Appendix: CAD Models to Real-World Images: A Practical Approach to Unsupervised Domain Adaptation in Industrial Object Classification

Dennis Ritter[1], Mike Hemberger[2], Marc Hönig[3], Volker Stopp[3], Erik Rodner[4], and Kristian Hildebrand[1]

[1] Berliner Hochschule für Technik
[2] nyris GmbH
[3] topex GmbH
[4] KI-Werkstatt/FB2, University of Applied Sciences Berlin

A Implementation Details

A.1 Adapting pretrained models to rendered images - implementation details

We use pretrained models *"google/vit-base-patch16-224-in21k"* (ViT) [2], *"microsoft/swinv2-base-patch4-window12-192-22k"* (SwinV2) [9], *"facebook/convnextv2-base-22k-224"* (ConvNextV2) [14], and *"facebook/deit-base-distilled-patch16-224"* (DeiT) [13] from Huggingface[5] for experiments using the VisDA-2017 dataset but only ViT and SwinV2 for our Topex-Printer dataset. ViT, SwinV2, and ConvNextV2 were pretrained on ImageNet22K, while DeiT has been pretrained on ImagNet1K. We perform three different training schemes, training the classification head only (CH), fine-tuning the full model (FT), and a combination of CH and FT, tuning the classification head first and continuing with full fine-tuning (CH-FT) inspired by [8].

1. For CH we use the Pytorch[6] SGD optimizer with learning rates [10.0, 0.1, 0.001], momentum 0.9, no weight decay, no learning rate scheduler, and no warmup.
2. For FT we use the Pytorch implementation of AdamW optimizer with learning rates [0.1, 0.001, 0.00001], weight decay 0.01, cosine annealing learning rate scheduler[7] [11] without restarts, and two warmup epochs (10% of total epochs).

For both datasets for data augmentation Pytorch 2.0.0 implementation [8] is used.

[5] https://huggingface.co/models
[6] https://pytorch.org/
[7] https://huggingface.co/docs/transformers/main_classes/optimizer_schedules
[8] https://pytorch.org/vision/main/generated/torchvision.transforms.AugMix.html

A.2 Adapting to real-world images with unsupervised domain adaptation - implementation details

For UDA experiments we start from the best source-domain-only trained CH checkpoint with respect to the model architecture and continue training using the same parameters as the best FT run for each model as described in the paper. We use Pytorch 2.0.0 implementations of image augmentations random resized crop, horizontal flip, and AugMix [3] with the same parameters described in the last paragraph of section A.1. We use the Transfer Learning Library (tllib) [5,7] implementations of CDAN (hidden size 1024) and MCC [6] (temperature 1.0) domain adaptation methods and also combine both using two different initial checkpoints for each model architecture. One initial checkpoint from Huggingface, pretrained on ImageNet22K [1] (*"google/vit-base-patch16-224-in21k"* (ViT) and *"microsoft/swinv2-base-patch4-window12-192-22k"* (SwinV2)) and the best-performing checkpoint after training only the classification head from our source-domain-only experiments. Again, we use global random seed 42 for all experiments and training is performed on a single Nvidia Tesla V100 PCIE 32GB GPU.

Different from other methods, we perform considerably better correctly identifying the *truck* class but underperform on the *motorcycle* and *person* class instead. The confusion matrix shown in figure 4 shows, that our trained model often mixes up motorcycle samples with bicycles (7%) and skateboards (10%) while the person class is mixed up rather uniformly (3%-4%) with skateboards, plants, motorcycles, and horses.

B Dataset samples

Fig. 1. 80 random samples of rendered images from the Topex-Printer dataset. Each image 512^2, featuring machine parts marked with bounding boxes, is trimmed according to these boxes, extended to form a rectangle, and padded with black if needed. Finally, all images are resized to a resolution of 256x256 pixels.

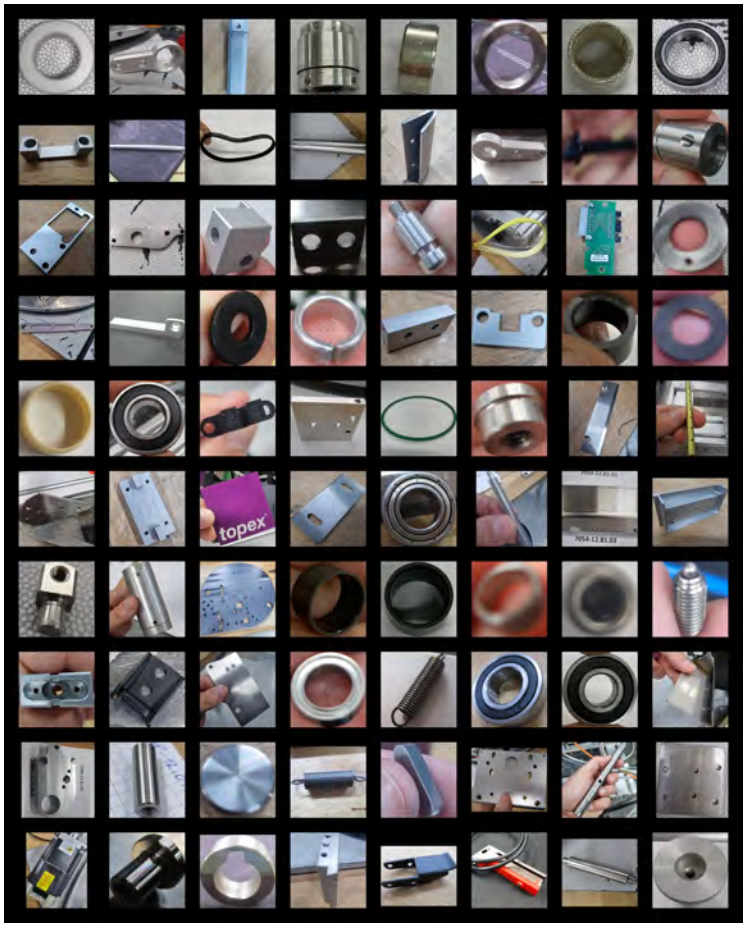

Fig. 2. 80 random samples of real images from the Topex-Printer dataset.

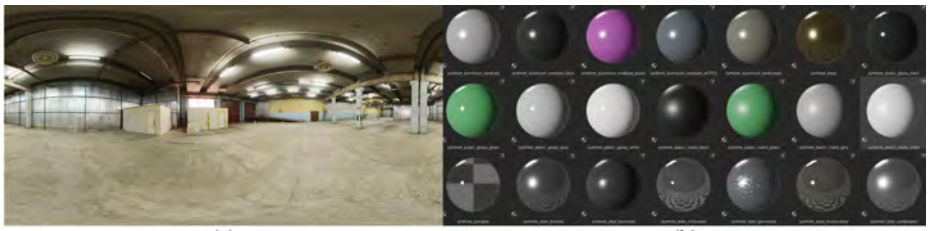

Fig. 3. (Best viewed in color) Left (a): HDRI of the warehouse environment map used in our rendering scene. Image by Sergej Majboroda [CC0], via Polyhaven. Right (b): Our handcrafted Blender material collection we used for the Topex-Printer dataset.

C Evaluation Results

Table 1. Acc@1 in % on target domain (real images) for all source-domain-only training experiments on VisDA-2017 classification dataset. Note that *base* transform means that random color jitter and random grayscale transforms are applied. Faded out rows are representing numerically instable runs that have been canceled due to NaN loss for example.

Model	**Pre-training**	**train scheme**	**transform**	**lr**	**Acc@1**
ViT-b	IN22K	CH	base	1e+1	63.31
ViT-b	IN22K	CH	base	1e-1	71.03
ViT-b	IN22K	CH	base	1e-3	80.37
ViT-b	IN22K	CH	AugMix	1e-3	80.18
ViT-b	IN22K	FT	base	1e-1	07.64
ViT-b	IN22K	FT	base	1e-3	17.69
ViT-b	IN22K	FT	base	1e-5	66.88
ViT-b	IN22K	FT	AugMix	1e-5	73.76
ViT-b	**IN22K**	**CH-FT**	**AugMix**	**1e-5**	**80.53**
SwinV2	IN22K	CH	base	1e+1	69.49
SwinV2	IN22K	CH	base	1e-1	72.02
SwinV2	IN22K	CH	base	1e-3	80.12
SwinV2	IN22K	CH	AugMix	1e-3	79.54
SwinV2	IN22K	FT	base	1e-3	18.84
SwinV2	IN22K	FT	base	1e-5	72.41
SwinV2	IN22K	FT	AugMix	1e-5	73.49
SwinV2	IN22K	CH-FT	AugMix	1e-5	76.96
ConvNextV2	IN22K	CH	base	1e+1	12.81
ConvNextV2	IN22K	CH	base	1e-1	12.42
ConvNextV2	IN22K	CH	base	1e-3	11.30
ConvNextV2	IN22K	CH	AugMix	1e-3	11.98
ConvNextV2	IN22K	FT	base	1e-1	10.04
ConvNextV2	IN22K	FT	base	1e-3	17.22
ConvNextV2	IN22K	FT	base	1e-5	19.82
ConvNextV2	IN22K	FT	AugMix	1e-5	11.98
ConvNextV2 CH-base-1e-3-e20	IN22K	CH-FT	AugMix	1e-5	25.43
DeiT	IN1K	CH	base	1e+1	59.21
DeiT	IN1K	CH	base	1e-1	59.50
DeiT	IN1K	CH	base	1e-3	75.13
DeiT	IN1K	FT	base	1e-1	12.32
DeiT	IN1K	FT	base	1e-3	21.00
DeiT	IN1K	FT	base	1e-5	69.34
DeiT	IN1K	FT	AugMix	1e-5	70.52
DeiT CH-base-1e-3-e20	IN1K	CH-FT	AugMix	1e-5	69.41
DeiT CH-base-1e-3-e1	IN1K	CH-FT	AugMix	1e-5	74.12

Table 2. Acc@1 in % on target domain (real images) for all source-domain-only training experiments on the Topex-Printer dataset. Note that *base* transform means that random color jitter and random grayscale transforms are applied. Faded out rows are representing numerically instable runs that have been canceled due to NaN loss for example.

Model	Pre-training	train scheme	transform	lr	Acc@1
ViT-b	IN22K	CH	base	1e+1	34.85
ViT-b	IN22K	CH	base	1e-1	40.69
ViT-b	IN22K	CH	base	1e-3	31.78
ViT-b	IN22K	FT	AugMix	1e-1	01.74
ViT-b	IN22K	FT	AugMix	1e-3	21.75
ViT-b	IN22K	FT	AugMix	1e-5	32.54
ViT-b	IN22K	CH-FT	AugMix	1e-5	45.90
SwinV2	IN22K	CH	base	1e+1	42.34
SwinV2	IN22K	CH	base	1e-1	45.15
SwinV2	IN22K	CH	base	1e-3	51.79
SwinV2	IN22K	FT	AugMix	1e-1	01.70
SwinV2	IN22K	FT	AugMix	1e-3	26.23
SwinV2	IN22K	FT	AugMix	1e-5	25.69
SwinV2	IN22K	CH-FT	AugMix	1e-3	51.79
SwinV2	**IN22K**	**CH-FT**	**AugMix**	**1e-5**	**59.21**

Table 3. Acc@1 in % on target domain (real images) for best results per model and training scheme in our source domain training experiments on VisDA-2017 classification dataset. Note that *base* transform means that random color jitter and random grayscale transforms are applied instead of AugMix (other augmentations stay the same as explained in section A.1).

Model	Pre-training	train scheme	transform	lr	Acc@1
ViT-b	IN22K	CH	base	1e-3	80.37
ViT-b	IN22K	FT	AugMix	1e-5	73.76
ViT-b	**IN22K**	**CH-FT**	**AugMix**	**1e-5**	**80.53**
SwinV2	IN22K	CH	base	1e-3	80.12
SwinV2	IN22K	FT	AugMix	1e-5	73.49
SwinV2	IN22K	CH-FT	AugMix	1e-5	76.96
DeiT	IN1K	CH	base	1e-3	75.13
DeiT	IN1K	FT	AugMix	1e-5	70.52
DeiT	IN1K	CH-FT	AugMix	1e-5	74.12

Table 4. Acc@1 in % on target domain (real images) for best results per model and training scheme in our source-domain-only training experiments on Topex-Printer dataset. Note that *base* transform means that random color jitter and random grayscale transforms are applied instead of AugMix (other augmentations stay the same as explained in section A.1).

Model	Pre-training	train scheme	transform	lr	Acc@1
ViT-b	IN22K	CH	base	1e-1	40.69
ViT-b	IN22K	FT	AugMix	1e-5	32.54
ViT-b	IN22K	CH-FT	AugMix	1e-5	45.90
SwinV2	IN22K	CH	base	1e-3	51.79
SwinV2	IN22K	FT	AugMix	1e-5	25.69
SwinV2	**IN22K**	**CH-FT**	**AugMix**	**1e-5**	**59.21**

Table 5. Acc@1 in % on target domain (real images) for all UDA experiments on VisDA-2017 classification dataset. Note that *init checkpoint* describes the model checkpoint used for the UDA experiments. CH refers to the best-performing CH training scheme from our DG experiments respecting the used model architecture and IN22K refers to the respective Huggingface model checkpoints described in section A.2.

Model	DA method	init checkpoint	Acc@1
ViT-b	CDAN	IN22K	61.96
ViT-b	CDAN	CH	88.78
ViT-b	MCC	IN22K	79.63
ViT-b	MCC	CH	88.88
ViT-b	CDAN-MCC	IN22K	75.26
ViT-b	CDAN-MCC	CH	89.38
SwinV2	CDAN	IN22K	71.21
SwinV2	CDAN	CH	80.12
SwinV2	MCC	IN22K	90.65
SwinV2	MCC	CH	91.88
SwinV2	CDAN-MCC	IN22K	91.99
SwinV2	**CDAN-MCC**	**CH**	**93.47**

Table 6. Acc@1 in % on target domain (real images) for all UDA experiments on the Topex-Printer dataset. Note that *init checkpoint* describes the model checkpoint used for the UDA experiments. *CH* refers to the best-performing CH training scheme from our source-domain-only training experiments respecting the used model architecture and IN22K refers to the respective Huggingface model checkpoints described in section A.2.

Model	DA method	init checkpoint	Acc@1
ViT-b	CDAN	IN22K	43.31
ViT-b	CDAN	CH	47.51
ViT-b	MCC	IN22K	32.95
ViT-b	MCC	CH	61.36
ViT-b	CDAN-MCC	IN22K	43.33
ViT-b	CDAN-MCC	CH	61.08
SwinV2	CDAN	IN22K	65.51
SwinV2	CDAN	CH	61.94
SwinV2	MCC	IN22K	72.86
SwinV2	MCC	CH	71.14
SwinV2	CDAN-MCC	IN22K	73.74
SwinV2	**CDAN-MCC**	**CH**	**74.86**

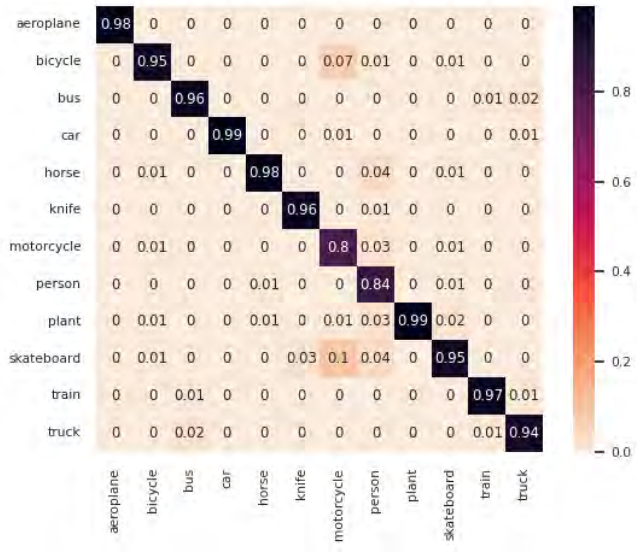

Fig. 4. Confusion matrix for our best-performing model on VisDA-2017: SwinV2-CH-CDAN-MCC

Table 7. Image classification top-1 accuracy in % on VisDA-2017 target domain (real images) across all classes compared to literature. We report our best source-domain-only and UDA runs for the ViT and SwinV2 architecture.

Method		Pl	Bcl	Bus	Car	Hrs	Knf	Mcy	Per	Plt	Skb	Trn	Tck	Mean
CDAN [10]	ResNet	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.9
MCC [6]	ResNet	88.1	80.3	80.5	71.5	90.1	93.2	85.0	71.6	89.4	73.8	85.0	36.9	78.8
SDAT [12]	ResNet	95.8	85.5	76.9	69.0	93.5	97.4	88.5	78.2	93.1	91.6	86.3	55.3	84.3
MIC [4]	ResNet	96.7	88.5	84.2	74.3	96.0	96.3	90.2	81.2	94.3	95.4	88.9	56.6	86.9
TVT [16]	ViT	92.9	85.6	77.5	60.5	93.6	98.2	89.3	76.4	93.6	92.0	91.7	55.7	83.9
CDTRANS [15]	ViT	97.1	90.5	82.4	77.5	96.6	96.1	93.6	88.6	97.9	86.9	90.3	62.8	88.4
SDAT [12]	ViT	98.4	90.9	85.4	82.1	98.5	97.6	96.3	86.1	96.2	96.7	92.9	56.8	89.8
MIC [4]	ViT	**99.0**	93.3	86.5	87.6	**98.9**	**99.0**	**97.2**	**89.8**	98.9	**98.9**	96.5	68.0	92.8
Source Only	ViT	96.48	71.82	90.14	**99.20**	94.66	77.71	87.28	44.45	95.12	83.64	94.05	40.76	80.54
Ours	ViT	94.82	93.49	92.80	95.89	90.95	88.51	77.46	75.42	96.27	97.32	94.74	88.03	89.38
Source Only	Swin	97.09	80.48	85.35	98.12	92.39	83.54	94.85	19.89	89.13	78.89	**97.03**	55.18	80.12
Ours	Swin	97.96	**95.15**	**95.81**	98.64	98.34	95.68	80.12	83.87	**99.39**	94.68	96.61	**93.85**	**93.47**

References

1. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. CVPR pp. 248–255 (2009)
2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
3. Hendrycks, D., Mu, N., Cubuk, E.D., Zoph, B., Gilmer, J., Lakshminarayanan, B.: Augmix: A simple data processing method to improve robustness and uncertainty. In: ICLR (2019)
4. Hoyer, L., Dai, D., Wang, H., Van Gool, L.: Mic: Masked image consistency for context-enhanced domain adaptation. In: CVPR. pp. 11721–11732 (2023)
5. Jiang, J., Shu, Y., Wang, J., Long, M.: Transferability in deep learning: A survey. ArXiv **abs/2201.05867** (2022)
6. Jin, Y., Wang, X., Long, M., Wang, J.: Minimum class confusion for versatile domain adaptation. In: ECCV (2019)
7. Junguang Jiang, Baixu Chen, B.F.M.L.: Transfer-learning-library. `https://github.com/thuml/Transfer-Learning-Library` (2020)
8. Kumar, A., Raghunathan, A., Jones, R.M., Ma, T., Liang, P.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: ICLR (2022)
9. Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., Wei, F., Guo, B.: Swin transformer v2: Scaling up capacity and resolution. CVPR pp. 11999–12009 (2021)
10. Long, M., Cao, Z., Wang, J., Jordan, M.I.: Conditional adversarial domain adaptation. In: NeurIPS (2017)
11. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv: Learning (2016)
12. Rangwani, H., Aithal, S.K., Mishra, M., Jain, A., Babu, R.V.: A closer look at smoothness in domain adversarial training. In: ICML (2022)
13. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jegou, H.: Training data-efficient image transformers amp; distillation through attention. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. vol. 139, pp. 10347–10357. PMLR (18–24 Jul 2021)
14. Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I.S., Xie, S.: Convnext v2: Co-designing and scaling convnets with masked autoencoders. ArXiv **abs/2301.00808** (2023)
15. Xu, T., Chen, W., Pichao, W., Wang, F., Li, H., Jin, R.: Cdtrans: Cross-domain transformer for unsupervised domain adaptation. In: ICLR (2021)
16. Yang, J., Liu, J., Xu, N., Huang, J.: Tvt: Transferable vision transformer for unsupervised domain adaptation. In: WACV. pp. 520–530 (2021)