Supplementary of "CA-Stream: Attention-based pooling for interpretable image recognition"

A. More on the connection between Attention and CAM

Following the explanation of Cross-Attention acting as a class agnostic version of CAM demonstrated in section 3.2, we provide a visual explanation of this connection in Figure 1.

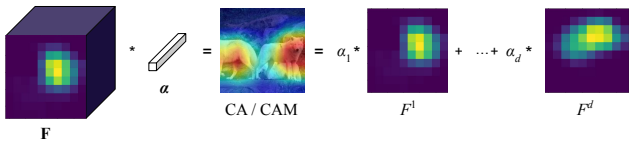

Figure 1. **Visualization of eq.(5)).** On the left, a feature tensor $\mathbf{F} \in \mathbb{R}^{w \times h \times d}$ is multiplied by the vector $\boldsymbol{\alpha} \in \mathbb{R}^d$ in the channel dimension, like in 1×1 convolution, where $w \times h$ is the spatial resolution and d is the number of channels. This is *cross attention* (CA) [1] between the query $\boldsymbol{\alpha}$ and the key $\mathbf{F}$. On the right, a linear combination of feature maps $F^1, \ldots, F^d \in \mathbb{R}^{w \times h}$ is taken with weights $\alpha_1, \ldots, \alpha_d$. This is a *class activation mapping* (CAM) [4] with class agnostic weights. Eq.(5)expresses the fact that these two quantities are the same, provided that $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ and $\mathbf{F}$ is reshaped as $F = (\mathbf{f}^1 \ldots \mathbf{f}^d) \in \mathbb{R}^{p \times d}$, where $p = wh$ and $\mathbf{f}^k = \text{vec}(F^k) \in \mathbb{R}^p$ is the vectorized feature map of channel k.

B. More on experimental setup

Implementation details Following the training recipes from the pytorch models[1], we choose the ResNet protocol given its simplicity. Thus, we train over 90 epochs with SGD optimizer with momentum 0.9 and weight decay 10^{-4}. We start our training with a learning rate of 0.1 and decrease it every 30 epochs by a factor of 10. Our models are trained on 8 V100 GPUs with a batch size 32 per GPU, thus global batch size 256. We follow the same protocol for both ResNet and ConvNeXt, though a different protocol might lead to improvements on ConvNeXt.

C. More Visualizations

In addition, Figure 2 shows examples of images from the MIT 67 Scenes dataset [2] along with raw attention maps obtained by CA-Stream. These images come from four classes that do not exist in ImageNet and the network sees them at inference for the first time. Nevertheless, the attention maps focus on objects of interest in general.

[1] https://github.com/pytorch/vision/tree/main/references/classification

D. More Architectures

Table Table 1 presents interpretability metrics for both ResNet18 and ConvNeXt-S. Complementary experiments are reported on Table 2 for CUB and Pascal VOC for ResNet 50.

Network	Attribution	Pooling	AD↓	AG↑	AI↑	I↑	D↓
ResNet-18	Grad-CAM	GAP	17.64	12.73	41.21	63.13	**10.66**
		CA	**16.99**	**17.22**	**44.95**	**65.94**	10.68
	Grad-CAM++	GAP	19.05	11.16	37.99	62.80	**10.75**
		CA	**19.02**	**14.76**	**40.82**	**65.53**	10.82
	Score-CAM	GAP	13.64	12.98	44.53	62.56	**11.37**
		CA	**11.53**	**18.12**	**50.32**	**65.33**	11.51
ConvNeXt-S	Grad-CAM	GAP	42.99	1.69	12.60	48.42	**30.12**
		CA	**22.09**	**14.91**	**32.65**	**84.82**	43.02
	Grad-CAM++	GAP	56.42	1.32	10.35	48.28	**33.41**
		CA	**51.87**	**9.40**	**20.55**	**84.28**	52.58
	Score-CAM	GAP	74.79	1.29	10.10	47.40	**38.21**
		CA	**64.21**	**8.81**	**18.96**	**82.92**	57.46

Table 1. of CA-Stream *vs*. baseline GAP for more networks and interpretability methods on ImageNet.

Results on CUB in Table 2 show that our CA-Stream consistently provides improvements when the model is fine-tuned on a smaller fine-grained dataset.

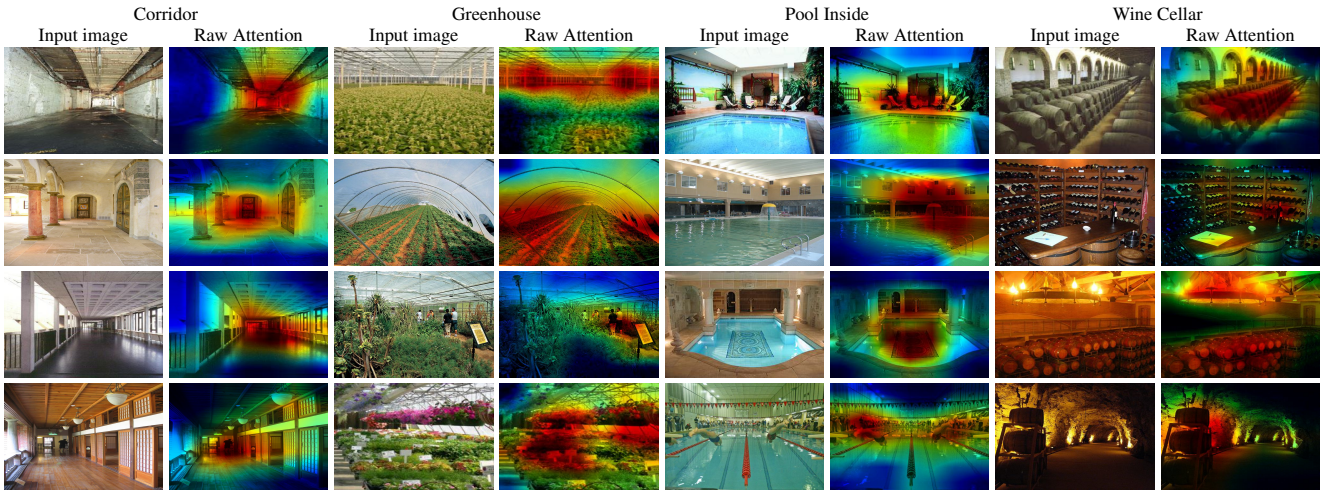

Figure 2. Raw attention maps obtained from our CA-Stream on images of the MIT 67 Scenes dataset [2] on classes that do not exist in ImageNet. The network sees them at inference for the first time.

CUB-200-2011 - ResNet-50						
	Pooling					Acc↑
	GAP					76.96
	CA					75.90
	Interpretability Metrics					
Method	Pooling	AD↓	AG↑	AI↑	I↑	D↓
Grad-CAM	GAP	10.87	10.29	45.81	65.71	**6.17**
	CA	**10.44**	**17.61**	**53.54**	**74.60**	6.56
Grad-CAM++	GAP	11.35	9.68	44.32	65.64	**5.92**
	CA	**11.01**	**16.50**	**51.63**	**74.64**	6.21
Score-CAM	GAP	9.05	10.62	48.90	65.58	5.94
	CA	**6.37**	**19.50**	**60.41**	**74.22**	**2.14**
Pascal VOC 2012 - ResNet-50						
	Pooling					mAP↑
	GAP					78.32
	CA					78.35
	Interpretability Metrics					
Method	Pooling	AD↓	AG↑	AI↑	I↑	D↓
Grad-CAM	GAP	**12.61**	9.68	27.88	**89.10**	59.39
	CA	12.77	**15.46**	**34.53**	88.53	**59.16**
Grad-CAM++	GAP	**12.25**	9.68	27.62	**89.34**	54.23
	CA	12.28	**16.76**	**34.87**	89.02	**53.34**
Score-CAM	GAP	14.8	6.76	36.41	71.10	**39.95**
	CA	**10.96**	**21.35**	**43.82**	**89.21**	51.44

Table 2. Accuracy, respectively mean Average Precision, and interpretability metrics of CA-Stream *vs*. baseline GAP for ResNet-50 on CUB and Pascal dataset.

E. Ablation Experiments

We conduct ablation experiments on ResNet50 because of its modularity and ease of modification. We investigate the effect of the cross attention block design, the placement of the CA-Stream relative to the backbone network.

Cross attention block design Following transformers [1, 3], it is possible to add more layers in the cross attention block. We consider a variant referred to as PROJ→CA, which uses linear projections $W_\ell^K, W_\ell^V \in \mathbb{R}^{d_\ell \times d_\ell}$ on the key and value

$$\text{CA}_\ell(\mathbf{q}_\ell, F_\ell) := (F_\ell W_\ell^V)^\top h_\ell(F_\ell W_\ell^K \mathbf{q}_\ell) \in \mathbb{R}^{d_\ell}, \quad (1)$$

while (10) remains.

Block Type	#Params	Accuracy
CA	6.96M	74.70
PROJ→CA	18.13M	74.41

Table 3. *Different cross attention block design for CA-Stream.* Classification accuracy and parameters using ResNet-50 on ImageNet. #Param: parameters of CA-Stream only.

Results are reported in Table 3. We observe that the stream made of vanilla CA blocks (6) offers slightly better accuracy than projections, while having less parameters. We also note that most of the computation takes place in the last residual stages, where the channel dimension is the largest. To keep our design simple, we choose the vanilla solution without projections (6) by default.

CA-Stream placement To validate the design of CA-Stream, we measure the effect of its depth on its performance *vs*. the baseline GAP in terms of both classification accuracy / number of parameters and classification metrics for interpretability. In particular, we place the stream in parallel to the network f, starting at stage ℓ and running through stage L, the last stage of f, where $0 \leq \ell \leq L$. Results are reported in Table 4.

Accuracy and Parameters					
	Placement	CLS dim	#Param	Acc↑	
	$S_0 - S_4$	64	6.96M	**74.70**	
	$S_1 - S_4$	256	6.95M	74.67	
	$S_2 - S_4$	512	6.82M	74.67	
	$S_3 - S_4$	1024	6.29M	74.67	
	$S_4 - S_4$	2048	4.20M	74.63	
Interpretability Metrics					
Method	Placement	AD↓	AG↑	AI↑	I↑	D↓
Grad-CAM	$S_0 - S_4$	**12.54**	**22.67**	48.56	75.53	13.50
	$S_1 - S_4$	12.69	22.65	48.31	75.53	13.41
	$S_2 - S_4$	**12.54**	21.67	**48.58**	75.54	13.50
	$S_3 - S_4$	12.69	22.28	47.89	**75.55**	13.40
	$S_4 - S_4$	12.77	20.65	47.14	74.32	**13.37**
Grad-CAM++	$S_0 - S_4$	13.99	19.29	44.60	75.21	13.78
	$S_1 - S_4$	13.99	19.29	44.62	75.21	13.78
	$S_2 - S_4$	13.71	**19.90**	**45.43**	75.34	13.50
	$S_3 - S_4$	13.69	19.61	45.04	**75.36**	13.50
	$S_4 - S_4$	**13.67**	18.36	44.40	74.19	**13.30**
Score-CAM	$S_0 - S_4$	**7.09**	23.65	54.20	74.91	14.68
	$S_1 - S_4$	**7.09**	23.65	54.20	74.92	14.68
	$S_2 - S_4$	**7.09**	**23.66**	**54.21**	74.91	14.68
	$S_3 - S_4$	7.74	23.03	52.92	**74.97**	14.65
	$S_4 - S_4$	7.52	19.45	50.45	74.19	**14.46**

Table 4. *Effect of stream placement* on accuracy, parameters and interpretability metrics for ResNet-50 on ImageNet. $S_\ell - S_L$: CA-Stream runs from stage ℓ to L (last); #Param: parameters of CA-Stream only.

From the interpretability metrics as well as accuracy, we observe that stream configurations that allow for iterative interaction with the network features obtain the best performance, although the effect of stream placement is small in general. In many cases, the lightest stream of only one cross attention block ($S_4 - S_4$) is inferior to options allowing for more interaction. Since starting the stream at early stages has little effect on the number of parameters and performance is stable, we choose to start the stream in the first stage ($S_0 - S_4$) by default.

Class-specific CLS As discussed in subsection 3.3, the formulation of single-query cross attention as a CAM-based saliency map (1) is class agnostic (single channel weights α_k), whereas the original CAM formulation (1) is class specific (channel weights α_k^c for given class of interest c). Here we consider a class specific extension of CA-Stream using one query vector per class. In particular, the stream is initialized by one learnable parameter $\mathbf{q}_0^c$ per class c, but only one query (CLS token) embedding is forwarded along the stream. At training, c is chosen according to the target class label, while at inference, the class predicted by the baseline classifier is used instead.

Accuracy and Parameters						
	Representation		#Param	Acc↑		
	Class agnostic		32.53M	74.70		
	Class specific		32.59M	74.68		
Interpretability Metrics						
Method	ThRepresentation	AD↓	AG↑	AI↑	I↑	D↓
Grad-CAM	Class agnostic	12.54	22.67	48.56	75.53	13.50
	Class specific	12.53	22.66	48.58	75.54	13.50
Grad-CAM++	Class agnostic	13.99	19.29	44.60	75.21	13.78
	Class specific	13.99	19.28	44.62	75.20	13.78
Score-CAM	Class agnostic	7.09	23.65	54.20	74.91	14.68
	Class specific	7.08	23.64	54.15	74.99	14.53

Table 5. *Effect of class agnostic* vs. *class specific representation* on accuracy, parameters and interpretability metrics of CA-Stream for ResNet-50 and different interpretability methods on ImageNet. #Param: parameters of CA-Stream only.

Results are reported in Table 5. We observe that the class specific representation for CA-Stream provides no improvement over the class agnostic representation, despite the additional complexity and parameters. We thus choose the class agnostic representation by default. The class specific approach is similar to [50] in being able to generate class specific attention maps, although no fine-tuning is required in our case.

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2

[2] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE conference on computer vision and pattern recognition*, 2009. 1, 2

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 2

[4] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 1