ECAP: EXTENSIVE CUT-AND-PASTE AUGMENTATION FOR UNSUPERVISED DOMAIN ADAPTIVE SEMANTIC SEGMENTATION

SUPPLEMENTARY MATERIAL

Erik Brorsson[⋆†] *Knut Åkesson*[†] *Lennart Svensson*[†] *Kristofer Bengtsson*[⋆]

[⋆] Global Trucks Operations, Volvo Group, Gothenburg, Sweden
[†]Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden

In this supplementary material, we analyze the predictions of ECAP qualitatively, motivate the introduction of the variant of MIC denoted as MIC†, and provide an extended analysis of our method and its limitations.

1. QUALITATIVE COMPARISON WITH STATE-OF-THE-ART

In this section, we compare the predictions of MIC+ECAP with those of MIC qualitatively. Specifically, Figure 1 shows the predictions of the two methods on one image from the (target domain) validation set for each of the evaluated benchmarks. Notably, in row 1 and 2, MIC+ECAP predicts more accurate masks for the classes wall and bus on Synthia→Cityscapes and GTA→Cityscapes respectively. Row 3 and 4 further illustrates failure cases of MIC+ECAP on Cityscapes→DarkZurich and Cityscapes→ACDC respectively. Specifically, in row 3, MIC+ECAP misclassifies the sidewalk as road, and in row 4, MIC+ECAP misclassifies the sky as road. These observations qualitatively explain the large drops in IoU for the classes *sidewalk* on Cityscapes→DarkZurich and *sky* and *road* on Cityscapes→ACDC that were presented in the main paper.

2. MOTIVATION OF MIC†

While DAFormer, HRDA and MIC refrain from training with pseudo-labels on the regions of the image corresponding to the ego-vehicle hood and the image boarders on GTA→Cityscapes and Synthia→Cityscapes, we find it beneficial to train with pseudo-labels in the entire image on Synthia→Cityscapes. As shown in the main paper, letting MIC train on pseudo-labels in the entire image (denoted by MIC†) increases the performance by 0.9 mIoU on Synthia→Cityscapes. In this section we provide an analysis of this phenomenon.

The first row in Figure 2 shows the predictions of MIC and MIC† on Synthia→Cityscapes, where MIC refrains from training on pseudo-labels in the mentioned regions (following the implementation of MIC) and MIC† instead trains on the entire target image. It can be seen that MIC makes ambiguous predictions while MIC† typically predicts the class *road* in the region corresponding to the ego-vehicle hood. Since this region is ignored during evaluation on the Cityscapes benchmark, this may seem like an insignificant detail. However, the ambiguous predictions of MIC are more prone to spilling over from the ego-vehicle hood to the road ahead. Furthermore, the ambiguous predictions in this region may correspond to rare classes such as bus, in which case these predictions may have a significant impact on the resulting mIoU score. By training on the pseudo-labels in this region, the predictions become more stable and tend not to spill over to the road ahead.

Although training on the whole image is beneficial for Synthia→Cityscapes, it is not for GTA→Cityscapes. The second row in 2 shows predictions of MIC and MIC† on GTA→Cityscapes. Also in this case, the predictions of MIC† are less sporadic than those of MIC in the region of the ego-vehicle hood. However, the predictions of MIC are not as prone to spilling over from the ego-vehicle hood when training on GTA→Cityscapes as they are when training on Synthia→Cityscapes. Therefore, it is not necessary to train on the pseudo-labels in this region for the GTA→Cityscapes benchmark. In fact, MIC† achieves an average mIoU score of 75.26 over three random seeds, which is inferior to the score of 75.9 achieved by MIC on GTA→Cityscapes. Apparently, it is better not to train on this region on GTA→Cityscapes, which probably is the reason why MIC adopts this strategy.

We hypothesize that training on pseudo-labels on the ego-vehicle hood is non-informative for the actual evaluation task, making it beneficial to ignore this region, unless this leads to unexpected problems as for Synthia→Cityscapes. Furthermore, we believe that the predictions on the ego-vehicle hood are more prone to spilling over in the case of Synthia→Cityscapes since there is typically no ego-vehicle hood in the Synthia dataset. Additionally, due to the perspective of the camera in Synthia, virtually any object class can appear in this region of the image. In the GTA dataset on the other hand, there is an ego-vehicle hood and typically a road segment around the ego-vehicle hood. This makes training on GTA more prone to predicting a sharp edge between ego-vehicle hood and the surrounding road segment, while training on Synthia results in more ambiguous predictions in this region. In the Dark Zurich and ACDC datasets, there exists no ego-vehicle hood in the test images, and the issue is avoided completely.

3. ECAP EXTENDED ANALYSIS

To gain a better understanding of ECAP, we study the memory bank in detail in this section. Figure 3 shows the five most confident samples, along with associated pseudo-labels, of each of the classes traffic sign, rider and bus in the memory bank of the median run of MIC†+ECAP on Synthia→Cityscapes presented in the main paper. It can be noted that the instances of the respective classes often are clearly visible in the images and are relatively close to the camera, which presumably is the reason why these samples generally are associated with high-quality pseudo-labels. It should also be noted that the images of the memory bank are not full-sized images. The reason for this is that they originate from the sampled target images in every iteration, which are cropped to size 1024×1024. Addition-

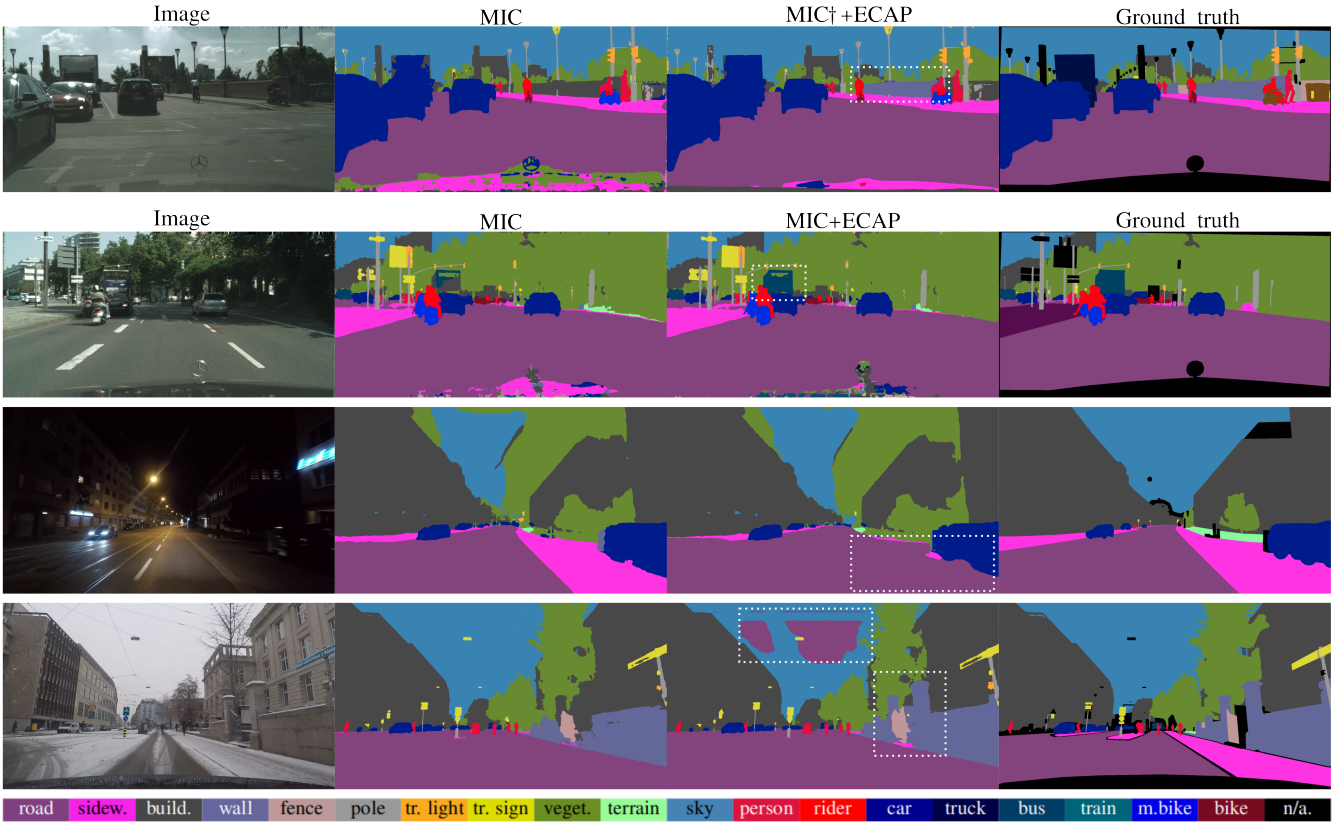

Fig. 1. Qualitative comparison of MIC and MIC†+ECAP on Synthia→Cityscapes (row 1), as well as MIC and MIC+ECAP on GTA→Cityscapes (row 2), Cityscapes→DarkZurich (row 3), and Cityscapes→ACDC (row 4).

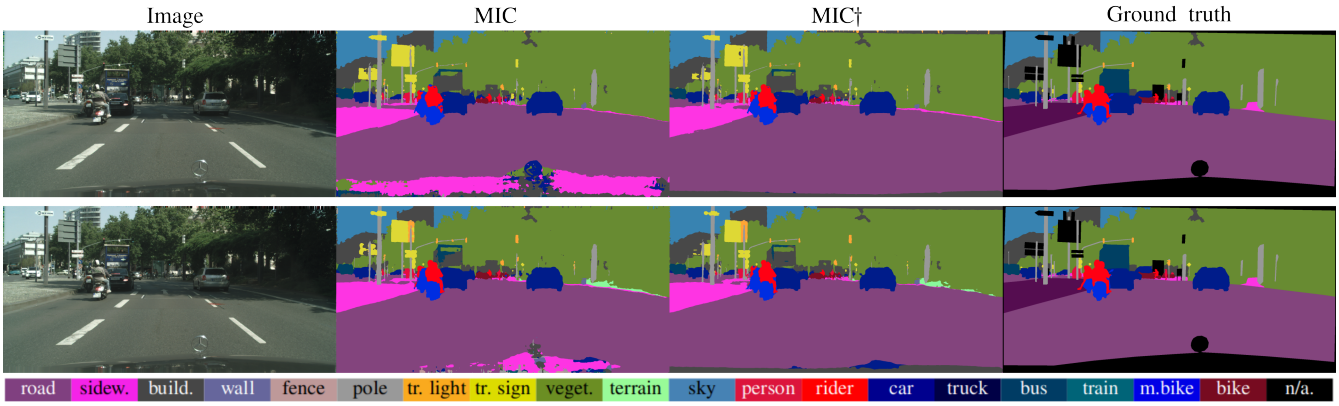

Fig. 2. Predictions on Cityscapes validation images following training on Synthia→Cityscapes (row 1) and GTA→Cityscapes (row 2).

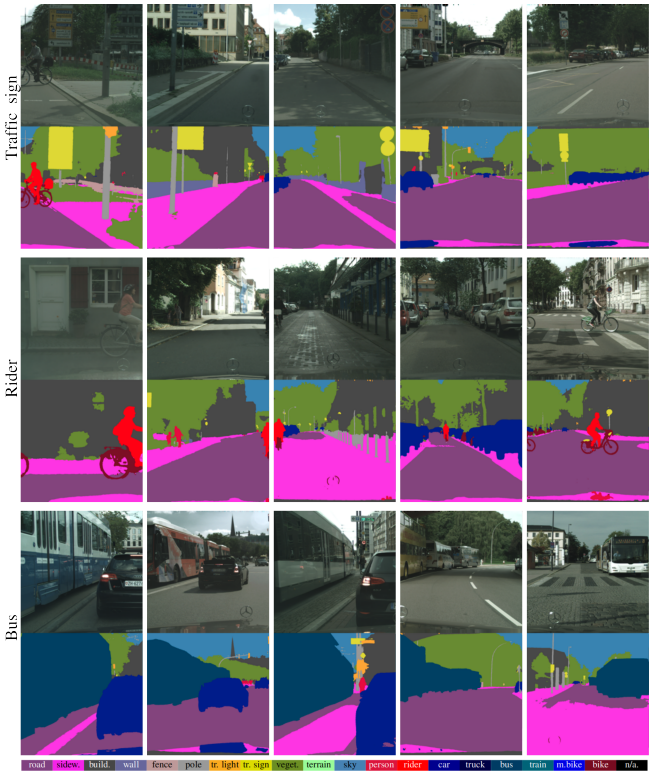

Fig. 3. The five most confident samples of the classes *traffic sign* (row 1), *rider* (row 2), and *bus* (row 3) in the ECAP memory bank. The samples are present in the memory bank at the end of training in the MIC+ECAP run on Synthia→Cityscapes with median performance.

ally, since the class *train* is not present in Synthia, it is reasonable that these are misclassified as *bus* on Synthia→Cityscapes, which explains why some trains are present in the third row of Figure 3.

Although the memory bank generally provides high quality pseudo-labels, which is a result from the positive correlation between accuracy and confidence, it also inevitably include some erroneous pseudo-labels. In Figure 4, two samples from the memory bank of class *rider* of the MIC+ECAP run on GTA with median performance are shown. These two samples where present in the memory bank in the end of the training and both display a high confidence of the class rider (≈ 0.97) although both are misclassified examples. This highlights a problem with using predicted confidence to identify accurate predictions, namely that even misclassified examples may display a high confidence. Intuitively, this constitutes a potential risk of ECAP as erroneous pseudo-labels may be used excessively in ECAP augmentation.

Figure 5 shows three training samples that have been generated through ECAP augmentation. The samples consist of a mix between the source and target images sampled in the current iteration as well as multiple samples from the memory bank. A few things are worth pointing out. First, many classes are present in the images since multiple classes are cut-and-pasted from the memory bank. This may facilitate learning different classes in the UDA setting and counteract the problem of self-training being dominated by pseudo-labels of certain easy-to-adapt classes. Second, when cut-and-pasting classes from the memory bank, they end up in a new context and the resulting images are typically unrealistic. While this could have the benefit of better learning to detect classes in unusual contexts, it may also be a drawback since it hampers learning of context information and certain prior knowledge.

Fig. 4. Two images (row 1) in the memory bank of class *rider* that has been assigned inaccurate pseudo-labels (row 2) during the MIC+ECAP run on GTA→Cityscapes with median performance.

Fig. 5. Augmented training examples of MIC†+ECAP run on Synthia→Cityscapes with median performance.

4. LIMITATIONS

The results of the main paper indicate that ECAP is not beneficial for day-to-nighttime (Cityscapes→DarkZurich) or clear-to-adverse-weather (Cityscapes→ACDC) unsupervised domain adaptation. As illustrated in Figure 1, ECAP struggles with, for example, sidewalk, road and sky, while MIC handles these classes better. This does not, however, apply to the benchmarks on synthetic-to-real domain adaptation. We believe that one reason for this is that the appearance of the road and sidewalk becomes more similar during nighttime. Similarly, a clouded sky has similar appearance as a snow-covered road, making it more difficult to distinguish them from each other than in clear weather conditions. Therefore, learning context information becomes increasingly important to attain good segmentation results on Cityscapes→DarkZurich and Cityscapes→ACDC. On the other hand, context information is of little importance in the augmented training examples of ECAP, which encourages the model to instead focus on the appearance of classes during training. In this sense, ECAP may hamper the learning of context information, and as a result, struggles with images in which context information and prior knowledge of the scene layout is pivotal for making an accurate segmentation.