SCOUTER: Slot Attention-based Classifier for Explainable Image Recognition

Liangzhi Li[1], Bowen Wang[2], Manisha Verma[1], Yuta Nakashima[1],
Ryo Kawasaki[3], Hajime Nagahara[1]
Osaka University, Japan
[1]{li, mverma, n-yuta, nagahara}@ids.osaka-u.ac.jp
[2]bowen.wang@is.ids.osaka-u.ac.jp [3]ryo.kawasaki@ophthal.med.osaka-u.ac.jp

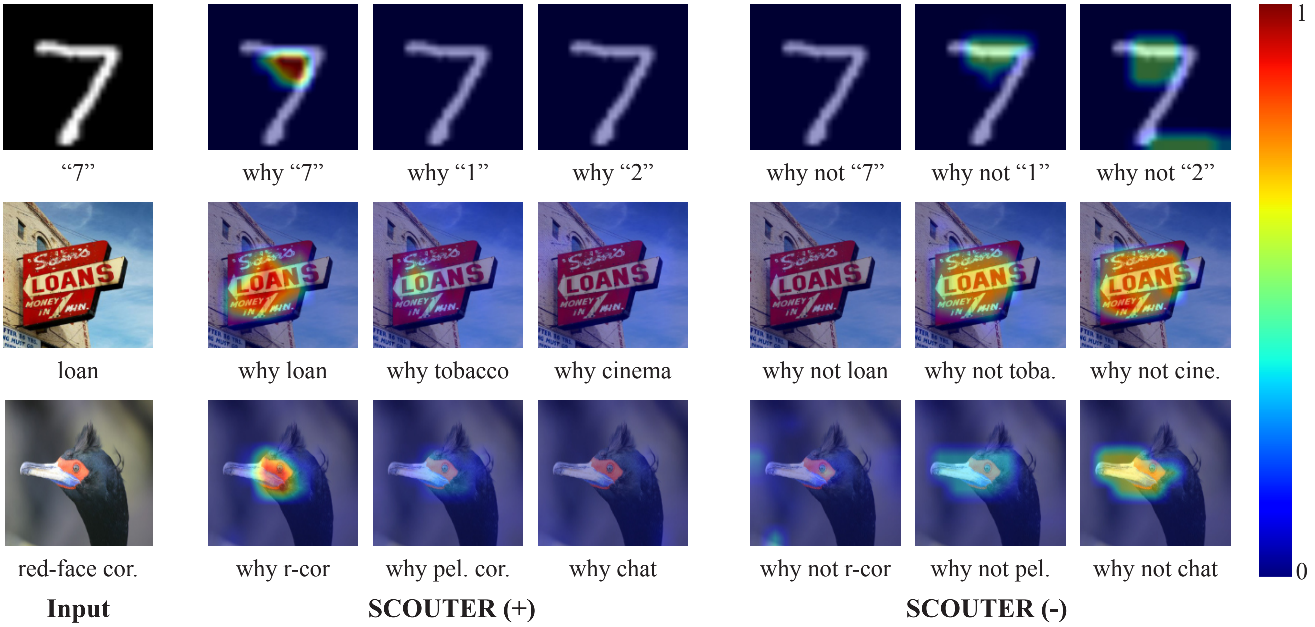


Figure 1. Positive and negative explanations. The images from top to down are from the test sets of MNIST [18], Con-text [16], and CUB-200 [34] datasets. The models trained with positive (+) and negative (−) SCOUTER losses can respectively highlight the positive and negative supports, based on which one can reason why or why not the images are classified into the corresponding categories.

Abstract

Explainable artificial intelligence has been gaining attention in the past few years. However, most existing methods are based on gradients or intermediate features, which are not directly involved in the decision-making process of the classifier. In this paper, we propose a slot attention-based classifier called SCOUTER for transparent yet accurate classification. Two major differences from other attention-based methods include: (a) SCOUTER's explanation is involved in the final confidence for each category, offering more intuitive interpretation, and (b) all the categories have their corresponding positive or negative explanation, which tells "why the image is of a certain category" or "why the image is not of a certain category." We design a new loss tailored for SCOUTER that controls the model's behavior to switch between positive and negative explanations, as well as the size of explanatory regions. Experimental results show that SCOUTER can give better visual explanations in terms of various metrics while keeping good accuracy on small and medium-sized datasets. Code is available[1].

1. Introduction

It is of great significance to know how deep models make predictions, especially for the fields like medical diagnosis, where potential risks exist when black-box models are adopted. Explainable artificial intelligence (XAI), which can give a close look into models' inference process, therefore has gained lots of attention.

[1] https://github.com/wbw520/scouter

The most popular paradigm in XAI is *attributive explanation*, which gives the contribution of pixels or regions to the final prediction [27, 7, 24, 26]. One natural question that arises here is *how these regions contribute to the decision*. For a better view of this, let $g_l(v) = w_l^\top v + b_l$ denotes a fully-connected (FC) classifier for category l, where w_l and b_l are trainable vector and scalar, respectively. Training this classifier may be interpreted as a process to find from the training samples a combination of discriminative patterns s_{li} with corresponding weight γ_i, *i.e.*,

$$g_l(v) = \left(\sum_i \gamma_i s_{li}^\top \right) v + b_l. \tag{1}$$

In general, these patterns can include *positive* and *negative* ones. Given v of an image of l, a positive pattern gives $s_{li}^\top v > 0$. A negative pattern, in contrast, gives $s_{li}^\top v < 0$ for v of any category other than l, which means that the presence of pattern described by s_{li} is a support of *not* being category l. Therefore, set $\mathcal{S}_l$ of all (linearly independent) patterns for l can be the union of sets $\mathcal{S}_l^+$ and $\mathcal{S}_l^-$ of all positive and negative patterns.

Differentiation of positive/negative patterns gives useful information on the decision. Figure 1(top) shows an MNIST image for example. One of positive patterns that makes the image being 7 can be the acute angle formed by white line segments that appears around the top-right corner, as in the second image. Meanwhile, the sixth image shows that the presence of the horizontal line is the support not being 1. A more practical application [2] in medical image analysis also points out the importance of visualizing positive/negative patterns. Nevertheless of the obvious benefit, recent mainstream methods like [44, 27, 23, 10] have not extensively studied this differentiation.

Positive and negative patterns lead to two interesting questions: (i) Can we provide *positive explanation* and *negative explanation* that visually show support regions in the image that correspond to positive and negative patterns? (ii) As the combination of patterns to be learned is rather arbitrary and any combination is possible as long as it is discriminative; can we provide preference on the combination in order to leverage prior knowledge on the task in training?

In this paper, we re-formulate explainable AI with an *explainable classifier*, coined SCOUTER (Slot-based COnfigUrable and Transparent classifiER), which tries to find either positive or negative patterns in images. This approach is similar to the attention-based approach (*e.g.* [17, 35]) rather than the post-hoc approaches [44, 27, 26]. Our newly proposed explainable slot attention (xSlot) module is the main building block of SCOUTER. This module is built on top of the recently-emerged slot attention [19], which offers an object-centric approach for image representation. The xSlot module identifies the spatial support of either positive or negative patterns for each category in the image, which is directly used as the confidence value of that category; the commonly-used FC classifier is no longer necessary. The xSlot module can also be used to visualize the support as shown in Fig. 1. SCOUTER is also characterized by its configurability over patterns to be learned, *i.e.*, the choice of positive or negative pattern and the desirable size of the pattern, which can incorporate the prior knowledge on the task. The controllable size of explanation can be beneficial for some applications, *e.g.*, disease diagnosis in medicine, defect recognition in manufacturing, *etc*.

Contribution Our transparent classifier, SCOUTER, explicitly models positive and negative patterns with a dedicated loss, allowing to set preference over the spatial size of patterns to be learned. We experimentally show that SCOUTER successfully learns both positive and negative patterns and visualize their support in the given image as the explanation, achieving state-of-the-art results in several commonly-used metrics like IAUC/DAUC [23]. Our case study in medicine also highlights the importance of both types of explanations as well as controlling the area size of explanatory regions.

2. Related Work

2.1. Explainable AI

There are mainly three XAI paradigms [38], *i.e. post-hoc*, *intrinsic*, and *distillation*. The post-hoc paradigm usually provides a heat map highlighting important regions for the decision (*e.g.* [27, 26]). The heat map is computed besides the forward path of the model. The intrinsic paradigm explores the important piece of information within the forward path of the model, *e.g.*, as attention maps (*e.g.* [17, 35, 20, 37]). *Distillation methods* are built upon model distillation [15]. The basic idea is to use an inherently transparent model to mimic the behaviors of a black-box model (*e.g.* [43, 25]).

The post-hoc paradigm has been extensively studied among them. The most popular type of methods is based on channel activation or back-propagation, including CAM [44], GradCAM [27], DeepLIFT [28], and their extensions [3, 21, 31, 30, 7]. Another type of method is perturbation-based, including Occlusion [40], RISE [23], meaningful perturbations [11], real-time saliency [5], extremal perturbations [10], I-GOS [24], IBA [26], *etc*. These methods basically give *attributive explanation*, which visualizes support regions of learned patterns for each category l in the set of all possible categories $\mathcal{L}$. This visualization can be done by finding regions in feature maps or the input image that give large impact on the score g_l. By definition, attributive explanation is the same as our positive explanation.

Some of the methods for attributive explanation thus can be extended to provide negative explanations by negating the sign of the score, feature maps, or gradients. It should be noted that the interpretation of visual explanation by gradient-based methods [27, 3, 21] may not be straightforward because of linearization of g_l for the given image; and thus the resulting visualization may not highlight the support regions for negative patterns. GradCAM [27] refers to its negative variant as *counterfactual explanation* that gives regions that can change the decision, emphasizing how it should be interpreted.

Discriminant explanation is a new type of XAI in the post-hoc paradigm, which appeared in [33] to show "why image x belongs to category l rather than l'." This can be interpreted using set $\mathcal{S}_l^+$ of all possible positive patterns for l and set $\mathcal{S}_{l'}^-$ of all possible negative patterns for l': It try to spot a (combination of) discriminative pattern s that is in the intersection $\mathcal{S}_l^+ \cap \mathcal{S}_{l'}^-$. Due to the unavailability of negative patterns, the method [33] first finds (a combination of) positive patterns and uses the complementary of the region containing the positive patterns as a proxy of negative patterns. Goyal *et al*. gives another line of counterfactual explanation in [13]. Given two images of categories l and l', they find the region in the image of l, of which replacement to a certain region in the image of l' changes the prediction from l to l'. This can be also implemented using discriminant explanation.

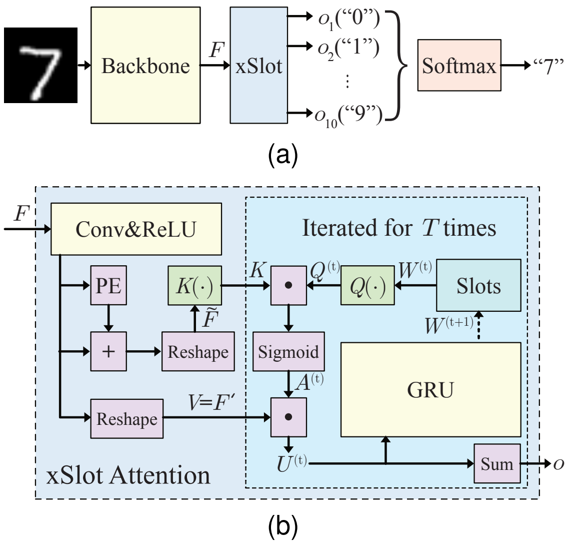
SCOUTER computes a heat map to spot regions important for the decision in the forward path, so it falls into the intrinsic paradigm. Together with the dedicated loss, it can directly identify positive and negative patterns with control over the size of patterns.

2.2. Self-attention in Computer Vision

Self-attention is first introduced in the Transformers [29], in which self-attention layers scan through the input elements one by one and update them using the aggregation over the whole input. Initially, self-attention is used in place of recurrent neural networks for sequential data, *e.g*., natural language processing [8]. Recently, self-attention is adopted to the computer vision field, *e.g*., Image Transformer [22], Axial-DeepLab [32], DEtection TRansformer (DETR) [1], Image Generative Pre-trained Transformer (Image GPT) [4], *etc*. Slot attention [19] is also based on this mechanism to extract object-centric features from images (there are some other works [14, 12] using the concept of *slot*); however, the original slot attention is tested only on some synthetic image datasets. SCOUTER is based on slot attention but is designed to be an explainable classifier applicable to natural images.

3. SCOUTER

Given an image x, the objective of a classification model is to find its most probable category l in category set $\mathcal{L} = \{l_1, l_2, \dots, l_n\}$. This can be done by first extracting features $F = B(x) \in \mathbb{R}^{c \times h \times w}$ using a backbone network B. F is then mapped into a score vector $o \in \mathbb{R}^n$, representing the confidence values, using FC layers and softmax as the classifier. However, such FC classifiers can learn an arbitrary (nonlinear) transformation and thus can be black-box.



(a)

(b)

Figure 2. Classification pipeline. (a) Overview of the classification model. (b) The xSlot Attention module in SCOUTER.

We replace such an FC classifier with our SCOUTER (Fig. 2(a)), consisting of the xSlot attention module, which produces the confidence for each category given features F. The whole network, including the backbone, is trained with the *SCOUTER loss*, which provides control over the size of support regions and switching between positive and negative explanations.

3.1. xSlot Attention

In the original slot attention mechanism [19], a *slot* is a representation of a local region aggregated based on the attention over the feature maps. A single slot attention module with multiple slots is attached on top of the backbone network B. Each slot produces its own feature as output. This configuration is handy when there are multiple objects of interest. This idea can be transferred to spot the supports in the input image that leads to a certain decision.

The main building block of SCOUTER is the xSlot attention module, which is a variant of the slot attention module tailored for SCOUTER. Each slot of the xSlot attention module is associated with a category and gives the confidence that the input image falls into the category. With the slot attention mechanism, the slot for category l is required to find support $\mathcal{S}_l$ in the image that directly correlates to l.

Given feature F, the xSlot attention module updates slot $w_l^{(t)}$ for T times, where $w_l^{(t)}$ represents the slot after t updates and $l \in \mathcal{L}$ is the category associated to this slot. The slot is initialized with random weights, *i.e.*,

$$w_l^{(0)} \sim \mathcal{N}(\mu, \mathrm{diag}(\sigma)) \in \mathbb{R}^{1\times c'}, \tag{2}$$

where μ and σ are the mean and variance of a Gaussian, and c' is the size of the weight vector. We denote the slots for all categories by $W^{(t)} \in \mathbb{R}^{n\times c'}$.

The slot $W^{(t+1)}$ is updated using $W^{(t)}$ and feature F. Firstly, F goes through a 1×1 convolutional layer to reduce the number of channels and the ReLU nonlinearity as $F' = \mathrm{ReLU}(\mathrm{Conv}(F)) \in \mathbb{R}_+^{c'\times d}$, with F's spatial dimensions being flattened ($d = hw$). F' is augmented by adding the position embedding to take the spatial information into account, following [29, 1], *i.e.* $\tilde{F} = F' + \mathrm{PE}$, where PE is the position embedding. We then use two multilayer perceptrons (MLPs) Q and K, each of which has three FC layers and the ReLU nonlinearity between them. This design is for giving more flexibility in the computation of *query* and *key* in the self-attention mechanism. Using

$$Q(W^{(t)}) \in \mathbb{R}^{n\times c'}, \quad K(\tilde{F}) \in \mathbb{R}^{c'\times d}, \tag{3}$$

we obtain the dot-product attention $A^{(t)}$ using sigmoid σ as

$$A^{(t)} = \sigma(Q(W^{(t)})K(\tilde{F})) \quad \in (0,1)^{n\times d}. \tag{4}$$

The attention is used to compute the weighted sum of features in the spatial dimensions by

$$U^{(t)} = A^{(t)}F'^{\top} \quad \in \mathbb{R}^{n\times c'}, \tag{5}$$

and slot $W^{(t)}$ is eventually updated through a gated recurrent unit (GRU) as

$$W^{(t+1)} = \mathrm{GRU}(U^{(t)}, W^{(t)}), \tag{6}$$

taking $U^{(t)}$ and $W^{(t)}$ as input and hidden state, respectively. Following the original slot attention module, we update the slot for $T = 3$ times.

The output of the xSlot attention module is the sum of all elements for category l in $U^{(T)}$, which is a function of F. Formally, the output of xSlot Attention module is:

$$\mathrm{xSlot}(F) = U^{(T)}\mathbf{1}_{c'} \quad \in \mathbb{R}_+^n, \tag{7}$$

where $\mathbf{1}$ is the column vector with all c' elements being 1.

From Eqs. (5) and (7), we have $\mathrm{xSlot}(F) = A^{(T)}F'^{\top}\mathbf{1}_{c'}$, where $F'^{\top}\mathbf{1}_{c'} \in \mathbb{R}^d$ is a reduction of F and is a class-agnostic map. The l-th row of $A^{(T)}$ can then be viewed as spatial weights over map $F'^{\top}\mathbf{1}_{c'}$ to spot where the support regions for category l is[2]. In order for visualizing the support regions, we reshape and resize each row of $A^{(T)}$ to the input image size.

Note that in the original slot attention module, a linear transformation is applied to the features, *i.e.*, $V(\tilde{F})$, which is then weighted using Eq. (5). However, the xSlot attention module omits this transformation as it already has a sufficient number of learnable parameters in Q, K, GRU, *etc.*, and thus the flexibility. Also, the confidences, given by Eq. (7), are typically computed by an FC layer, while SCOUTER just sums up the output of xSlot attention module, which is actually the presence of learned supports for each category. This simplicity is essential for a transparent classifier as discussed in Section 3.3.

3.2. SCOUTER Loss

The whole model, including the backbone network, can be trained by simply applying softmax to $\mathrm{xSlot}(F)$ and minimizing cross-entropy loss ℓ_{CE}. However, there is a phenomenon that, in some cases, the model prefers attending to a broad area (*e.g.*, a slot covers a combination of several supports that occupy large areas in the image) depending on the content of the image. As argued in Section 1, it can be beneficial to have control over the area of support regions to constrain the coverage of a single slot.

Therefore, we design the SCOUTER loss to limit the area of support regions. The SCOUTER loss is defined by

$$\ell_{\mathrm{SCOUTER}} = \ell_{\mathrm{CE}} + \lambda\ell_{\mathrm{Area}}, \tag{8}$$

where ℓ_{Area} is the area loss, λ is a hyper-parameter to adjust the importance of the area loss. The area loss is defined by

$$\ell_{\mathrm{Area}} = \mathbf{1}_n^{\top}A^{(T)}\mathbf{1}_d, \tag{9}$$

which simply sums up all the elements in $A^{(T)}$. With larger λ, SCOUTER attends smaller regions by selecting fewer and smaller supports. On the contrary, it prefers a larger area with smaller λ.

3.3. Switching Positive and Negative Explanation

The model with the SCOUTER loss in Eq. (8) can only provide positive explanation since larger elements in $A^{(T)}$ means the prediction is made based on the corresponding features. We introduce a hyper-parameter $e \in \{+1, -1\}$ in Eq. (7), *i.e.*,

$$o = \mathrm{xSlot}_e(F) = e \cdot U^{(T)}\mathbf{1}_{c'} \quad \in \mathbb{R}_+^n, \tag{10}$$

where $o = \{o_1, \dots, o_n\}$. This hyper-parameter configures the xSlot attention module to learn to find either positive or negative supports.

With the softmax cross-entropy loss, the model learns to give the largest confidence o_l corresponding to ground-truth (GT) category l and a smaller value $o_{l'}$ to wrong category $l' \neq l$. For $e = +1$, all elements given by xSlot is non-negative since both $A^{(T)}$ and F' are non-negative and thus

[2] $F'^{\top}\mathbf{1}_{c'}$ can be viewed as a single map that includes a mixture of supports for all categories.

$U^{(T)}$ is. For arbitrary non-negative F', thanks to simple reduction in Eq. (7), larger o_l can be produced only when some elements in $a_l^{(T)}$, the row vector in $A^{(T)}$ corresponding to l, is close to 1, whereas a smaller $o_{l'}$ is given when all elements in $a_l^{(T)}$ are close to 0. Therefore, by setting e to $+1$, the model learns to find the positive supports $\mathcal{S}_l^+$ among the images of the GT category. The visualization of $a_l^{(T)}$ thus serves as positive explanation, as in Fig. 1 (left).

On the contrary, for $e = -1$, all elements in o are negative and thus the prediction by Eq. (10) gives o_l close to 0 for correct category l and smaller $o_{l'}$ for non-GT category l'. To make o_l close to 0, all elements in $a_l^{(T)}$ must be close to 0, and a smaller $o_{l'}$ is given when $a_{l'}^{(T)}$ has some elements close to 1. For this, the model learns to find the negative supports $\mathcal{S}_-$ that do not appear in the images of the GT category. As a result, $a_{l'}^{(T)}$ can be used as negative explanation, as shown in Fig. 1 (right).

4. Experiments

4.1. Experimental Setup

We chose to use the ImageNet dataset [6] for a detailed evaluation of SCOUTER, because of the following three reasons: (i) It is commonly used in the evaluation of classification models. (ii) There are many classes with similar semantics and appearances, and the relationships among them are available in the synsets of the WordNet, which can be used to evaluate positive and negative explanations. (iii) Bounding boxes are available for foreground objects, which helps measure the accuracy of visual explanation. In experiments, we use subsets of ImageNet by extracting the first n $(0 < n \leq 1,000)$ categories in the ascending order of the category IDs. Also, we present classification performance on Con-text [16] and CUB-200 [34] datasets and illustrate glaucoma diagnosis using quantitative and qualitative results on ACRIMA [9] dataset.

The size of images is 260×260. The feature F extracted by the backbone network is mapped into a new feature F' with the channel number $c' = 64$. The models were trained on the training set for 20 epochs and the performance scores are computed on the validation set with the trained models after the last epoch. All the quantitative results are obtained by averaging the scores from three independent runs.

4.2. Explainability

To evaluate the quality of visual explanation, we use bounding boxes provided in ImageNet as a proxy of the object regions and compute the percentage of the pixels located inside the bounding box over the total pixel numbers in the whole explanation. Specifically, for set I of all pixels in the input image and set D of all pixels in the bounding box, we define the explanation precision as $\mathrm{Precision}_l = \frac{\sum_{i \in D} a_i^l}{\sum_{i \in I} a_i^l}$, where category $l \in \mathcal{L}$ and $a_i^l \in [0, 1]$ is the value of pixel i in $\bar{A}_l$, which is attention map A resized to the same size as the input image by bilinear interpolation. We compute this metric on the visualization results of the GT category for positive explanations and on the least similar class (LSC) (using Eq. 11) for negative explanations, as LSC images usually show strong and consistent negative explanations. This precision metric actually is a generalization of the pointing game [42], which counts one *hit* when the point with the largest value on the heat map locates in the bounding box and the final score is calculated as $\frac{\#\text{Hits}}{\#\text{Hits} + \#\text{Misses}}$.

We also adopt several other metrics, *i.e.*, (i) insertion area under curve (IAUC) [23], which measures the accuracy gain of a model when gradually adding pixels according to their importance given in the explanation (heat map) to a synthesized input image; (ii) deletion area under curve (DAUC) [23], which measures the performance drop when gradually removing important pixels from the input image; (iii) infidelity [39], which measures the degree to which the explanation captures how the prediction changes in response to input perturbations; and (iv) sensitivity [39], which measures the degree to which the explanation is affected by the input perturbations. In addition, we calculate the (v) overall size of the explanation areas by $\mathrm{Area}_l = \sum_{i \in I} a_i^l$, as for some applications, a smaller value is better to pinpoint the supports to differentiate one class from the others.

We conduct the explainability experiments with the ImageNet subset with the first 100 classes. We train seven models with (1) an FC classifier, (2)–(4) SCOUTER$_+$ $(\lambda = 1, 3, 10)$, and (5)–(7) SCOUTER$_-$ $(\lambda = 1, 3, 10)$ using ResNeSt 26 [41] as the backbone. The results of competing methods are obtained from the FC classifier-based model. In addition, as introduced in Section 2.1, some of the existing works can give negative explanations. Therefore, we also implement and compare our results with their negative variants by using negative feature maps/gradients or modifying their objective functions.

The numerical results are shown in Table 1. We can see that SCOUTER can generate explanations with different area sizes while achieving good scores in all metrics. These results demonstrate that the visualization by SCOUTER is preferable in terms of controlling area sizes, high precision, insensitive to noises (sensitivity), and with good explainability (infidelity, IAUC, and DAUC). Among the competing methods, extremal perturbation [10], I-GOS [24], and IBA [26] also take the size of support regions into account, and thus some of them give smaller explanatory regions. Extremal perturbation's explanatory regions cover some parts of foreground objects. This leads to a high precision score, but the performance over other metrics is not satisfactory. I-GOS and IBA give small explanation areas. I-GOS results in low IAUC and sensitivity scores. IBA gets relatively low scores of IAUC and DAUC, which means its

Table 1. Evaluation of the explanations. Positive explanations are from the GT class, while negative is from the least similar class (LSC).

	Methods	Year	Type	Area Size	Precision ↑	IAUC ↑	DAUC ↓	Infidelity ↓	Sensitivity ↓
Positive	CAM [44]	2016	Back-Prop	0.0835	0.7751	0.7185	0.5014	0.1037	0.1123
	GradCAM [27]	2017	Back-Prop	0.0838	0.7758	0.7187	0.5015	0.1038	0.0739
	GradCAM++ [3]	2018	Back-Prop	0.0836	0.7861	0.7306	0.4779	0.1036	0.0807
	S-GradCAM++ [21]	2019	Back-Prop	0.0868	0.7983	0.6991	0.4896	0.1548	0.0812
	Score-CAM [31]	2020	Back-Prop	0.0818	0.7714	0.7213	0.5247	0.1035	0.0900
	SS-CAM [30]	2020	Back-Prop	0.1062	0.7902	0.7143	0.4570	0.1109	0.1183
	⌞ w/ threshold	2020	Back-Prop	0.0496	0.8243	0.6010	0.7781	0.1079	0.0790
	RISE [23]	2018	Perturbation	0.3346	0.5566	0.6913	0.4903	0.1199	0.1548
	Extremal Perturbation [10]	2019	Perturbation	0.1458	0.8944	0.7121	0.5213	0.1042	0.2097
	I-GOS [24]	2020	Perturbation	0.0505	0.8471	0.6838	0.3019	0.1106	0.6099
	IBA [26]	2020	Perturbation	0.0609	0.8019	0.6688	0.5044	0.1039	0.0894
	SCOUTER$_+$ ($\lambda = 1$)		Intrinsic	0.1561	0.8493	0.7512	0.1753	**0.0799**	0.0796
	SCOUTER$_+$ ($\lambda = 3$)		Intrinsic	0.0723	0.8488	**0.7650**	**0.1423**	0.0949	**0.0608**
	SCOUTER$_+$ ($\lambda = 10$)		Intrinsic	0.0476	**0.9257**	0.7647	0.2713	0.0840	0.1150
Negative	CAM [44]	2016	Back-Prop	0.1876	0.3838	0.6069	0.6584	0.1070	0.0617
	GradCAM [27]	2017	Back-Prop	0.0988	0.6543	0.6289	0.7281	0.1060	0.5493
	GradCAM++ [3]	2018	Back-Prop	0.0879	0.6280	0.6163	0.6017	0.1047	0.3114
	S-GradCAM++ [21]	2019	Back-Prop	0.1123	0.6477	0.6036	0.5430	0.1071	0.0590
	RISE [23]	2018	Perturbation	0.4589	0.4490	0.4504	0.7078	0.1064	0.0607
	Extremal Perturbation [10]	2019	Perturbation	0.1468	0.6390	0.2089	0.7626	0.1068	0.8733
	SCOUTER$_-$ ($\lambda = 1$)		Intrinsic	0.0643	0.8238	**0.7343**	**0.1969**	0.0046	**0.0567**
	SCOUTER$_-$ ($\lambda = 3$)		Intrinsic	0.0545	**0.8937**	0.6958	0.4286	0.0196	0.1497
	SCOUTER$_-$ ($\lambda = 10$)		Intrinsic	0.0217	0.8101	0.6730	0.7333	**0.0014**	0.1895

Table 2. Area sizes of the explanations ($\lambda = 10$).

Methods	Target Classes			
	GT	Highly-similar	Similar	Dissimilar
SCOUTER$_+$	0.0476	0.0259	0.0093	0.0039
SCOUTER$_-$	0.0097	0.0141	0.0185	0.0204

explanations cannot give correct attention to the pixels and thus does not have enough explainability.

It is arguable that area size can be controlled by thresholding the heatmap. In order to verify this, we set a threshold ($a \geq 0.2$) to one of the back-propagation-based methods (SS-CAM) to get explanations with smaller size. We can see that this variant suffers a deterioration in IAUC and DAUC (significantly worse than I-GOS, IBA, and SCOUTER), which represents a large explainability drop and hampers its uses in actual applications requiring small explanations.

To further explore the explanation for non-GT categories, we define the semantic similarity between categories based on [36], which uses WordNet, as

$$\text{Similarity} = 2\frac{d(\text{LCS}(l, l'))}{d(l) + d(l')}, \tag{11}$$

where $d(\cdot)$ gives the depth of category l in WordNet, and $\text{LCS}(l, l')$ is to find the least common subsumer of two arbitrary categories l and l'. We define the highly-similar categories as the category pairs with a similarity score no less than 0.9, similar categories as with a score in $[0.7, 0.9)$, and the remaining categories are regarded as dissimilar categories. Table 2 summarizes the area sizes of the explanatory regions for GT, highly-similar, similar, and dissimilar categories. We see a clear trend: SCOUTER$_+$ decreases the area size when the inter-category similarity becomes lower, while SCOUTER$_-$ gives larger explanatory regions for the dissimilar categories.

Some visualization results are given in Figs. 3 and 4. It can be seen that SCOUTER gives reasonable and accurate explanations. Comparing SCOUTER$_+$'s explanation with SS-CAM [31], and IBA [26], we find that SCOUTER$_+$ can give explanations with more flexible shapes which fit the target objects better. For example, in the first row of Fig. 3, SCOUTER$_+$ gives more accurate attention around the neck. In the second row, it accurately finds the individual entities. Compared with SS-CAM, IBA shows smaller explanatory regions. However, IBA is less precise and less reasonable, which is consistent with the numerical results in Table 1.

In Fig. 4, SCOUTER$_-$ can also find the negative supports, *e.g.*, the wattle of the hen, and the hammerhead and the fin of the shark. In addition, although the negative variation of S-GradCAM++ performs well on the first row, its explanation in the second row does not well fit the object's shape and fails to pinpoint the key difference (the head).

4.3. Classification Performance

We compare SCOUTER and FC classifiers with several commonly used backbone networks with respect to the classification accuracy. The results are summarized in Fig. 5. With the increase of the category number, both the FC classifier and SCOUTER show a performance drop. They show

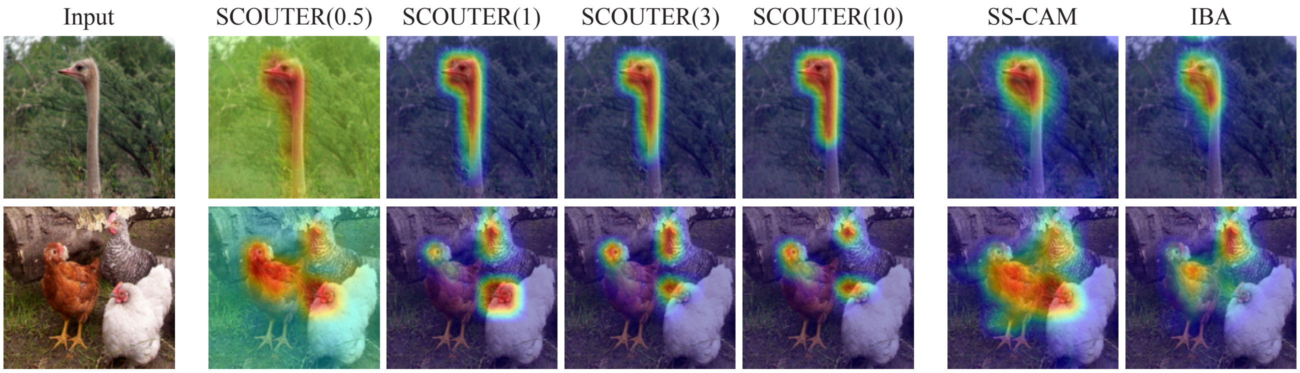
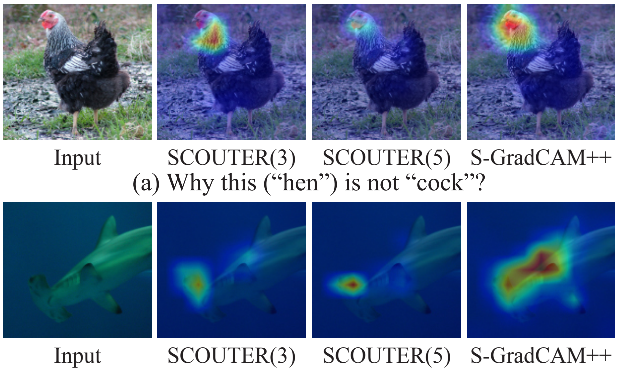


Figure 3. Visualized positive explanations using SCOUTER$_+$ and existing methods. The numbers in the parentheses are the λ values used in the SCOUTER training.



(a) Why this ("hen") is not "cock"?

(b) Why this ("hammerhead shark") is not "tiger shark"?

Figure 4. Visualized negative explanations using SCOUTER$_-$ and an existing method. The numbers in the parentheses are the λ values used in the SCOUTER training.

similar trends with respect to the category number.

The relationship between λ, which controls the size of explanatory regions, and the classification accuracy is shown in Fig. 6 for ResNeSt 26 model with $n = 100$. A clear pattern is that the area sizes of both SCOUTER$_+$ and SCOUTER$_-$ drop quickly with the increase of λ. However, there is no significant trend in the classification accuracy, which should be because the cross-entropy loss term works well regardless of λ.

Also, according to the visualization results in Figs. 3 and 4, a larger λ does not simply decrease the explanatory regions' sizes. Instead, SCOUTER shifts its focus from some larger supports to fewer, smaller yet also decisive supports. For example, in the first row of Fig. 4, when λ is small, SCOUTER$_-$ can easily make a decision that the input image is not a cock because of unique feathers on the neck. With a larger λ, SCOUTER finds smaller combinations of supports (*i.e.*, its wattle) and thus the explanation changes from the (larger) neck to the (smaller) wattle region.

We also summarize the classification performance of the FC classifier, SCOUTER$_+$ ($\lambda = 10$), and SCOUTER$_-$ ($\lambda = 10$) over ImageNet [6], Con-text [16], and CUB-200 [34] datasets in Table 3. The subsets with $n = 100$ are adopted for ImageNet and CUB-200, while all 30 categories are used for the Con-text. The results show that SCOUTER can be generalized to different domains and has a comparable performance with the FC classifier over all datasets. Also, SCOUTER's number of parameters is comparable to FC's (more details in supplementary material).

One drawback of SCOUTER is that its training is unstable when $n > 100$. This is possibly because of the increasing difficulty in finding effective supports that consistently appear in all images of the same category but are not shared by other categories. This drawback limits the application of SCOUTER to small- or medium-sized datasets.

4.4. Case Study

SCOUTER uses the area loss, which constrains the size of support. This constraint can benefit some applications, including the classification of medical images, since small support regions can precisely show the symptoms and are more informative in some cases. Also, there was no method that could give the negative explanation but it is actually needed (doctors need reasons to deny some diseases). SCOUTER was designed upon these needs and is being tested in hospitals for glaucoma (Fig. 7), artery hardening (supplementary material), *etc*.

For glaucoma diagnosis, we tested SCOUTER with $\lambda = 10$ over a publicly available dataset, *i.e.*, ACRIMA [9], which has two categories (normal and glaucoma). ResNeSt 26 is used as backbone. The results are shown in Table 4. We can see that both SCOUTER$_+$ and SCOUTER$_-$ get better performances than the FC classifier. Besides, SCOUTER is preferred in this task as doctors are eager to know the precise regions in the optic disc that lead to the machine diagnosis. We can see that, in the visualization results in Fig. 7, SCOUTER shows more precise and reasonable explanations that locate on some vessels in the optic disc and show clinical meanings (vessel shape change due to the enlarged optic cup), which are verified by doctors. Although IBA also gives small regions, they cover

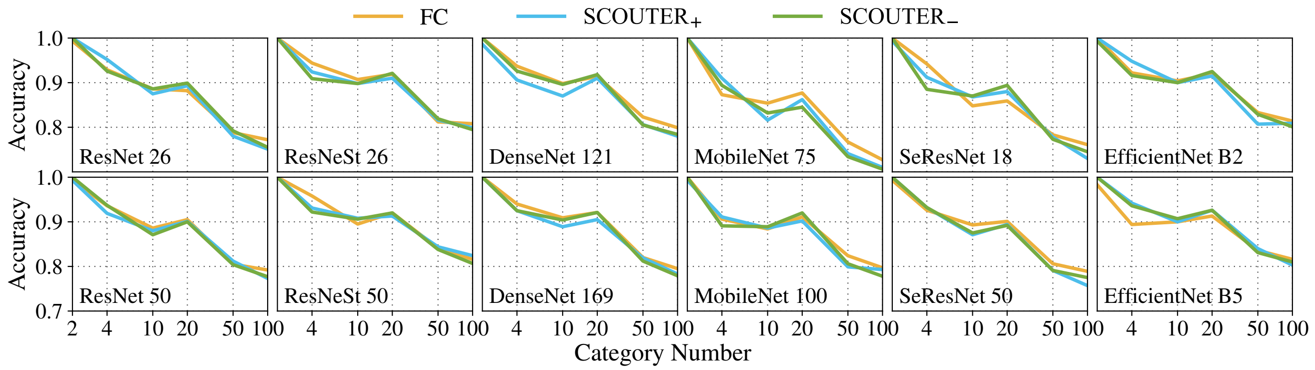


Figure 5. Classification performance of different models with FC classifier, SCOUTER$_+$ ($\lambda = 10$), and SCOUTER$_-$ ($\lambda = 10$). The horizontal axis is the category number n (in the logarithmic scale), which is used to generate the training and test set with the first n categories of ImageNet dataset; the vertical axis is the accuracy of the model.

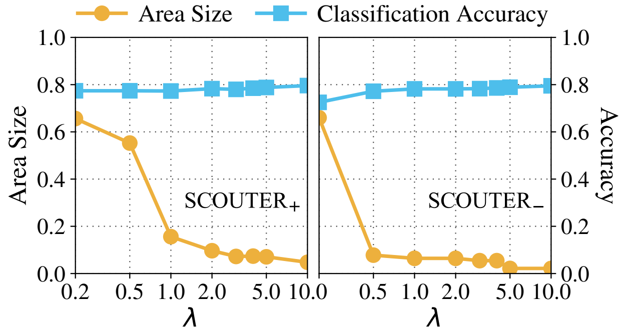


Figure 6. Relationships between λ and explanation area sizes and between λ and classification accuracy for the GT (SCOUTER$_+$, left) and LSC (SCOUTER$_-$, right) when $n = 100$. The horizontal axis is in the logarithmic scale.

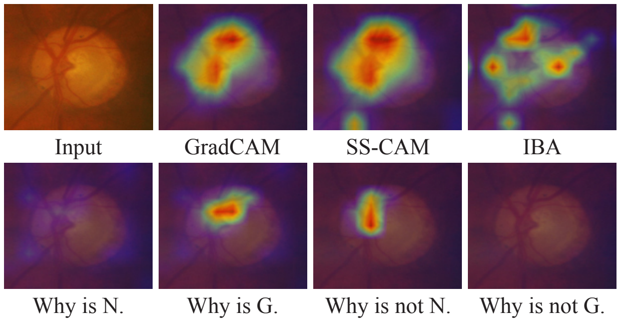
Table 3. Classification accuracy on various datasets.

Models	ImageNet	Con-text	CUB-200
ResNeSt 26 (FC)	**0.8080**	0.6732	**0.7538**
ResNeSt 26 (SCOUTER$_+$)	0.7991	**0.6870**	0.7362
ResNeSt 26 (SCOUTER$_-$)	0.7946	0.6866	0.7490
ResNeSt 50 (FC)	0.8158	0.6918	**0.7739**
ResNeSt 50 (SCOUTER$_+$)	**0.8242**	**0.6943**	0.7397
ResNeSt 50 (SCOUTER$_-$)	0.8066	0.6922	0.7600
ResNeSt 101 (FC)	0.8255	0.7038	**0.7804**
ResNeSt 101 (SCOUTER$_+$)	0.8251	**0.7131**	0.7428
ResNeSt 101 (SCOUTER$_-$)	**0.8267**	0.7062	0.7643

some unrelated or uninformative locations. In addition, it is notable that the facts to admit category "Glaucoma" need not to match with the facts to deny "Normal", as they are only subsets of the support sets and an on-purpose negative explanation is especially helpful for the doctors when the machine decisions are against their expectations.

5. Conclusion

An explainable classifier is proposed in this paper, with two variants, *i.e.*, SCOUTER$_+$ and SCOUTER$_-$, which can respectively give positive or negative explanation of the classification process. SCOUTER adopts an explainable variant of the slot attention, namely, xSlot attention, which is also based on the self-attention. Moreover, a loss is designed to control the size of explanatory regions. Experimental results prove that SCOUTER can give accurate explanations while keeping good classification performance.


Input GradCAM SS-CAM IBA

Why is N. Why is G. Why is not N. Why is not G.

Figure 7. Explanations for a positive sample in the glaucoma diagnosis dataset. Bottom row is from SCOUTER +/- for normal (N.) and glaucoma (G.).

Table 4. Classification Performance on ACRIMA Dataset [9].

Methods	AUC	Acc.	Prec.	Rec.	F1	Kappa
FC	0.9997	0.9857	0.9915	0.9831	0.9872	0.9710
SCOUTER$_+$	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
SCOUTER$_-$	0.9999	0.9952	**1.0000**	0.9915	0.9957	0.9903

6. Acknowledgments

This work was supported by Council for Science, Technology and Innovation (CSTI), cross-ministerial Strategic Innovation Promotion Program (SIP), "Innovative AI Hospital System" (Funding Agency: National Institute of Biomedical Innovation, Health and Nutrition (NIBIOHN)). This work was also supported by JSPS KAKENHI Grant Number 19K10662, 20K23343, 21K17764, MHLW AC Program Grant Number 20AC1006, and JST CREST Grant Number JPMJCR20D3, Japan.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020. 3, 4

[2] Jooyoung Chang, Jinho Lee, Ahnul Ha, Young Soo Han, Eunoo Bak, Seulggie Choi, Jae Moon Yun, Uk Kang, Il Hyung Shin, Joo Young Shin, et al. Explaining the rationale of deep learning glaucoma decisions with adversarial examples. *Ophthalmology*, 128(1):78–88, 2021. 2

[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE WACV*, pages 839–847, 2018. 2, 3, 6

[4] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020. 3

[5] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *NeurIPS*, pages 6967–6976, 2017. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE CVPR*, pages 248–255, 2009. 5, 7

[7] Saurabh Desai and Harish G. Ramaswamy. Ablation-CAM: Visual explanations for deep convolutional network via gradient-free localization. In *IEEE WACV*, pages 972–980, 2020. 2

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[9] Andres Diaz-Pinto, Sandra Morales, Valery Naranjo, Thomas Köhler, Jose M Mossi, and Amparo Navea. CNNs for automatic glaucoma assessment using fundus images: an extensive validation. *Biomedical Engineering Online*, 18(1):29, 2019. 5, 7, 8

[10] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *IEEE ICCV*, pages 2950–2958, 2019. 2, 5, 6

[11] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *IEEE ICCV*, pages 3429–3437, 2017. 2

[12] Anirudh Goyal, Alex Lamb, Phanideep Gampa, Philippe Beaudoin, Sergey Levine, Charles Blundell, Yoshua Bengio, and Michael Mozer. Object files and schemata: Factorizing declarative and procedural knowledge in dynamical systems. *arXiv preprint arXiv:2006.16225*, 2020. 3

[13] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *ICML*, 2019. 3

[14] Mikael Henaff, Jason Weston, Arthur Szlam, Antoine Bordes, and Yann LeCun. Tracking the world state with recurrent entity networks. In *ICLR*, 2017. 3

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[16] Sezer Karaoglu, Ran Tao, Jan van Gemert, and Theo Gevers. Con-Text: Text detection for fine-grained object classification. *IEEE TIP*, 26(8):3965–3980, 2017. 1, 5, 7

[17] Jinkyu Kim and John Canny. Interpretable learning for self-driving cars by visualizing causal attention. In *IEEE ICCV*, pages 2942–2950, 2017. 2

[18] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1

[19] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *arXiv preprint arXiv:2006.15055*, 2020. 2, 3

[20] David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *IEEE CVPR*, pages 4942–4950, 2018. 2

[21] Daniel Omeiza, Skyler Speakman, Celia Cintas, and Komminist Weldermariam. Smooth Grad-CAM++: An enhanced inference level visualization technique for deep convolutional neural network models. *arXiv preprint arXiv:1908.01224*, 2019. 2, 3, 6

[22] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *ICML*, 2018. 3

[23] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: Randomized input sampling for explanation of black-box models. In *BMVC*, 2018. 2, 5, 6

[24] Zhongang Qi, Saeed Khorram, and Li Fuxin. Visualizing deep networks by optimizing with integrated gradients. In *AAAI*, volume 34, pages 11890–11898, 2020. 2, 5, 6

[25] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should I trust you? Explaining the predictions of any classifier. In *KDD*, pages 1135–1144, 2016. 2

[26] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. In *ICLR*, 2020. 2, 5, 6

[27] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *IEEE ICCV*, pages 618–626, 2017. 2, 3, 6

[28] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *ICML*, page 3145–3153, 2017. 2

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 3, 4

[30] Haofan Wang, Rakshit Naidu, Joy Michael, and Soumya Snigdha Kundu. SS-CAM: Smoothed Score-CAM for sharper visual feature localization. *arXiv preprint arXiv:2006.14255*, 2020. 2, 6

[31] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-CAM:

Score-weighted visual explanations for convolutional neural networks. In *IEEE CVPR Workshops*, pages 24–25, 2020. 2, 6

[32] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. *arXiv preprint arXiv:2003.07853*, 2020. 3

[33] Pei Wang and Nuno Vasconcelos. SCOUT: Self-aware discriminant counterfactual explanations. In *IEEE CVPR*, pages 8981–8990, 2020. 3

[34] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010. 1, 5, 7

[35] Zbigniew Wojna, Alex Gorban, Dar-Shyang Lee, Kevin Murphy, Qian Yu, Yeqing Li, and Julian Ibarz. Attention-based extraction of structured information from street view imagery. In *ICDAR*, pages 844–850, 2017. 2

[36] Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *ACL*, page 133–138, 1994. 6

[37] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*, 2019. 2

[38] Ning Xie, Gabrielle Ras, Marcel van Gerven, and Derek Doran. Explainable deep learning: A field guide for the uninitiated. *arXiv preprint arXiv:2004.14545*, 2020. 2

[39] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in)fidelity and sensitivity of explanations. In *NeurIPS*, pages 10967–10978, 2019. 5

[40] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014. 2

[41] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Muller, R. Manmatha, Mu Li, and Alexander Smola. ResNeSt: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 5

[42] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018. 5

[43] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. Interpreting CNNs via decision trees. In *IEEE CVPR*, pages 6261–6270, 2019. 2

[44] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE CVPR*, pages 2921–2929, 2016. 2, 6