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

Ruchika Chavhan   Ankit Jha   Biplab Banerjee   Subhasis Chaudhuri

Indian Institute of Technology, Bombay, India

Winter Conference on Applications of Computer Vision (WACV 2021)

Overview

Motivation

- What if supervision is available for only certain tasks in a domain?
- How to perform inference for the task in a domain for which there exists no ground truth labels ?
- Can we transfer task-related information across domains?
- How can we leverage information from strongly correlated tasks across domains ?
- We need an intersection between domain transfer and task transfer

Problem Statement

- Consider two dense predictions tasks: monocular depth estimation and semantic segmentation denoted by $\mathcal{T}_1$ and $\mathcal{T}_2$ interchangeably
- Source domain: $\mathcal{A}$ Target domain: $\mathcal{B}$
- Supervision is available for both $\mathcal{T}_1$ and $\mathcal{T}_2$ for domain $\mathcal{A}$
  Supervision for $\mathcal{T}_2$ is unavailable for domain $\mathcal{B}$
- $\{\mathcal{X}_j^{\mathcal{A}/\mathcal{B}}, \mathcal{Y}_j^{\mathcal{A}/\mathcal{B}}\}$ $(j \in 1, 2)$ denotes training samples (for $\mathcal{T}_1$ and $\mathcal{T}_2$ for $\mathcal{A}$ and $\mathcal{T}_1$ for $\mathcal{B}$)
- How to obtain predictions for $\mathcal{T}_2$ for domain $\mathcal{B}$ using information from $\mathcal{T}_1$ and $\mathcal{A}$?
- How can we perform cross-domain cross-task knowledge transfer ?

Prior Work

"*Learning across tasks and domains*" Ramirez et al, ICCV, 2019 [ZRTSDS19]

- First work in cross task cross domain knowledge transfer

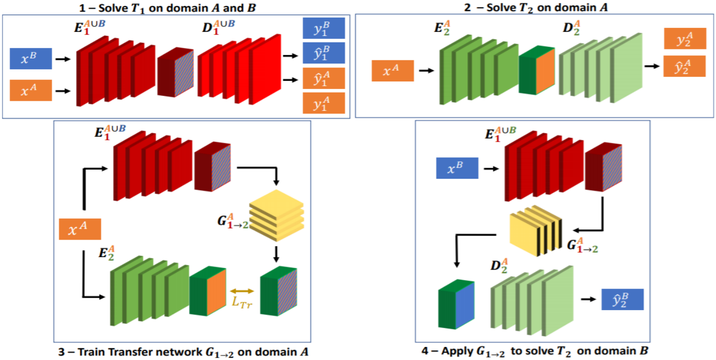

Figure: Architectural overview of Across Tasks Domain Transfer (ATDT) [ZRTSDS19].

"Learning transferable features with deep adaptation networks" Long et.al, JMLR, 2015 [LCWJ15].

- Enhance the transferability of features from task-specific layers

"Cycada: Cycle consistent adversarial domain adaptation" Hoffman et.al, [Hof18].

- Cycle-Consistent Adversarial Domain Adaptation model
- Domain adaptation for semantic segmentation

FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation, Hoffman et al [HWYD16].

- Domain adaptive semantic segmentation method
- Global and category specific adaptation

Shortcomings of the ATDT [ZRTSDS19] model

Domain shift

- To perform task-transfer, domain-agnostic features are required
- The base model in ATDT is trained on the union of the two datasets
- This causes domain discrepancy when the domains considered are real (CityScapes) and synthetic (CARLA)
- For domain-invariant features, encoder needs to made deeper leading to increase in the number of trainable parameters
- **Adversarial training to ensure that the domain-gaps are minimized**

Transfer Function

- Can we employing a better transfer function that captures contextual information relevant to the task?
- **Transformation mapping in the form of U-Net with spatial attention**

ADA-AT/DT Architecture

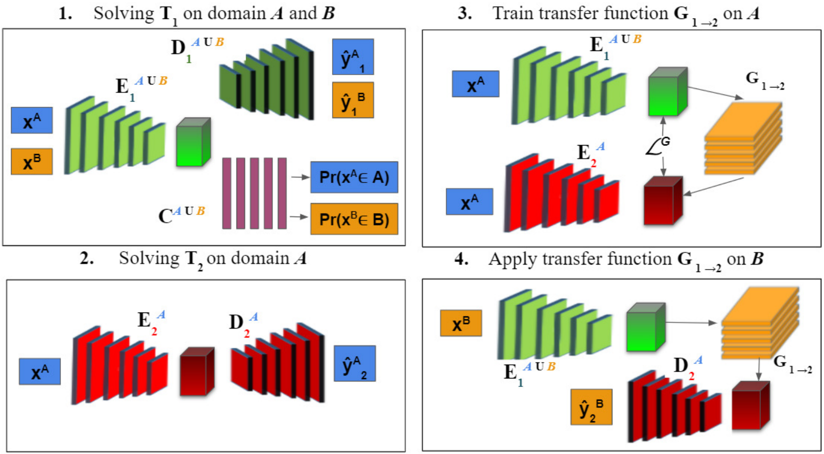

Figure: Overview of the ADA-AT/DT framework

Methodology

Step I: Training base model on $\mathcal{T}_1$

- $\mathcal{E}_1^{A \cup B}$: Encoder network trained on $\mathcal{T}_1$ on domains $\mathcal{A}$ and $\mathcal{B}$
- $\mathcal{D}_1^{A \cup B}$: Decoder network trained on $\mathcal{T}_1$ on domains $\mathcal{A}$ and $\mathcal{B}$
- $\mathcal{N}_1^{A \cup B}(\cdot)$: $\mathcal{D}_1^{A \cup B}(\mathcal{E}_1^{A \cup B}(\cdot))$
- $\mathcal{C}_1^{A \cup B}$: Binary domain classifier

$$
\begin{aligned}
\mathcal{L}_{adv}^{\mathcal{A}}(x) &=_{x \sim \mathcal{A}} [\log C(E_1^{A \cup B}(x))] \\
\mathcal{L}_{adv}^{\mathcal{B}}(x) &=_{x \sim \mathcal{B}} [\log 1 - C(E_1^{A \cup B}(x))] \\
\mathcal{L}_{adv}(x) &= \mathcal{L}_{adv}^{\mathcal{A}}(x) + \mathcal{L}_{adv}^{\mathcal{B}}(x) \\
\mathcal{L}_{task}(x, y) &= \mathcal{L}_T(x, y)
\end{aligned} \tag{1}
$$

The total optimization objective is:

$$
\min_{E_1, \mathcal{D}_1} \max_{\mathcal{C}} (1 - \alpha)\mathcal{L}_{adv} + \alpha \mathcal{L}_{task} \tag{2}
$$

Methodology

Step II: Training base model on $\mathcal{T}_2$

- $\mathcal{E}_2^A$: Encoder network trained on $\mathcal{T}_2$ on domain $\mathcal{A}$
- $\mathcal{D}_2^A$: Decoder network trained on $\mathcal{T}_2$ on domains $\mathcal{A}$
- $\mathcal{N}_2^A(\cdot)$: $\mathcal{D}_2^A(\mathcal{E}_2^A(\cdot))$

The network for task $\mathcal{T}_2$ is trained only on domain $\mathcal{A}$.

Step III: Training the transfer function $G_{1\rightarrow 2}^{\mathcal{A}}$

- Training a mapping function to transform deep features extracted from $\mathcal{T}_1$ to $\mathcal{T}_2$

$$\mathcal{L}_G = |G_{1\rightarrow 2}^{\mathcal{A}}(E_1^{\mathcal{A}\cup\mathcal{B}}(x_{\mathcal{A}})) - E_2^{\mathcal{A}}(x_{\mathcal{A}})| \tag{3}$$

Step IV: Testing the transfer function on $\mathcal{B}$

- The output for $\mathcal{T}_2$ for domain $\mathcal{B}$ is obtained by transforming features using $G_{1\rightarrow 2}^{\mathcal{A}}$.

$$\hat{y}_2^{\mathcal{B}} = \mathcal{D}_2^{\mathcal{A}}(G_{1\rightarrow 2}^{\mathcal{A}}(E_1^{\mathcal{A}\cup\mathcal{B}}(x_{\mathcal{B}}))) \tag{4}$$

Results

$\mathcal{T}_1$: Depth estimation and $\mathcal{T}_2$: Semantic Segmentation

$\mathcal{A}$	$\mathcal{B}$	Method	Road	Sidewalk	Walls	Fence	Person	Poles	Vegetation	Vehicle	Tr. Sign	Building	Sky	mIoU	Acc
Synthia	CityScapes	AT/DT [ZRTSDS19]	**85.77**	29.40	1.23	0.00	3.72	14.55	1.87	8.85	0.38	42.79	**67.06**	23.34	64.03
Synthia	CityScapes	CYCADA [Hof18]	72.89	21.74	0.00	0.00	0.77	**21.03**	0.00	16.66	1.34	32.67	46.71	16.23	69.03
Synthia	CityScapes	FCN wild [HWYD16]	60.23	30.33	0.48	0.00	0.53	3.64	4.30	39.44	2.72	53.48	59.06	19.51	65.93
Synthia	CityScapes	DAN [LCWJ15]	54.09	3.67	0.00	0.00	0.00	0.00	2.23	18.88	0.00	1.79	41.59	9.87	51.65
Synthia	CityScapes	**Ours (conv)**	70.21	21.78	3.71	0.00	1.36	1.79	3.69	17.31	2.85	69.82	47.04	21.77	69.93
Synthia	CityScapes	**Ours (U-Net)**	77.30	29.51	7.67	0.00	4.74	4.63	7.37	27.36	6.01	72.44	51.08	26.19	73.99
Synthia	CityScapes	**Ours (U-Net + att.)**	84.39	**38.56**	**10.71**	0.00	**9.11**	8.14	**11.15**	**41.86**	**8.95**	**77.94**	58.69	**31.78**	**79.68**
Carla	CityScapes	AT/DT [ZRTSDS19]	76.44	32.24	4.75	5.58	24.49	**24.95**	68.98	40.49	10.78	69.38	78.19	39.66	76.37
Carla	CityScapes	CYCADA [Hof18]	73.94	47.53	0.00	2.50	1.61	0.00	56.64	21.85	0.63	18.46	52.03	24.56	68.68
Carla	CityScapes	FCN wild [HWYD16]	57.26	53.76	3.72	0.42	0.65	0.12	30.17	4.411	0.00	31.11	6.177	18.16	65.06
Carla	CityScapes	DAN [LCWJ15]	68.61	23.89	0.00	0.00	0.00	0.00	40.08	42.11	0.00	3.52	52.53	17.82	62.37
Carla	CityScapes	Cycle-GAN [ZRTSDS19]	81.58	39.15	6.08	5.31	**30.22**	21.73	**77.71**	50.00	8.33	68.35	77.22	42.33	80.93
Carla	CityScapes	AT/DT (DA)[ZRTSDS19]	85.19	41.37	5.44	3.02	29.90	24.07	71.93	58.09	7.53	70.90	77.78	43.20	81.92
Carla	CityScapes	**Ours (conv)**	62.07	28.47	5.69	2.2	2.16	1.71	41.54	23.77	0.00	27.38	39.51	21.31	54.31
Carla	CityScapes	**Ours (U-Net)**	**85.54**	**68.54**	12.31	**6.71**	23.22	22.46	71.72	**66.66**	**13.79**	**74.45**	80.01	**47.76**	**84.92**
Carla	CityScapes	**Ours (U-Net + att.)**	81.41	58.42	**12.45**	4.08	14.8	13.54	63.81	56.05	12.09	69.45	**81.09**	42.69	84.52

Table: Quantitative results of semantic segmentation for 13 classes on the CityScapes dataset as target domains

Results

$\mathcal{T}_1$: Semantic Segmentation and $\mathcal{T}_2$: Depth estimation

			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	AT/DT [ZRTSDS19]	0.316	5.485	11.712	0.458	0.553	0.785	0.880
Synthia	Carla	**Ours (conv)**	0.2126	1.9635	2.2695	0.3755	0.8029	0.9125	0.9538
Synthia	Carla	**Ours (U-Net)**	0.451	1.2699	1.122	0.2483	0.878	0.9638	0.9834
Synthia	Carla	**Ours (U-Net + att.)**	**0.0796**	**0.19125**	**0.6885**	**0.1937**	**0.9128**	**0.9732**	**0.987**
Carla	CityScapes	AT/DT [ZRTSDS19]	0.394	5.837	13.915	0.435	0.337	0.749	0.899
Carla	CityScapes	Cycle-GAN [ZRTSDS19]	0.943	27.026	21.666	0.695	0.218	0.478	0.690
Carla	CityScapes	AT/DT (DA) [ZRTSDS19]	0.563	10.789	15.636	0.489	0.247	0.668	0.861
Carla	CityScapes	**Ours(conv)**	1.2016	19.893	7.599	0.7528	0.6334	0.8046	0.8754
Carla	CityScapes	**Ours (U-Net)**	0.3397	3.264	3.2385	0.5119	0.7233	0.8753	0.9275
Carla	CityScapes	**Ours (U-Net + att.)**	**0.1894**	**0.9078**	**1.224**	**0.3169**	**0.8245**	**0.9303**	**0.9602**
Carla	Kitti	AT/DT [ZRTSDS19]	0.439	8.263	9.148	0.421	0.483	0.788	0.891
Carla	Kitti	**Ours(conv)**	0.5631	7.0686	3.2385	0.5325	0.7038	0.8681	0.9268
Carla	Kitti	**Ours (U-Net)**	0.233	1.8054	1.785	0.3845	0.7886	0.9208	0.9578
Carla	Kitti	**Ours (U-Net + att.)**	**0.1389**	**0.5381**	**0.7905**	**0.2476**	**0.8655**	**0.9566**	**0.9781**

Table: Quantitative results of depth estimation on Carla, CityScapes and KITTI datasets as target domains

Ablation on binary domain classifier

Selecting the best domain-classifier

- Need to select an optimization objective that provides task-discriminative and domain-invariant features
- Experiments have been conducted on different optimization objectives:
  - Binary cross entropy
  - Mean square error
  - Wasserstein distance
- Experiments conducted to select a good value for the hyperparameter α in Equation 2

Ablation on binary domain classifier

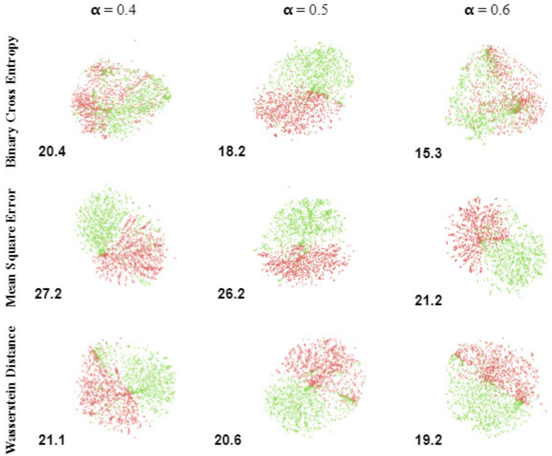

Figure: The hyper-parameter α in Equation 2 and optimization objective for domain classifier has to be chosen carefully for stable training. The Mean Intersection over Union is shown in the bottom left corner of each setup.

Feature Visualization

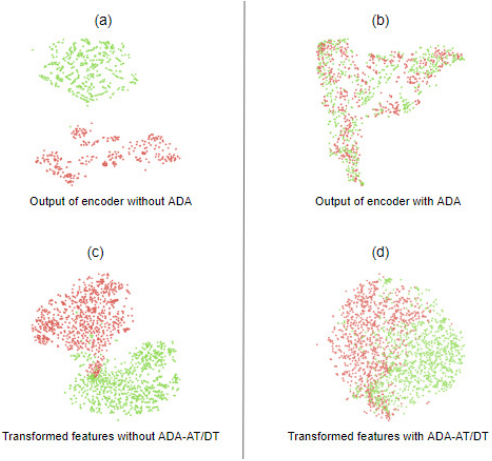

Figure: t-SNE visualization of features before and after the transformation for base models. (a) and (c) denote the features before and after the transformation mapping is applied on features obtained on a model trained without ADA. (b) and (d) show the features before and after the transformation mapping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

Discussion

- To counter the problem of domain-shift between source and target domains, we employ an adversarial domain adaptation training setup using a domain classifier
- We perform extensive experiments on the architecture and the optimization objective of the binary domain classifier
- We demonstrate that our proposed method significantly outperforms [ZRTSDS19] by using models with 81% fewer trainable parameters
- In addition, we perform experiments on a transformation mapping similar to U-Net to ensure maximum exploitation of features for task transfer

References I

Cycada: Cycle consistent adversarial domain adaptation, International Conference on Machine Learning (ICML), 2018.

Judy Hoffman, Dequan Wang, F. Yu, and Trevor Darrell, *Fcns in the wild: Pixel-level adversarial and constraint-based adaptation*, ArXiv **abs/1612.02649** (2016).

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan, *Learning transferable features with deep adaptation networks*, Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, ICML'15, JMLR.org, 2015, p. 97–105.

Pierluigi Zama Ramirez, Alessio Tonioni, Samuele Salti, and Luigi Di Stefano, *Learning across tasks and domains*, 04 2019.

The End