A. Limitations

One limitation of our OT-VP approach is its reliance on the quality of pseudo labels for computing the Optimal Transport distance. As visualized in our t-SNE plots 3, there's a risk of occasional misalignment due to inaccurate pseudo-labeling, which can adversely affect the model's ability to accurately bridge the source and target domain gap. While implementing entropy-based filtering akin to T3A [16] could mitigate this by filtering out high-entropy, less reliable pseudo labels, the fundamental limitation remains: OT-VP's capacity to perform effective test-time adaptation may be significantly hindered if the pseudo labels are entirely unreliable or carry no meaningful information about the true class distribution.

B. Full Results

B.1.

In this section, we present the computation time for OT-VP on the PACS dataset. We optimize only 4 prompt tokens over 5 epochs, utilizing a 20% hold-out split from both source and target data. While the multi-source setting involves processing triple the data to compute source representations compared to the single-source setting, the time required for both is nearly identical. Specifically, the average time is 39.5 seconds for the multi-source and 38.7 seconds for the single-source setting on the PACS dataset on our hardware. Moreover, the computational time is slightly influenced by the size of the datasets but remains relatively quick. For instance, in the PACS dataset, domain S, which has more than double the data of domain P, requires more processing time—51.7 seconds for S versus 34.5 seconds for P in the multi-source setting. Full results for PACS can be found in Table 5 in Appendix. In conclusion, OT-VP can efficiently learn prompts in both single and multi-source settings without significant computational overhead. For the single-source setting, the average computation time is calculated across three different sources.

Setting	A	C	P	S	Avg
Single-Source	32.6	37.5	33.7	50.8	38.7
Multi-Source	33.9	38.0	34.5	51.7	39.5

Table 5. Average computation time (seconds) for OT-VP on PACS dataset.

B.2.

In this section, we provide details for the three stylistic datasets and one corrupted dataset. **PACS** [23] is composed of four domains: **P**hotos, **A**rt, **C**artoon, and **S**ketch, containing 9,991 images in 7 classes. **VLCS** [8] comprises four real-world photographic datasets: **V**OC2007, **L**abelMe, **C**altech, and **S**UN09, containing 10,729 images in 5 classes. **OfficeHome** [48] consists of four domains: **A**rt, **C**lipart, **P**roduct, **R**eal, containing 15,588 images in 65 classes. **ImageNet-C** comprises corrupted images in 15 types of corruption. We use the highest level of corruption (*i.e.* severity 5).

B.3.

We present the implementation details of our experiments. Following [14], we partition the data from each domain into training and validation splits of 80% and 20%, respectively, utilizing the larger split for training and the smaller one for model selection. Our training approach for ERM adheres to the hyperparameters specified by [52], incorporating a dropout rate of 0.1 and a weight decay of 10^{-2}. The learning rate is 5×10^{-6} for PACS and VLCS, and 10^{-5} for OfficeHome.

For all baseline methods except for DePT, we use their official implementation[2] [3] [4]. For the implementation of DoPrompt, we set the prompt length to 4, with the coefficient λ explored over the set $\{0.1, 1, 10\}$. The M parameter for T3A is chosen from $\{1, 5, 20, 50, 100, \mathrm{N/A}\}$, while the configuration for Tent is determined from combinations of $\{0.1, 1.0, 10.0\}$ and $\{1, 3\}$. For DePT, we implement DePT-Group with $M = 4$ stages and 50 prompts, adhering to the same hyperparameters specified for ImageNet-C in [11].

In the single-source scenario, the model is trained on one domain and then adapted to another. The average accuracy is calculated across all 12 domain pairings for each trial. In the multi-source setting, one domain is designated as the target while the remaining three serve as sources.

B.4.

In this section, we present the comprehensive outcomes in Tables 2 and 3. The experiments were conducted using three different seeds {0, 1, 2} within the `DomainBed` framework. Tables 6, 8, and 10 display average results from three rounds for each source-target pair on PACS, VLCS, and OfficeHome, respectively, in the single-source setting. Similarly, Tables 7, 9, and 11 show the average results across three rounds for each target domain on PACS, VLCS, and OfficeHome in the multi-source setting.

B.5.

OT-VP implicitly reduces prediction entropy. Consistent with prior research [49], there's an observed correlation between prediction entropy and accuracy—lower entropy often signifies more accurate and confident predic-

[2] https://github.com/DequanWang/tent

[3] https://github.com/matsuolab/T3A

[4] https://github.com/zhengzangw/DoPrompt

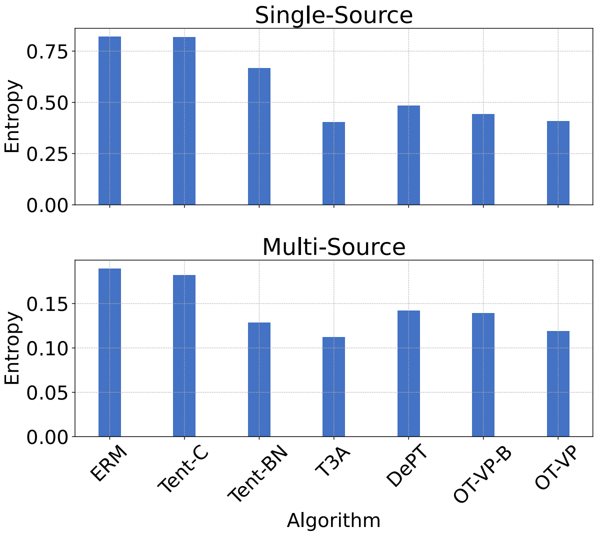

Figure 5. Prediction entropy across TTA Algorithms in Single-Source and Multi-Source settings on PACS. In both settings, OT-VP demonstrates a marked reduction in entropy, outperforming Tent-C and Tent-BN, which target entropy minimization directly.

tions. Unlike traditional approaches that explicitly target entropy reduction by adjusting model parameters [42, 49], OT-VP achieves this indirectly through the strategic application of Optimal Transport. This involves leveraging a cost metric that encompasses both features and labels 7, aiming to align the target distribution more closely with the source distribution, thereby enhancing model confidence near the decision boundary. This alignment is visually supported by representations such as those depicted in Fig. 3, a t-SNE visualization for source A and the target C (A → C) within the PACS dataset.

A comparative analysis of prediction entropy among ERM, Tent-C, Tent-BN, and OT-VP—illustrated in Fig. 5—demonstrates that OT-VP can significantly lower entropy through the refined optimization of prompts. Remarkably, it does so even when compared with methods like Tent-C and Tent-BN, which pursue entropy minimization directly. It's important to note that the improvements achieved by Tent-C and Tent-BN result from carefully balancing accuracy and entropy reduction when selecting their hyperparameters.

Algo.		**A**	**C**	**P**	**S**
ERM	**A**	-	64.5	98.9	56.4
	C	83.9	-	89.6	69.2
	P	74.2	44.4	-	34.1
	S	50.8	58.4	49.5	-
DoPrompt	**A**	-	64.6	98.5	56.5
	C	84.1	-	90.1	74.0
	P	75.6	46.2	-	35.2
	S	46.4	55.0	45.1	-
Tent-C	**A**	-	64.6	98.9	56.3
	C	83.9	-	89.6	69.0
	P	74.4	44.5	-	33.7
	S	50.4	58.3	49.0	-
Tent-BN	**A**	-	71.9	98.9	66.6
	C	84.9	-	91.3	71.7
	P	78.0	56.0	-	41.8
	S	56.3	62.9	47.5	-
T3A	**A**	-	70.2	98.6	67.9
	C	86.3	-	94.4	71.1
	P	80.2	53.9	-	35.9
	S	69.0	69.9	56.9	-
DePT	**A**	-	70.1	98.4	64.4
	C	83.9	-	90.1	69.3
	P	76.6	49.7	-	36.1
	S	51.4	63.4	52.3	-
OT-VP-B	**A**	-	76.7	98.3	66.8
	C	84.4	-	92.2	69.8
	P	77.8	56.8	-	63.9
	S	44.7	58.1	40.4	-
OT-VP	**A**	-	81.8	99.0	72.2
	C	84.4	-	92.6	69.5
	P	80.4	64.3	-	67.2
	S	56.0	64.6	50.5	-

Table 6. Single-Source Full Results on PACS in Table 2

Algo.	**A**	**C**	**P**	**S**	**Gain**
ERM	91.3	82.3	98.9	75.6	87.0
DoPrompt	91.4	81.8	**99.5**	77.1	87.5
Tent-C	91.6	82.7	98.9	75.7	87.2
Tent-BN	91.1	82.4	98.3	76.8	87.2
T3A	91.5	81.8	99.0	77.4	87.4
DePT	91.1	81.7	99.2	77.3	87.3
OT-VP-B	91.2	81.8	99.4	**77.4**	87.3
OT-VP	**92.0**	**83.0**	99.2	76.4	**87.7**

Table 7. Multi-Source Full Results on PACS in Table 3

Algo.		C	L	S	V
ERM	**C**	-	50.7	47.9	47.0
	L	62.9	-	55.8	63.1
	S	67.5	59.9	-	67.7
	V	96.5	66.1	80.3	-
DoPrompt	**C**	-	53.4	50.0	50.5
	L	71.7	-	57.8	70.1
	S	67.8	62.5	-	66.2
	V	98.6	62.0	78.8	-
Tent-C	**C**	-	50.4	48.3	47.0
	L	70.3	-	55.8	63.2
	S	67.2	59.8	-	67.9
	V	96.5	66.0	88.2	-
Tent-BN	**C**	-	38.3	46.9	52.7
	L	50.0	-	42.7	49.9
	S	60.9	62.3	-	69.3
	V	85.9	66.0	77.5	-
T3A	**C**	-	51.8	52.1	54.3
	L	83.6	-	62.7	64.3
	S	71.1	60.5	-	67.4
	V	97.3	66.8	80.3	-
DePT	**C**	-	54.6	50.8	48.5
	L	78.4	-	56.4	64.1
	S	68.4	61.2	-	67.6
	V	96.7	67.1	80.2	-
OT-VP-B	**C**	-	55.7	50.0	47.0
	L	73.1	-	56.4	60.7
	S	67.1	60.8	-	67.3
	V	96.8	68.4	79.1	-
OT-VP	**C**	-	59.9	51.3	48.9
	L	90.8	-	56.3	63.8
	S	69.6	64.2	-	68.8
	V	96.8	69.3	80.8	-

Table 8. Single-Source Full Results on VLCS in Table 2

Algo.	C	L	S	V	Gain
ERM	96.5	65.5	75.2	76.7	78.5
DoPrompt	**98.2**	67.8	75.3	**79.9**	80.3
Tent-C	97.7	65.2	75.3	76.9	78.8
Tent-BN	86.3	66.2	68.8	72.6	73.5
T3A	97.3	65.6	**78.0**	79.3	80.0
DePT	96.6	69.2	76.7	77.8	80.1
OT-VP-B	96.8	71.9	75.2	76.9	80.2
OT-VP	96.8	**73.1**	76.8	77.0	**80.9**

Table 9. Multi-Source Full Results on VLCS in Table 3

Algo.		A	C	P	R
ERM	**A**	-	54.3	71.4	77.0
	C	67.4	-	70.0	73.2
	P	62.9	47.8	-	78.9
	R	70.3	49.2	78.5	-
DoPrompt	**A**	-	52.1	71.7	79.0
	C	67.4	-	71.7	75.5
	P	66.8	47.8	-	79.0
	R	72.2	48.8	79.6	-
Tent-C	**A**	-	54.5	69.9	76.9
	C	66.9	-	69.6	73.6
	P	62.9	47.9	-	78.6
	R	71.0	47.1	79.9	-
Tent-BN	**A**	-	56.7	70.5	77.5
	C	67.9	-	70.4	72.9
	P	65.4	48.6	-	79.1
	R	72.8	49.5	79.9	-
T3A	**A**	-	55.1	71.2	76.6
	C	67.9	-	71.5	74.8
	P	67.1	48.7	-	80.3
	R	72.9	49.8	80.9	-
DePT	**A**	-	54.9	70.2	75.9
	C	68.1	-	70.2	73.5
	P	64.7	48.6	-	79.2
	R	71.0	49.6	79.1	-
OT-VP-B	**A**	-	55.1	70.5	75.0
	C	65.8	-	69.4	73.1
	P	64.6	49.1	-	77.4
	R	71.1	52.1	79.6	-
OT-VP	**A**	-	55.0	71.4	76.9
	C	67.6	-	70.1	73.6
	P	68.7	49.7	-	79.9
	R	71.3	52.2	80.8	-

Table 10. Single-Source Full Results on OfficeHome in Table 2

Algo.	A	C	P	S	Gain
ERM	73.8	57.3	80.3	83.0	73.6
DoPrompt	73.4	58.8	81.7	**84.8**	74.7
Tent-C	73.5	57.3	80.4	83.2	73.6
Tent-BN	73.5	58.8	81.9	84.0	74.6
T3A	74.2	58.3	81.8	84.6	74.7
DePT	73.9	59.3	82.2	83.7	74.8
OT-VP-B	74.0	58.9	80.7	83.5	74.3
OT-VP	**74.2**	**59.6**	**82.3**	84.1	**75.1**

Table 11. Multi-Source Full Results on OfficeHome in Table 3