A. Appendix

A.1. Method - Details of region-scale/contextual-relation pseudo labels and regularizer weight

We would share more details about the region-scale/contextual-relation pseudo labels and the weight of regularizer used in this paper. For the **source domain**, the sizes of the input image for datasets GTA5 and SYNTHIA are 720×1280 and 760×1280 , respectively. In this paper, we use two types of regions with two different sizes. The first sizes of regions for datasets GTA5 and SYNTHIA are 18×32 and 19×32, respectively. The second sizes of regions for datasets GTA5 and SYNTHIA are 36×64 and 38×64, respectively. For the **target domain** (dataset Cityscapes), the size of input image is 512×1024. The sizes of regions are 16×32 and 32×64, respectively. We use two independent contextual-relations (CR) classifiers to deal with these two types of regions with two different sizes. The weight of the regularizer in adaptive entropy max-minimizing adversarial learning scheme decreases with training iteration, which is expressed as: $\lambda_R = (1 - \frac{iter}{max\_iter})^{power}$ with $power = 0.9$.

A.2. Method - Traditional Losses

For the source domain, traditional approaches learn a supervised segmentation model G that aims to minimize a segmentation loss. For the target domain, UDA networks using adversarial learning train G to extract domain-invariant features though the minimaxing game between G and a domain discriminator D. The overall loss in the UDA networks can therefore be formulated by:

$$\mathcal{L}(X_s, X_t) = \mathcal{L}_{seg}(G) + \mathcal{L}_{adv}(G, D) \tag{1}$$

A.3. Method - Loss in Multi-Scale Adaptation

Source Flow: In our contextual-relation consistent domain adaptation (Cr-CDA) with multi-scale form, the source-domain data contribute to $\mathcal{L}_{seg}$, $\mathcal{L}_{cr}$ and $\mathcal{L}_D$. Given a source-domain image $x_s \subset X_s$ and the corresponding pixel-scale label $y_s \subset Y_s$, region-scale (contextual-relations) pseudo label $y_{s\_cr} \subset Y_{s\_cr}$, $P_s^{(h,w,c)} = C_{seg}(E(x_s))$ is the predicted probability map $w.r.t$ each pixel over C classes; $P_{s\_cr}^{(i,j,n)} = C_{cr}(E(x_s))$ is the predicted probability map $w.r.t$ each region over N classes. The layout probability map $P_{s\_layout}^{(h,w,c+n)}$ is generated by concatenating $P_s^{(h,w,c)}$ and up-sampled $P_{s\_cr}^{(i,j,n)}$. $\mathcal{L}_{seg}$ and $\mathcal{L}_{cr}$ are provided in the submitted manuscript. $\mathcal{L}_{s_d}$ is formulated as follows:

$$\mathcal{L}_{s_d}(E, C_{seg}, C_{cr}, C_D) = \sum_{h,w} E[\log C_D(P_{s\_layout}^{(h,w,c+n)})] \tag{2}$$

Target Flow: As the target label is not accessible, we design an adversarial training scheme between feature extractor E and classifiers (C_{seg}, C_{cr} and C_D) that extracts discriminative features via max-minimizing entropy in the target domain. Given a target image $x_t \subset X_t$, $P_t^{(h,w,c)} = C_{seg}(E(x_t))$ is the predicted probability map $w.r.t$ each target pixel over C classes; $P_{t\_cr}^{(i,j,n)} = C_{cr}(E(x_t))$ is the predicted probability map $w.r.t$ each target region over N classes. The layout probability map $P_{t\_layout}^{(h,w,c+n)}$ of the target-domain image is generated by concatenating $P_t^{(h,w,c)}$ and up-sampled $P_{t\_cr}^{(i,j,n)}$. $\mathcal{L}\_ent\_pix$ and $\mathcal{L}_{ent\_cr}$ are provided in the submitted manuscript. $\mathcal{L}_{t_d}$ is expressed as:

$$\mathcal{L}_{t_d}(E, C_{seg}, C_{cr}, C_D) = \sum_{h,w} E[\log(1 - C_D(P_{t\_layout}^{(h,w,c+n)}))] \tag{3}$$

Therefore, the overall global alignment loss is expressed as:

$$\mathcal{L}_D(E, C_{seg}, C_{cr}, C_D) = \mathcal{L}_{s_d} + \mathcal{L}_{t_d} + Ent_{s_d} + Ent_{t_d} \tag{4}$$

where domain classifier entropy is $Ent_{s_d} = -C_D(P_{s\_layout}^{(h,w,c+n)}) \log C_D(P_{s\_layout}^{(h,w,c+n)})$ for source domain; similarly, $Ent_{t_d} = -C_D(P_{t\_layout}^{(h,w,c+n)}) \log C_D(P_{t\_layout}^{(h,w,c+n)})$ for target domain.

A.4. Experiment - More Qualitative Results

We share more qualitative experimental results for GTA5 $\rightarrow$ Cityscapes as shown in Fig. 1. As Fig. 1 shows, our CrCDA aligns both low-level features (*e.g.*, boundaries of sidewalk, car and person *etc.*) and high-level features by multi-scale adversarial learning. As a comparison, AdvEnt neglects low-level information which focuses more on high-level features. As a result, CrCDA achieves both local and global consistencies in segmentation while AdvEnt achieves global consistency only.

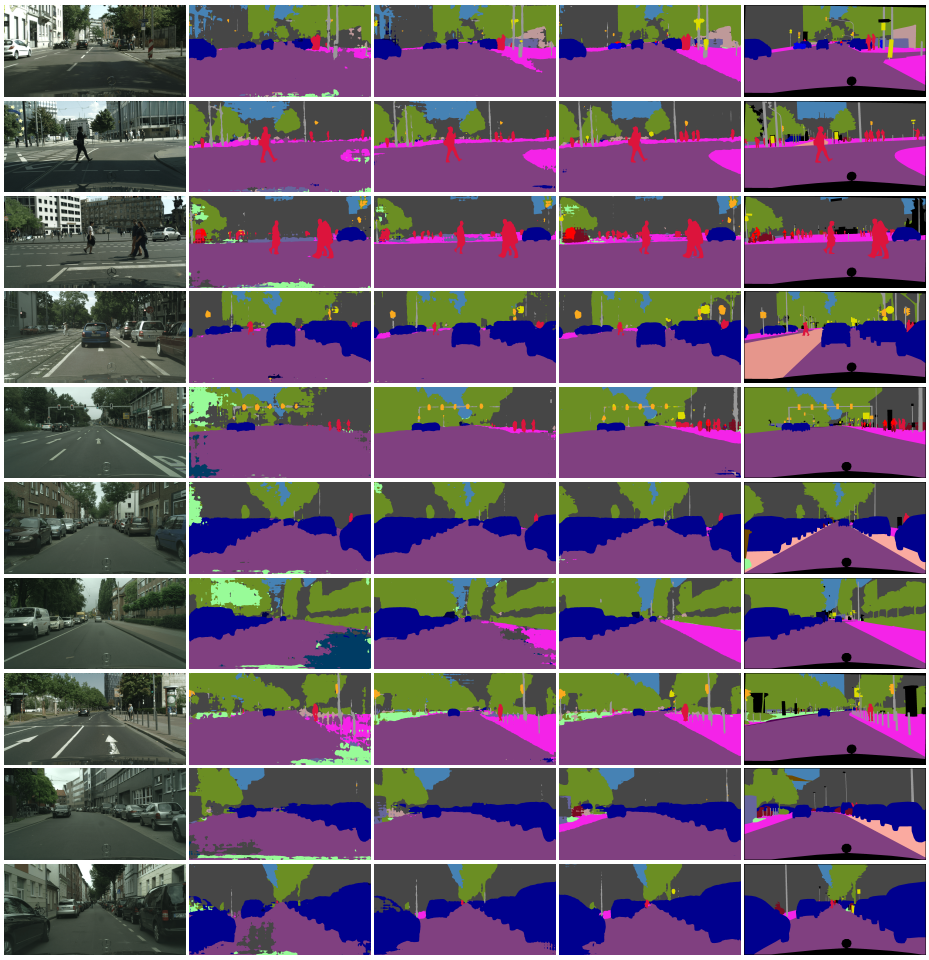

Fig. 1. Qualitative results for GTA5 → Cityscapes. Our approach (CrCDA) aligns low-level features (*e.g.*, boundaries of sidewalk, car and person *etc.*) as well as high-level features by multi-scale adversarial learning. In contrast, AdvEnt ignores low-level information because global alignment focuses more on high-level information. Thus, as shown above, CrCDA achieves both local and global consistencies while AdvEnt only achieves global consistency.