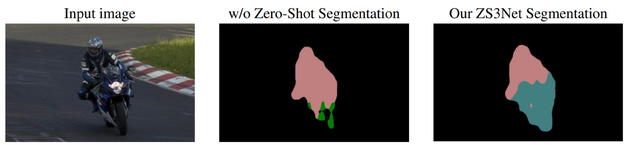
Zero-Shot Semantic Segmentation

Maxime Bucher[(1)], Tuan-Hung Vu[(1)], Matthieu Cord[(1,2)], Patrick Pérez[(1)]
(1) valeo.ai, (2) Sorbonne Université, firstname.lastname@valeo.com

I. Zero-Shot Learning

Zero-Shot Learning (ZSL) aims to classify unseen objects by leveraging auxiliary knowledge (eg. word2vec).



In this example, there are no 'motorbike' examples in the training set. As a consequence, a supervised model (middle) fails on this object, seeing it as a mix of the seen classes person, bicycle and background. With proposed ZS3Net method (right), pixels of the never-seen motorbike class are recognized.

II. Contributions

- Introduce the new task of zero-shot semantic segmentation and propose an architecture (ZS3Net), to address it
- Extend the model by exploiting contextual cues from spatial region relationship
- Propose a novel self-training step in a relaxed zero-shot setup where unlabelled pixels from unseen classes are already available at training time
- Report evaluations on two datasets with varying numbers of unseen classes.

III. Approach: ZS3Net

Motivated by the fact that, without adaptation, the prediction on target is uncertain as reflected in *entropy* of the prediction, we propose entropy minimization for domain adaptation.

Two ways to minimize entropy for domain adaptation:

- **Direct entropy minimization**

  Minimizing the entropy of model prediction $\mathcal{L}_{ent}(\boldsymbol{x})$ given by

  $$\boldsymbol{E}_{\boldsymbol{x}}^{(h,w)} = \frac{-1}{\log(C)} \sum_{c=1}^{C} \boldsymbol{P}_{\boldsymbol{x}}^{(h,w,c)} \log \boldsymbol{P}_{\boldsymbol{x}}^{(h,w,c)},$$

  where $\boldsymbol{P}_{\boldsymbol{x}}$ is the output of the model after softmax for input image $\boldsymbol{x}$, C is the number of semantic classes and (h, w) index the pixels of the output.

- **Entropy minimization with Adversarial training**

  Alignment with source by adversarial training on *weighted self-information*:

  $$\boldsymbol{I}_{\boldsymbol{x}}^{(h,w)} = -\boldsymbol{P}_{\boldsymbol{x}}^{(h,w)} \log \boldsymbol{P}_{\boldsymbol{x}}^{(h,w)}.$$

  Adversarial training:
  Train the discriminator D to classify domain of the input by

  $$\min_{\theta_D} \frac{1}{|\mathcal{X}_s|} \sum_{\boldsymbol{x}_s} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_s}, 1) + \frac{1}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_t}, 0),$$

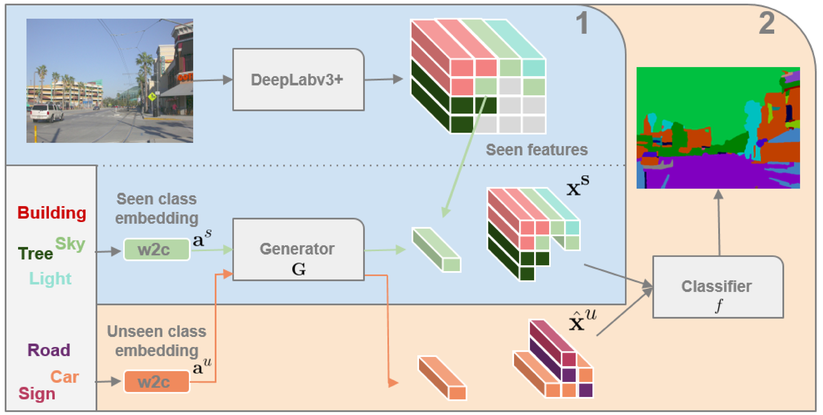
  and train the model to fool D by

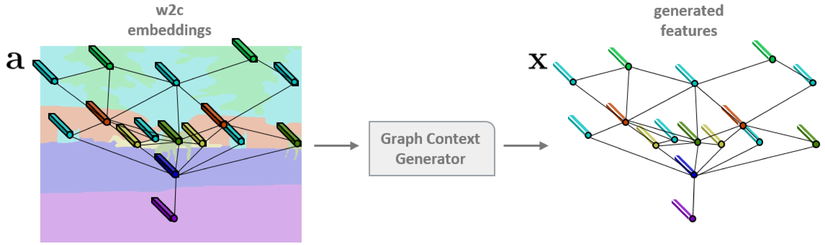
  $$\min_{\theta_F} \frac{1}{|\mathcal{X}_t|} \sum_{\boldsymbol{x}_t} \mathcal{L}_D(\boldsymbol{I}_{\boldsymbol{x}_t}, 1).$$

References

[1] Learning to adapt structured output space for semantic segmentation, YH. Tsai, WC. Hung, S. Schulter, K. Sohn, MH. Yang, M. Chandraker, CVPR 2018

IV. ZS3Net Architecture and Graph-context encoding





The segmentation mask is encoded as an adjacency graph of semantic connected components (represented as nodes with different colors in the graph). Each semantic node is attached to its corresponding word2vec embedding vector. The generative process is conditioned on this graph. The generated output is also a graph with the same structure as the input's, except that attached to each output node is a generated visual feature.

V. Results

GTA5 → Cityscapes

Method	Approach	mIoU
Adapt-SegMap [1]	Adv	42.4
Ours (MinEnt)	Ent	42.3
Ours (MinEnt+ER)	Ent	43.1
Ours (AdvEnt)	Adv	43.8
Ours (AdvEnt+MinEnt)	Adv+Ent	**45.5**

SYNTHIA → Cityscapes

Method	Approach	mIoU
Adapt-SegMap [1]	Adv	46.7
Ours (MinEnt)	Ent	44.2
Ours (AdvEnt)	Adv	47.6
Ours (AdvEnt+MinEnt)	Adv+Ent	**48.0**

Table 1: **Segmentation performance in mIoU with ResNet-101 based model and Deeplab-V2 [2] as the segmentation framework.**

Cityscapes → Cityscapes Foggy

Method	Approach	mIoU
SSD-300	-	14.7
Ours (MinEnt)	Ent	16.9
Ours (AdvEnt)	Adv	**26.2**

Table 2: **Object detection performance on Cityscapes Foggy.**