remaining attributes show some level of location dependency (e.g. *farming* has some positive impact when in the bottom half of the image but a negative one when it is located on the top third of the image) but have an overall weaker impact on scenicity.

**Activation maps.** We analyze the images from the SoN test set in which our model and the baseline disagree the most. For each image, we show the eight predicted attribute maps that contribute the most to the scenicity, both positively (with a green frame) and negatively (with a magenta frame). The thickness of the frame around each map represents the magnitude of the contribution to the scenicity score. For the baseline and the proposed model we show the total activation maps, which show how the contributions are distributed spatially.

Figure 5 illustrates some examples where our proposed model (SIAM) performs well in attribute detection. The spatial distribution of the scenicity is similar in both SIAM and the baseline, but the elements that induce the latter to fail are not straightforward to discern, showcasing how explanations by localization might not always be satisfactory.

Figure 6 shows cases in which the error in scenicity can be easily attributed to misclassifications at the attribute level, allowing to correctly guess whether the model is over- or underestimating the score. In the top example, the reflection in the water is misclassified as road, impacting the prediction negatively. In the middle, the top of the phone booth is also misclassified as road, driving down the estimation of the score. In the bottom case, the reflection on the lake is predicted as ice, which is assigned a large positive contribution.

The examples in Fig. 7 represent cases where the baseline model captures subtleties related to attributes (or object classes) that are not explicitly considered. In this case, SIAM remains blind to those contributions, since it is constrained to use only the pre-selected classes in the interpretable bottleneck. In the top and middle examples, the attribute classes *metal* and *transport* are not able to capture subtleties such as the added aesthetic value of a vintage tram or a pleasant boat. In the last one, the positive scores of the *rugged* and *mountain* features overwhelm the dirtiness detected in a landfill.

These examples show that a fix and predefined set of attributes might not suffice, and a method for the discovery of

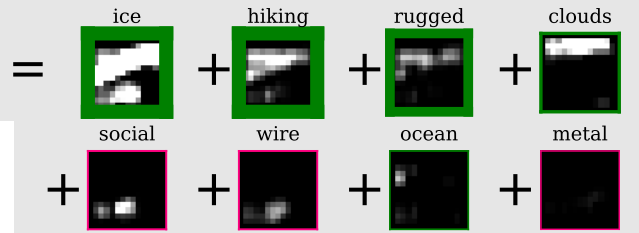
Crowd scenicness: 7.00



Our model: 7.12



Baseline: 5.21



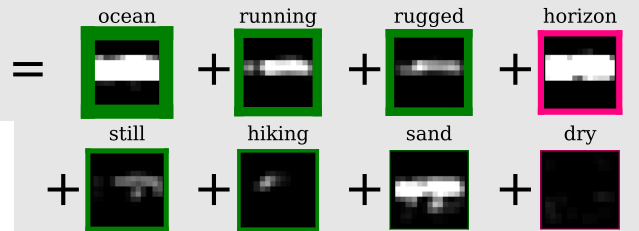
Crowd scenicness: 8.00



Our model: 7.36



Baseline: 5.12



Crowd scenicness: 2.60



Our model: 2.56



Baseline: 3.70

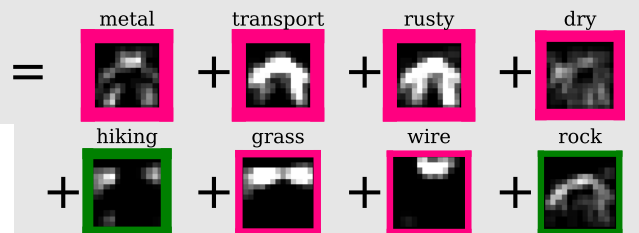


Figure 5. Examples in which SIAM predicts well both attributes and scenicness.

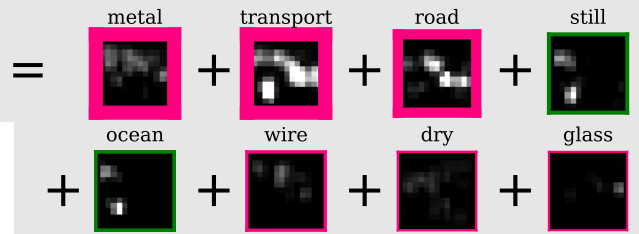
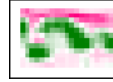
Crowd scenicness: 5.29



Our model: 3.35



Baseline: 4.54



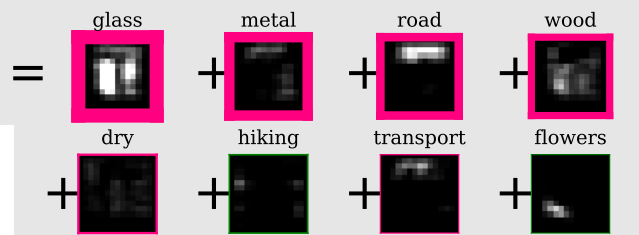
Crowd scenicness: 3.00



Our model: 2.58



Baseline: 4.09



Crowd scenicness: 6.33



Our model: 6.91



Baseline: 5.16

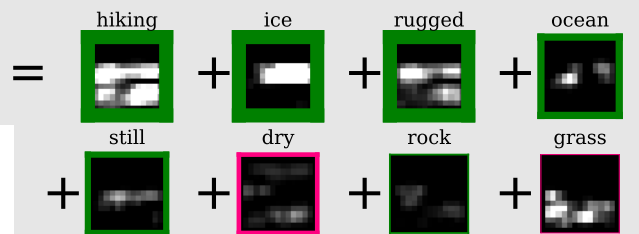


Figure 6. Examples in which SIAM mispredicts scenicness, but where this can be easily detected using the attribute prediction.



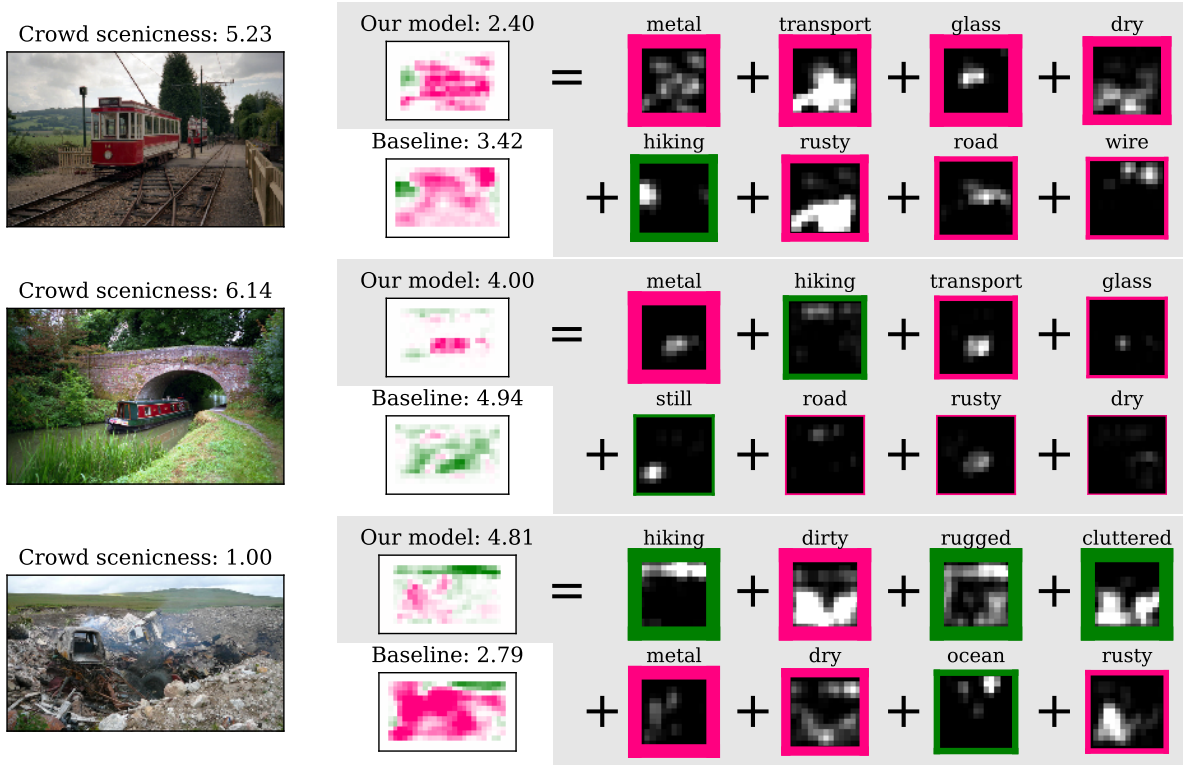


Figure 7. Failure cases in which SIAM mispredicts the scenicity without obvious mistakes on attributes prediction.

potentially useful attributes could play a role in the selection of additional attribute classes. In addition to this, we often see that, although reasonable, the activation maps do not always match well with the semantics of the image. This is due to the weakly supervised nature of the attribute learning process, and some additional supervision, in the form of segmentation maps, could help solve this issue.

## 5. Conclusion

We propose the use of a semantic bottleneck made of Semantically Interpretable Activation Maps (SIAM) to provide an explanation of a CNN’s output. These maps inform about what objective elements are relevant, where in the image they are, and how they contribute to the final prediction. We applied this method to the subjective task of landscape scenicity estimation, by forcing the model to use an information bottleneck that is jointly trained to predict a set of 33 landscape related attributes from the SUN Attributes database. Firstly, looking at the model layers that use the attribute maps as input, we can understand how the output will react to the presence of a given class at a given location on the images. Secondly, when an image is shown to the model, the activation maps and their contribution to the final score can help the user understand which elements are being used to construct the final score and get a hint about potential sources of errors. Despite a small loss of performance

(smaller than 2.5% in terms of RMSE) in scenicity estimation, we observed a boost in the attribute detection and, more importantly, a much richer source of interpretation of the predicted value, without needing additional annotation.

## References

- [1] K. E. Ak, A. A. Kassim, J. Hwee Lim, and J. Yew Tham. Learning attribute representations with localization for flexible fashion search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7708–7717, 2018.
- [2] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6541–6549, 2017.
- [3] M. Bucher, S. Herbin, and F. Jurie. Semantic bottleneck for computer vision tasks. In *Asian Conference on Computer Vision (ACCV)*, 2018.
- [4] Z. A. Daniels and D. Metaxas. Scenarionet: An interpretable data-driven model for scene understanding. In *IJCAI Workshop on XAI*, page 33, 2018.
- [5] V. Escorcia, J. Carlos Niebles, and B. Ghanem. On the relationship between visual attributes and convolutional networks. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1256–1264, 2015.
- [6] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVF/IEEE Conference*

- on *Computer Vision and Pattern Recognition (CVPR)*, pages 1778–1785, 2009.
- [7] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal. Explaining explanations: An overview of interpretability of machine learning. In *International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018.
  - [8] A. Gonzalez-Garcia, D. Modolo, and V. Ferrari. Do semantic parts emerge in convolutional neural networks? *International Journal of Computer Vision*, 126(5):476–494, 2018.
  - [9] M. Harradon, J. Druce, and B. Ruttenberg. Causal learning and explanation of deep neural networks via autoencoded activations. *arXiv preprint arXiv:1802.00541*, 2018.
  - [10] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell. Generating visual explanations. In *European Conference on Computer Vision (ECCV)*, pages 3–19. Springer, 2016.
  - [11] L. A. Hendricks, R. Hu, T. Darrell, and Z. Akata. Grounding visual explanations. In *European Conference on Computer Vision (ECCV)*, pages 269–286. Springer, 2018.
  - [12] B. Herman. The promise and peril of human evaluation for model interpretability. *arXiv preprint arXiv:1711.07414*, 2017.
  - [13] D. Huk Park, L. Anne Hendricks, Z. Akata, A. Rohrbach, B. Schiele, T. Darrell, and M. Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8779–8788, 2018.
  - [14] V. Kalogeiton, P. Weinzaepfel, V. Ferrari, and C. Schmid. Joint learning of object and action detectors. In *CVF/IEEE International Conference on Computer Vision (ICCV)*, pages 4163–4172, 2017.
  - [15] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*, pages 2673–2682, 2018.
  - [16] W. Knight. The dark secret at the heart of AI. *MIT Technology Review*, 2017.
  - [17] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014.
  - [18] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *Association for the Advancement of Artificial Intelligence*, volume 1, page 3, 2008.
  - [19] M. Losch, M. Fritz, and B. Schiele. Interpretability beyond classification output: Semantic bottleneck networks. *arXiv preprint arXiv:1907.10882*, 2019.
  - [20] G. Montavon, W. Samek, and K.-R. Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.
  - [21] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev. The building blocks of interpretability. *Distill*, 2018.
  - [22] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1410–1418, 2009.
  - [23] G. Patterson, C. Xu, H. Su, and J. Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81, 2014.
  - [24] V. Petsiuk, A. Das, and K. Saenko. Rise: Randomized input sampling for explanation of black-box models. In *British Machine Vision Conference (BMVC)*, 2018.
  - [25] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should I trust you?: Explaining the predictions of any classifier. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1135–1144. ACM, 2016.
  - [26] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *CVF/IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.
  - [27] C. I. Seresinhe, T. Preis, and H. S. Moat. Using deep learning to quantify the beauty of outdoor places. *Royal Society open science*, 4(7):170170, 2017.
  - [28] B. Simpson, F. Dutil, Y. Bengio, and J. P. Cohen. GradMask: Reduce overfitting by regularizing saliency. *arXiv preprint arXiv:1904.07478*, 2019.
  - [29] A. Stone, H. Wang, M. Stark, Y. Liu, D. Scott Phoenix, and D. George. Teaching compositionality to cnns. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5058–5067, 2017.
  - [30] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, 63(2):113–140, 2005.
  - [31] S. Workman, R. Souvenir, and N. Jacobs. Understanding and mapping natural beauty. In *CVF/IEEE International Conference on Computer Vision (CVPR)*, pages 5589–5598, 2017.
  - [32] Q. Yu, J. Liu, H. Cheng, A. Divakaran, and H. Sawhney. Multimedia event recounting with concept based representation. In *ACM International Conference on Multimedia*, pages 1073–1076. ACM, 2012.
  - [33] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, pages 818–833. Springer, 2014.
  - [34] Q. Zhang, Y. Nian Wu, and S.-C. Zhu. Interpretable convolutional neural networks. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8827–8836, 2018.
  - [35] Z. Zhao, H. Ma, and S. You. Single image action recognition using semantic body part actions. In *CVF/IEEE International Conference on Computer Vision (ICCV)*, pages 3391–3399, 2017.
  - [36] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVF/IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.
  - [37] B. Zhou, Y. Sun, D. Bau, and A. Torralba. Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–134, 2018.