Improving Predictors via Combination Across Diverse Task Categories – Supplemental Document

Kwang In Kim [1]

In this supplementary document, we present 1) details of the predictor combination process for the given reference evaluations aligned with the target (Eq. 18 of the main paper; Sec. 1); 2) the complete results of our experiments (Sec. 2); and 3) examples of image translation via cycle-consistent translation networks (Zhu et al., 2017) as a potential alternative to our chart map learning approach (Sec. 3). Some content from the main paper has been reproduced to make this document self-contained.

1. Predictor combination process given aligned references

Suppose that we are given a sample set $X^f = \{\mathbf{x}_1^f, \dots, \mathbf{x}_N^f\}$ and the evaluation of the initial target predictor f^I on X^f: $\mathbf{f}^I = [f^I(\mathbf{x}_1^f), \dots, f^I(\mathbf{x}_N^f)]^\top$. Using the chart maps $\{q^{fr}\}_{r=1}^R$ (Sec. 2.2.1 of the main paper), we can construct the corresponding reference evaluations aligned with the target: $G = \{\mathbf{g}^r\}_{r=1}^R$ with $\mathbf{g}^r = [g^r((q^{fr})^{-1}(\mathbf{x}_1^f)), \dots, g^r((q^{fr})^{-1}(\mathbf{x}_N^f))]^\top$.

Our algorithm improves $\mathbf{f}^I$ by applying an iterative averaging process: Given the solution $\mathbf{f}^t$ at time t, the new solution $\mathbf{f}^{t+1}$ is obtained by maximizing

$$\widehat{\mathcal{O}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{f}^t + \lambda \mathbf{f}^\top \mathbf{Q}_G \mathbf{f} \tag{1}$$

for a positive definite matrix $\mathbf{Q}_G$ and a hyperparameter λ. Adopting the framework of Kim et al. (Kim et al., 2020), we define $\mathbf{Q}_G$ such that $\mathbf{f}^\top \mathbf{Q}_G \mathbf{f}$ becomes the accuracy of predicting $\mathbf{f}$ based on Gaussian process (GP) regression using $\mathbf{G} = [\mathbf{g}^1, \dots, \mathbf{g}^R]$ as the input data matrix (Kim et al., 2020).

First, we build a kernel matrix $\mathbf{H}$ of references $\mathbf{G}$:

$$[\mathbf{H}]_{m,n} = h_G(\mathbf{G}_{[m,:]}^\top, \mathbf{G}_{[n,:]}^\top) \tag{2}$$

where h_G is an anisotropic Gaussian kernel

$$h_G(\mathbf{p}, \mathbf{p}') = \exp\left(-(\mathbf{p} - \mathbf{p}')^\top \mathbf{\Sigma} (\mathbf{p} - \mathbf{p}')\right), \tag{3}$$

$\mathbf{G}_{[i,:]}$ denotes the i-th row of $\mathbf{G}$, and $\mathbf{\Sigma}$ is a diagonal matrix of positive entries optimized by maximizing the regularized marginal likelihood $p(\mathbf{f}|\mathbf{G}, \mathbf{\Sigma})$ (Eq. 20 of the main paper).

Now, the Gaussian process regression function $m(\cdot)$[1] that estimates $\mathbf{f}$ based on $\mathbf{G}$ is obtained as follows (Rasmussen & Williams, 2006)

$$m(\cdot) = \sum_{n=1}^{N} [\mathbf{a}]_n h_G(\mathbf{G}_{[n,:]}^\top, \cdot), \tag{4}$$

where the coefficient vector $\mathbf{a}$ can be calculated as

$$\mathbf{a} = (\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \mathbf{f}, \tag{5}$$

with a regularization hyperparameter λ^{GP} (or equivalently, the noise variance).

Using this model, the prediction accuracy of $m(\cdot)$ on $\mathbf{f}$, normalized by the variance of the $\mathbf{f}$ entries is calculated as:

$$\begin{aligned}
\mathcal{O}^{\text{GP}}(m) =& 1 - \frac{\text{mean squared error}(m(\mathbf{G}), \mathbf{f})}{\text{variance}(\mathbf{f})}, \\
=& 1 - \frac{\sum_{n=1}^N ([\mathbf{f}]_n - m(\mathbf{G}_{[n,:]}^\top))^2}{\sum_{n=1}^N ([\mathbf{f}]_n - \sum_{l=1}^N [\mathbf{f}]_l / N)^2}, \\
=& 1 - \frac{\mathbf{f}^\top \mathbf{f}}{\mathbf{f}^\top \mathbf{C} \mathbf{f}} + \frac{\mathbf{f}^\top 2\mathbf{H}(\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \mathbf{f}}{\mathbf{f}^\top \mathbf{C} \mathbf{f}}, \\
& - \frac{\mathbf{f}^\top (\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \mathbf{H} \mathbf{H} (\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \mathbf{f}}{\mathbf{f}^\top \mathbf{C} \mathbf{f}},
\end{aligned}$$

with $\mathbf{C} = \mathbf{I} - \frac{1}{N} \mathbf{1} \mathbf{1}^\top$ and $\mathbf{1} = [1, \dots, 1]^\top$.

The final objective (Eq. 18 of the main paper) is obtained by rewriting $\mathcal{O}^{\text{GP}}$ as a function of $\mathbf{f}$, explicitly centering $\mathbf{f}$ (i.e. replacing $\mathbf{f}$ with $\mathbf{C}\mathbf{f}$), and replacing $\mathbf{f}^\top \mathbf{Q}_G \mathbf{f}$ in $\widehat{\mathcal{O}}$ (Eq. 1) with the resulting objective $\mathcal{O}^{\text{GP}}$ (Eq. 1):

$$\widehat{\mathcal{O}}(\mathbf{f}) = \frac{\mathbf{f}^\top \mathbf{C} \mathbf{A} \mathbf{C} \mathbf{f}}{\mathbf{f}^\top \mathbf{C} \mathbf{f}}, \tag{6}$$

where

$$\begin{aligned}
\mathbf{A} =& \mathbf{f}^t (\mathbf{f}^t)^\top + \lambda \mathbf{S}, \\
\mathbf{S} =& 2\mathbf{H}(\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \\
& - (\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1} \mathbf{H} \mathbf{H} (\mathbf{H} + \lambda^{\text{GP}} \mathbf{I})^{-1}.
\end{aligned}$$

The maximizer of $\widehat{\mathcal{O}}$ is the first eigenvector (the eigenvector corresponding to the largest eigenvalue) of $\mathbf{A} \in \mathbb{R}^{N \times N}$. For

[1] UNIST, Ulsan, Korea. Correspondence to: Kwang In Kim <kimki@unist.ac.kr>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

[1] More precisely, the mean of the predictive distribution under i.i.d. Gaussian noise.

large-scale problems ($N \gg 0$), calculating this eigenvector would be infeasible. A computationally efficient approximation is obtained by using a finite-rank approximation $\hat{h}_G$ of h_G similarly to Eq. 15 in the main paper:

$$\begin{aligned} \hat{h}_G(\mathbf{p},\mathbf{p}') &= \mathbf{h}_\mathbf{p}^\top \mathbf{H}_{BB}^{-1}\mathbf{h}_{\mathbf{p}'}, \\ \mathbf{h}_\mathbf{p} &= [h_G(\mathbf{p},\mathbf{b}_1),\dots,h_G(\mathbf{p},\mathbf{b}_{300})]^\top, \\ [\mathbf{H}_{BB}]_{m,n} &= h_G(\mathbf{b}_m,\mathbf{b}_n), \end{aligned} \tag{7}$$

where the basis set $B = \{\mathbf{b}_1,\dots,\mathbf{b}_{300}\}$ is constructed by linearly sampling 300 rows from all rows of $\mathbf{G}$. The resulting approximation of $\mathbf{H}$ is given as

$$\widehat{\mathbf{H}} \approx \mathbf{H}_{GB}\mathbf{H}_{BB}^{-1}\mathbf{H}_{GB}^\top \tag{8}$$

with $\mathbf{H}_{GB}^\top = [\mathbf{h}_{\mathbf{G}_{[1,:]}^\top},\dots,\mathbf{h}_{\mathbf{G}_{[N,:]}^\top}]$ and $\mathbf{G}_{[i,:]}$ is the i-th row of $\mathbf{G}$. Now, replacing $\mathbf{H}$ with $\widehat{\mathbf{H}}$ in Eq. 6 we obtain $\mathbf{A} \approx \mathbf{Y}\mathbf{Y}^\top$, where

$$\begin{aligned} \mathbf{Y} &= \left[\mathbf{C}\mathbf{f}^t, \sqrt{\lambda}\mathbf{H}_{GB}\mathbf{T}^{\frac{1}{2}}\right] \in \mathbb{R}^{N\times 301} \\ \mathbf{T} &= 2\mathbf{P} - \mathbf{P}\mathbf{H}_{GB}^\top\mathbf{H}_{GB}\mathbf{P} \\ \mathbf{P} &= (\mathbf{H}_{GB}^\top\mathbf{H}_{GB} + \lambda\mathbf{H}_{BB})^{-1}. \end{aligned} \tag{9}$$

The square-root $\mathbf{T}^{\frac{1}{2}}$ of $\mathbf{T}$ can be obtained by performing the Cholesky decomposition of $\mathbf{T}$. Finally, the (approximate) first eigenvector of $\mathbf{A}$ can be calculated as $\mathbf{Y}\mathbf{e}$ with $\mathbf{e}$ being the first eigenvector of $\mathbf{Y}^\top\mathbf{Y} \in \mathbb{R}^{301\times 301}$.

2. Complete ranking results

In the main paper, we compared our PC algorithm with three alternative design choices that use different chart map learning strategies: *Id* (identity maps), *IT* (image translation), and *MMD* (MMD only). Here, we also compare with a parametric adaptation (*Param*) of Kim et al.'s PC algorithm (Kim et al., 2020) and (Mejjati et al., 2018)'s non-parametric multi-task learning (*MTL*) algorithm.

Our *Param* adaptation uses rank support vector machines for all predictors (Chapelle & Keerthi, 2010; Parikh & Grauman, 2011): The target and reference predictors are represented based on the corresponding parameters:

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{x}^\top\mathbf{w}^f, \\ g^r(\mathbf{x}) &= \mathbf{x}^\top\mathbf{w}^r \text{ for } 1 \le r \le R. \end{aligned}$$

This algorithm improves the parameter $\mathbf{w}^I$ of the initial predictor f^I by replacing $\mathbf{f}$ and $\{\mathbf{g}^r\}_{r=1}^R$ with $\mathbf{w}$ and $\{\mathbf{w}^r\}_{r=1}^R$, respectively in Eqs. 1 and 6, and maximizing the resulting objective $\widehat{\mathcal{O}}$. The interpretation of this algorithm is similar to that of the original nonparametric version (Kim et al., 2020): The reference g^r is considered *relevant* when its parameter $\mathbf{w}^r$ makes a significant contribution to predicting the target predictor parameter $\mathbf{w}^f$. We evaluated this model only on *AWA2*, *CUB*, *PubFig (ResNet)*, *Shoes (ResNet)*, *OSR (ResNet)* as it requires that all predictors are defined on the same input space: $\mathcal{X} = \mathcal{X}^f = \mathcal{X}^r$. Still, in this case, the corresponding probability distributions differ from each other.

***MTL* (Mejjati et al., 2018)** does not require access to the forms of individual predictors and therefore, it can be directly applied to the PC problems: For each target f, we performed joint MTL on f and the references $\{g_i\}_{i=1}^R$. Similarly to (Kim et al., 2020) as a nonparametric algorithm, *MTL* relies on a common dataset X on which all predictors are evaluated: We aligned the evaluations of all predictors using our MMD and HSIC-based data alignment scheme.

Tables 2-6 show the complete results. The results of statistical significance tests are summarized in Table 1. All four nonparametric PC algorithms (*Id*, *IT*, *MMD*, *Ours*) frequently achieved significant performance gain over the initial predictors $\mathbf{f}^I$, confirming the utility and possibility of predictor combination across heterogeneous task categories. Apart from one dataset (*AWA2*) where *Id* ranked second best (after our final algorithm; *Ours*), *IT* and *MMD* outperformed *Id* demonstrating that when the target and reference domains are significantly different, aligning the target data with the distributions of references can help extract *useful* evaluations from the references.

As demonstrated in the main paper (Fig. 1), using only the MMD can fail to preserve the local structure of the original target data. By explicitly addressing this with the additional HSIC regularizer, *Ours* showed further significant improvements. While in principle, image translation has the capability of matching images across different domains (e.g. matching aerial photographs to maps (Zhu et al., 2017)), we observed that training such translation networks is challenging when the two domains differ substantially as in our PC scenario (example translation results are shown in Sec. 3). As a result, *IT* recorded only a comparable level of performance to *MMD*. Further, our preliminary experiments suggested that the *IT* training process is prone to mode collapse, making the transfer networks generate identical images, requiring human intervention, e.g. to restart with a new initialization and to stop before mode collapse starts. The parametric adaptation of (Kim et al., 2020)'s PC algorithm *Param* did not show any noticeable improvement over $\mathbf{f}^I$ demonstrating that adapting PC algorithms to heterogeneous tasks is not straightforward. *MTL* achieved better performance improvement than *Param* but all PC algorithms demonstrated more significant improvements.

Overall, *Ours* statistically significantly improved $\mathbf{f}^I$, *Param*, *MTL*, *Id*, *IT*, and *MMD* for 71.84%, 70.75%, 74.71%, 60.54%, 62.59%, and 55.75% of the total target attributes, respectively as shown in Table 1. Importantly, *Ours* did not significantly degraded performance in any of the total 174 target attributes.

Table 1. Results of statistical significance tests of different PC methods compared to the initial predictors $\mathbf{f}^I$ (vs. $\mathbf{f}^I$) and our method (*Ours* vs.). For (vs. $\mathbf{f}^I$), we show # attributes where the respective algorithms are statistically significantly better (first column) or worse (second column) than $\mathbf{f}^I$. For (*Ours* vs.), # attributes where *Ours* is better or worse than the corresponding algorithms are shown.

Dataset	vs. $\mathbf{f}^I$												*Ours* vs.										# targets
	Param		*MTL*		*Id*		*IT*		*MMD*		*Ours*		*Param*		*MTL*		*Id*		*IT*		*MMD*		
AWA2	2	6	9	0	29	2	10	2	17	1	55	0	56	0	62	0	41	0	53	0	48	0	80
CUB	0	2	3	0	4	2	0	3	4	1	23	0	24	0	20	0	24	0	22	0	20	0	40
PubFig (ResNet)	0	0	1	1	2	0	5	0	5	0	9	0	9	0	9	0	8	0	7	0	5	0	11
PubFig	*N/A*		1	1	*N/A*		*N/A*		0	1	9	0	*N/A*		9	0	*N/A*		*N/A*		8	0	11
Shoes (ResNet)	0	0	2	0	1	0	8	0	6	0	9	0	9	0	9	0	10	0	5	0	4	0	10
Shoes	*N/A*		2	0	*N/A*		*N/A*		3	0	8	0	*N/A*		9	0	*N/A*		*N/A*		7	0	10
OSR (ResNet)	0	2	1	0	3	0	5	0	6	2	6	0	6	0	6	0	6	0	5	0	3	0	6
OSR	*N/A*		1	0	*N/A*		*N/A*		4	0	6	0	*N/A*		6	0	*N/A*		*N/A*		2	0	6
Total (%)	1.36	6.80	11.49	1.15	26.53	2.72	19.05	3.40	25.86	2.87	71.84	0	70.75	0	74.71	0	60.54	0	62.59	0	55.75	0	174

3. Image translation examples

Image translation methods (e.g. cycle-consistent translation networks (Zhu et al., 2017); *IT*) provide a powerful mechanism to capture characteristics of image datasets and transfer such characteristics across different domains. These methods are especially effective in recovering *underlying connections* between different image collections, e.g. in converting an aerial photograph to a map, or a photograph of a place taken in summer and to the corresponding winter photograph (Fig. 1).

However, when such inherent connections do not exist, *IT* can establish spurious connections as exemplified in Fig. 2: The classes of objects appearing in *AWA2* and *CUB* differ significantly and in this case, *IT* characterized and transferred these images mainly based on their local brightness and texture. Similar results are also observed in the cases of transferring between *CUB* and *OSR*, and *AWA2* and *OSR*: Here, one domain (*OSR*) does not have any foreground object while the other domains *AWA2* and *CUB* contain specific foreground objects (animals and birds, respectively). The images in *Shoes* have distinct foreground objects on white backgrounds, and *IT* identified white color of the background as distinct characteristics: Translating *OSR* and *AWA2* images to *Shoes* selectively replaced pixels of source images with white pixels. On the other hand, mapping *Shoes* to other domains tended to keep the foreground shoes while filling in the backgrounds with certain texture patterns matching the respective target domains.

To summarize, we observed that when the source and target domains differ substantially, *IT* tends to find and focus on spurious image characteristics. While we do not claim that this is a fundamental limitation of image translation approaches, we take our findings as an indication that applying these techniques to predictor combination across heterogeneous task categories is not straightforward.

References

Chapelle, O. and Keerthi, S. S. Efficient algorithms for ranking with SVMs. *Information Retrieval*, 13(3):201–215, 2010.

Kim, K. I., Richardt, C., and Chang, H. J. Combining task predictors via enhancing joint predictability. In *ECCV*, 2020.

Mejjati, Y. A., Cosker, D., and Kim, K. I. Multi-task learning by maximizing statistical dependence. In *CVPR*, pp. 3465–3473, 2018.

Parikh, D. and Grauman, K. Relative attributes. In *ICCV*, pp. 503–510, 2011.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

Table 2. Accuracy of different PC and MTL algorithms on *AWA2* (first 50 attributes): 1) a parametric adaptation of (Kim et al., 2020)'s algorithm (*Param*), 2) (Mejjati et al., 2018)'s non-parametric multi-task learning algorithm that uses our data alignment method (*MTL*), and 3-6) PC algorithms that use identity chart maps (*Id*), image translation networks (*IT*), *MMD*, and a combination of MMD and HSIC (*Ours*), respectively. For the initial predictors $\mathbf{f}^I$, 100×Kendall's τ coefficients are shown, while for the PC algorithms, the accuracy increases from $\mathbf{f}^I$ are presented. The numbers in parentheses show standard deviations. For each target attribute, the best and second best results are highlighted in **boldface** and with underline, respectively. The results of statistical significance tests of each PC method compared to $\mathbf{f}^I$ are presented in three categories: significantly better, significantly worse, and insignificantly different.

Target	$\mathbf{f}^I$	*Param*	*MTL*	*Id*	*IT*	*MMD*	*Ours*
1	76.99 (3.04)	0.20 (1.32)	0.15 (0.34)	1.77 (1.60)	0.59 (1.19)	1.80 (1.29)	**5.30 (2.01)**
2	84.95 (2.91)	0.20 (0.77)	0.05 (0.33)	0.73 (2.16)	0.48 (1.10)	0.11 (1.46)	**1.82 (1.91)**
3	98.33 (0.85)	-0.03 (0.33)	**0.03 (0.06)**	-0.08 (0.28)	-0.18 (0.28)	0.00 (0.23)	-0.03 (0.54)
4	86.19 (2.87)	-0.83 (0.71)	0.10 (0.36)	1.35 (1.33)	0.12 (0.84)	0.19 (2.73)	**3.20 (1.46)**
5	87.19 (3.01)	-0.14 (0.99)	-0.07 (0.29)	0.75 (1.24)	0.32 (1.00)	1.00 (0.93)	**2.42 (1.62)**
6	97.80 (1.28)	-0.04 (0.35)	-0.04 (0.09)	-0.12 (0.17)	-0.14 (0.19)	-0.16 (0.29)	**0.15 (0.71)**
7	98.23 (1.27)	0.01 (0.32)	-0.02 (0.03)	-0.03 (0.27)	-0.23 (0.82)	-0.09 (0.19)	**0.09 (0.53)**
8	82.02 (2.85)	-0.85 (1.30)	0.25 (0.40)	0.79 (1.64)	0.44 (0.94)	0.68 (1.06)	**3.54 (1.49)**
9	78.82 (3.94)	0.67 (1.57)	0.04 (0.42)	1.18 (1.66)	-0.22 (1.09)	-0.23 (2.94)	**2.26 (2.20)**
10	98.82 (0.60)	-0.14 (0.44)	0.00 (0.04)	-0.08 (0.20)	-0.34 (0.46)	-0.79 (0.90)	**0.06 (0.37)**
11	96.38 (1.22)	0.24 (0.50)	0.13 (0.35)	0.28 (0.40)	-0.12 (0.43)	0.22 (0.45)	**0.71 (0.86)**
12	93.92 (2.03)	0.06 (0.53)	0.03 (0.10)	0.20 (0.34)	-0.05 (0.55)	0.03 (0.65)	**0.86 (0.41)**
13	94.22 (1.80)	-0.06 (0.48)	0.13 (0.24)	0.33 (0.75)	0.35 (0.41)	0.05 (0.43)	**0.47 (1.59)**
14	94.45 (1.51)	-0.31 (0.37)	-0.03 (0.25)	0.17 (0.49)	-0.34 (0.75)	0.39 (0.51)	**0.91 (0.92)**
15	93.83 (1.45)	-0.24 (0.73)	0.16 (0.22)	0.46 (0.34)	-0.16 (0.38)	0.33 (0.34)	**0.95 (1.24)**
16	85.50 (1.60)	-0.43 (1.21)	0.20 (0.20)	1.48 (1.06)	0.02 (1.00)	0.29 (1.27)	**2.88 (1.43)**
17	86.82 (2.20)	0.07 (0.83)	0.21 (0.46)	0.97 (0.84)	0.40 (0.52)	1.08 (0.89)	**1.84 (2.15)**
18	98.98 (0.52)	0.11 (0.19)	0.03 (0.06)	-0.05 (0.26)	-0.02 (0.12)	-0.02 (0.33)	**0.11 (0.48)**
19	99.49 (0.48)	**0.01 (0.05)**	-0.01 (0.04)	-0.00 (0.05)	-0.04 (0.07)	-0.64 (1.74)	-0.07 (0.34)
20	97.74 (0.73)	-0.02 (0.28)	0.07 (0.35)	0.12 (0.42)	-0.02 (0.32)	**0.17 (0.53)**	0.13 (0.99)
21	92.97 (1.80)	-0.54 (0.71)	0.21 (0.25)	0.25 (0.58)	0.29 (0.56)	0.12 (1.02)	**1.54 (0.80)**
22	96.36 (1.24)	-0.06 (0.26)	0.12 (0.20)	0.46 (0.46)	-0.11 (0.67)	0.33 (0.51)	**0.59 (1.11)**
23	91.35 (2.38)	0.02 (0.41)	0.05 (0.26)	0.73 (0.84)	-0.15 (0.96)	-0.03 (1.79)	**1.48 (1.55)**
24	93.87 (1.55)	-0.14 (0.52)	0.16 (0.35)	-0.40 (0.46)	-0.65 (1.15)	0.41 (0.89)	**1.10 (1.29)**
25	86.73 (3.68)	-0.12 (0.93)	0.43 (0.93)	**1.04 (1.17)**	0.52 (0.77)	-0.49 (1.91)	0.87 (1.25)
26	80.87 (3.71)	0.15 (0.68)	-0.43 (1.43)	1.61 (1.58)	0.85 (0.81)	0.35 (1.31)	**3.89 (1.61)**
27	90.46 (1.91)	-0.18 (0.77)	0.07 (0.37)	1.27 (0.92)	0.34 (0.71)	-0.25 (1.06)	**1.60 (0.95)**
28	89.60 (3.77)	0.21 (0.29)	-0.05 (0.13)	0.65 (0.72)	0.32 (0.90)	1.20 (0.66)	**3.33 (1.45)**
29	95.24 (1.20)	-0.14 (0.99)	-0.00 (0.17)	-0.09 (0.51)	0.07 (0.21)	-0.50 (1.21)	**0.34 (0.80)**
30	95.68 (1.44)	-0.05 (0.36)	0.04 (0.17)	-0.15 (0.54)	-0.15 (0.37)	**0.27 (0.40)**	0.23 (0.80)
31	93.98 (1.72)	-0.14 (0.44)	0.16 (0.24)	0.60 (0.42)	0.09 (0.77)	0.12 (0.63)	**1.08 (0.88)**
32	96.19 (1.38)	-0.15 (0.40)	-0.00 (0.20)	**0.08 (0.34)**	-0.03 (0.22)	0.02 (0.63)	0.04 (0.51)
33	82.69 (3.47)	0.82 (0.88)	0.40 (0.68)	1.85 (2.06)	0.58 (1.83)	0.89 (1.82)	**3.96 (1.70)**
34	98.88 (0.68)	-0.24 (0.41)	0.02 (0.13)	-0.01 (0.34)	**0.06 (0.24)**	-0.09 (0.64)	-0.06 (0.62)
35	98.72 (0.39)	**0.11 (0.07)**	-0.02 (0.14)	0.00 (0.24)	-0.04 (0.18)	-0.04 (0.24)	0.05 (0.26)
36	97.04 (0.89)	-0.19 (0.48)	0.03 (0.06)	-0.05 (0.15)	-0.08 (0.17)	**0.08 (0.21)**	-0.07 (0.26)
37	97.10 (0.73)	-0.01 (0.35)	0.04 (0.14)	-0.03 (0.28)	-0.26 (0.64)	-0.44 (0.86)	**0.21 (0.57)**
38	94.92 (1.28)	-0.05 (0.15)	-0.01 (0.12)	0.17 (0.34)	0.05 (0.37)	0.01 (0.27)	**0.72 (0.44)**
39	87.19 (2.47)	-0.13 (0.49)	0.25 (0.26)	1.36 (1.74)	0.93 (0.60)	2.17 (0.87)	**3.44 (1.26)**
40	93.86 (2.58)	-0.13 (0.31)	0.06 (0.27)	0.25 (0.43)	-0.00 (0.54)	-0.08 (0.90)	**1.23 (1.07)**
41	92.27 (1.57)	-0.13 (0.36)	0.02 (0.15)	0.12 (0.49)	-0.12 (0.38)	-0.14 (0.65)	**0.29 (0.88)**
42	84.52 (4.06)	-0.19 (1.00)	0.12 (0.36)	**0.96 (0.98)**	0.54 (0.78)	-0.31 (1.86)	0.87 (2.19)
43	95.56 (1.35)	-0.28 (0.87)	-0.07 (0.42)	0.06 (0.50)	-0.06 (0.39)	0.36 (0.60)	**0.87 (0.76)**
44	98.54 (0.56)	-0.04 (0.24)	0.01 (0.09)	-0.14 (0.21)	-0.12 (0.22)	-0.59 (1.45)	**0.05 (0.31)**
45	89.79 (2.22)	-0.35 (0.76)	0.10 (0.19)	1.06 (0.58)	0.38 (0.97)	1.52 (1.04)	**2.43 (1.83)**
46	82.61 (3.51)	-0.38 (1.16)	0.06 (0.26)	2.12 (1.69)	0.70 (1.11)	2.99 (1.70)	**5.40 (1.75)**
47	85.90 (3.20)	0.53 (1.00)	0.09 (0.21)	0.30 (1.01)	0.05 (0.58)	0.90 (0.82)	**2.76 (1.96)**
48	89.68 (2.79)	-0.31 (0.45)	0.64 (1.19)	0.17 (1.21)	0.17 (0.66)	0.89 (1.33)	**2.30 (2.36)**
49	87.24 (2.13)	-0.02 (0.46)	0.08 (0.19)	0.77 (1.15)	0.21 (0.75)	0.01 (2.50)	**2.68 (1.94)**
50	93.94 (2.22)	0.08 (0.62)	0.05 (0.18)	0.36 (0.83)	0.07 (0.87)	0.27 (0.50)	**0.99 (0.89)**

Table 3. Accuracy of different PC and MTL algorithms on *AWA2* (last 30 attributes): 1) a parametric adaptation of (Kim et al., 2020)'s algorithm (*Param*), 2) (Mejjati et al., 2018)'s non-parametric multi-task learning algorithm that uses our data alignment method (*MTL*), and 3-6) PC algorithms that use identity chart maps (*Id*), image translation networks (*IT*), *MMD*, and a combination of MMD and HSIC (*Ours*), respectively. For the initial predictors $\mathbf{f}^I$, 100×Kendall's τ coefficients are shown, while for the PC algorithms, the accuracy increases from $\mathbf{f}^I$ are presented. The numbers in parentheses show standard deviations. For each target attribute, the best and second best results are highlighted in **boldface** and with underline, respectively. The results of statistical significance tests of each PC method compared to $\mathbf{f}^I$ are presented in three categories: significantly better, significantly worse, and insignificantly different.

Target	$\mathbf{f}^I$	*Param*	*MTL*	*Id*	*IT*	*MMD*	*Ours*
51	89.51 (2.71)	0.24 (0.48)	0.31 (0.33)	0.91 (1.15)	0.78 (0.87)	0.27 (0.99)	**1.00 (1.06)**
52	93.10 (2.48)	-0.12 (0.63)	0.10 (0.33)	0.59 (0.37)	0.29 (0.38)	0.78 (0.99)	**1.26 (1.36)**
53	**93.47 (1.59)**	-0.18 (0.91)	-0.01 (0.09)	-0.08 (0.30)	-0.30 (0.63)	-0.26 (0.54)	-0.01 (0.48)
54	89.05 (2.44)	-0.11 (0.84)	0.06 (0.14)	1.16 (0.92)	0.18 (0.77)	0.71 (2.08)	**2.90 (1.66)**
55	94.14 (2.52)	0.14 (0.84)	0.00 (0.06)	0.74 (0.72)	0.32 (0.52)	0.19 (0.74)	**0.81 (0.75)**
56	87.93 (3.19)	-0.09 (0.47)	0.15 (0.35)	1.58 (1.06)	0.78 (0.65)	0.84 (1.16)	**2.50 (1.28)**
57	91.21 (1.96)	-0.43 (0.67)	0.13 (0.22)	-0.20 (0.71)	-0.20 (0.73)	-0.59 (1.40)	**1.00 (0.53)**
58	95.04 (1.60)	0.16 (0.24)	0.13 (0.19)	-0.10 (1.13)	-0.10 (0.66)	0.25 (0.30)	**0.70 (0.96)**
59	92.79 (2.01)	-0.35 (0.79)	0.02 (0.24)	0.25 (0.44)	-0.23 (0.67)	0.26 (0.54)	**0.56 (0.66)**
60	90.79 (1.81)	-0.59 (0.40)	0.05 (0.20)	0.31 (0.81)	0.41 (0.72)	0.18 (0.22)	**1.38 (1.16)**
61	90.73 (3.36)	-0.23 (0.74)	-0.04 (0.18)	0.24 (0.43)	-0.22 (0.59)	0.05 (0.55)	**1.19 (1.13)**
62	97.71 (1.31)	-0.01 (0.40)	0.04 (0.14)	-0.07 (0.19)	-0.20 (0.42)	-0.42 (0.78)	**0.09 (0.38)**
63	91.90 (2.23)	-1.04 (0.77)	0.12 (0.15)	0.51 (0.68)	0.10 (0.76)	0.36 (1.19)	**2.21 (1.13)**
64	90.49 (2.17)	0.30 (0.83)	0.13 (0.43)	0.66 (0.79)	-0.39 (0.67)	-0.61 (2.18)	**1.43 (1.30)**
65	90.03 (1.87)	-0.44 (0.93)	0.04 (0.23)	**1.09 (0.89)**	0.12 (0.64)	0.64 (1.04)	0.74 (1.59)
66	91.75 (2.00)	-0.06 (0.31)	0.07 (0.20)	0.38 (1.29)	0.03 (0.83)	0.54 (0.90)	**1.98 (1.21)**
67	94.03 (1.74)	-0.34 (0.52)	0.05 (0.07)	0.05 (0.66)	-0.02 (0.50)	0.14 (0.52)	**0.56 (0.73)**
68	88.67 (2.04)	-1.10 (2.55)	0.09 (0.33)	0.74 (0.71)	-0.14 (0.82)	1.01 (0.91)	**2.67 (0.76)**
69	99.00 (0.62)	0.02 (0.11)	0.05 (0.11)	0.06 (0.25)	0.03 (0.13)	-0.83 (2.46)	**0.13 (0.35)**
70	97.20 (0.86)	0.11 (0.27)	0.17 (0.24)	-0.16 (0.41)	-0.08 (0.16)	0.22 (0.38)	**0.38 (0.39)**
71	98.59 (0.69)	**0.08 (0.13)**	-0.10 (0.17)	-0.07 (0.29)	-0.08 (0.32)	-0.43 (0.83)	-0.02 (0.26)
72	96.42 (1.75)	-0.13 (0.53)	0.06 (0.13)	0.06 (0.28)	0.10 (0.24)	-0.29 (0.75)	**0.53 (0.57)**
73	91.56 (2.12)	0.03 (0.42)	-0.11 (0.27)	0.01 (0.47)	-0.07 (0.44)	-0.42 (0.98)	**0.80 (0.75)**
74	80.77 (2.96)	0.03 (0.67)	0.41 (1.21)	1.67 (2.24)	0.91 (1.07)	0.54 (1.89)	**3.46 (2.73)**
75	81.26 (1.97)	-0.07 (0.62)	0.43 (0.43)	**2.39 (0.82)**	1.19 (0.87)	0.13 (1.53)	2.21 (1.30)
76	81.83 (3.24)	0.28 (0.50)	-0.12 (0.67)	0.31 (1.98)	-0.02 (0.78)	0.86 (1.21)	**3.07 (2.11)**
77	85.51 (4.92)	-0.32 (0.52)	0.28 (0.54)	0.70 (1.15)	0.40 (0.86)	0.67 (1.67)	**2.32 (1.46)**
78	83.53 (4.57)	0.16 (1.25)	0.10 (0.51)	0.45 (1.03)	0.09 (1.55)	0.56 (1.30)	**2.40 (1.90)**
79	87.50 (2.96)	-0.05 (1.09)	0.08 (0.17)	0.42 (1.09)	0.05 (0.77)	1.15 (1.38)	**2.56 (1.67)**
80	85.63 (3.38)	-1.51 (1.12)	0.64 (0.59)	1.21 (1.17)	0.65 (0.54)	1.35 (1.44)	**2.19 (1.40)**

Table 4. Accuracy of different PC and MTL algorithms on *CUB*: 1) a parametric adaptation of (Kim et al., 2020)'s algorithm (*Param*), 2) (Mejjati et al., 2018)'s non-parametric multi-task learning algorithm that uses our data alignment method (*MTL*), and 3-6) PC algorithms that use identity chart maps (*Id*), image translation networks (*IT*), *MMD*, and a combination of MMD and HSIC (*Ours*), respectively. For the initial predictors $\mathbf{f}^I$, 100×Kendall's τ coefficients are shown, while for the PC algorithms, the accuracy increases from $\mathbf{f}^I$ are presented. The numbers in parentheses show standard deviations. For each target attribute, the best and second best results are highlighted in **boldface** and with underline, respectively. The results of statistical significance tests of each PC method compared to $\mathbf{f}^I$ are presented in three categories: significantly better, significantly worse, and insignificantly different.

Target	$\mathbf{f}^I$	*Param*	*MTL*	*Id*	*IT*	*MMD*	*Ours*
1	65.20 (4.90)	-0.38 (1.06)	-0.11 (0.58)	-0.64 (1.57)	-0.25 (0.90)	-0.10 (1.02)	**1.56 (1.52)**
2	69.77 (4.05)	-0.35 (0.63)	-0.04 (0.43)	0.42 (0.92)	0.04 (0.69)	0.26 (0.69)	**0.91 (0.69)**
3	74.87 (3.59)	-0.46 (0.93)	0.09 (0.26)	-0.14 (0.82)	-0.24 (0.49)	-0.35 (1.02)	**0.66 (0.77)**
4	63.75 (3.36)	-0.21 (1.28)	0.10 (0.20)	0.58 (0.69)	0.56 (1.23)	0.43 (0.71)	**1.23 (2.77)**
5	70.85 (2.09)	-0.38 (1.08)	0.42 (0.86)	0.26 (0.68)	-0.09 (0.52)	-0.21 (2.26)	**0.91 (1.12)**
6	59.24 (3.05)	-0.59 (1.84)	-0.09 (0.78)	-0.15 (1.84)	0.44 (0.97)	0.33 (0.83)	**0.82 (1.14)**
7	70.85 (2.32)	-0.08 (0.35)	0.26 (0.77)	0.04 (0.94)	0.03 (0.45)	-0.78 (1.26)	**0.76 (1.48)**
8	53.24 (3.57)	-0.88 (1.35)	-0.05 (0.22)	-0.40 (1.46)	-0.89 (1.88)	-0.76 (1.05)	**0.43 (1.35)**
9	70.18 (4.45)	-0.18 (0.45)	0.08 (0.36)	0.10 (0.74)	-0.78 (0.65)	0.03 (1.27)	**0.20 (1.17)**
10	58.98 (2.13)	-0.45 (0.88)	0.20 (0.41)	-0.04 (0.85)	-0.38 (1.41)	-0.49 (1.15)	**0.53 (2.11)**
11	55.03 (4.28)	-0.58 (1.11)	-0.74 (2.57)	0.22 (1.46)	0.22 (0.72)	-0.59 (1.94)	**1.54 (2.06)**
12	66.35 (2.94)	0.01 (0.50)	0.27 (0.36)	-0.24 (0.42)	-0.15 (0.40)	0.17 (0.40)	**0.47 (1.01)**
13	72.93 (2.32)	-0.14 (0.29)	-0.02 (0.20)	-0.09 (0.69)	0.16 (0.42)	-0.01 (0.33)	**1.35 (1.47)**
14	69.60 (2.15)	-0.39 (1.06)	-0.02 (0.33)	-0.64 (0.64)	-0.39 (0.73)	-0.31 (0.51)	**0.58 (0.89)**
15	65.14 (3.20)	-0.63 (0.75)	0.03 (0.80)	0.58 (0.77)	-0.14 (0.88)	0.33 (0.37)	**1.54 (1.02)**
16	63.22 (3.16)	0.07 (0.88)	-0.01 (0.64)	0.65 (0.60)	0.10 (0.62)	0.96 (1.43)	**2.82 (1.04)**
17	48.14 (4.43)	-0.39 (1.81)	0.64 (0.73)	-0.22 (1.79)	0.11 (0.81)	0.36 (3.32)	**2.75 (1.58)**
18	50.80 (4.61)	-0.34 (1.69)	0.33 (1.44)	0.99 (1.50)	0.34 (1.04)	1.68 (3.29)	**3.69 (3.20)**
19	71.50 (2.78)	-0.13 (0.80)	0.03 (0.43)	-0.23 (0.85)	-0.36 (0.61)	-0.47 (1.27)	**0.52 (0.76)**
20	73.06 (2.64)	-0.04 (0.41)	0.01 (0.36)	0.04 (0.69)	0.04 (0.40)	-0.29 (0.65)	**0.35 (0.79)**
21	70.96 (3.39)	-0.24 (0.72)	0.10 (0.31)	-0.30 (0.52)	-0.08 (0.81)	0.08 (0.83)	**0.66 (0.80)**
22	70.11 (4.54)	0.19 (0.64)	**0.26 (0.56)**	-0.86 (0.62)	-0.27 (0.30)	-0.73 (1.39)	-0.23 (1.62)
23	62.12 (3.41)	-0.92 (2.07)	0.02 (0.39)	-0.39 (1.03)	0.26 (1.01)	0.34 (0.61)	**1.16 (1.08)**
24	72.73 (2.89)	-0.70 (0.98)	-0.14 (0.87)	-0.01 (0.70)	-0.11 (0.49)	-0.30 (1.43)	**1.46 (1.50)**
25	62.18 (5.89)	-0.01 (0.87)	0.39 (0.73)	0.28 (0.82)	0.43 (1.11)	1.82 (1.46)	**3.80 (2.68)**
26	54.74 (3.72)	-0.05 (1.21)	0.11 (0.92)	0.45 (1.17)	0.23 (1.29)	0.75 (1.31)	**3.09 (1.95)**
27	66.21 (4.25)	-0.26 (0.95)	-0.08 (0.29)	-0.16 (1.10)	-0.11 (0.76)	-0.15 (1.12)	**0.80 (0.84)**
28	57.51 (4.59)	-0.67 (1.45)	-0.52 (1.87)	**0.59 (0.81)**	-0.68 (1.87)	-0.54 (1.86)	0.45 (0.78)
29	54.71 (4.05)	-1.35 (2.15)	-0.24 (0.73)	-0.54 (1.71)	-0.14 (1.04)	**0.37 (1.20)**	-0.31 (2.40)
30	53.34 (3.21)	-0.16 (0.38)	0.17 (1.19)	0.15 (1.01)	-0.37 (1.05)	-0.06 (1.14)	**0.37 (1.65)**
31	70.12 (3.50)	-0.61 (0.88)	0.26 (0.66)	-0.32 (0.97)	-0.27 (0.86)	-0.58 (1.71)	**0.60 (0.93)**
32	43.41 (3.69)	-0.55 (1.37)	-0.09 (0.39)	-0.03 (2.10)	-0.64 (2.92)	0.17 (1.14)	**3.06 (1.87)**
33	57.79 (3.31)	-0.19 (0.86)	0.39 (0.49)	-0.16 (1.39)	0.88 (1.76)	1.25 (1.47)	**1.54 (1.93)**
34	44.75 (5.48)	-0.08 (1.10)	0.16 (0.60)	0.60 (1.47)	0.20 (1.59)	0.72 (2.45)	**3.94 (2.73)**
35	47.12 (4.23)	-0.40 (0.91)	0.40 (1.08)	-0.32 (1.89)	-0.47 (0.73)	0.95 (1.68)	**3.07 (2.34)**
36	49.86 (3.78)	-0.31 (1.09)	-0.32 (0.93)	0.07 (1.55)	0.24 (0.66)	1.56 (1.23)	**3.22 (2.36)**
37	37.42 (3.33)	0.28 (0.74)	0.07 (0.38)	0.14 (1.24)	0.01 (0.94)	0.99 (2.60)	**2.83 (2.87)**
38	64.66 (3.97)	0.09 (1.32)	0.33 (0.78)	-0.37 (1.04)	-0.76 (0.95)	-0.08 (1.02)	**0.71 (1.03)**
39	74.71 (2.29)	-0.54 (0.43)	**0.27 (0.46)**	-0.57 (1.03)	-0.26 (0.65)	-0.05 (0.77)	0.19 (0.73)
40	65.19 (4.41)	-0.03 (0.78)	-0.05 (0.11)	-0.37 (1.70)	-0.33 (0.53)	0.15 (0.79)	**1.25 (2.26)**

Table 5. Accuracy of different PC and MTL algorithms on *PubFig (ResNet)*, *PubFig*, *Shoes (ResNet)*, and *Shoes*: 1) a parametric adaptation of Kim et al.'s algorithm (Kim et al., 2020) (*Param*), and 2-5) PC algorithms that use identity chart maps (*Id*), image translation networks (*IT*), *MMD*, and a combination of MMD and HSIC (*Ours*), respectively. For the initial predictors $\mathbf{f}^I$, 100×Kendall's τ coefficients are shown while for the PC algorithms, the accuracy increases from $\mathbf{f}^I$ are presented. The numbers in parentheses show standard deviations. For each target attribute, the best and second best results are highlighted in **boldface** and with underline, respectively. The results of statistical significance tests of each PC method compared to $\mathbf{f}^I$ are presented in three categories: significantly better, significantly worse, and insignificantly different.

Target	$\mathbf{f}^I$	*Param*	*MTL*	*Id*	*IT*	*MMD*	*Ours*
			PubFig (ResNet)				
1	78.43 (2.16)	-0.09 (0.28)	-0.06 (0.18)	-0.26 (0.38)	-0.01 (0.60)	-0.15 (0.38)	**0.21 (0.68)**
2	63.72 (3.09)	-0.18 (0.53)	-0.13 (0.64)	0.54 (0.63)	0.88 (0.87)	0.67 (1.26)	**1.60 (0.89)**
3	72.62 (2.32)	0.21 (0.29)	0.20 (0.43)	0.18 (1.22)	0.97 (0.96)	0.83 (1.07)	**1.10 (1.05)**
4	69.96 (4.04)	0.07 (0.18)	-0.04 (0.42)	1.03 (1.39)	0.45 (1.19)	1.64 (1.88)	**2.62 (1.78)**
5	69.01 (2.19)	-0.30 (0.58)	-0.23 (0.54)	-0.30 (1.01)	0.32 (0.84)	-0.29 (1.06)	**1.58 (1.51)**
6	82.38 (1.50)	-0.09 (0.42)	0.16 (0.52)	0.24 (0.65)	-0.40 (1.13)	0.87 (1.15)	**1.24 (0.93)**
7	65.65 (3.29)	-0.05 (0.93)	-0.23 (0.75)	-0.08 (1.36)	0.07 (1.14)	0.85 (1.49)	**2.54 (2.28)**
8	66.54 (1.98)	-0.17 (0.42)	0.37 (0.51)	-0.60 (1.09)	0.73 (0.86)	0.93 (1.28)	**2.27 (0.81)**
9	65.93 (2.52)	-0.26 (0.43)	-0.22 (0.59)	0.67 (1.19)	0.90 (0.79)	3.08 (1.36)	**4.43 (1.19)**
10	74.13 (3.39)	-0.00 (0.46)	-0.06 (0.06)	0.27 (0.83)	0.07 (0.68)	0.59 (0.84)	**0.94 (1.00)**
11	76.29 (2.23)	-0.24 (0.64)	0.05 (0.19)	0.19 (0.60)	**1.05 (0.74)**	-0.30 (1.94)	0.72 (1.35)
			PubFig				
1	64.22 (3.39)	N/A	0.08 (0.39)	N/A	N/A	-0.02 (0.59)	**0.96 (0.76)**
2	60.74 (3.22)	N/A	0.02 (0.16)	N/A	N/A	-0.18 (0.84)	**0.05 (0.52)**
3	65.39 (2.01)	N/A	-0.26 (0.43)	N/A	N/A	-0.02 (0.62)	**0.94 (1.06)**
4	62.67 (4.00)	N/A	0.05 (0.28)	N/A	N/A	-0.27 (0.55)	**1.85 (1.46)**
5	58.47 (2.75)	N/A	-0.15 (0.87)	N/A	N/A	0.25 (0.53)	**0.94 (1.25)**
6	77.67 (1.59)	N/A	0.24 (0.39)	N/A	N/A	0.17 (0.84)	**0.77 (0.68)**
7	61.33 (4.04)	N/A	-0.06 (0.18)	N/A	N/A	-0.48 (1.15)	**0.50 (1.23)**
8	60.51 (2.57)	N/A	-0.13 (0.24)	N/A	N/A	-0.42 (0.54)	**0.92 (0.56)**
9	56.39 (3.73)	N/A	0.03 (0.14)	N/A	N/A	0.39 (0.68)	**4.13 (1.91)**
10	61.04 (3.22)	N/A	0.23 (0.44)	N/A	N/A	0.00 (0.69)	**2.31 (1.73)**
11	65.79 (4.16)	N/A	-0.03 (0.34)	N/A	N/A	0.05 (0.41)	**1.62 (1.45)**
			Shoes (ResNet)				
1	74.34 (1.32)	-0.02 (0.08)	0.03 (0.16)	-0.05 (0.25)	0.41 (0.34)	**0.72 (0.24)**	0.66 (0.51)
2	73.45 (1.31)	-0.09 (0.25)	0.05 (0.06)	0.14 (0.29)	0.71 (0.32)	0.73 (0.35)	**1.23 (0.43)**
3	44.62 (2.32)	-0.08 (0.37)	0.07 (0.16)	0.68 (0.86)	1.23 (1.00)	1.42 (1.42)	**2.70 (1.10)**
4	56.71 (2.21)	-0.32 (0.82)	-0.05 (0.19)	0.25 (0.40)	1.23 (1.08)	0.46 (1.14)	**1.43 (1.08)**
5	71.46 (1.40)	-0.06 (0.26)	0.04 (0.04)	-0.01 (0.22)	0.45 (0.33)	0.47 (0.48)	**0.57 (0.52)**
6	70.06 (1.38)	-0.17 (0.28)	-0.01 (0.05)	-0.06 (0.32)	0.49 (0.45)	0.79 (1.01)	**1.20 (0.70)**
7	67.45 (2.15)	-0.28 (0.55)	0.00 (0.07)	0.27 (0.66)	0.63 (0.78)	0.88 (1.68)	**1.44 (0.76)**
8	62.47 (1.59)	0.09 (0.36)	-0.07 (0.31)	0.26 (0.43)	0.40 (0.65)	0.62 (0.83)	**1.03 (0.52)**
9	73.89 (1.04)	0.05 (0.41)	0.01 (0.10)	-0.08 (0.27)	0.23 (0.49)	0.20 (0.40)	**0.55 (0.45)**
10	74.03 (1.37)	0.00 (0.17)	0.21 (0.31)	0.11 (0.52)	0.31 (0.31)	0.56 (0.85)	**0.76 (1.07)**
			Shoes				
1	70.85 (1.11)	N/A	0.01 (0.07)	N/A	N/A	0.14 (0.87)	**0.46 (0.66)**
2	61.91 (2.01)	N/A	0.09 (0.18)	N/A	N/A	0.88 (0.87)	**2.21 (1.31)**
3	37.60 (2.34)	N/A	0.08 (0.47)	N/A	N/A	-0.03 (2.21)	**2.97 (1.60)**
4	49.98 (2.74)	N/A	-0.02 (0.25)	N/A	N/A	0.27 (0.88)	**0.71 (1.38)**
5	64.57 (1.22)	N/A	0.05 (0.07)	N/A	N/A	0.33 (0.39)	**1.29 (0.91)**
6	65.72 (1.59)	N/A	0.11 (0.17)	N/A	N/A	0.04 (0.91)	**1.56 (1.27)**
7	58.75 (1.96)	N/A	-0.01 (0.16)	N/A	N/A	-0.32 (1.55)	**2.00 (1.45)**
8	54.08 (2.03)	N/A	0.06 (0.10)	N/A	N/A	0.31 (0.68)	**0.65 (0.76)**
9	65.96 (1.28)	N/A	0.19 (0.34)	N/A	N/A	0.38 (0.57)	**0.91 (0.69)**
10	69.78 (1.37)	N/A	0.08 (0.22)	N/A	N/A	0.69 (0.78)	**1.06 (0.86)**

Table 6. Accuracy of different PC and MTL algorithms on *OSR (ResNet)* and *OSR*: 1) a parametric adaptation of Kim et al.'s algorithm (Kim et al., 2020) (*Param*), and 2-5) PC algorithms that use identity chart maps (*Id*), image translation networks (*IT*), *MMD*, and a combination of MMD and HSIC (*Ours*), respectively. For the initial predictors $\mathbf{f}^I$, 100×Kendall's τ coefficients are shown while for the PC algorithms, the accuracy increases from $\mathbf{f}^I$ are presented. The numbers in parentheses show standard deviations. For each target attribute, the best and second best results are highlighted in **boldface** and with underline, respectively. The results of statistical significance tests of each PC method compared to $\mathbf{f}^I$ are presented in three categories: significantly better, significantly worse, and insignificantly different.

Target	$\mathbf{f}^I$	*Param*	*MTL*	*Id*	*IT*	*MMD*	*Ours*
				OSR (ResNet)			
1	98.72 (0.30)	-0.02 (0.10)	-0.01 (0.06)	0.00 (0.17)	0.17 (0.15)	0.21 (0.13)	**0.21 (0.19)**
2	92.79 (0.65)	-0.02 (0.09)	0.03 (0.09)	0.28 (0.22)	0.50 (0.20)	0.64 (0.29)	**0.96 (0.29)**
3	91.10 (0.50)	-0.27 (0.17)	0.15 (0.15)	0.42 (0.58)	0.91 (0.35)	0.93 (0.43)	**1.32 (0.44)**
4	92.95 (0.37)	0.00 (0.11)	-0.03 (0.22)	0.29 (0.48)	0.62 (0.38)	1.48 (0.32)	**1.73 (0.48)**
5	94.48 (0.46)	0.04 (0.19)	0.01 (0.10)	0.76 (0.24)	0.71 (0.20)	1.41 (0.27)	**1.44 (0.32)**
6	96.19 (0.62)	-0.14 (0.09)	0.01 (0.15)	0.18 (0.30)	0.22 (0.39)	0.53 (0.34)	**0.58 (0.27)**
				OSR			
1	90.71 (0.53)	N/A	0.07 (0.14)	N/A	N/A	0.66 (0.34)	**0.86 (0.40)**
2	83.99 (0.73)	N/A	**0.46 (0.30)**	N/A	N/A	0.37 (0.54)	0.45 (0.50)
3	74.85 (0.85)	N/A	0.18 (0.19)	N/A	N/A	0.39 (0.70)	**0.77 (0.54)**
4	74.84 (1.33)	N/A	-0.11 (0.26)	N/A	N/A	1.00 (0.73)	**1.40 (0.41)**
5	78.12 (1.06)	N/A	0.09 (0.14)	N/A	N/A	0.80 (0.58)	**1.92 (0.96)**
6	78.85 (1.22)	N/A	-0.04 (0.15)	N/A	N/A	0.31 (0.60)	**1.16 (1.01)**

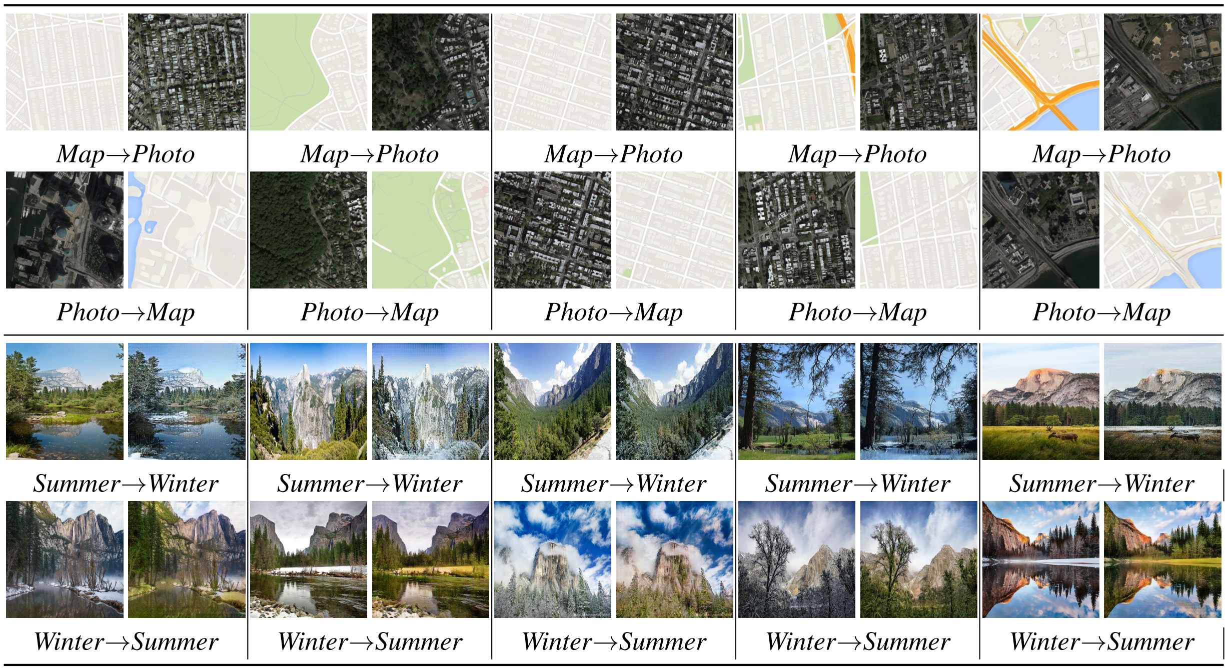

Figure 1. Examples of image translation between *Aerial photographs* and *Google maps* (top two rows), and *Summer* and *Winter* Yosemite photographs (bottom two rows). Images are provided by Zhu et al. (Zhu et al., 2017).

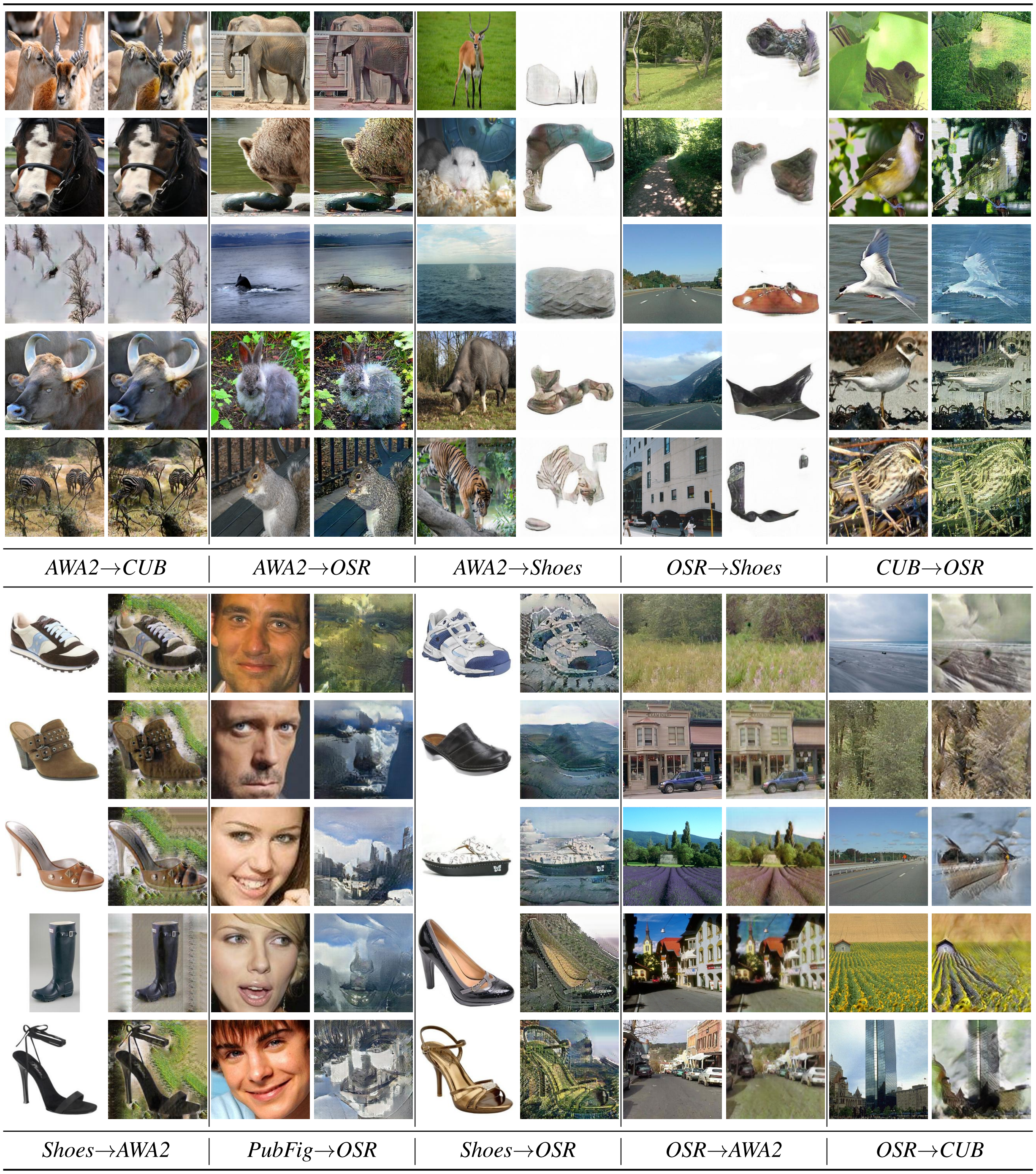

Figure 2. Examples of image translation across datasets of heterogeneous categories.