Pointly-Supervised Weak-Shot Semantic Segmentation via Dual Mapping Transfer

Wenhui Jiang[1] Ruikang Luo[1] Zeyu Luo[1] Xiaowei Zhao[2] Junjie Chen[1] Yuming Fang[1]

[1]Jiangxi University of Finance and Economics [2]Sany Heavy Industry Co.,Ltd

Introduction

- We are the first to explore pointly-supervised weak-shot semantic segmentation, where the model jointly learns from full annotations of base classes and point-level annotations of novel classes.
- We design a novel model transferring query-to-mask mapping and keypoint-to-mask mapping, in which the latter mapping could help to learn the former mapping for novel classes.
- We design a confident memory bank to alleviate the object ambiguity of novel classes.
- Extensive experiments on two datasets demonstrate the effectiveness of our method.

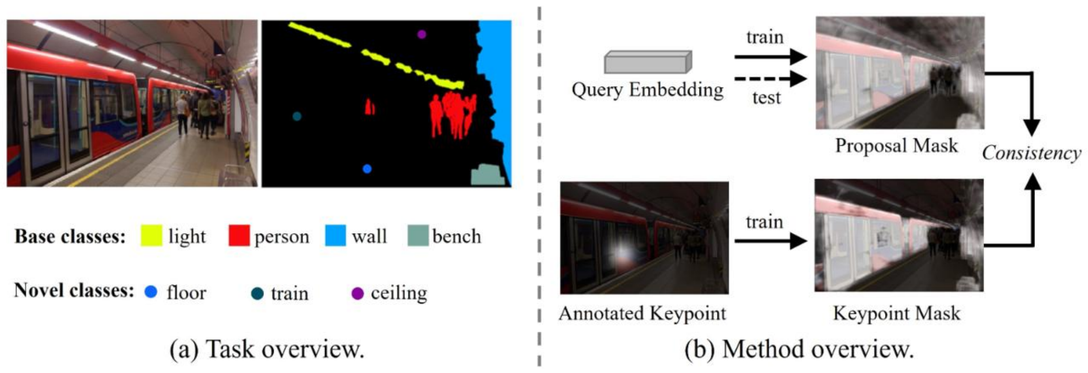

Figure 1: Task&Method overview

Method Overview

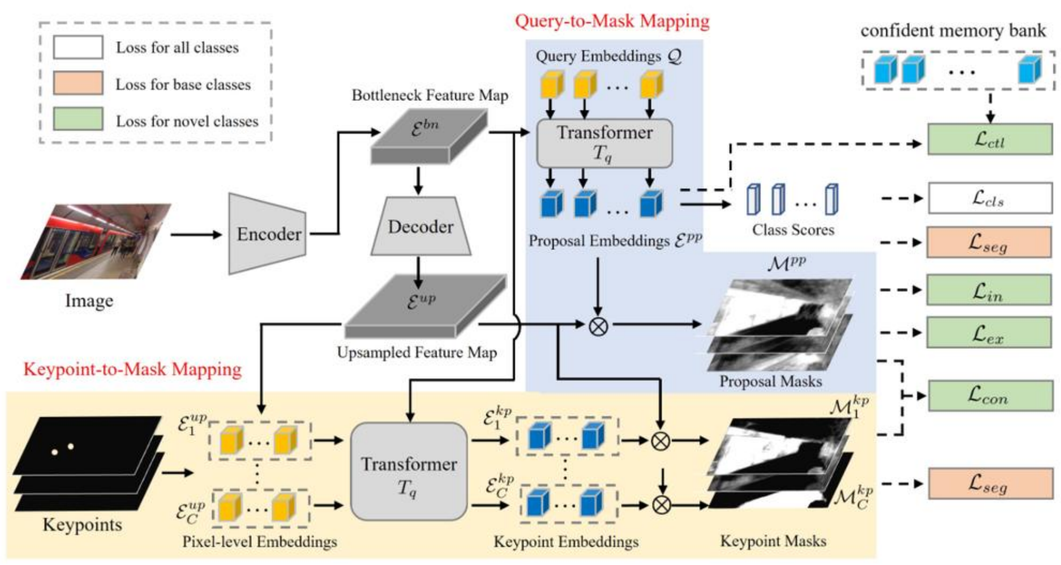

Figure 2: The illustration of our framework.

Keypoint-to-Mask Mapping

Generates class-specific masks: For each keypoint embedding of class c, compute its similarity with all pixel embeddings (dot product + sigmoid) to produce a keypoint mask; union all masks of class c yields the class-specific mask.

Query-to-Mask Mapping

Assigns class labels to proposals: Converts N query embeddings into N proposal masks; assigns class labels using assignment cost (classification + segmentation for base classes; adds inclusion/exclusion costs for novel classes using known positive/negative points).

Confident Memory Bank

AReduces object ambiguity: Employs contrastive loss L_{ctl} to align novel-class proposal embeddings with prototypes; computes novel-class confidence as $1-\omega$ (where ω is co-occurrence probability from class-pair statistics).

Loss Function

Contrastive loss

$$\mathcal{L}_{ctl} = -\log \frac{\exp(\mathcal{E}^{pp}[i,:]\cdot\mathcal{B}[c,:]^T/\tau)}{\sum_{c'\in\mathcal{C}^{novel}}\exp(\mathcal{E}^{pp}[i,:]\cdot\mathcal{B}[c',:]^T/\tau)}$$

Inclusion loss、Exclusion loss and Consistency loss

$$\mathcal{L}_{in}=\mathcal{L}_{\subseteq}(\hat{\mathcal{M}}_{\delta(i)},\mathcal{M}_i^{pp})\ \mathcal{L}_{ex}=\mathcal{L}_{\subseteq}(\mathcal{M}_i^{pp},\mathbf{1}-\bigcup_{c'\neq\delta(i)}\hat{\mathcal{M}}_{c'})\ \mathcal{L}_{con}=\|\mathcal{M}_i^{pp}-\mathcal{M}_{\delta(i)}^{kp}\|_1$$

Base class total loss

$$\mathcal{L}_b=\mathcal{L}_{cls}(\delta(n),\mathbf{y}[n])+\mathcal{L}_{seg}(\boldsymbol{M}^*[\delta(n)],\boldsymbol{M}^{pp}[n])+\mathcal{L}_{seg}(\boldsymbol{M}^*[\delta(n)],\boldsymbol{M}^{kp}[\delta(n)])$$

Novel class total loss

$$\mathcal{L}_n=\mathcal{L}_{cls}(\delta(n),\mathbf{y}[n])+\mathcal{L}_{ctl}+\lambda_1\cdot\mathcal{L}_{in}(\boldsymbol{M}^*[\delta(n)],\boldsymbol{M}^{pp}[n])$$
$$+\lambda_2\cdot\mathcal{L}_{ex}\left(\bigcup_{c'\neq\delta(n)}\boldsymbol{M}^*[c'],\boldsymbol{M}^{pp}[n]\right)+\lambda_3\mathcal{L}_{con}(\boldsymbol{M}^{kp}[\delta(n)],\boldsymbol{M}^{pp}[n])$$

Experimental Results

Quantitative comparison results

We evaluate different methods on two benchmark datasets: COCO-Stuff-10K and ADE20K. COCO-Stuff-10K has 9k training images and 1k test images from 171 classes.ADE20K has 20k training images and 2k validating images from 150 classes. For each dataset, we split all the classes into base classes and novel classes with the split ratio $|C_{base}| : |C_{novel}| = 3:1$.

Table 1: The mIoU results of various methods on two benchmark datasets.

Published in	Method	COCO-Stuff-10K[3] split1	split2	split3	split4	Mean	ADE20K[48] split1	split2	split3	split4	Mean
TPAMI-21	A^2GNN[41]	21.6	18.0	23.6	19.1	20.6	28.2	24.1	21.3	21.9	23.9
	A^2GNN*[41]	26.8	22.8	28.4	23.1	25.3	33.4	28.7	23.8	26.3	28.0
CVPR-22	TEL[20]	24.0	19.6	25.1	21.2	22.5	29.8	27.0	21.8	23.7	25.6
	TEL*[20]	28.1	23.2	30.5	25.2	26.8	34.2	30.2	25.7	27.7	29.5
CVPR-23	AGMM[38]	25.1	21.4	28.2	23.6	24.6	31.3	28.4	23.3	24.4	26.9
	AGMM*[38]	29.3	25.0	32.9	26.9	28.5	35.8	32.6	28.3	29.2	31.5
AAAI-24	CDSP[46]	30.1	25.5	30.7	28.2	28.6	35.2	32.7	27.3	29.1	31.1
	CDSP*[46]	34.3	30.1	36.5	32.9	33.5	39.6	38.3	30.6	32.0	35.1
BMVC-22	RETAB†[49]	27.5	23.1	29.8	25.6	26.5	34.6	31.3	25.7	25.9	29.4
	RETAB* [49]	28.6	24.5	31.2	27.3	27.9	36.0	32.3	28.2	28.9	31.4
NeurIPS-22	SimFormer†[6]	33.5	27.4	33.7	31.7	31.6	37.6	35.3	29.5	31.8	33.6
	SimFormer* [6]	34.1	29.3	35.9	32.1	32.9	39.1	38.1	30.7	32.2	35.0
Our submit	Ours	36.6	32.6	40.1	34.6	36.0	43.4	40.7	33.4	33.9	37.9
	Ours*	**37.0**	**33.3**	**41.3**	**35.3**	**36.7**	**44.1**	**42.3**	**34.9**	**34.4**	**39.0**
	FullyOracle†	37.5	34.0	43.8	36.0	37.8	47.8	44.3	38.3	36.2	41.7

Ablation Studies

We investigate the impact of each loss term (L_{in}, L_{ex}, L_{con}, L_{ctl}) and each assignment cost term (J_{in}, J_{ex}, J_{con}) for novel classes in table 2. We also explore different variants of network designs in Table 3.

Table 2: Ablation studies on each loss term and each assignment cost term on COCO-Stuff-10K

Row	$\mathcal{L}_{in}$	$\mathcal{J}_{in}$	$\mathcal{L}_{ex}$	$\mathcal{J}_{ex}$	$\mathcal{L}_{con}$	$\mathcal{J}_{con}$	$\mathcal{L}_{ctl}$	Split1	Split2	Split3	Split4
1	✓							32.2	26.6	33.9	30.3
2	✓	✓						32.9	27.3	34.4	31.0
3	✓	✓	✓					34.2	28.9	36.6	32.1
4	✓	✓	✓	✓				34.6	29.3	37.3	32.7
5	✓	✓	✓	✓	✓			35.7	31.7	38.9	34.0
6	✓	✓	✓	✓	✓	✓		36.2	32.1	39.4	34.3
7	✓	✓	✓	✓	✓	✓	✓	36.6	32.6	40.1	34.6

Table 3: The mIoU results of different variants of network designs on COCO-Stuff-10K

Query	Keypoint	Split1	Split2	Split3	Split4
T_q		35.0	29.6	37.9	33.2
T_q	T_q	36.6	32.6	40.1	34.6
T_q	T_k	36.7	32.8	40.2	34.7

Visual comparison

Table 4: Visual comparison of various methods on COCO-STuff-10K and ADE20K datasets.