Extracting Class Activation Maps from Non-Discriminative Features as well

CVPR 2023

Paper Reading by Yiwei Sun

2023.07.11

智能多媒体内容计算实验室
Intelligent Multimedia Content Computing Lab

- 作者介绍
- 研究背景
- 解决方法
- 实验效果
- 总结反思

作者介绍

Qianru Sun 孙倩茹

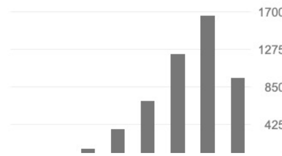
Assistant Professor of Computer Science, Singapore Management University
在 smu.edu.sg 的电子邮件经过验证 - 首页
Meta Learning Continual Learning Image Generation Semantic Segmentation Computer Vision

关注

标题	引用次数	年份
Meta-Transfer Learning for Few-Shot Learning Q Sun, Y Liu, TS Chua, B Schiele Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern …	1023	2019
Pose Guided Person Image Generation L Ma, X Jia, Q Sun, B Schiele, T Tuytelaars, L Van Gool 31st Conference on Neural Information Processing Systems (NIPS 2017)	839	2017
Disentangled Person Image Generation L Ma, Q Sun, S Georgoulis, L Van Gool, B Schiele, M Fritz IEEE Conference on Computer Vision and Pattern Recognition, 99-108		
Learning to Self-Train for Semi-Supervised Few-Shot Classification X Li, Q Sun, Y Liu, S Zheng, Q Zhou, TS Chua, B Schiele 33rd Annual Conference on Neural Information Processing Systems (NeurIPS 2019)		
Causal intervention for weakly-supervised semantic segmentation D Zhang, H Zhang, J Tang, X Hua, Q Sun 34th Annual Conference on Neural Information Processing Systems (NeurIPS 2020)		

引用次数

	总计	2018 年至今
引用	5194	5088
h 指数	26	26
i10 指数	38	32



新加坡管理大学 助理教授

研究方向：元学习、因果学习

Zhaozheng Chen

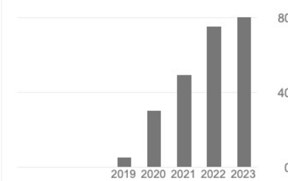
Singapore Management University
在 phdcs.smu.edu.sg 的电子邮件经过验证 - 首页
Computer Vision

关注

标题	引用次数	年份
Explicit interaction model towards text classification C Du, Z Chen, F Feng, L Zhu, T Gan, L Nie Proceedings of the AAAI conference on artificial intelligence 33 (01), 6359-6366	82	2019
Meta-transfer learning through hard tasks Q Sun, Y Liu, Z Chen, TS Chua, B Schiele IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)	57	2020
Virtually trying on new clothing with arbitrary poses N Zheng, X Song, Z Chen, L Hu, D Cao, L Nie Proceedings of the 27th ACM International Conference on Multimedia (ACMMM …	47	2019
Class Re-Activation Maps for Weakly-Supervised Semantic Segmentation Z Chen, T Wang, X Wu, XS Hua, H Zhang, Q Sun IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)	44	2022
Urban perception: Sensing cities via a deep interactive multi-task learning framework W Guan, Z Chen, F Feng, W Liu, L Nie ACM Transactions on Multimedia Computing, Communications, and Applications …	8	2021
Extracting Class Activation Maps from Non-Discriminative Features as well Z Chen, Q Sun IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)	1	2023

引用次数

	总计	2018 年至今
引用	239	239
h 指数	5	5
i10 指数	4	4



开放获取的出版物数量 查看全部

0 篇文章 4 篇文章

无法查看的文章 可查看的文章

根据资助方的强制性开放获取政策

合著作者

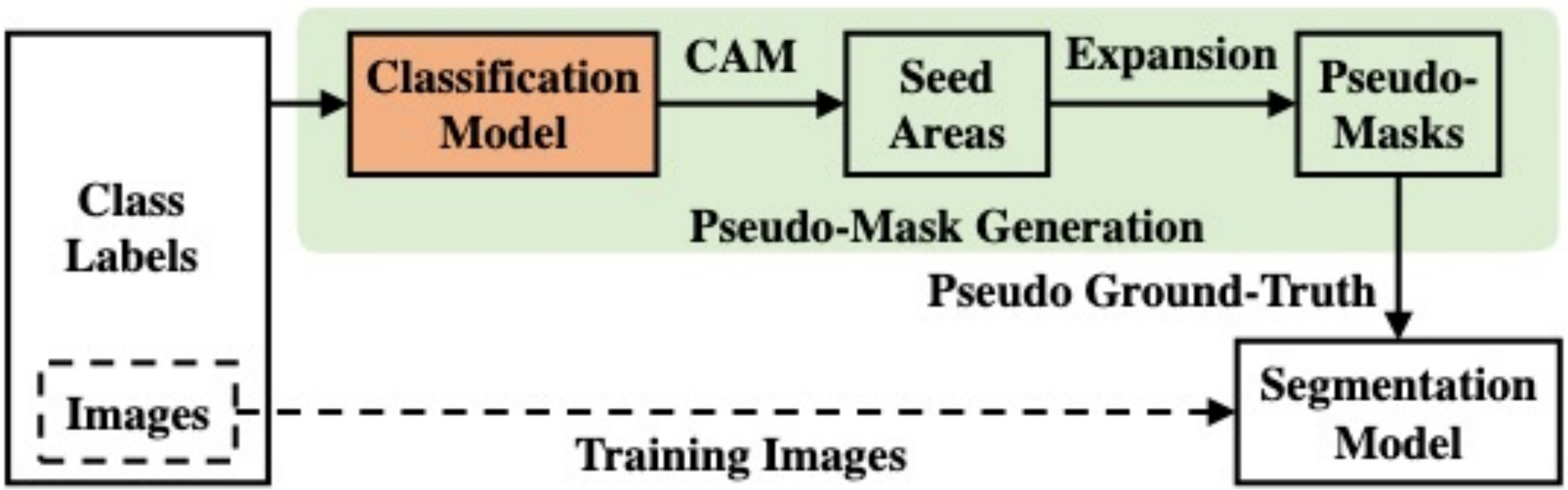
新加坡管理大学 20级博士生

研究方向：弱监督图像分割

- 作者介绍
- 研究背景
- 解决方法
- 实验效果
- 总结反思

研究背景

弱监督语义分割（WSSS）：仅依赖图像级标签完成对语义分割网络的训练。



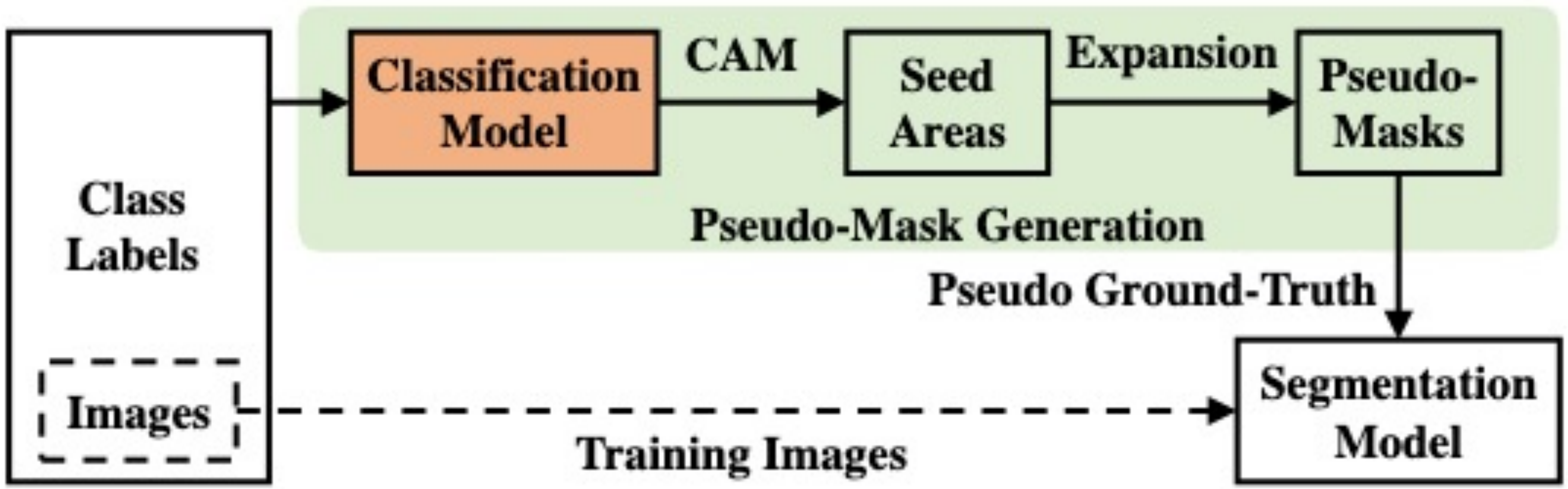
1. 通过语义标签训练分类模型；
2. 得到对应标签的类激活图，将其中的高响应区域作为Seed Areas；
3. 优化Seed Areas，形成相对可靠的分割标签，即Pseudo Masks；
4. 根据Pseudo Masks训练语义分割模型。

研究背景


```python
if args.train_cam_pass is True:
    # 训练分类模型
    import step.train_cam

    timer = pyutils.Timer('step.train_cam:')
    step.train_cam.run(args)

if args.make_cam_pass is True:
    # 计算CAM
    import step.make_cam

    timer = pyutils.Timer('step.make_cam:')
    step.make_cam.run(args)

if args.make_lpcam_pass is True:
    import step.make_lpcam

    timer = pyutils.Timer('step.make_lpcam:')
    step.make_lpcam.run(args)

if args.eval_cam_pass is True:
    import step.eval_cam

    timer = pyutils.Timer('step.eval_cam:')
    step.eval_cam.run(args)
```

```python
if args.cam_to_ir_label_pass is True:
    import step.cam_to_ir_label

    timer = pyutils.Timer('step.cam_to_ir_label:')
    step.cam_to_ir_label.run(args)

if args.train_irn_pass is True:
    # 训练掩膜优化模型
    import step.train_irn

    timer = pyutils.Timer('step.train_irn:')
    step.train_irn.run(args)

if args.make_sem_seg_pass is True:
    # 训练分割模型
    import step.make_sem_seg_labels
    args.sem_seg_bg_thres = float(args.sem_seg_bg_thres)
    timer = pyutils.Timer('step.make_sem_seg_labels:')
    step.make_sem_seg_labels.run(args)

if args.eval_sem_seg_pass is True:
    import step.eval_sem_seg

    timer = pyutils.Timer('step.eval_sem_seg:')
    step.eval_sem_seg.run(args)
```

研究背景



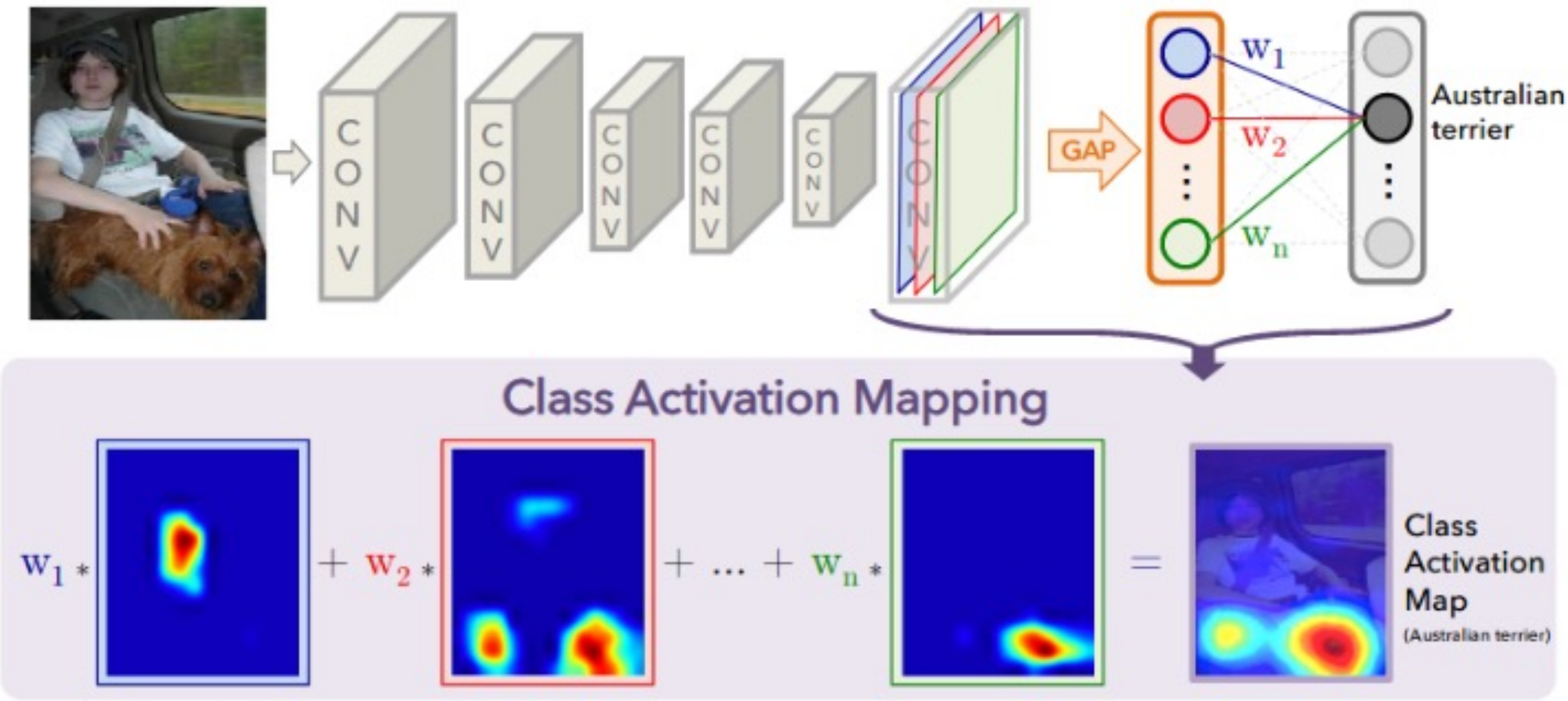
总结：根据CAM构建分割标签对模型进行监督训练。

研究重点：

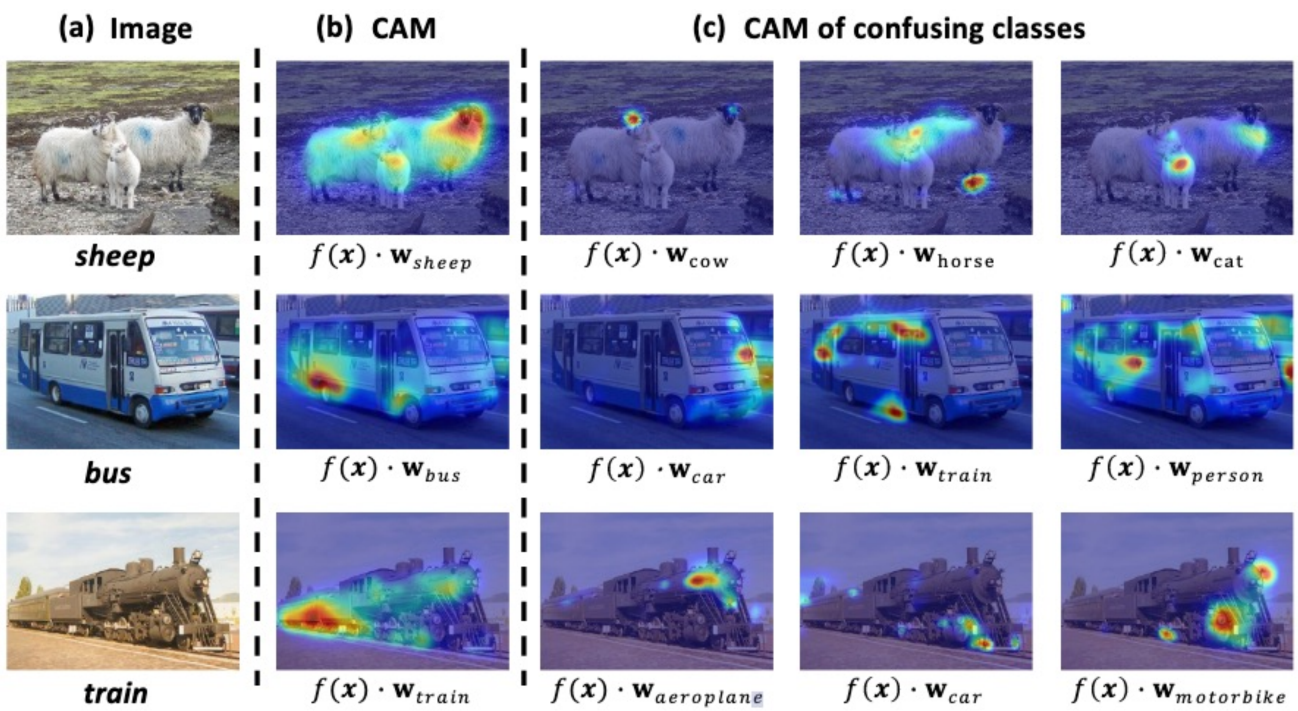
1. 如何得到CAM（本文的研究内容）；
2. 如何优化CAM。

研究背景

CAM的计算：

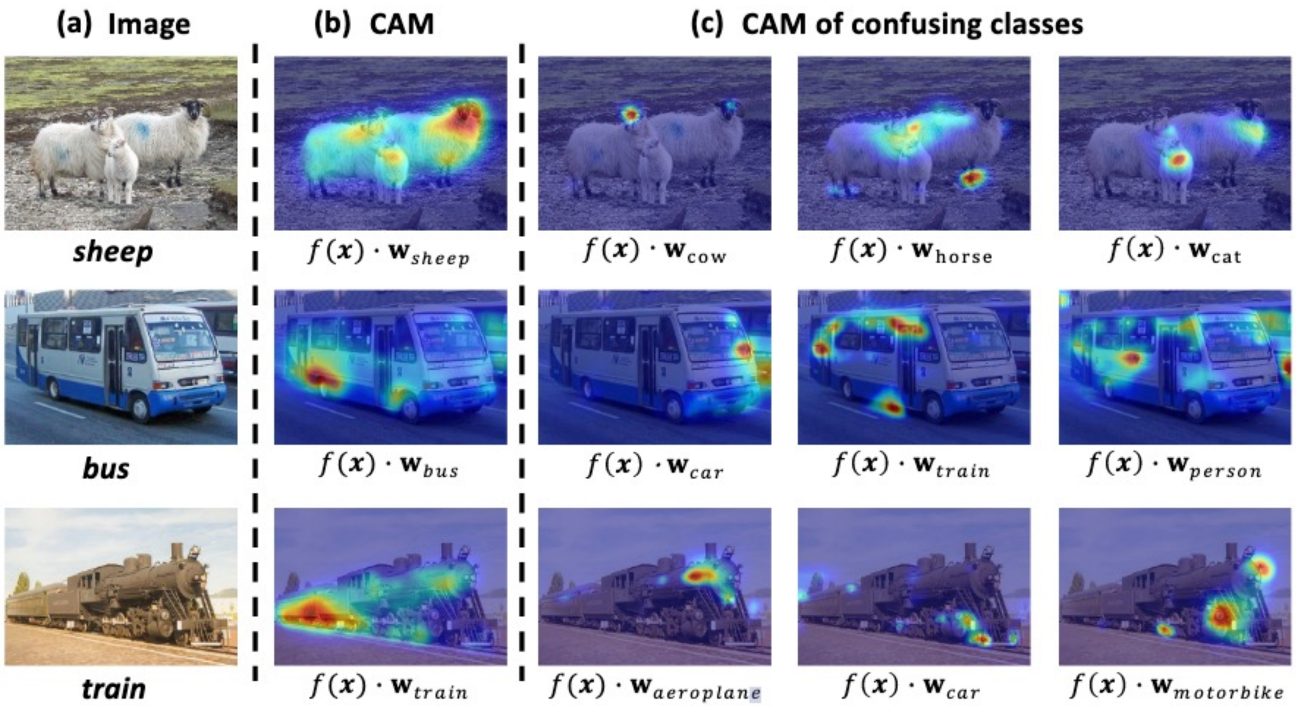

CONV
CONV
CONV
CONV
CONV
CONV
GAP
w_1
w_2
w_n
Australian terrier
Class Activation Mapping
$w_1 *$ + $w_2 *$ + ... + $w_n *$ =
Class Activation Map
(Australian terrier)

研究背景



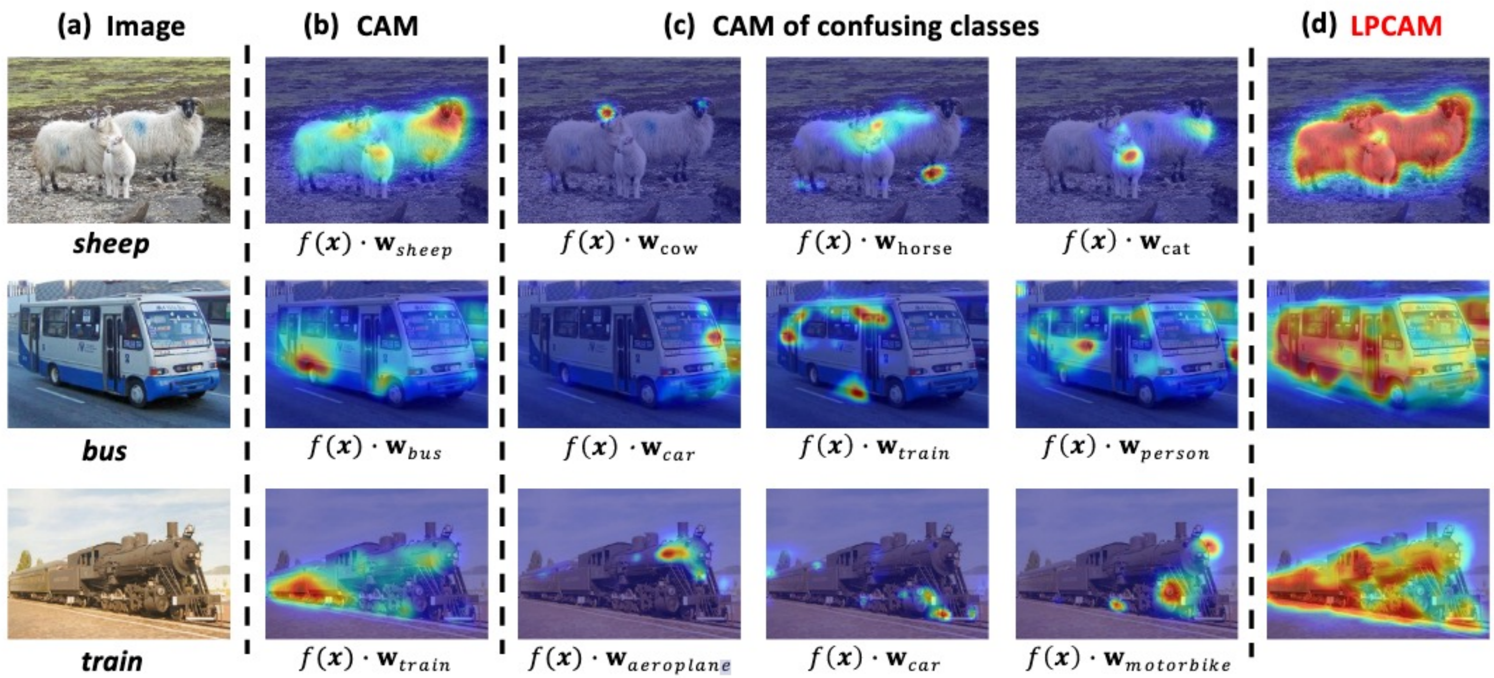
问题提出：传统的CAM存在前景区域覆盖率低的问题，如图(b)。

研究背景



问题分析：CAM是通过判别模型得到的，这一类模型自然会关注具有强区分性的区域，并且丢弃弱区分性及无区分性区域（对分类起混淆作用），即(c)。

研究背景

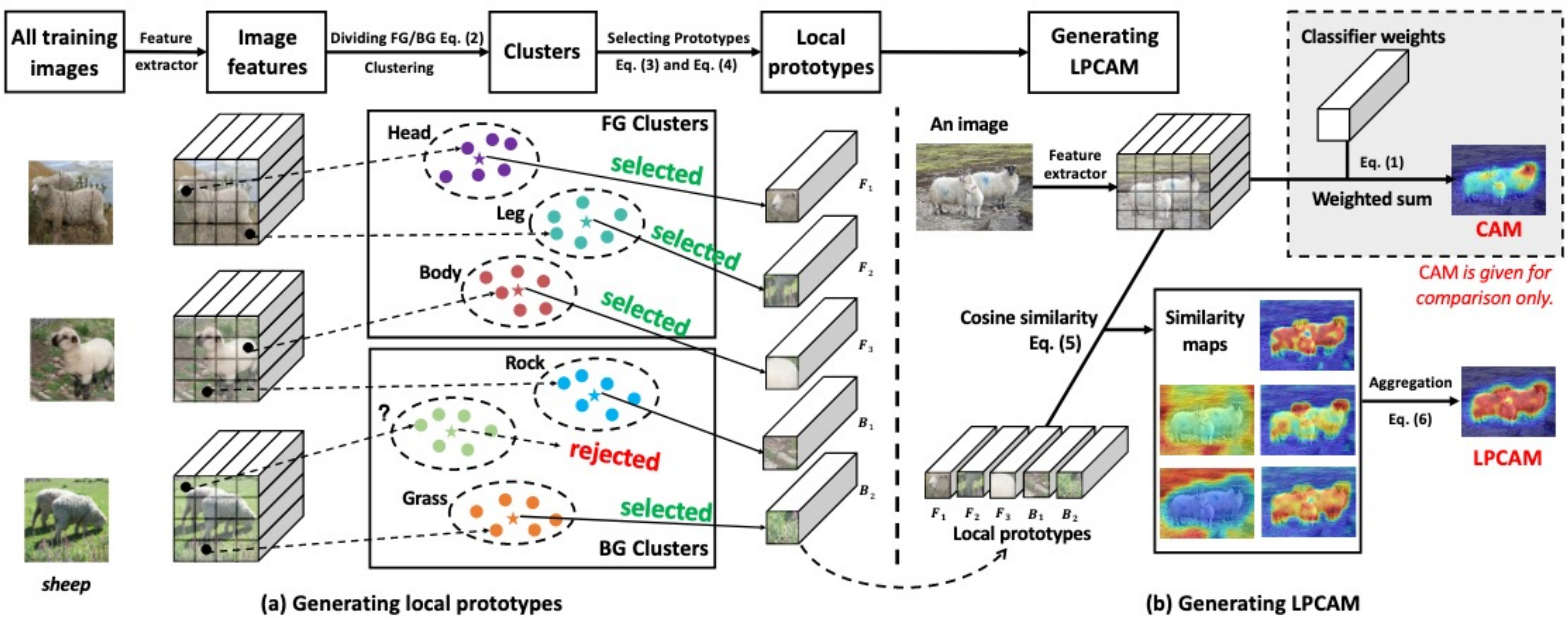


问题解决：

1. 判别模型的分类器是有偏好的，这体现在它只关注分类对象的强判别性局部；
2. 分类器的偏好体现在线性模型的系数当中，这是构建CAM的重要组成部分；
3. 因此，一个直观的想法是构建一种无偏的CAM计算方式。效果如(d)所示。

- 作者介绍
- 研究背景
- 解决方法
- 实验效果
- 总结反思

解决方法



1. 对每一类的前景像素块集合和背景像素块集合分别执行K-means算法，各得到K个聚类中心，文中称其为“局部原型”；
2. 每一个原型与特征块的每一个位置计算余弦相似度，得到余弦相似度图，并聚合为最终的类响应图。

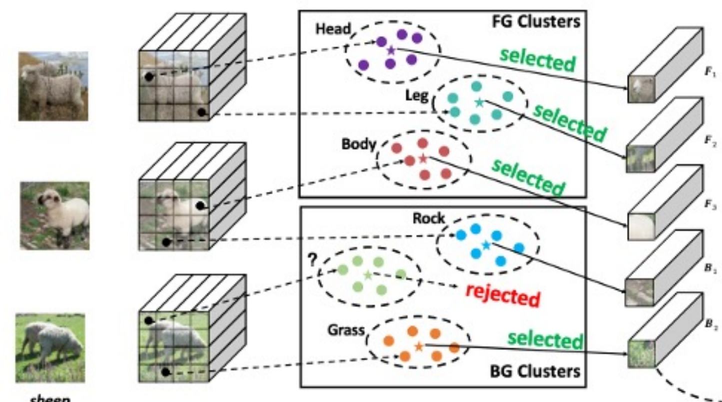
解决方法

如何得到“局部原型”？（以类别n为例）

Step1：筛选满足条件的前景和背景像素块。

$$\mathrm{CAM}_n(\boldsymbol{x}) = \frac{\mathrm{ReLU}\left(\boldsymbol{A}_n\right)}{\max\left(\mathrm{ReLU}\left(\boldsymbol{A}_n\right)\right)}, \quad \boldsymbol{A}_n = \mathbf{w}_n^{\top} f(\boldsymbol{x}).$$

$$f(\boldsymbol{x})^{i,j} \in \begin{cases} \mathcal{F}, & \text{if } \mathrm{CAM}_n^{i,j}(\boldsymbol{x}) \geq \tau \\ \mathcal{B}, & \text{otherwise} \end{cases}$$



(a) Generating local prototypes

此时我们得到了前景像素集合和背景像素集合$F = \{F_1, \dots, F_N\}$ $B = \{B_1, \dots, B_M\}$

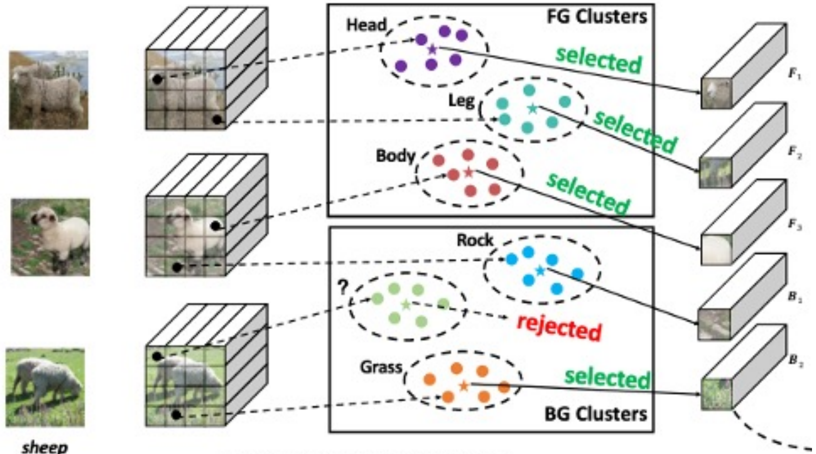
解决方法

如何得到“局部原型”？（以类别n为例）

Step2：执行K-Means算法，得到K个候选局部原型。
Step3：进一步筛选，得到最终的局部原型。

$$z_i = \frac{\exp(\mathbf{F}_i \cdot \mathbf{w}_n)}{\sum_j \exp(\mathbf{F}_i \cdot \mathbf{w}_j)}.$$

$$z_i = \frac{\exp(\mathbf{B}_i \cdot \mathbf{w}_n)}{\sum_j \exp(\mathbf{B}_i \cdot \mathbf{w}_j)}.$$



(a) Generating local prototypes

将$z_i > \mu_f$的前景像素块保留，将$z_i < \mu_b$的背景像素块保留。

解决方法

可能存在的问题：

Q1：如何构建训练数据的前景像素集合和背景像素集合？

```python
img_selected = torch.nonzero(tensor_label[:,class_id])[:,0].numpy() # 返回非零元素的索引，即返回标签为class_id在数据中的全部索引
feature_selected = []
feature_not_selected = []
for idx in img_selected:
    name = id2name[idx]
    # 取出单个样本的CAM信息
    cam = np.load(osp.join(mask_dir, name+'.npy'), allow_pickle=True).item()
    # 信息1:CAM
    mask = cam['high_res']
    # 信息2:包含的类别
    valid_cat = cam['keys']
    feature_map = tensor_feature[idx].permute(1,2,0)
    size = feature_map.shape[:2]
    mask = F.interpolate(torch.tensor(mask).unsqueeze(0),size)[0]
    # 如果包含的类别正是目前计算的K-Means的Target，那么计算F和B
    for i in range(len(valid_cat)):
        if valid_cat[i]==class_id:
            mask = mask[i]
            position_selected = mask>select_thres          # F
            position_not_selected = mask<select_thres      # B
            feature_selected.append(feature_map[position_selected])
            feature_not_selected.append(feature_map[position_not_selected])
feature_selected = torch.cat(feature_selected,0)
feature_not_selected = torch.cat(feature_not_selected,0)

# class center
cluster_ids_x, cluster_centers = kmeans(X=feature_selected, num_clusters=num_cluster, distance='cosine', device=torch.device('cuda:0'), tol=tol)
cluster_ids_x2, cluster_centers2 = kmeans(X=feature_not_selected, num_clusters=num_cluster, distance='cosine', device=torch.device('cuda:0'), tol=tol)
```

解决方法

可能存在的问题：

Q2：强判别性区域是单一的，那么最后会不会得到的得到语义相似的局部原型（没有意义的结果）？

我的理解：计算LPCAM的过程是一种后处理，所以涉及到一类样本。分类器在这类样本中的每一个个体上得到的强判别性区域是不同的，因此通常情况下会产生多个不同语义局部原型。

补充说明：多个局部原型是建立在分类器的偏好下生成的，因此用分类器的系数去衡量局部原型的可靠性是合理的，即Step3。

$$z_i = \frac{\exp(\mathbf{F}_i \cdot \mathbf{w}_n)}{\sum_j \exp(\mathbf{F}_i \cdot \mathbf{w}_j)}.$$

解决方法

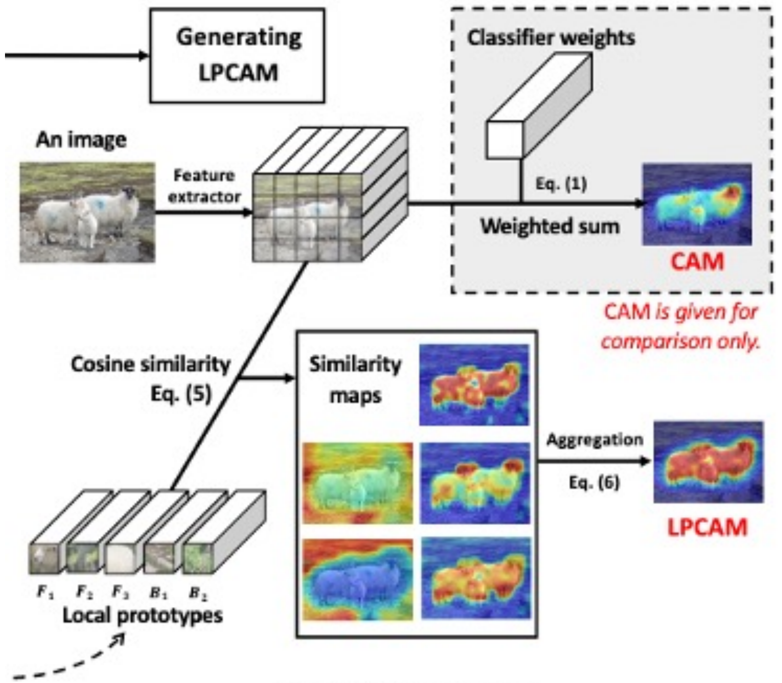
如何得到LPCAM？（以类别n为例）

$$\boldsymbol{FG}_n = \frac{1}{K_1'} \sum_{\mathbf{F}_\mathrm{i}' \in \mathbf{F}'} sim(f(\boldsymbol{x}), \mathbf{F}_\mathrm{i}'),$$

$$\boldsymbol{BG}_n = \frac{1}{K_2'} \sum_{\mathbf{B}_\mathrm{i}' \in \mathbf{B}'} sim(f(\boldsymbol{x}), \mathbf{B}_\mathrm{i}'),$$

$$\mathrm{LPCAM}_n(\boldsymbol{x}) = \frac{\mathrm{ReLU}\left(\boldsymbol{A}_n\right)}{\max\left(\mathrm{ReLU}\left(\boldsymbol{A}_n\right)\right)},$$

$$\boldsymbol{A}_n = \boldsymbol{FG}_n - \boldsymbol{BG}_n,$$



(b) Generating LPCAM

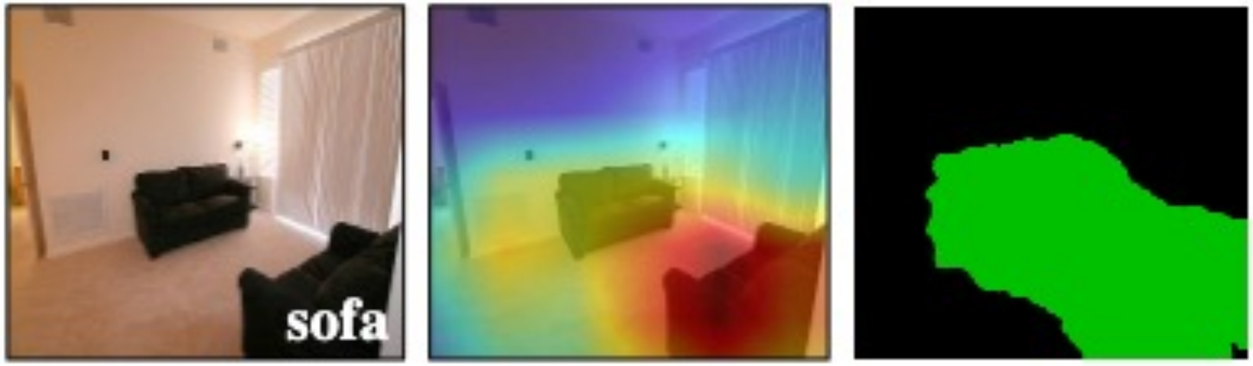
如何无偏？

包容每一种偏好以达成无偏。

解决方法

可能存在的问题：

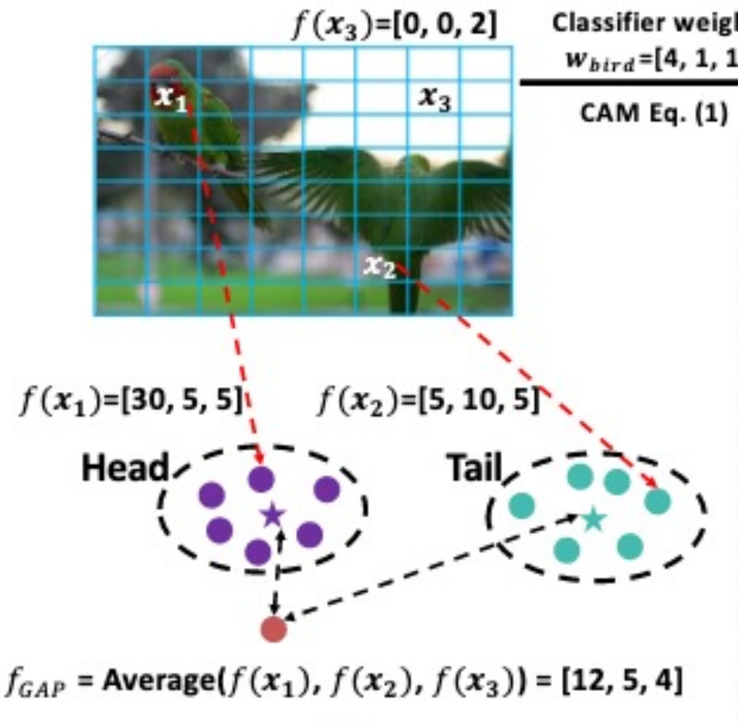
Q3：明明突出前景集合，为什么还要构建背景的局部原型并进行减处理。



判别模型的另一个问题是：对强共现性事物难以区分。即“沙发”和“地毯”经常一起出现，因此模型会对两者一同产生很强的响应。图A的正响应在图B中或许是负响应，该操作的目的是为了通过B中得到的负响应去消除A中的正响应。

解决方法

通过数值分析LPCAM的有效性



(a)

Before Normalization

	x_1	x_2	x_3
CAM	130	35	2

Local prototypes

	x_1	x_2	x_3
head	1.0	0.1	0.0
tail	0.2	0.9	0.1
sky	0.0	0.0	0.9
LPCAM	0.60	0.50	-0.85

After Normalization

	x_1	x_2	x_3
CAM	1.00	0.27	0.02
LPCAM	1.00	0.83	0.00

(b)

1. 有偏分类器是由于全局平均池化造成的；
2. 归一化后，明显能看到弱区分性特征的响应值与强区分性特征的差距缩小。

- 作者介绍
- 研究背景
- 解决方法
- 实验效果
- 总结反思

实验效果

CAM、LPCAM(不消除背景)、LPCAM之间的性能差异

	FP	FN	mIoU	Prec.	Recall
CAM	26.5	26.2	48.8	65.0	65.2
LPCAM-F	33.1+6.6	16.2-10.0	52.1+3.3	61.3-3.7	76.6+11.4
LPCAM	29.8+3.3	16.7-9.5	54.9+6.1	64.9-0.1	77.2+12.0

Table 1. An ablation study on VOC dataset. "-F" denotes only the "Foreground" term $\boldsymbol{FG}_n$ is used in Eq. 6. *Please refer to the supplementary materials for the results on MS COCO.*

FP提升，Precision下降：混杂背景。背景消除之后FP下降、Precision提升。

FN下降，Recall提升：弱区分性特征得到关注。

实验效果

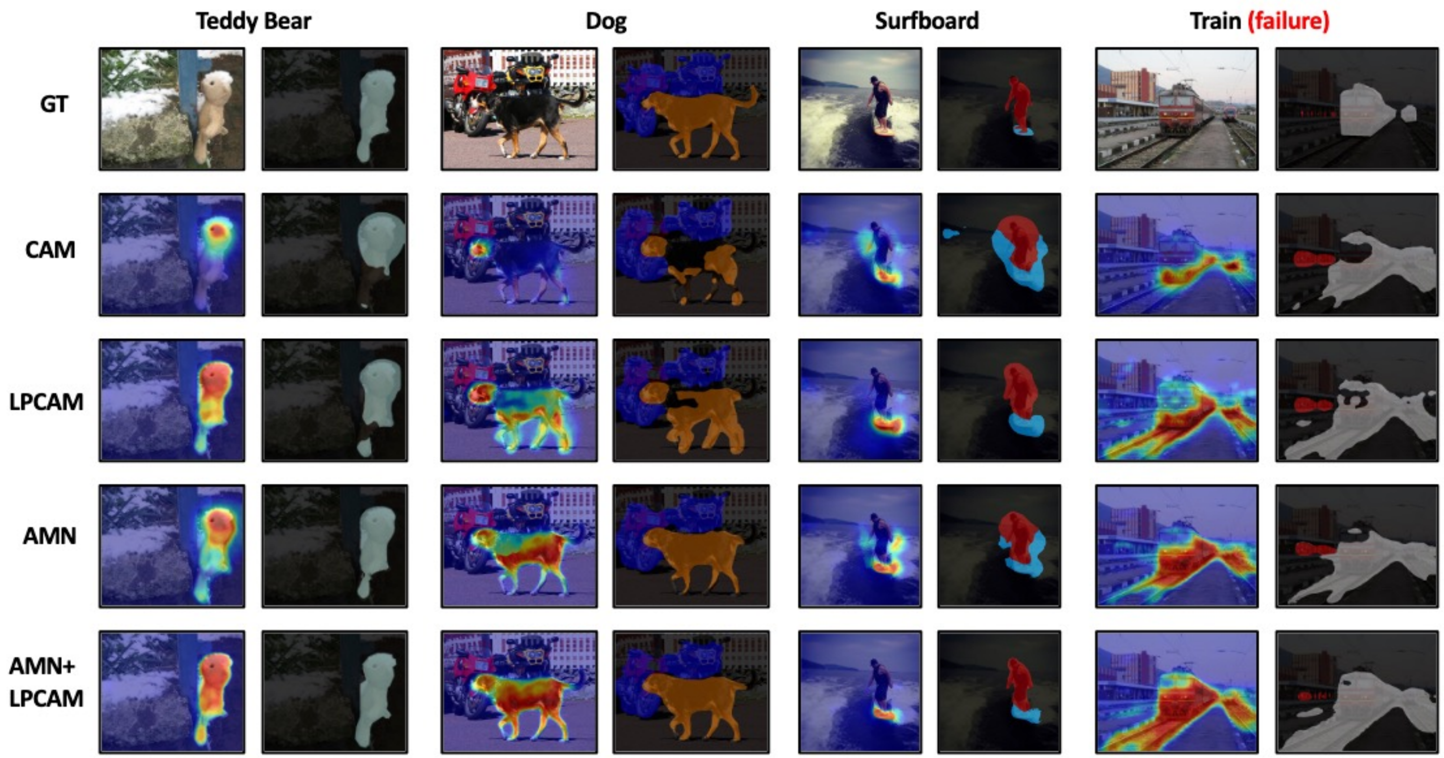
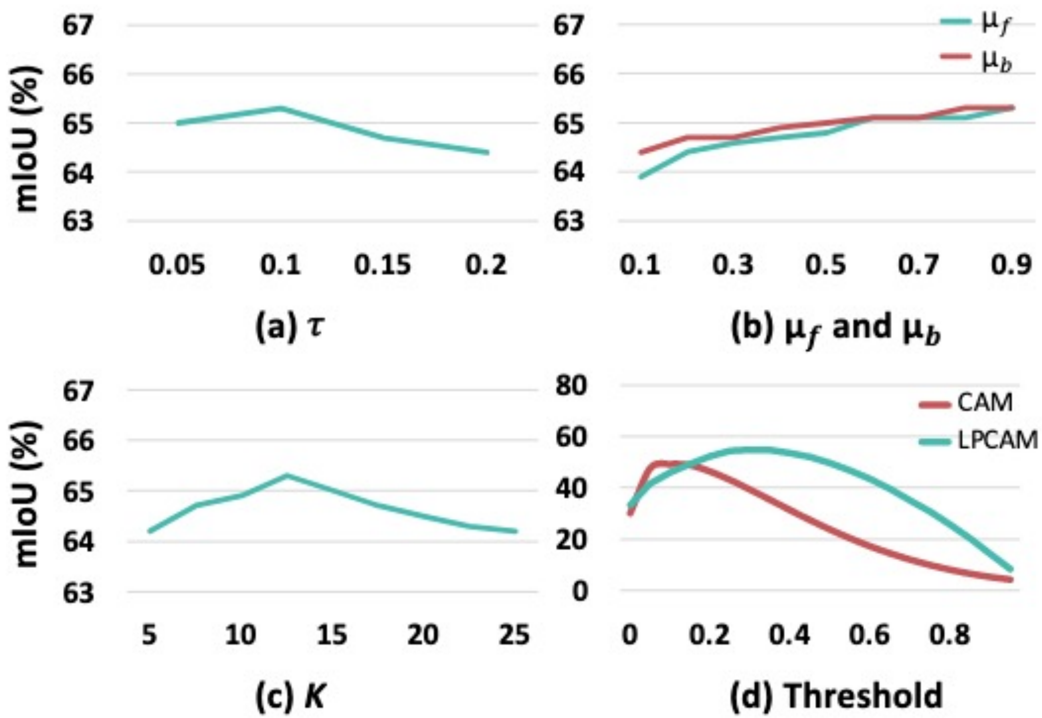


Figure 4. Qualitative results on MS COCO. In each example pair, the left is heatmap and the right is seed mask. *Please refer to the supplementary materials for the qualitative results on VOC.*

1. 弱区分性得到关注；
2. 无法解决超强共现性造成的误分类问题。

实验效果



τ:在CAM上划分前景和背景；

$\mu_{f/\mathrm{b}}$:从聚类中心筛选可靠中心；

K：聚类中心个数;

Threshold：在激活图上划分前景和背景；

实验效果

	Methods	Seed Mask		Pseudo Mask	
		CAM	LPCAM	CAM	LPCAM
VOC	IRN [1]	48.8	54.9+6.1	66.5	71.2+4.7
	EDAM [39]	52.8	54.9+2.1	68.1	69.6+1.5
	MCTformer [45]	61.7	63.5+1.8	69.1	70.8+1.7
	AMN [26]	62.1	65.3+3.2	72.2	72.7+0.5
COCO	IRN [1]	33.1	35.8+2.7	42.5	46.8+4.3
	AMN [26]	40.3	42.5+2.2	46.7	47.7+1.0

Table 2. Taking LPCAM as a substitute of CAM in state-of-the-art WSSS methods. Except MCTformer [45] using DeiT-S [35], other methods all use ResNet-50 as feature extractor.

1. 比较掩膜的效果；
2. 不同的分类模型，相同的分割模型。

	Methods	Sal.	VOC		MS COCO
			val	test	val
ResNet-50	IRN [1] CVPR'19		63.5	64.8	42.0
	LayerCAM [19] TIP'21		63.0	64.5	-
	AdvCAM [24] CVPR'21		68.1	68.0	44.2
	RIB [22] NeurIPS'21		68.3	68.6	44.2
	ReCAM [8] CVPR'22		68.5	68.4	42.9
	IRN+LPCAM		68.6	68.7	44.5
	SIPE [7] CVPR'22		68.8	69.7	40.6
	OOD [25]+Adv CVPR'22		69.8	69.9	-
	AMN [26] CVPR'22		69.5	69.6	44.7
	AMN+LPCAM		70.1	70.4	45.5
	ESOL [28] NeurIPS'22		69.9*	69.3*	42.6
	CLIMS [42] CVPR'22		70.4*	70.0*	-
	EDAM [39] CVPR'21	✓	70.9*	71.8*	-
	EDAM+LPCAM	✓	71.8*	72.1*	42.1
WideResNet-38	Spatial-BCE [38] ECCV'22		70.0	71.3	35.2
	BDM [43] ACMMM'22	✓	71.0	71.0	36.7
	RCA [51]+OOA CVPR'22	✓	71.1	71.6	35.7
	RCA [51]+EPS CVPR'22	✓	72.2	72.8	36.8
	HGNN [47] ACMMM'22	✓	70.5*	71.0*	34.5
	EPS [27] CVPR'21	✓	70.9*	70.8*	-
	RPIM [34] ACMMM'22	✓	71.4*	71.4*	-
	L2G [18] CVPR'22	✓	72.1*	71.7*	44.2
DeiT-S	MCTformer [45] CVPR'22		71.9†	71.6†	42.0
	MCTformer+LPCAM		72.6†	72.4†	42.8

Table 4. The mIoU results (%) based on DeepLabV2 on VOC and MS COCO. The side column shows three backbones of multi-label classification model. "Sal." denotes using saliency maps. * denotes the segmentation model is pre-trained on MS COCO. † denotes the segmentation model is pre-trained on VOC.

实验效果

Methods	VOC								MS COCO			
	DeepLabV2				UperNet-Swin				DeepLabV2		UperNet-Swin	
	CAM		LPCAM		CAM		LPCAM		CAM	LPCAM	CAM	LPCAM
	val	test	val	test	val	test	val	test	val	val	val	val
IRN [1]	63.5	64.8	$68.6_{+5.1}$	$68.7_{+3.9}$	65.9	67.7	$71.1_{+5.2}$	$71.8_{+4.1}$	42.0	$44.5_{+2.5}$	44.0	$47.0_{+3.0}$
AMN [26]	69.5	69.6	$70.1_{+0.6}$	$70.4_{+0.8}$	71.7	71.8	$73.1_{+1.4}$	$73.4_{+1.6}$	44.7	$45.5_{+0.8}$	47.1	$48.3_{+1.2}$
EDAM [39]	70.9^{*}	70.6^{*}	$71.8^{*}_{+0.9}$	$72.1^{*}_{+1.5}$	71.2	71.0	$72.7_{+1.6}$	$72.5_{+1.5}$	40.6	$42.1_{+1.5}$	41.7	$43.0_{+1.3}$
MCTformer [45]	$71.9^{\dagger}$	$71.6^{\dagger}$	$72.6^{\dagger}_{+0.7}$	$72.4^{\dagger}_{+0.8}$	70.6	70.3	$72.0_{+1.4}$	$72.5_{+2.2}$	-	-	-	-

Table 3. The mIoU results (%) of WSSS using different segmentation models on VOC and MS COCO. Seed masks are generated by either CAM or LPCAM, and mask refinement methods are row titles. * denotes the segmentation model (ResNet-101 based DeepLabV2) is pre-trained on MS COCO. † denotes the segmentation model (WideResNet-38 based DeepLabV2) is pre-trained on VOC.

3. 相同的分类模型，不同的分割模型。

- 作者介绍
- 研究背景
- 解决方法
- 实验效果
- 总结反思

总结反思

1. 模型对强区分性特征的过度关注是一种shortcut learning的体现，这会制约模型的泛化能力；
2. 本文的初衷是为了扩大CAM在目标上的响应区域，这样一种引导模型关注弱区分性特征的构想是值得学习的；
3. 美中不足的是，构建过程是多阶段的，十分繁琐，仅仅适用于目前的WSSS任务。