

# AdaGraph: Unifying Predictive and Continuous Domain Adaptation through Graphs

## Supplementary Material

Massimiliano Mancini<sup>1,2</sup>, Samuel Rota Bulò<sup>3</sup>, Barbara Caputo<sup>4,5</sup>, Elisa Ricci<sup>2,6</sup>

<sup>1</sup>Sapienza University of Rome, <sup>2</sup>Fondazione Bruno Kessler, <sup>3</sup>Mapillary Research,

<sup>4</sup>Politecnico di Torino, <sup>5</sup>Italian Institute of Technology, <sup>6</sup>University of Trento

mancini@diag.uniroma1.it,samuel@mapillary.com,barbara.caputo@polito.it,eliricci@fbk.eu

### 1. Metadata Details

**CompCars.** For the experiments with the CompCars dataset [2], we have two domain information: the car production year and the viewpoint. We encode the metadata through a 2-dimensional integer vector where the first integer encodes the year of production (between 2009 and 2014) and the second the viewpoint. While encoding the production year is straightforward, for the viewpoint we use the same criterion adopted in [3], *i.e.* we encode the viewpoint through integers between 1-5 in the order: *Front, Front-Side, Side, Rear-Side, Rear*.

**Portraits.** For the experiments with the Portraits dataset [1], we have again two domain information: the year and the region where the picture has been taken. To allow for a bit more precise geographical information we encode the metadata through a 3-dimensional integer vector.

As for the CompCars dataset, the first integer encodes the decade of the image (8 decades between 1934 and 2014), while the second and third the geographical position. For the geographical position we simplify the representation through a coarse encoding involving 2 directions: est-west (from 0 to 1) and north-south (from 0 to 3). In particular we assign the following value pairs ([north-south, east-west]): *Mid-Atlantic*  $\rightarrow$  [0, 1], *Midwestern*  $\rightarrow$  [0, 2], *New England*  $\rightarrow$  [0, 0], *Pacific*  $\rightarrow$  [0, 3] and *Southern*  $\rightarrow$  [1, 1]. Each component of the vector has been normalized in the range 0-1.

### 2. Additional Analysis

#### 2.1. ResNet-18 on CompCars

Here we apply *AdaGraph* to the ResNet-18 architecture in the CompCars dataset [2]. As for the other experiments, we apply *AdaGraph* by replacing each BN layer of the network with its GBN counterpart.

Table 1: CompCars dataset [2]. Results with ResNet-18 architecture.

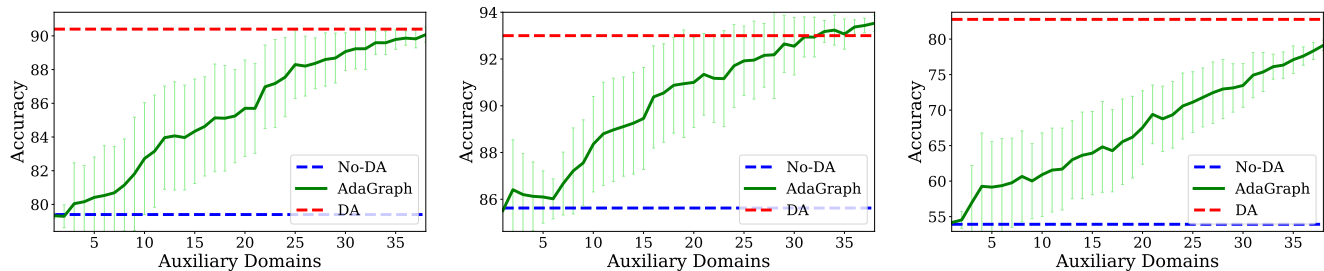
Method	Avg. Accuracy
Baseline	56.8
AdaGraph	<b>65.1</b>
Baseline + Refinement	65.3
AdaGraph + Refinement	<b>66.7</b>
DA upper bound	66.9

The network is initialized with the weights of the model pretrained on ImageNet. We train the network for 6 epochs on the source dataset, employing Adam as optimizer with a weight decay of  $10^{-6}$  and a batch-size of 16. The learning rate is set to  $10^{-3}$  for the classifier and  $10^{-4}$  for the rest of the network and it is decayed by a factor of 10 after 4 epochs. We extract domain-specific parameters by training the network for 1 epoch on the union of source and auxiliary domains, keeping the same optimizer and hyperparameters. The batch size is kept to 16, building each batch with elements of a single pair production year-viewpoint belonging to one of the domains available during training (either auxiliary or source).

The results are shown in Table 1. As the table shows, *AdaGraph* largely increases the performance of the *Baseline* model. Coherently with previous experiments, our refinement strategy is able to further increase the performances of *AdaGraph*, filling almost entirely the gap with the DA upper bound.

#### 2.2. Performances vs Number of Auxiliary Domains

In this section, we analyze the impact of varying the number of available auxiliary domains on the performances of our model. We employ the ResNet-18 architecture on the Portraits dataset, with the same setting and set of hyper-



(a) From 1954 Mid-Atlantic to 1984 Pacific. (b) From 2004 Midwestern to 1944 South-ern. (c) From 1974 Mid-Atlantic to 1994 New England.

Figure 1: Portraits dataset: performances of *AdaGraph* with respect to the number of auxiliary domains available for different source-target pairs. The years reported in the captions indicate the starting year of source and target decades.

parameters described in the experimental section. However, differently from the previous experiments, we vary the number of available auxiliary domains, from 1 to 38. We repeat the experiments 20 times, randomly sampling the available auxiliary domains each time.

The results are shown in Figure 1. As expected, increasing the number of auxiliary domains leads to an increase in the performance of the model. In general, as we have more than 20 domains available, the performance of our model are close to the DA upper bound. While these results obviously depend on the relatedness between the auxiliary domains and the target, the plots show that having a large set of auxiliary domains may not be strictly necessary for achieving good performances.

## References

- [1] Shiry Ginosar, Kate Rakelly, Sarah Sachs, Brian Yin, and Alexei A Efros. A century of portraits: A visual historical record of american high school yearbooks. In *ICCV-WS*, 2015. 1
- [2] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *CVPR*, 2015. 1
- [3] Yongxin Yang and Timothy M Hospedales. Multivariate regression on the grassmannian for predicting novel domains. In *CVPR*, 2016. 1