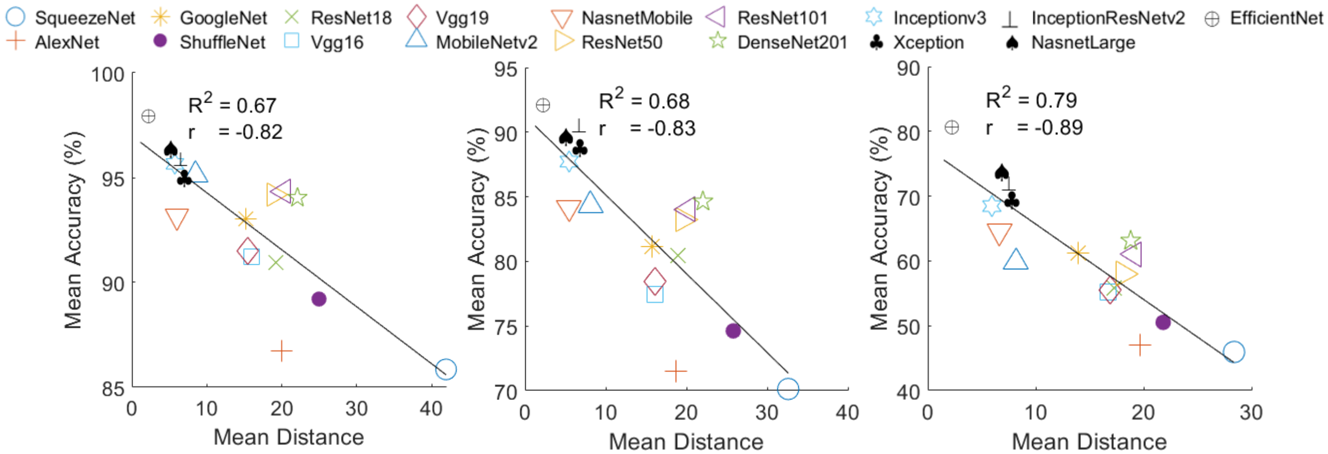
Supplementary Material: Efficient Pre-trained Features and Recurrent Pseudo-Labeling in Unsupervised Domain Adaptation

Youshan Zhang   Brian D. Davison
Computer Science and Engineering, Lehigh University, Bethlehem, PA, USA
{yoz217, bdd3}@lehigh.edu



(a) Office + Caltech-10

(b) Office-31

(c) Office-Home

Figure 1: Relationship between mean distance of different ImageNet models and mean accuracy on three datasets using proposed mean distance $Dist_k^{Pre}$ (R^2: R squared value, r: correlation).

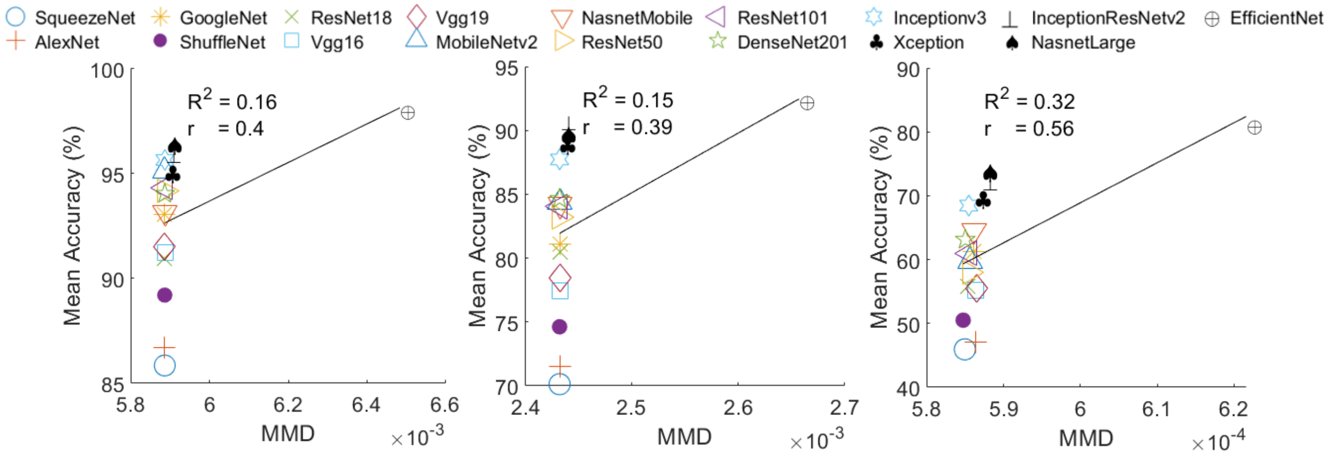
1. ImageNet Mean Distance Validation

In Tab. 1 of main paper, we list the mean distance values to decide the best backbone network. Therefore, we conduct another experiment to show the effectiveness of different pre-trained features on mean target domain accuracy.

Fig. 1 shows the relationship between the mean distance of different ImageNet models and the mean target domain accuracy on three benchmarking domain adaptation datasets with our proposed PRPL model. We report the $R^2 \in [0, 1]$ values for fitting the linear line and correlation score $r \in [-1, 1]$. The higher the R^2 value, the better of the model fits data. $r = -1$ means there is strong negative relationship, while $r = 1$ represents strong positive relationship, and $r = 0$ indicates there is no correlation between mean distance and mean accuracy. We can observe that overall domain transfer performance from three datasets is negatively correlated with the increase of mean distance, which implies that a lower mean distance will lead to higher performance in domain adaptation. In addition, EfficientNet features always outperform other features, which further validates that EfficientNet is best among all 17 ImageNet models. We note that the performance of pre-trained features from an ImageNet model is slightly lower than fine-tuning or retraining that model (since we only extract features based on models that only trained with the ImageNet dataset). However, the purpose of this paper is to reduce computation time and find the best pre-trained features for UDA which will achieve the highest performance.

2. Different Distance Functions Comparison

To show the superiority of our proposed $Dist_k^{Pre}$ mean distance function, we also compare it with MMD (Eq. 1) and mean cosine distance (Eq. 2). From Fig. 2, we can find that the R^2 values and r values are relatively small, which illustrates that the MMD function cannot suggest a strong relationship between its estimated distance and the mean

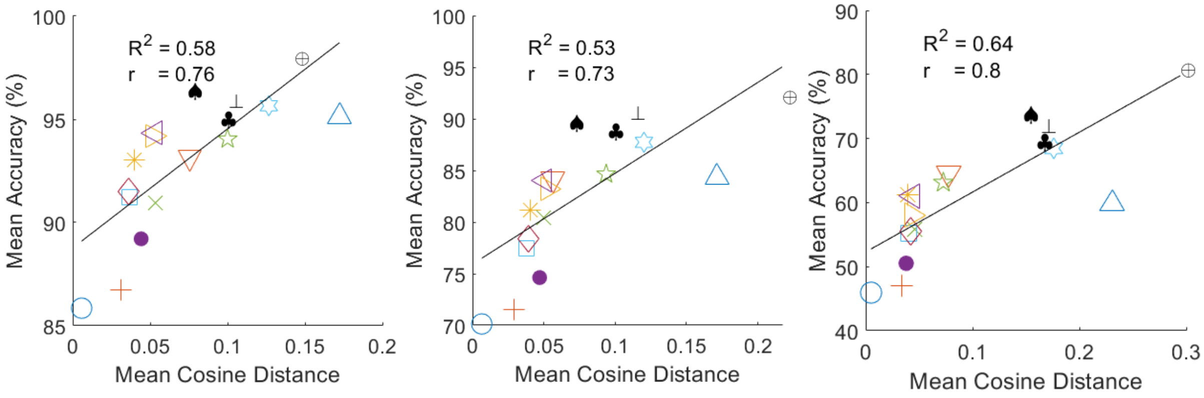


(a) Office + Caltech-10

(b) Office-31

(c) Office-Home

Figure 2: Relationship between mean distance of different ImageNet models and mean accuracy on three datasets using MMD (R^2: R square value, r: correlation).



(a) Office + Caltech-10

(b) Office-31

(c) Office-Home

Figure 3: Relationship between mean distance of different ImageNet models and mean accuracy on three datasets using mean cosine distance (R^2: R square value, r: correlation).

accuracy. Also, we cannot select the best pre-trained features if we use the MMD function. In addition, we observe that mean cosine distance in Fig. 3 shows an unusual trend that, with the increase of distance between two domains, the accuracy is increased, which is clearly an invalid observation. First of all, compared with Fig. 1, the R^2 values and the absolute r values of mean cosine distance are all lower than mean distance $Dist_k^{Pre}$. Secondly, Fig. 3 violates the truth that a smaller domain discrepancy will lead to a higher accuracy on the target domain. The underlying reason is that mean cosine distance relies on the similarity between the two domains. In comparison, the similarity is not a proper function to determine the distance between two domains for pre-trained features. Therefore, we can conclude that our proposed mean distance $Dist_k^{Pre}$ is effective and useful in selecting the best pre-trained features for the recurrent pseudo-labeling procedure.

$$\mathcal{L}_{\mathcal{MMD}} = \frac{1}{\mathcal{N}_\mathcal{S}^2}\sum_{i,j}^{\mathcal{N}_\mathcal{S}} \kappa(\Phi_k(\mathcal{X}_\mathcal{S}^i),\Phi_k(\mathcal{X}_\mathcal{S}^j)) + \frac{1}{\mathcal{N}_\mathcal{T}^2}\sum_{i,j}^{\mathcal{N}_\mathcal{T}} \kappa(\Phi_k(\mathcal{X}_\mathcal{T}^i),\Phi_k(\mathcal{X}_\mathcal{T}^j)) - \frac{2}{\mathcal{N}_\mathcal{S}\cdot\mathcal{N}_\mathcal{T}}\sum_{i,j}^{\mathcal{N}_\mathcal{S},\mathcal{N}_\mathcal{T}} \kappa(\Phi_k(\mathcal{X}_\mathcal{S}^i),\Phi_k(\mathcal{X}_\mathcal{T}^j)), \tag{1}$$

where κ is the mean of linear combination of multiple RBF kernels, and Φ_k is the $k^{th} \in \{1,2,\cdots 17\}$ feature extractor from seventeen pre-trained models.

$$Dist_k^{Pre} = cosd(\frac{1}{\mathcal{N}_\mathcal{S}}\sum_{i=1}^{\mathcal{N}_\mathcal{S}}\Phi_k(\mathcal{X}_\mathcal{S}^i) - \frac{1}{\mathcal{N}_\mathcal{T}}\sum_{j=1}^{\mathcal{N}_\mathcal{T}}\Phi_k(\mathcal{X}_\mathcal{T}^j)), \tag{2}$$

where $cosd(m,n) = 1 - \frac{m\cdot n}{||m||\times||n||}$.