Contrastive Model Adaptation for Cross-Condition Robustness in Semantic Segmentation

David Bruggemann Christos Sakaridis Tim Brödermann Luc Van Gool
ETH Zurich, Switzerland
{brdavid, csakarid, timbr, vangool}@vision.ee.ethz.ch

Abstract

Standard unsupervised domain adaptation methods adapt models from a source to a target domain using labeled source data and unlabeled target data jointly. In model adaptation, on the other hand, access to the labeled source data is prohibited, i.e., only the source-trained model and unlabeled target data are available. We investigate normal-to-adverse condition model adaptation for semantic segmentation, whereby image-level correspondences are available in the target domain. The target set consists of unlabeled pairs of adverse- and normal-condition street images taken at GNSS-matched locations. Our method—CMA—leverages such image pairs to learn condition-invariant features via contrastive learning. In particular, CMA encourages features in the embedding space to be grouped according to their condition-invariant semantic content and not according to the condition under which respective inputs are captured. To obtain accurate cross-domain semantic correspondences, we warp the normal image to the viewpoint of the adverse image and leverage warp-confidence scores to create robust, aggregated features. With this approach, we achieve state-of-the-art semantic segmentation performance for model adaptation on several normal-to-adverse adaptation benchmarks, such as ACDC and Dark Zurich. We also evaluate CMA on a newly procured adverse-condition generalization benchmark and report favorable results compared to standard unsupervised domain adaptation methods, despite the comparative handicap of CMA due to source data inaccessibility. Code is available at https://github.com/brdav/cma.

1. Introduction

Adverse visual conditions, such as fog, heavy rain, or snowfall, represent a challenge for autonomous systems expected to navigate "in the wild". To achieve full autonomy, systems require perception algorithms that perform robustly in every condition. However, due to their infrequent occurrence, inclement weather conditions are often underrepresented in common, finely annotated outdoor datasets (*e.g.*, BDD100K [44] or Mapillary Vistas [28]). As a result, state-of-the-art recognition methods are biased towards "normal" visual conditions (*i.e.*, daytime and clear weather), which causes them to fail for edge cases. Furthermore—in particular for detailed, pixel-level tasks like semantic segmentation—high-quality annotations for adverse-condition images are difficult and expensive to obtain. In fact, they require specialized annotation protocols due to ambiguities arising from aleatoric uncertainty [33]. To bypass these issues, researchers have investigated unsupervised domain adaptation (UDA) from normal to adverse conditions as an alternative to full supervision [6, 11, 14, 30, 41], where a model is jointly trained on labeled source-domain data and unlabeled target-domain data.

This paper instead targets the more general problem of source-free domain adaptation—also known as *model adaptation*—for semantic segmentation. In model adapta-

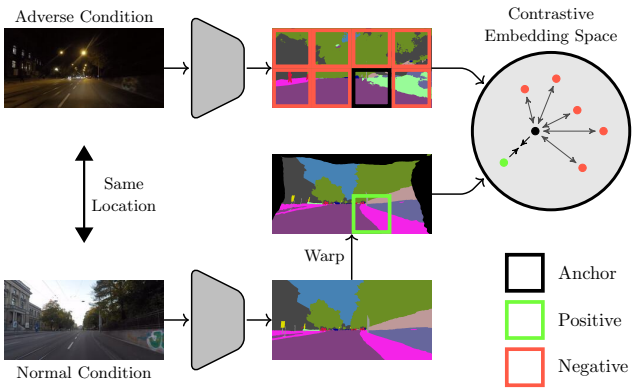

Figure 1. CMA exploits image-level correspondences to learn condition-invariant features. Two images of the same location (but captured under different visual conditions) are encoded, and the normal-condition image features are warped to get spatially aligned with the adverse-condition image features. Our contrastive loss then creates an embedding space where patches of adverse features (black) are *closer to their corresponding normal patches* (green) than to other adverse patches (red).

tion, only (i) the model pre-trained on source images and (ii) unlabeled target images are available. This pertains to many real-world use cases when the labeled source data is proprietary or inaccessible due to privacy concerns. The complete absence of fine ground-truth annotations represents a significant challenge, as the model can easily drift and unlearn important concepts during adaptation. To bolster the adaptation process, we leverage another form of weak supervision, which is far easier and cheaper to collect than pixel-wise semantic annotations. In particular, multiple recent driving datasets—such as RobotCar [26], ACDC [33] and Boreas [3]—traverse the same route several times under varying weather conditions, and provide GNSS-matched frames. Each adverse-condition target image can thus be paired with a corresponding *reference* image depicting roughly the same scene under normal conditions. While also unlabeled, the reference images bridge the domain gap between the source and target domain by overlapping both with the source domain in terms of visual condition and with the target domain in terms of geography and sensor characteristics.

Our proposed method, named Contrastive Model Adaptation (CMA), leverages the reference predictions through a unified embedding space. Assuming the reference and target images are sufficiently aligned, co-located features should be similar between the two—neglecting dynamic objects and slight shifts in static content (*e.g.*, missing leaves on a tree). Accordingly, we posit that for a given target feature, its reference feature at the same spatial location should be *closer in the embedding space than most other target features*. An embedding space fulfilling this assumption would effectively eliminate condition-specific information, but simultaneously preserve semantic content. We aim to create such an embedding space through contrastive learning, where dense spatial embeddings of the target image serve as *anchors* (black patch in Fig. 1). Each anchor is pulled towards a single *positive*, *i.e.*, the embedding of the reference image corresponding to the same location (green patch in Fig. 1). Since the pre-trained source model is expected to produce semantic features of higher quality on the reference images than on the target images (for a qualitative comparison see Sec. E of the suppl. material), this clustering step helps to correct less reliable anchor semantics. Conversely, the anchor is pushed apart from the *negatives*, which are simply target embeddings at other spatial locations (or from other target images, red patches in Fig. 1), to counteract mode collapse. Through spatial alignment of the reference and target images and custom, confidence-modulated feature aggregation, we create robust embeddings for optimization with our cross-domain contrastive loss.

CMA yields state-of-the-art results for model adaptation on several normal-to-adverse semantic segmentation benchmarks. It even outperforms recent standard UDA methods on these benchmarks, despite its data handicap compared to the latter methods. Attesting to our successful cross-domain embedding alignment, CMA delivers exceptionally *robust* results, as shown by evaluations on the newly compiled Adverse-Condition Generalization (ACG) benchmark.

2. Related Work

Model adaptation or source-free domain adaptation lifts the assumption of standard unsupervised domain adaptation that data from the source domain are accessible at adaptation time, which renders the former task more challenging. In the absence of labeled data for providing supervision, model adaptation methods for semantic segmentation typically rely on loss-based constraints on the features and/or outputs of the network, which are computed for target images. Such losses promote robustness of the network to missing features [25, 35] or to perturbations of the inputs and features [25], or aim at minimizing entropy in the network outputs [35, 38]. Some works focus primarily on the normalization layers of the involved networks, encouraging consistency of the statistics of these layers across the initial source-trained model and the final adapted model [24] and optimizing channel-level affine transformations of the normalized features with respect to output entropy [38]. Best "upstream" practices for training source models for model adaptation are explored in [19]. One previous work on model adaptation in semantic segmentation [17] has considered contrastive learning similar to ours. However, that work contrasts features within individual images across the model adaptation cycle, whereas we contrast features across domains due to multiple corresponding views.

Contrastive learning is a fundamental unsupervised framework based on instance discrimination for extracting informative representations. Seminal works on contrastive learning include [29], which introduced the widely used InfoNCE loss, and [5], which proposed a simple framework for visual contrastive learning. While such fundamental works focus on the setting of unsupervised pre-training for image classification, there has been a body of recent literature examining contrastive learning for domain adaptive semantic segmentation, by primarily leveraging *class-wise* contrast. [18, 21, 42] employ partially dense contrast between classes using pixel features as anchors and class-level prototype vectors as positives and negatives. Along similar lines, [22, 42] implement partially dense contrast between classes using pixel features as anchors and estimated class-level *distributions* as positives and negatives, while [27] use class prototypes both as anchors and as positives/negatives. These approaches are prone to false positive/negative samples which contaminate the contrastive loss due to potential errors in the target-domain pseudo-labels, which are used both to determine the anchors and to compute the

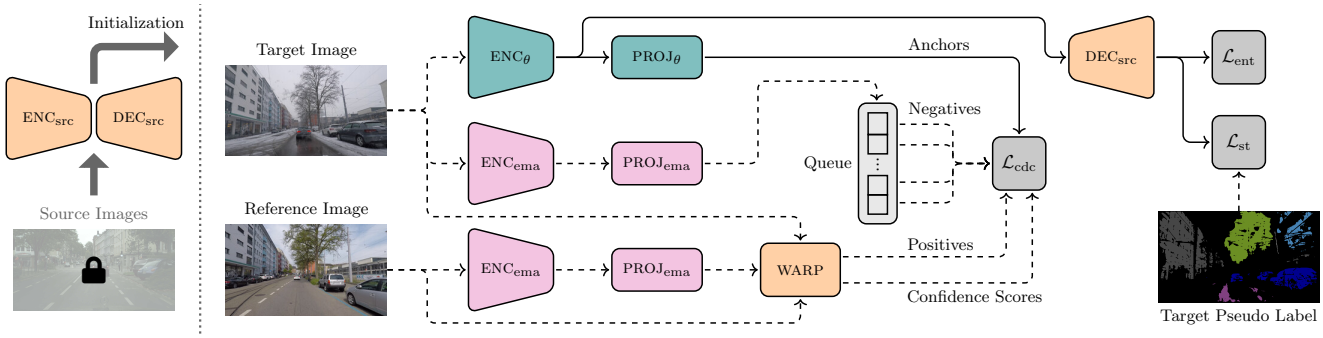

Figure 2. Overview of the CMA architecture. (Left) The segmentation network (ENC and DEC) is initialized with weights pre-trained on the source domain, however, access to the source data itself is prohibited. (Right) The model is trained with pairs of target and reference images. In addition to standard entropy minimization ($\mathcal{L}_{\text{ent}}$) and self-training ($\mathcal{L}_{\text{st}}$), we propose a cross-domain contrastive (CDC) loss ($\mathcal{L}_{\text{cdc}}$) to align features across domains. Dense embeddings are extracted from both images through projection heads PROJ. The CDC loss pulls the target anchors close to corresponding reference embeddings (positives), while pushing them apart from other target embeddings stored in a queue (negatives). Crucially, the positives are obtained through spatial alignment (WARP) and robust feature aggregation (see Sec. 3.3). Frozen modules are in orange, trainable modules in blue, and exponential moving average modules in pink. Gradients are only backpropagated through solid arrows.

class prototypes that serve as positives and negatives. By contrast, we propose a novel *domain-wise* contrast, which does not rely on class (pseudo-)labels to construct the contrastive loss and avoids the above issue. Moreover, our cross-domain contrastive loss is *fully* dense, in the sense that both the anchors and the positives/negatives are densely extracted in patches from the input images. In that regard, our method is closely related to [39], which also uses correspondences between matching views to contrast dense, locally aggregated features—albeit for unsupervised pre-training. Whereas correspondences between views are heuristically determined in [39] through the similarities of backbone features, we have a dedicated, externally pre-trained module for computing correspondences via dense matching. Besides, contrary to [39], we explicitly combat false positive pairs owing to errors in the correspondences by incorporating confidence-guided feature aggregation: when grouping features locally, we weigh them by the confidence associated with the respective correspondences. In addition, low-confidence anchors and positives are filtered out.

Cross-condition image-level correspondences are provided by several driving datasets [3, 26, 33, 34] and can be utilized for normal-to-adverse condition domain adaptive semantic segmentation: [20] find sparse, pixel-level correspondences and apply a consistency loss on semantic predictions. Other works establish dense correspondences through two-view geometry [34] or end-to-end dense matching [2, 40]. The dense correspondences are subsequently used to enforce prediction consistency [40] or to fuse cross-condition predictions [2, 34]. While we also use end-to-end dense matching to find correspondences, we instead use contrastive learning to create a condition-invariant, discriminative embedding space.

3. Contrastive Model Adaptation (CMA)

The aim of CMA is to fine-tune a given pre-trained semantic segmentation model on unlabeled, adverse-condition images. In contrast to standard unsupervised domain adaptation, access to the labeled source data—with which the model was originally trained—is assumed to be prohibited, *e.g.*, due to privacy concerns. In the experimental setup for CMA, we are specifically given (i) a semantic segmentation model, pre-trained on a source dataset recorded under normal conditions, *e.g.*, Cityscapes [9], and (ii) a set of unlabeled adverse-condition target samples, *e.g.*, from ACDC [33], to whose population we aim to adapt the model. Moreover, (iii) an unlabeled reference image is available for each target image, which depicts approximately the same scene as the adverse-condition target image, albeit under normal visual conditions.

3.1. Architecture Overview

Fig. 2 shows our model adaptation architecture. The pre-trained source-model weights are used to initialize the encoder ENC and decoder DEC. Since CMA focuses on generating condition-invariant, discriminative encoder features, decoder weights are kept frozen to preserve source-domain knowledge. The encoder ENC_θ is adapted by three loss functions: entropy minimization ($\mathcal{L}_{\text{ent}}$, Sec. 3.4), self-training ($\mathcal{L}_{\text{st}}$, Sec. 3.4), and the proposed cross-domain contrastive (CDC) loss ($\mathcal{L}_{\text{cdc}}$, Sec. 3.3). We describe the individual modules in detail in the next sections.

3.2. Spatial Alignment

Although the reference-target image pairs depict the same scene, their viewpoint can differ substantially as they are only GNSS-matched. The resulting correspondence dis-

crepancies can have a detrimental effect when working on pixel-accurate tasks such as semantic segmentation. We therefore densely warp the reference image into the viewpoint of the target image to obtain more accurate matches.

In particular, we choose the existing dense matching network UAWarpC [2] (WARP module in Fig. 2), since it provides a confidence score for each displaced pixel, which is an important component of our downstream CDC loss (see Sec. 3.3). UAWarpC was independently trained in a self-supervised way on MegaDepth [23], a large-scale collection of multi-view internet photos. When training CMA, UAWarpC is frozen and warps the reference image features. We tried to either warp the image before feeding it into the encoder ENC_θ or warp the dense feature maps and found that the latter works better.

3.3. Cross-Domain Contrastive Loss

The cross-domain contrastive (CDC) loss is the central component of CMA. It incentivizes the encoder to learn features that discriminatively reflect the semantics, but are invariant to the visual condition. To this end, spatial target image features represent *anchors*, which are pulled towards spatially corresponding reference image features—the *positives*. The anchors and positives are assumed to represent similar semantics in distinct visual conditions. Simultaneously, the anchors are contrasted to other target image features—the *negatives*—to prevent mode collapse. Although the CDC loss could in principle be applied to encoder features directly, we first project the dense features to a dedicated 128-dimensional embedding space, as per standard practice [5]. The projection head PROJ consists of two 1×1 convolutions with a ReLU non-linearity in-between. As shown in Fig. 2, the embeddings of the trainable model $\text{PROJ}_\theta \circ \text{ENC}_\theta$ serve as anchors, while—as proposed in [13]—positives and negatives are obtained by an exponential moving average model $\text{PROJ}_{\text{ema}} \circ \text{ENC}_{\text{ema}}$ to improve their consistency. Furthermore, we use a *queue* to accumulate negatives [13]. This enables the use of a large number of negatives during instance discrimination, which encourages the learning of meaningful representations by making the discrimination more challenging. Finally, the positives are spatially warped to align them with the anchors, as detailed in Sec. 3.2.

Despite the warping, anchors and positives might not always depict the same semantic content, because (i) the warping described in Sec. 3.2 is not exact, *e.g.*, street poles and other small or thin objects are often not perfectly aligned, and (ii) dynamic objects such as cars and pedestrians differ between the reference and target images. Such false positives introduce excessive noise into the contrastive loss, which worsens generalization [36]. To mitigate this issue, we use two strategies: patch-level grouping and confidence modulation.

Patch-Level Grouping. Inspired by [39], we average-pool spatial embeddings across square patches for anchors, positives, and negatives. However, differently from [39], we directly pool the embeddings, instead of applying pooling earlier in the model. Due to the averaging, larger patches are more forgiving towards small errors in the warping, as well as small semantic discrepancies due to dynamic objects. On the other hand, very large patch sizes do not promote the learning of local discriminative features. The grid size dictating the employed grouping is thus a key hyperparameter; we choose a 7×7 grid for square full-height crops of street images. A desirable side-effect of patch-level grouping is a significant reduction in memory and computational cost.

Confidence Modulation. The confidence scores provided by the WARP module can be leveraged to refine and filter patch embeddings for anchors and positives. We propose to use *weighted* average pooling to create patch embeddings, where each pixel is weighted by its confidence score. Accordingly, low-confidence correspondences within a single patch (*e.g.*, resulting from pixels of a dynamic object) contribute less to the aggregated patch embedding. In addition, patches with an average confidence of below 0.2 are deemed false positives and discarded altogether.

To formalize those steps, we define the set $\mathcal{N}_i$ to comprise all indices of pixels in the pooling receptive field of patch i. $\mathbf{z}_j^a, \mathbf{z}_j^p \in \mathbb{R}^{128}$ are the PROJ head outputs at pixel index j for anchor and (warped) positive respectively. $c_j \in [0, 1]$ is the corresponding warp confidence score (which is identical for anchor and positive). Importantly, $c_j = 0$ for pixels without a valid correspondence. Unnormalized patch embeddings for anchors $\tilde{\mathbf{a}}_i$ and positives $\tilde{\mathbf{p}}_i$ are computed through the weighted sums

$$\tilde{\mathbf{a}}_i = \sum_{j \in \mathcal{N}_i} c_j \mathbf{z}_j^a, \quad \tilde{\mathbf{p}}_i = \sum_{j \in \mathcal{N}_i} c_j \mathbf{z}_j^p. \tag{1}$$

The embeddings are subsequently L2-normalized to obtain $\mathbf{a}_i$ and $\mathbf{p}_i$. Meanwhile, negative embeddings $\mathbf{n}_j$ are obtained through simple average pooling, followed by L2-normalization.

To create an embedding space where, for each patch i, the anchor $\mathbf{a}_i$ is pulled towards the positive $\mathbf{p}_i$ and pushed away from M negatives $\mathbf{n}_j$ (sourced from the queue of length M), we use the InfoNCE [29] loss:

$$\mathcal{L}_{\text{cdc},i} = -\log \frac{\exp\left(\mathbf{a}_i^T \mathbf{p}_i / \tau\right)}{\exp\left(\mathbf{a}_i^T \mathbf{p}_i / \tau\right) + \sum_{j=1}^{M} \exp\left(\mathbf{a}_i^T \mathbf{n}_j / \tau\right)}. \tag{2}$$

τ is a temperature hyperparameter that scales the sensitivity of the loss function. Finally, when aggregating the patch-wise losses, low-confidence patches are discarded:

$$\mathcal{L}_{\text{cdc}} = \frac{\sum_i \mathcal{L}_{\text{cdc},i} [\bar{c}_i \geq 0.2]}{\sum_i [\bar{c}_i \geq 0.2]}, \tag{3}$$

where $[\cdot]$ denotes the Iverson bracket and $\bar{c}_i$ is the average-pooled confidence of patch i:

$$\bar{c}_i = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} c_j. \tag{4}$$

The effect of patch-level grouping and confidence modulation is illustrated in Fig. 3. In the two center columns, each pixel is whited out according to its confidence. Within each patch, the "confident", visible pixels are subsequently aggregated. Orange patches are eliminated due to their overall low confidence. Notice that the remaining features correspond well between the two images, despite the initial differences in viewpoint, dynamic objects, occlusions, *etc*. In fact, the warping confidence is rather conservative.

3.4. Complete Training Loss

Besides the proposed CDC loss (Sec. 3.3) we employ two commonly used loss functions.

Self-Training. We follow the pseudo-labeling strategy of CBST [46] to create class-balanced pseudo-labels from confident predictions. The pseudo-labels are created once before training by the source model, to inject regularization to the source. We retain the most confident 20% of pixels and all other pixels are ignored. During model adaptation, we use a cross-entropy loss $\mathcal{L}_{\text{st}}$ for self-training.

Entropy Minimization. We use entropy minimization as a regularizer during training. $\mathcal{L}_{\text{ent}}$ is the mean normalized entropy of the predicted class probabilities over all pixels.

Finally, the complete training loss consists of a weighted sum of the three losses: $\mathcal{L}_{\text{tot}} = \mathcal{L}_{\text{st}} + \lambda_{\text{ent}}\mathcal{L}_{\text{ent}} + \lambda_{\text{cdc}}\mathcal{L}_{\text{cdc}}$. λ_{ent} and λ_{cdc} are hyperparameters that determine the relative importance given to the individual losses.

4. Experiments

In this section, we present extensive experimental results, comparing CMA to state-of-the-art model adaptation (Sec. 4.2) and standard unsupervised domain adaptation (Sec. 4.3) methods. Moreover, we analyze generalization performance (Sec. 4.4) and ablate various components (Sec. 4.5) of the method.

4.1. Setup

Datasets. We use Cityscapes [9] as a source dataset, and various target datasets: ACDC [33] (train/val/test: 1600/406/2000 images), Dark Zurich [34] (train/val/test: 2416/50/151 images), RobotCar Correspondence [20, 26] (train/val/test: 6511/27/27 images), and CMU Correspondence [1, 20] (train/val/test: 28766/25/33 images). All target datasets contain corresponding pairs of normal- and adverse-condition street images in the training set. Adverse conditions vary between datasets: ACDC contains images in fog, night, rain, and snow; Dark Zurich consists of night images; RobotCar and CMU contain variable conditions as well as seasonal changes. Unless otherwise stated, we report performance on the test sets for all datasets.

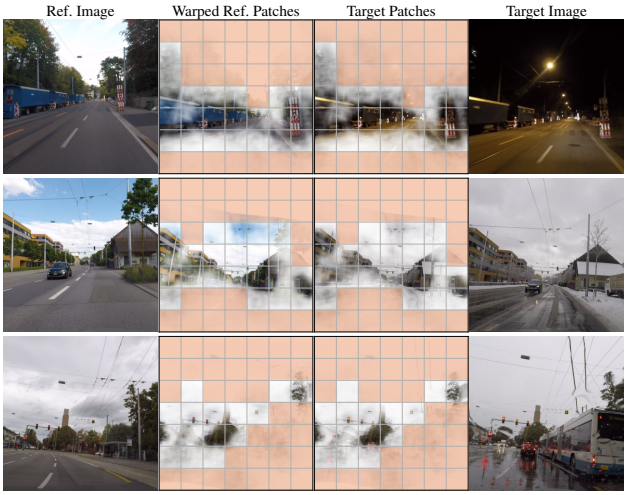

Figure 3. Visualization of matched patches. The second column shows the warped reference image, with low-confident regions whited out. The drawn grid illustrates the patch-level grouping. Within each patch, features are aggregated proportionally to their confidence. Patches with low average confidence are discarded altogether, as shown by orange shading.

ACG Benchmark. To assess the generalization performance of trained adverse-condition models, we present an adverse-condition generalization (ACG) benchmark consisting of diverse samples from several public street-scene segmentation datasets. We inspected all labeled images of WildDash2 [45], BDD100K [44], Foggy Zurich [10, 31], and Foggy Driving [32], selected adverse-condition samples (featuring fog, night, rain, snow, or a combination of those), and manually verified the quality of each corresponding ground truth. Samples with evident ground-truth inaccuracies were eliminated. For Foggy Zurich and Foggy Driving, we also meticulously cross-checked every image for overlap with the ACDC dataset, as all three datasets were recorded in the same region. We refer to Sec. B in the supplementary material for more details about the sample selection process and the resulting dataset statistics. ACG consists of a highly *diverse* set of 922 adverse-condition driving images from various geographical regions in Europe and North America. We additionally divide ACG into 121 fog, 225 rain, 276 snow, and 300 night images, to allow for condition-wise evaluation. Importantly, ACG-night also includes adverse-weather images, *e.g.*, snowy nighttime scenes. The curated list of ACG image filenames is publicly available via https://github.com/brdav/cma.

Architectures and Hyperparameters. We conduct the bulk of our experiments using the state-of-the-art Seg-

Table 1. Comparison to the state of the art in model adaptation on Cityscapes→ACDC, with reported performance on the ACDC test set.

Method		ACDC IoU↑																			
		road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	**mean**
Source model	DeepLabv2 [4]	76.6	40.5	56.0	12.0	27.3	35.6	40.2	45.6	69.8	38.2	76.2	21.3	12.4	65.6	25.2	29.2	28.1	15.2	34.6	39.5
TENT [38]	DeepLabv2 [4]	84.1	48.0	56.9	21.1	30.3	43.7	56.3	53.7	69.6	36.7	61.8	55.0	33.0	78.7	40.6	43.0	48.7	30.5	39.3	49.0
HCL [17]	DeepLabv2 [4]	80.5	42.9	57.6	14.7	29.4	40.3	49.0	51.1	72.4	35.6	78.3	39.7	31.8	76.0	35.4	42.7	42.5	25.7	43.0	46.8
URMA [35]	DeepLabv2 [4]	85.4	52.9	62.9	20.4	34.4	39.9	36.7	43.9	74.9	46.9	85.1	27.2	22.4	76.0	40.5	41.5	38.9	20.6	46.2	47.2
URMA + SimT [12]	DeepLabv2 [4]	83.5	52.7	60.7	19.6	33.7	42.0	43.1	47.4	75.0	42.5	85.8	39.8	19.6	76.9	39.6	42.7	41.1	24.0	43.1	48.0
CMA	DeepLabv2 [4]	83.1	52.7	65.4	18.7	30.5	44.5	56.3	53.9	76.7	39.7	79.0	54.2	31.2	76.7	40.2	39.3	47.4	29.8	38.6	50.4
Source model	SegFormer [43]	85.7	51.0	76.6	36.4	37.1	45.2	55.7	57.5	77.7	52.0	84.1	60.3	34.8	82.9	61.6	65.4	73.4	37.9	52.5	59.4
TENT [38]	SegFormer [43]	84.0	51.5	75.4	36.8	37.2	46.2	56.0	57.7	77.9	52.9	81.7	59.9	36.0	82.9	60.8	65.5	73.6	38.3	52.4	59.3
HCL [17]	SegFormer [43]	86.4	53.5	78.5	38.8	38.1	48.0	57.8	58.9	78.1	52.4	85.1	61.7	37.1	83.7	64.1	66.6	74.5	39.1	53.3	60.8
URMA [35]	SegFormer [43]	89.2	60.4	84.3	48.7	42.5	53.8	65.4	63.8	76.3	57.3	85.9	63.4	43.9	85.8	**68.8**	73.2	82.8	46.3	48.4	65.3
URMA + SimT [12]	SegFormer [43]	90.0	65.7	80.6	46.0	41.7	**56.3**	65.2	62.7	75.9	55.6	84.4	66.4	**46.6**	85.4	68.4	72.3	80.0	**46.8**	58.0	65.7
CMA	SegFormer [43]	**94.0**	**75.2**	**88.6**	**50.5**	**45.5**	54.9	**65.7**	**64.2**	**87.1**	**61.3**	**95.2**	**67.0**	45.2	**86.2**	68.6	**76.6**	**83.9**	43.3	**60.5**	**69.1**

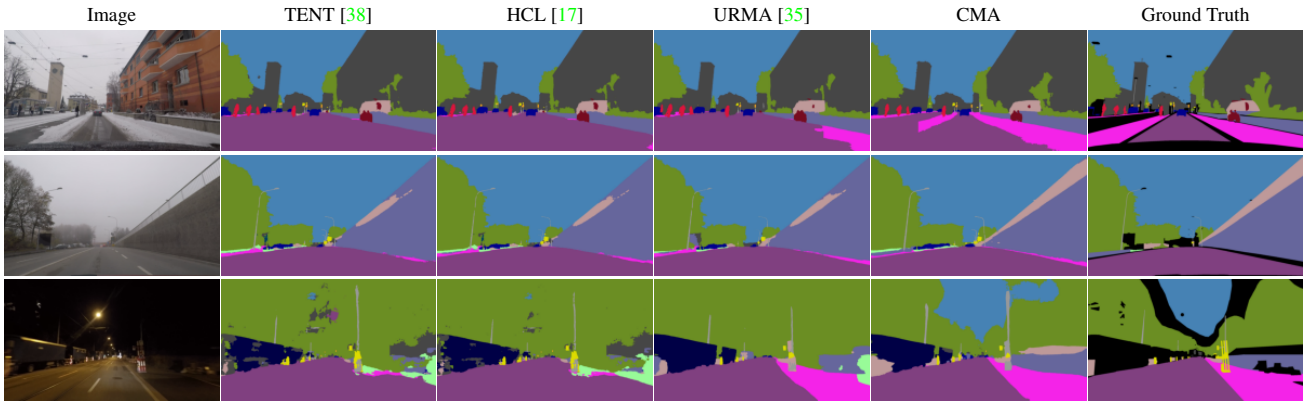

Figure 4. Qualitative segmentation results of SegFormer-based model adaptation methods on ACDC validation images.

Former [43] architecture, however, we also include experiments using DeepLabv2 [4]. For all datasets and architectures, we train CMA for 10k iterations, where for the first 2.5k iterations we stop gradient flow from PROJ_θ back to ENC_θ to "warm up" PROJ_θ. Since PROJ_θ is the only randomly initialized module, we use a 10× learning rate for it w.r.t. ENC_θ. We find that a high momentum of 0.9999 works best for the exponential moving average components, presumably preserving source knowledge better during the adaptation process. All models were trained on a single TITAN RTX GPU. More details on training configurations are in Sec. A of the supplementary material.

4.2. Comparison to Model Adaptation Methods

We benchmark CMA against the state-of-the-art model adaptation methods TENT [38], HCL [17], URMA [35], and SimT [12] on Cityscapes→ACDC and report the ACDC test set scores in Table 1. In this comparison, CMA is the only method using reference images. Using a SegFormer architecture, CMA sets the new state of the art for model adaptation from Cityscapes to ACDC with a mIoU of 69.1%, substantially outperforming all other methods. Note that CMA outperforms competing methods both on static and dynamic classes, even though our contrastive loss does not explicitly target dynamic objects. We hypothesize that the universal performance gain is enabled through the learned invariance to global condition-level variations, *e.g.*, with respect to changes in scene illumination or reflectance. For DeepLabv2, CMA obtains 50.4% mIoU, still outperforming other methods, although in this case, all compared methods bring a substantial improvement over the source model. Due to the large absolute performance difference, we conduct the rest of the experiments in this paper on the SegFormer architecture. We show qualitative segmentation results for the different SegFormer-based methods on three ACDC validation set images in Fig. 4. CMA predicts higher-quality segmentation maps than other methods, *e.g.*, on the snowy sidewalk in the top image, the fence in the middle image, or the wall on the right in the bottom image.

We further evaluate CMA on three other adaptation settings: Cityscapes→Dark Zurich, Cityscapes→RobotCar, and Cityscapes→CMU in Table 2. For all investigated sce-

Table 2. Comparison to the state of the art in model adaptation on Cityscapes→Dark Zurich, Cityscapes→RobotCar, and Cityscapes→CMU. All models use a SegFormer architecture.

Method		mIoU↑		
		Dark Zurich [34]	RobotCar [20, 26]	CMU [1, 20]
Source model	SegFormer [43]	41.7	50.0	80.0
TENT [38]		42.8	50.1	78.9
HCL [17]		42.7	50.1	80.2
URMA [35]		49.3	51.6	82.8
URMA + SimT [12]		50.1	52.4	83.9
CMA		**53.6**	**54.3**	**92.0**

Table 3. Comparison of CMA to state-of-the-art standard UDA methods. Adaptation from Cityscapes as source dataset.

Method	Source-Free	mIoU↑	
		ACDC [33]	Dark Zurich [34]
DAFormer [15]		55.4	53.8
SePiCo [42]		59.1	54.2
HRDA [16]		68.0	**55.9**
CMA	✓	**69.1**	53.6

narios, CMA significantly outperforms other methods.

4.3. Comparison to Standard UDA Methods

Table 3 shows a comparison of CMA to standard unsupervised domain adaptation (UDA) methods, which use the labeled source data during the adaptation process. Standard UDA methods are thus not susceptible to forgetting source knowledge. Despite this handicap, CMA compares favorably to DAFormer [15], SePiCo [42], and HRDA [16] on Cityscapes→ACDC, while only slightly falling behind on the more challenging Cityscapes→Dark Zurich.

4.4. Generalization and Robustness

To evaluate the generalization performance of trained adverse-condition models, we test the Cityscapes→ACDC models on our diverse ACG benchmark. We report the condition-wise mIoU for fog, night, rain, and snow, as well as the overall mIoU, in Table 4. CMA achieves the best generalization performance compared to other model adaptation and UDA methods, with a mIoU of 51.3% for all ACG samples. Interestingly, CMA performs exceptionally well on the most challenging ACG-night split, which also contains combinations of conditions (*e.g.*, night and rain), which are absent from the training set. This corroborates that CMA learns highly *robust* representations through our proposed contrastive cross-domain feature alignment.

4.5. Ablation Study and Further Analysis

All the numbers in this section are from the ACDC validation set. We report the mean performance over 3 repeated runs for each experiment, to reduce variance.

Table 4. ACG benchmark generalization performance of models adapted from Cityscapes to ACDC.

Method	Source-Free	ACG mIoU↑				
		fog	night	rain	snow	all
TENT [38]	✓	52.6	27.7	47.5	41.1	40.0
HCL [17]	✓	54.2	28.3	48.2	42.4	40.8
URMA [35]	✓	54.1	31.0	51.9	45.5	44.4
DAFormer [15]		52.6	21.5	47.5	33.6	40.1
SePiCo [42]		53.9	20.6	46.3	36.1	38.6
HRDA [16]		**60.0**	27.1	56.2	43.3	48.9
CMA	✓	59.7	**40.0**	**59.6**	**52.2**	**51.3**

Ablation Study. Table 5 shows the ablation study for several important CMA components. Row 1 represents CMA without the CDC loss, solely relying on entropy minimization and self-training for adaptation. A comparison of row 1 to the final model in row 7 reveals the large performance increase of 7.1% mIoU owing to the CDC loss. In row 2 the contrastive loss is added, but the embeddings are obtained by global average pooling over the entire image. Nevertheless, this improves performance substantially by 4.6% mIoU. Next, we add patch-level grouping on a 7×7 grid combined with reference image warping in row 5, which brings a 1.2% mIoU improvement over the global embeddings of row 2. Interestingly, using patches without warping decreases the performance, as shown in row 3. This can be explained by the excessive noise introduced to the contrastive loss due to patch misalignment between reference and target. A comparison of rows 4 and 5 reveals that, although warping is responsible for the majority of the performance gain, patch-level grouping brings further improvements by producing more locally discriminative features. In row 6, we show that adding our confidence modulation to patch forming leads to another 1% mIoU increase. Finally, row 7 refers to the complete model, which involves estimating positives and negatives through an exponential moving average model, instead of simply using the source model and a random projection head.

Effect of Reference Images. Compared to other model adaptation methods, CMA uses extra reference images through the CDC loss. We, therefore, train two baseline models—CMA without the CDC loss and URMA—by including the reference images in two ways: (i) mixing the reference and target images randomly, and adapting to the combination of both, and (ii) adapting in a curriculum, *i.e.*, first adapting from the source to the reference domain, and then using the same method to continue adapting from the reference to the target domain. Table 6 shows that our contrastive approach clearly outperforms both of these simple strategies. The performance benefits of our approach are thus not attainable by naively introducing reference images.

Alternative Contrastive Loss. In addition to the mech-

Table 5. Ablation study on the ACDC validation set, reporting IoU. "EMA": exponential moving average model for positives and negatives.

	$\mathcal{L}_{cdc}$	patch-level grouping	warp	confidence-modulation	EMA	road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	**mean**
1						90.1	64.6	79.0	39.0	33.4	54.0	73.6	56.0	71.0	35.9	82.1	61.8	37.2	84.9	71.9	69.7	55.1	46.9	35.7	60.1
2	✓					92.2	69.1	82.5	45.9	40.8	56.0	73.4	58.8	78.2	40.2	87.6	64.0	41.4	85.4	79.5	72.8	65.2	48.5	46.9	64.7
3	✓	✓				92.3	68.9	82.8	44.2	39.0	54.1	73.1	58.2	81.6	39.8	91.0	63.8	41.9	83.0	74.0	72.5	64.9	45.0	37.8	63.6
4	✓		✓			92.5	70.0	82.7	46.0	40.9	56.8	73.9	59.5	77.7	40.2	86.9	64.4	41.8	85.5	80.2	72.9	66.3	48.2	48.8	65.0
5	✓	✓	✓			93.3	72.0	84.7	47.4	41.2	57.8	75.1	60.7	83.1	42.8	92.4	64.4	40.4	84.7	77.9	73.9	64.7	48.8	47.7	65.9
6	✓	✓	✓	✓		93.3	72.3	84.9	47.7	41.4	59.1	75.9	61.3	84.1	44.2	93.3	65.9	40.8	85.2	81.6	74.0	65.8	50.1	49.5	66.9
7	✓	✓	✓	✓	✓	94.7	75.6	85.4	48.0	43.3	59.4	75.9	61.3	84.8	44.3	93.6	65.8	39.6	85.7	81.6	73.8	69.0	48.8	47.1	67.2

Table 6. Comparison of different strategies for involving reference images in model adaptation, reporting ACDC validation scores.

Model	Target	Reference	Type	mIoU ↑
Source model [43]			-	56.6
URMA [35]	✓		-	63.2
URMA [35]	✓	✓	mixed	62.9
URMA [35]	✓	✓	curriculum	64.1
CMA w/o CDC loss	✓		-	60.1
CMA w/o CDC loss	✓	✓	mixed	60.3
CMA w/o CDC loss	✓	✓	curriculum	61.5
CMA	✓	✓	contrastive	67.2

Table 7. CMA with alternative contrastive loss functions.

	Debiased [8]	RINCE [7]	InfoNCE [29]
ACDC val mIoU	66.3	67.4	67.2

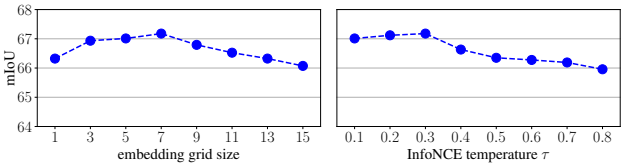

Figure 5. Hyperparameter study of the embedding grid size and the temperature in the InfoNCE loss.

anisms discussed in Sec. 3.3, we explored swapping the InfoNCE loss for more robust alternatives, to reduce the detrimental effects of false positives or false negatives. We picked the debiased contrastive loss of [8] to account for false negatives, and the robust InfoNCE (RINCE) [7] loss to account for false positives. As shown in Table 7, neither alternative shows significant improvements over InfoNCE. Even though RINCE performs slightly better overall, it introduces extra complexity, which prompts us to prefer the simpler InfoNCE loss. The negligible benefit of RINCE implies that patch-level grouping and confidence modulation already effectively mitigate the false positive rate.

Hyperparameter Sensitivity. Fig. 5 shows the sensitivity of CMA performance to changes in two central, method-specific hyperparameters: the embedding grid size, and the InfoNCE temperature τ. Note that the performance is quite insensitive to either hyperparameter.

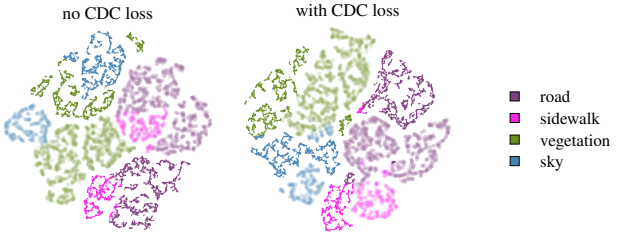

Figure 6. t-SNE plots showing semantic features of a pair of corresponding ACDC validation set images (adverse↔sharp, normal↔blurry), for a CMA model trained without (left) and with (right) the CDC loss. For clarity, only four classes are shown.

Embedding Space Visualization. The t-SNE [37] visualizations in Fig. 6 show semantic features extracted from a corresponding pair of adverse- and normal-condition images of ACDC. The features are color-coded by ground truth class, whereby adverse-condition features are plotted sharp and normal-condition features blurry (the "ground truth" for the normal-condition image was obtained through pseudo-labeling). For clarity, only road, sidewalk, vegetation, and sky features are plotted. The left plot shows the features of CMA without the CDC loss. Note that sky (blue) and sidewalk (pink) features are scattered. By contrast, with the CDC loss, the features of these classes are correctly grouped together across conditions in the right plot.

5. Conclusion

We present CMA, a model adaptation method for cross-condition semantic segmentation. CMA leverages image-level correspondences to learn condition-invariant features through a contrastive loss. This fosters a shared embedding space, where adverse-condition image features are clustered with semantically corresponding normal-condition features. As experimentally shown, this leads to large performance gains in normal-to-adverse model adaptation, with CMA setting the new state of the art on several benchmarks.

Acknowledgment. This work was supported by the ETH Future Computing Laboratory (EFCL), financed by a donation from Huawei Technologies.

References

[1] Hernán Badino, Daniel Huber, and Takeo Kanade. Visual topometric localization. In *IEEE Intelligent vehicles symposium (IV)*, 2011. 5, 7

[2] David Bruggemann, Christos Sakaridis, Prune Truong, and Luc Van Gool. Refign: Align and refine for adaptation of semantic segmentation to adverse conditions. In *WACV*, 2023. 3, 4

[3] Keenan Burnett, David J Yoon, Yuchen Wu, Andrew Z Li, Haowei Zhang, Shichen Lu, Jingxing Qian, Wei-Kang Tseng, Andrew Lambert, Keith YK Leung, et al. Boreas: A multi-season autonomous driving dataset. *The International Journal of Robotics Research*, 42(1-2):33–42, 2023. 2, 3

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 40(4):834–848, 2017. 6

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2, 4

[6] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, 2018. 1

[7] Ching-Yao Chuang, R Devon Hjelm, Xin Wang, Vibhav Vineet, Neel Joshi, Antonio Torralba, Stefanie Jegelka, and Yale Song. Robust contrastive learning against noisy views. In *CVPR*, 2022. 8

[8] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *NeurIPS*, 2020. 8

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 3, 5

[10] Dengxin Dai, Christos Sakaridis, Simon Hecker, and Luc Van Gool. Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. *IJCV*, 128(5):1182–1204, 2020. 5

[11] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2018. 1

[12] Xiaoqing Guo, Jie Liu, Tongliang Liu, and Yixuan Yuan. Simt: Handling open-set noise for domain adaptive semantic segmentation. In *CVPR*, 2022. 6, 7

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 4

[14] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016. 1

[15] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. DAFormer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022. 7

[16] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. HRDA: Context-aware high-resolution domain-adaptive semantic segmentation. In *ECCV*, 2022. 7

[17] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *NeurIPS*, 2021. 2, 6, 7

[18] Zhengkai Jiang, Yuxi Li, Ceyuan Yang, Peng Gao, Yabiao Wang, Ying Tai, and Chengjie Wang. Prototypical contrast adaptation for domain adaptive semantic segmentation. In *ECCV*, 2022. 2

[19] Jogendra Nath Kundu, Akshay Kulkarni, Amit Singh, Varun Jampani, and R. Venkatesh Babu. Generalize then adapt: Source-free domain adaptive semantic segmentation. In *ICCV*, 2021. 2

[20] Mans Larsson, Erik Stenborg, Lars Hammarstrand, Marc Pollefeys, Torsten Sattler, and Fredrik Kahl. A cross-season correspondence dataset for robust semantic segmentation. In *CVPR*, 2019. 3, 5, 7

[21] Geon Lee, Chanho Eom, Wonkyung Lee, Hyekang Park, and Bumsub Ham. Bi-directional contrastive learning for domain adaptive semantic segmentation. In *ECCV*, 2022. 2

[22] Shuang Li, Binhui Xie, Bin Zang, Chi Harold Liu, Xinjing Cheng, Ruigang Yang, and Guoren Wang. Semantic distribution-aware contrastive adaptation for semantic segmentation. *arXiv preprint arXiv:2105.05013*, 2021. 2

[23] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 4

[24] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. In *CVPR*, 2021. 2

[25] Xin Luo, Wei Chen, Chen Li, Bin Zhou, and Yusong Tan. Multi-level consistency learning for source-free model adaptation. *IEEE Robotics and Automation Letters*, 7(4):12419–12426, 2022. 2

[26] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017. 2, 3, 5, 7

[27] Robert A. Marsden, Alexander Bartler, Mario Döbler, and Bin Yang. Contrastive learning and self-training for unsupervised domain adaptation in semantic segmentation. In *International Joint Conference on Neural Networks (IJCNN)*, 2022. 2

[28] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The Mapillary Vistas dataset for semantic understanding of street scenes. In *ICCV*, 2017. 1

[29] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 4, 8

[30] Christos Sakaridis, David Bruggemann, Fisher Yu, and Luc Van Gool. Condition-invariant semantic segmentation. *arXiv preprint arXiv:2305.17349*, 2023. 1

[31] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In *ECCV*, 2018. 5

[32] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *IJCV*, 126(9):973–992, 2018. 5

[33] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. ACDC: The Adverse Conditions Dataset with Correspondences for semantic driving scene understanding. In *ICCV*, 2021. 1, 2, 3, 5, 7

[34] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *TPAMI*, 44(6):3139–3153, 2022. 3, 5, 7

[35] Prabhu Teja S and Francois Fleuret. Uncertainty reduction for model adaptation in semantic segmentation. In *CVPR*, 2021. 2, 6, 7, 8

[36] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? In *NeurIPS*, 2020. 4

[37] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008. 8

[38] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. 2, 6, 7

[39] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021. 3, 4

[40] Xinyi Wu, Zhenyao Wu, Lili Ju, and Song Wang. A one-stage domain adaptation network with image alignment for unsupervised nighttime semantic segmentation. *TPAMI*, 45(1):58–72, 2021. 3

[41] Markus Wulfmeier, Alex Bewley, and Ingmar Posner. Addressing appearance change in outdoor robotics with adversarial domain adaptation. In *IROS*, 2017. 1

[42] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. SePiCo: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *TPAMI*, 2023. 2, 7

[43] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 6, 7, 8

[44] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 1, 5

[45] Oliver Zendel, Matthias Schörghuber, Bernhard Rainer, Markus Murschitz, and Csaba Beleznai. Unifying panoptic segmentation for autonomous driving. In *CVPR*, 2022. 5

[46] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 5