Supplementary for ADA-AT/DT: An Adversarial Approach for Cross-Domain and Cross-Task Knowledge Transfer

Ruchika Chavhan Ankit Jha Biplab Banerjee Subhasis Chaudhuri
{chavhanruchika2801, ankitjha16, getbiplab}@gmail.com, sc@ee.iitb.ac.in
Indian Institute of Technology Bombay, India

1. Introduction

In this supplementary document for ADA-AT/DT, we report the following:

- Additional experiments on the architecture of the domain classifier along with a detailed quantitative comparison in terms of standard evaluation metrics for both depth estimation and semantic segmentation.
- Visualisation of features extracted by the encoder of $\mathcal{N}_1^{A\cup B}$ with and without adversarial domain domain adaptation at the deepest spatial level of the encoder.
- Experiments for the symmetric scenario where for every pair of domains ($\mathcal{A}$, $\mathcal{B}$), we also report the performance on ($\mathcal{B}$, $\mathcal{A}$)

2. Experiments with $\mathcal{N}_1^{A\cup B}$

For the task-specific base models $\mathcal{N}_1^{A\cup B}$ and $\mathcal{N}_2^{A}$, we employ the model architecture proposed in [2]. The model utilized in [2] consists of significantly fewer trainable parameters as compared to [1]. Moreover, the model used in [2] is used for unpaired image-to-image translation ensuring that it extracts superior higher dimensional abstract representations. However, this model faces the problem of domain shift when trained on a real and synthetic domains concurrently. We have shown that integration of adversarial domain adaptation with the model architecture introduced in [2] has outperformed [1] for both semantic segmentation and depth estimation. The major merit of our proposed method is superior performance for both tasks with notably less number of parameters.

We have performed experiments with the architecture of the domain classifier $\mathcal{C}^{\mathcal{A}\cup\mathcal{B}}$ to select the best performing model. We consider three types of architectures:

- Firstly, we consider a discriminator with convolution layers followed by fully connected layers which output the probability that the input image belongs to either of the domains. The combination of fully connected and convolutional layers are known to provide greater stability in a min-max optimisation setup. We denote this model by "conv + fc" (Convolution + Fully Connected) in Table1 and 2.
- We experiment with a domain classifier consisting only of only convolutional layers. A fully convolutional discriminator without the use of max-pooling layers has been shown to provide greater training stability and significantly accurate results. We denote this model by "fcn" (Fully Convolutional Network) in Table 1 and 2.
- The binary domain classifier identifies the class label of the features provided by the encoder while the decoder outputs the task-specific output. The domain classifier extracts domain-specific information and promotes the normalization of this information in these features promoting domain-invariance. Therefore, convolution layers in the domain classifier propagate domain-specific information. Since two domains are involved domain-specific information may prove beneficial for task-specific predictions due to sheer difference in the sources of data. We experiment with a model in which the parameters of convolutional layers of the domain classifier are decoder are shared. This implies that feature maps are shared between the domain classifier and decoder. This network is denoted by "shared" in Table 1 and 2.

2.1. Semantic Segmentation

We have performed experiments on four combinations of domains for $\mathcal{A}$ and $\mathcal{B}$ for which labeled data for semantic segmentation is available. From Table 1-(a), where we train $\mathcal{N}^{\mathcal{A}\cup\mathcal{B}}$ on two synthetic domains (Synthia and Carla), we observe that the fully convolution domain classifier surpasses the other two models in terms of both mIoU and pixel wise accuracy. It is observed that a domain classifier with convolutional layers and fully connected layers performs significantly better in the case Table 1-(b) and 1-(d). In case of Table 1-(c), we observe that the model with

shared convolutional layers among the decoder and the domain classifier provides better mIoU and accuracy. However, this model performs poorly in case of Table 1-(d) for classes like Person, Poles, and Traffic Signs. To perform experiments on task transfer across domains using the transfer function $G^{\mathcal{A}}_{1\to 2}$, we employ the domain classifier with both convolutional and fully connected layers to train the base models as it provides more accurate results in most cases especially for combinations of real and synthetic datasets.

2.2. Depth Estimation

Experiments are performed on three combinations of domains $\mathcal{A}$ and $\mathcal{B}$ for which annotated data for depth estimation is available. As observed in Table 2-(a), domain classifier with only convolutional layers performs marginally better than the other two cases in terms of relative losses and accuracies. In case of Table 2-(b), it is observed that the domain classifier consisting of a fully convolutional network outperforms the other models in terms of all evaluation metrics. The domain classifier with both convolutional layers and fully connected layers provides more accurate results in the case reported in Table 2-(c). However, we can note that in all cases, all the three proposed models provide equally good results, proving that all the models are proficient for depth estimation. To maintain consistency in the models used across tasks, we employ both convolutional layers and fully connected layers for the domain classifier similar to similar to semantic segmentation.

3. Visualising features with ADA-AT/DT

Given that our problem deals with two domains, domain-invariance is a critical property for corresponding abstract representations. We argue that the deep features extracted from the two domains are disjoint when the model is trained concurrently on the two domains without adversarial domain adaptation at an intermediate level. We demonstrate the above with t-SNE visualisation of the abstract features before and after the task transformation mapping $G^{\mathcal{A}}_{1\to 2}$ is applied. We visualise the intermediate features generated by our proposed method before and after the task transformation mapping is applied and prove that the features are indeed domain-invariant. To prove the concreteness of our method, we choose to perform visualizations with one synthetic and one real dataset. Figure 1 shows four t-SNE visualization plots in which we consider two datasets: Carla (red) and CityScapes (green) datasets. Figure 1-(a) shows that the t-SNE plots for the encoder of the model which is trained on $\mathcal{A} \cup \mathcal{B}$ without Adversarial Domain Adaptation (ADA) are disjoint. Consequently, the transformed features in this case are also disjoint as shown in Figure 1-(c). This hinders the generalisation ability of the transformation mapping on the target domain. As seen in Figure 1-(b), the feature space provided by our proposed method involving domain adaptation in an adversarial setup is significantly less disjoint. Accordingly, the transformed features obtained by $G^{\mathcal{A}}_{1\to 2}$ with adversarial domain adaptation are less disjoint as seen in Figure 1-(d). The indistinguishability of these intermediate deep features promotes superior results on the target domain for which supervision is unavailable for $\mathcal{T}_2$.

4. Symmetric Scenario

The cross task cross domain transfer is designed to leverage information from synthetic domains to real domains due to the abundance of labeled data which is often tedious to obtain for real life data. We deel that it is helpful to perform experiments on the symmetric scenario, where for every pair $\mathcal{A}$, $\mathcal{B}$, we also report results for the pair $\mathcal{B}$, $\mathcal{A}$. The results are stated in Table 3 and 4.

References

[1] Pierluigi Zama Ramirez, Alessio Tonioni, Samuele Salti, and Luigi di Stefano. Learning across tasks and domains. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8109–8118, 2019.

[2] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.

	$\mathcal{A}$	$\mathcal{B}$	Method	Road	Sidewalk	Walls	Fence	Person	Poles	Vegetation	Vehicles	Tr. Sign	Building	Sky	mIoU	Acc
	Synthia	Carla	conv + fc	77.68	45.84	3.334	3.782	1.085	1.869	37.63	44.59	3.566	62.88	86.09	33.486	77.43
(a)	Synthia	Carla	fcn	**87.56**	**64.21**	**29.11**	**12.6**	**24.31**	8.164	**56.12**	**58.35**	**14.92**	**74.18**	**92.64**	**47.46**	**81.96**
	Synthia	Carla	shared	84.56	56.73	16.14	4.182	0.3861	**25.04**	46.01	52.11	6.84	69.51	89.22	40.97	81.26
	Synthia	CityScapes	conv + fc	**80.36**	**49.16**	**33.31**	5.905	**8.711**	**11.94**	**60.84**	59.96	**20.17**	70.09	81.17	**43.78**	**77.41**
(b)	Synthia	CityScapes	fcn	76.85	47.97	24.61	**13.77**	8.036	10.45	58.69	**62.92**	15.42	**71.99**	**83.34**	43.09	70.03
	Synthia	CityScapes	shared	74.21	41.61	18.47	8.05	3.732	6.96	49.73	49.33	20.05	62.08	75.36	37.23	75.31
	Carla	CityScapes	conv + fc	**77.3**	52.21	45.93	**21.83**	**13.95**	**16.08**	68.7	59.02	11.45	61.37	87.89	46.88	**78.99**
(c)	Carla	CityScapes	fcn	69.54	40.43	30.81	9.981	3.435	5.145	55.37	43.41	11.38	46.43	82.5	36.22	69.52
	Carla	CityScapes	shared	76.65	**54.61**	**51.21**	23.2	12.47	12.7	**70.35**	**66.15**	**20.26**	**67.27**	**91.34**	**49.65**	76.55
	Carla	KITTI	conv + fc	**87.07**	**73.67**	**64.01**	**35.71**	**25.63**	**28.23**	**76.18**	**74.83**	**14.72**	**78.57**	**96.31**	**59.53**	**88.51**
(d)	Carla	KITTI	fcn	84.39	61.74	45.59	14.63	2.901	10.07	64.31	59.16	10.49	65.71	93.06	46.55	80.45
	Carla	KITTI	shared	79.65	54.45	22.11	7.52	0.3521	1.44	50.07	42.94	0.1124	54.97	88.34	36.54	79.82

Table 1: Quantitative experiments of experiments performed with domain classifier for semantic segmentation

				Lower is better				Higher is better		
	$\mathcal{A}$	$\mathcal{B}$	Method	Abs Rel	Sq Rel	RMSE	RMSE log	δ_1	δ_2	δ_3
	Synthia	Carla	conv + fc	0.1432	**0.459**	0.714	0.2621	0.8539	0.9517	**0.9763**
(a)	Synthia	Carla	fcn	**0.136**	1.045	**0.4754**	**0.2247**	**0.8904**	**0.9521**	0.9751
	Synthia	Carla	shared	0.3087	2.5729	2.0145	0.3917	0.7831	0.9122	0.9488
	Synthia	CityScapes	conv + fc	0.5367	4.4732	4.8909	0.6735	0.3741	0.6156	0.71223
(b)	Synthia	CityScapes	fcn	**0.221**	**0.8119**	**0.5653**	**0.3095**	**0.6358**	**0.7633**	**0.8145**
	Synthia	CityScapes	shared	0.9064	10.92	7.2573	0.854	0.3567	0.5319	0.6474
	Carla	CityScapes	conv + fc	**0.3296**	**1.8972**	**2.4429**	**0.4578**	**0.5691**	**0.7414**	**0.8043**
(c)	Carla	CityScapes	fcn	0.4409	3.2385	2.6137	0.5098	0.541	0.6871	0.7603
	Carla	CityScapes	shared	0.5654	4.5161	3.5725	0.6138	0.4335	0.6308	0.7374

Table 2: Quantitative experiments of experiments performed with domain classifier for depth estimation

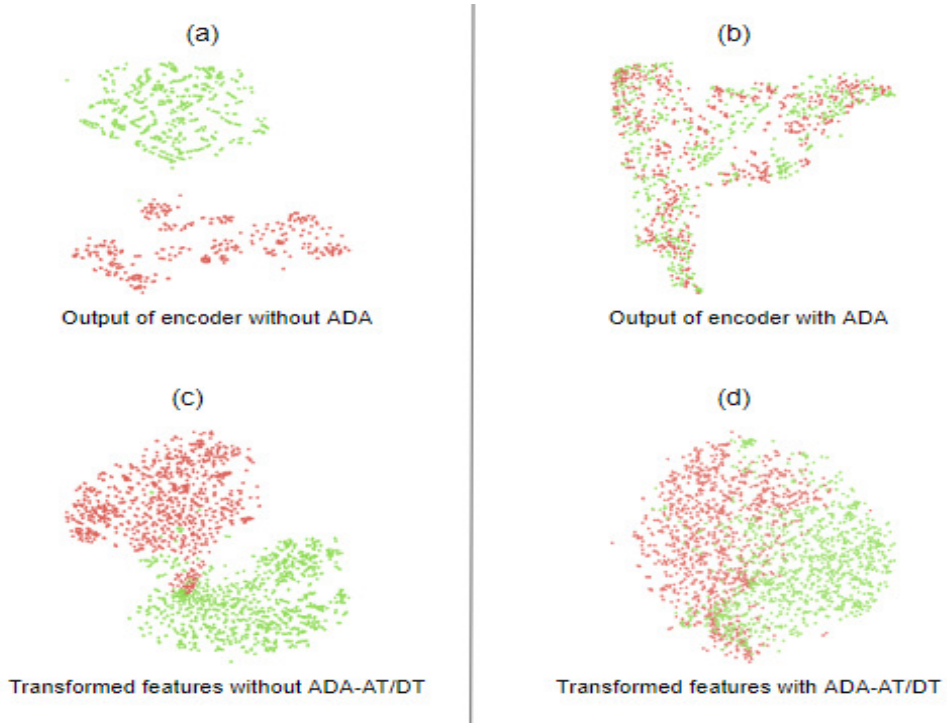

Figure 1: t-SNE visualization of features before and after the transformation for base models. (a) and (c) denote the features before and after the transformation mapping is applied on features obtained on a model trained without ADA. (b) and (d) show the features before and after the transformation mapping is applied on features obtained from our proposed method.

	$\mathcal{A}$	$\mathcal{B}$	Method	Road	Sidewalk	Walls	Fence	Person	Poles	Vegetation	Vehicles	Tr. Sign	Building	Sky	mIoU	Acc
	CityScapes	Synthia	**conv**	83.66	33.57	0.00	3.01	6.21	5.71	26.16	39.07	0.00	11.11	74.56	25.73	77.91
(b)	CityScapes	Synthia	**U-Net**	83.89	37.69	5.62	4.62	8.48	9.21	33.41	42.62	0.00	15.32	76.56	28.85	78.93
	CityScapes	Synthia	**U-Net + att.**	85.58	41.81	1.75	6.81	4.11	3.85	46.74	41.65	0.00	14.19	79.39	29.62	81.68
	CityScapes	Carla	**conv**	67.03	22.22	0.708	0.8322	0.00	0.1246	39.36	16.48	0.00	14.84	64.54	20.55	65.02
	CityScapes	Carla	**U-Net**	73.64	38.47	6.04	6.72	0.00	5.293	48.51	28.47	0.00	23.99	71.59	27.52	72.56
(c)	CityScapes	Carla	**U-Net + att.**	72.86	44.64	1.306	1.744	0.00	2.188	55.22	26.78	0.00	23.34	71.86	27.26	73.99

Table 3: Quantitative results obtained from depth estimation to semantic segmentation for symmetric scenario

				Lower is better				Higher is better		
	$\mathcal{A}$	$\mathcal{B}$	Method	Abs Rel	Sq Rel	RMSE	RMSE log	δ_1	δ_2	δ_3
	Carla	Synthia	**conv**	1.5194	25.3	7.293	0.7835	0.5691	0.7539	0.8366
(a)	Carla	Synthia	**U-Net**	0.543	5.783	3.621	0.5321	0.6072	0.8356	0.8968
	Carla	Synthia	**U-Net + att.**	0.6594	7.596	3.264	0.5599	0.6609	0.8281	0.8933
	CityScapes	Carla	**conv**	0.3095	2.685	12.67	0.9444	0.5939	0.7259	0.8101
(b)	CityScapes	Carla	**U-Net**	0.2083	1.212	3.723	0.513	0.7091	0.8418	0.9083
	CityScapes	Carla	**(U-Net + att.**	0.5396	1.351	1.5045	0.3256	0.7999	0.9151	0.9565

Table 4: Quantitative results obtained from semantic segmentation to depth estimation for symmetric scenario