Supplementary Material
Geodesic Flow Kernel for Unsupervised Domain Adaptation

Boqing Gong, Yuan Shi, Fei Sha
Dept. of Computer Science
U. of Southern California
{boqinggo, yuanshi, feisha}@usc.edu

Kristen Grauman
Dept. of Computer Science
U. of Texas at Austin
grauman@cs.utexas.edu

Throughout the main text, many algorithmic details and empirical results were omitted and only discussed briefly so as to observe the limit on the number of pages. In this file, we expand the discussions in the main text and provide more details on

- the derivation of our geodesic flow kernel (**GFK**) (Sec. A), explaining how eq.(5) and eq.(6) in the main text are derived (section 3.3).
- how to compute the rank of domain (ROD) metric (Sec. B); the idea was only sketched in section 3.5 of the main text.
- empirical studies of domain adaptation between 3 domains: Amazon, DSLR and Webcam (Sec. C).

  We had conducted two parallel empirical studies, one on the 3 domains and the other one on the 4 domains obtained from expanding the 3 with the dataset of Caltech-256. While both sets of empirical studies have reached the *same* findings that validate our methods, we chose to focus on domain adaptation among the 4 domains to demonstrate that our methods are robust to the additional diversity beyond the original 3.

  To be comprehensive, we report our results on those 3 domains as they provide a worthy reference point to contrast our work directly to published ones.
- empirical studies of domain adaptation between Caltech, Amazon, Webcam and DSLR (Sec. D). In the main text (section 4), we reported only 8 of 12 possible pairs of source and target domains. This Suppl. reports the remaining 4 pairs.
- characterizing the datasets of PASCAL, ImageNet, and Caltech-101 with our ROD metric (Sec. E). The metric corroborates our empirical findings on the cross-dataset generalization performances of these 3 domains (section 4.6, and especially Table 3).

A. The derivation of the geodesic flow kernel

Let $\boldsymbol{\Omega}^{\mathrm{T}}$ denote the following matrix

$$\boldsymbol{\Omega}^{\mathrm{T}} = [\boldsymbol{P}_{\mathcal{S}} \;\; \boldsymbol{R}_{\mathcal{S}}] \begin{bmatrix} \boldsymbol{U}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{U}_2 \end{bmatrix}. \tag{1}$$

The geodesic flow $\boldsymbol{\Phi}(t)$, $t \in (0,1)$, between $\boldsymbol{P}_{\mathcal{S}}$ and $\boldsymbol{P}_{\mathcal{T}}$ can be written as

$$\boldsymbol{\Phi}(t) = \boldsymbol{P}_{\mathcal{S}}\boldsymbol{U}_1\boldsymbol{\Gamma}(t) - \boldsymbol{R}_{\mathcal{S}}\boldsymbol{U}_2\boldsymbol{\Sigma}(t) = \boldsymbol{\Omega}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\Gamma}(t) \\ -\boldsymbol{\Sigma}(t) \end{bmatrix}. \tag{2}$$

Recall that the geodesic flow kernel (**GFK**) is defined as,

$$\langle \boldsymbol{z}_i^{\infty}, \boldsymbol{z}_j^{\infty} \rangle = \int_0^1 (\boldsymbol{\Phi}(t)^{\mathrm{T}}\boldsymbol{x}_i)^{\mathrm{T}}(\boldsymbol{\Phi}(t)^{\mathrm{T}}\boldsymbol{x}_j)\, dt = \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{G}\boldsymbol{x}_j, \tag{3}$$

where

$$\boldsymbol{G} = \int_0^1 \boldsymbol{\Phi}(t)\boldsymbol{\Phi}(t)^{\mathrm{T}} dt. \tag{4}$$

Substituting the expression of $\boldsymbol{\Phi}(t)$ of eq. (2) into above, we have (ignoring $\boldsymbol{\Omega}$ for the moment),

$$\boldsymbol{G} \propto \int_0^1 \begin{bmatrix} \boldsymbol{\Gamma}(t)\boldsymbol{\Gamma}(t) & -\boldsymbol{\Gamma}(t)\boldsymbol{\Sigma}(t) \\ -\boldsymbol{\Sigma}(t)\boldsymbol{\Gamma}(t) & \boldsymbol{\Sigma}(t)\boldsymbol{\Sigma}(t) \end{bmatrix} dt \tag{5}$$

Both $\boldsymbol{\Gamma}(t)$ and $\boldsymbol{\Sigma}(t)$ are diagonal matrices with elements being $\cos(t\theta_i)$ and $\sin(t\theta_i)$. Thus, we can integrate in close-form,

$$\lambda_{1i} = \int_0^1 \cos^2(t\theta_i)dt = 1 + \frac{\sin(2\theta_i)}{2\theta_i}, \tag{6}$$

$$\lambda_{2i} = -\int_0^1 \cos(t\theta_i)\sin(t\theta_i)dt = \frac{\cos(2\theta_i)-1}{2\theta_i} \tag{7}$$

$$\lambda_{3i} = \int_0^1 \sin^2(t\theta_i)dt = 1 - \frac{\sin(2\theta_i)}{2\theta_i}, \tag{8}$$

which become the i-th diagonal elements of diagonal matrices $\boldsymbol{\Lambda}_1$, $\boldsymbol{\Lambda}_2$, and $\boldsymbol{\Lambda}_3$ respectively. In terms of these matrices, the inner product eq. (3) is a linear kernel $\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{G}\boldsymbol{x}_j$ with the matrix $\boldsymbol{G}$ given by

$$\boldsymbol{G} = \boldsymbol{\Omega}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\Lambda}_1 & \boldsymbol{\Lambda}_2 \\ \boldsymbol{\Lambda}_2 & \boldsymbol{\Lambda}_3 \end{bmatrix} \boldsymbol{\Omega}. \tag{9}$$

B. How to compute rank of domain (ROD)

B.1. Principal angles and vectors

Let $\boldsymbol{P}_{\mathcal{S}}$ and $\boldsymbol{P}_{\mathcal{T}}$ be the basis of two subspaces. The *principal angles* θ_i between the two subspaces are recursively defined as,

$$\cos(\theta_i) = \max_{\boldsymbol{s}_i \in \mathrm{span}(\boldsymbol{P}_{\mathcal{S}})} \max_{\boldsymbol{t}_i \in \mathrm{span}(\boldsymbol{P}_{\mathcal{T}})} \frac{\langle \boldsymbol{s}_i, \boldsymbol{t}_i \rangle}{\|\boldsymbol{s}_i\| \|\boldsymbol{t}_i\|}, \tag{10}$$

such that

$$\begin{matrix} \boldsymbol{s}_k \in \mathrm{span}(\boldsymbol{P}_{\mathcal{S}}), & \boldsymbol{s}_i \perp \boldsymbol{s}_k, \\ \boldsymbol{t}_k \in \mathrm{span}(\boldsymbol{P}_{\mathcal{T}}), & \boldsymbol{t}_i \perp \boldsymbol{t}_k, \end{matrix} \quad k = 1, 2, \cdots i-1.$$

In the above, $\boldsymbol{s}_i$ and $\boldsymbol{t}_i$ are called the *principal vectors* associated with θ_i. Essentially, principal vectors are new basis for the two subspaces such that after the change of the basis, the two subspaces maximally overlap. The degrees of overlapping are measured in the principal angles — the smallest angles between basis.

Given the singular value decomposition,

$$\boldsymbol{P}_{\mathcal{S}}^{\mathrm{T}} \boldsymbol{P}_{\mathcal{T}} = \boldsymbol{U}_1 \boldsymbol{\Gamma} \boldsymbol{V}^{\mathrm{T}} \tag{11}$$

both the principal angles and vectors can be computed efficiently

$$\theta_i = \arccos \gamma_i, \boldsymbol{s}_i = (\boldsymbol{P}_{\mathcal{S}} \boldsymbol{U}_1)_{\cdot, i}, \boldsymbol{t}_i = (\boldsymbol{P}_{\mathcal{T}} \boldsymbol{V})_{\cdot, i}, \tag{12}$$

where γ_i is the i-th diagonal element of the diagonal matrix $\boldsymbol{\Gamma}$. $(\boldsymbol{M})_{\cdot,i}$ returns the i-th column of the matrix $\boldsymbol{M}$.

B.2. Computing ROD

Let $\boldsymbol{X}_{\mathcal{S}} \in \mathbb{R}^{\mathsf{N}_{\mathcal{S}} \times \mathsf{D}}$ and $\boldsymbol{X}_{\mathcal{T}} \in \mathbb{R}^{\mathsf{N}_{\mathcal{T}} \times \mathsf{D}}$ denote the data from the source and the target domains. We use their PCA subspaces to compute the ROD metric. The optimal dimensionality d^* of the subspaces is selected with our subspace disagreement measure, described in section 3.4 in the main text.

The ROD metric integrates both geometrical and statistical information between two domains by

$$\mathcal{R}(\mathcal{S}, \mathcal{T}) = \frac{1}{\mathsf{d}^*} \sum_{i}^{\mathsf{d}^*} \theta_i \left[KL(\mathcal{S}_i \| \mathcal{T}_i) + KL(\mathcal{T}_i \| \mathcal{S}_i) \right], \tag{13}$$

where $\mathcal{S}_i$ and $\mathcal{T}_i$ are two one-dimensional distributions of $\boldsymbol{X}_{\mathcal{S}}^{\mathrm{T}} \boldsymbol{s}_i$ and $\boldsymbol{X}_{\mathcal{T}}^{\mathrm{T}} \boldsymbol{t}_i$ respectively. In other words, we project data onto the principal vectors and compare how (dis)similar the data are distributed across domains.

We approximate these two distributions with one-dimensional Gaussians. Note that $\boldsymbol{X}_{\mathcal{S}}$ and $\boldsymbol{X}_{\mathcal{T}}$ have zero-means. We thus need only to compute the variances in order to specify the Gaussians. These variances can be readily computed from the projections and the covariance matrices of the original data:

$$\sigma_{i\mathcal{S}}^2 = \frac{1}{\mathsf{N}_{\mathcal{S}}} \boldsymbol{s}_i^{\mathrm{T}} \boldsymbol{X}_{\mathcal{S}}^{\mathrm{T}} \boldsymbol{X}_{\mathcal{S}} \boldsymbol{s}_i, \quad \sigma_{i\mathcal{T}}^2 = \frac{1}{\mathsf{N}_{\mathcal{T}}} \boldsymbol{t}_i^{\mathrm{T}} \boldsymbol{X}_{\mathcal{T}}^{\mathrm{T}} \boldsymbol{X}_{\mathcal{T}} \boldsymbol{t}_i, \tag{14}$$

In terms of the approximating Gaussians, the ROD metric is computed in close-form

$$\mathcal{R}(\mathcal{S}, \mathcal{T}) = \frac{1}{\mathsf{d}^*} \sum_{i}^{\mathsf{d}^*} \theta_i \left[\frac{1}{2} \frac{\sigma_{i\mathcal{S}}^2}{\sigma_{i\mathcal{T}}^2} + \frac{1}{2} \frac{\sigma_{i\mathcal{T}}^2}{\sigma_{i\mathcal{S}}^2} - 1 \right]. \tag{15}$$

C. Results on Amazon, Webcam & DSLR

These 3 domains have been studied and benchmarked in [3, 2, 1]. We report our own empirical studies of domain adaptation among them, thus, offer a direct comparison to published results.

We stress that results on these 3 domains arrive at the *same* findings as those identified in section 4 of the main text, where we report results on 4 domains (the 3 domains augmented with Caltech-256). In particular, results in this section validate the advantages of our **GFK** based methods for domain adaptation.

C.1. Setup

As in the previous work, we report results on Amazon → Webcam, DLSR → Webcam, and Webcam → DSLR in this section. We used the features provided by K. Saenko[1] and followed the same experimental setting as in [3] to split data. We randomly split data 20 times and report averaged results.

We conduct extensive evaluations of various baseline approaches as well as those proposed in [3, 1]. The methods we have studied include:

OrigFeat where we use original features, ie., without learning a new representation for domain adaptation.

PCA$_{\mathcal{S}}$ the PCA subspace learned from the source domain. We project the original features into this subspace, and then use the resulting representation for classification.

PCA$_{\mathcal{T}}$ the PCA subspace learned from the target domain.

PCA$_{\mathcal{S}+\mathcal{T}}$ the PCA subspace learned from the dataset that combines the source and the target domain directly.

PLS$_{\mathcal{S}}$ the PLS subspace learned from the source domain, which takes label information into consideration.

[1]The dataset and features are downloaded from http://www.icsi.berkeley.edu/∼saenko/projects.html#data.

Table 1. Recognition accuracies on target domains with *unsupervised* adaptation (A: Amazon, W: Webcam, and D: DSLR).

Method	A → W	D→ W	W → D
OrigFeat	10.7± 0.4	29.5 ±0.3	32.7±0.4
PCA$_{\mathcal{S}}$	*13.9*±0.4	35.4±0.3	43.4±0.4
PCA$_{\mathcal{T}}$	*13.8*±0.4	**46.9**±0.4	*47.2*±0.6
PCA$_{\mathcal{S}+\mathcal{T}}$	*14.0*±0.4	43.5±0.3	*47.7*±0.5
PLS$_{\mathcal{S}}$	*13.3*±0.4	31.6±0.3	35.6±0.6
SGF (rept.) [1]	39±2.0	26±0.8	19±1.2
SGF (impl.)	**14.2**±0.4	37.4±0.5	45.3±0.5
SGF (opti.)	**14.6**±0.4	43.1±0.3	*47.1*±0.5
GFK(PCA, PCA)	**14.8** ±0.4	42.7±0.3	*47.2*±0.5
GFK(PLS, PCA)	**15.0**±0.4	*44.6*±0.3	**49.7**±0.5

PLS$_{\mathcal{T}}$ the PLS subspace learned from the target domain by leveraging a small set of labeled target data.

PLS$_{\mathcal{S}+\mathcal{T}}$ the PLS subspace learned from the union of labeled source data and the small set of labeled target data.

Metric the metric learning method for domain adaptation proposed in [3]. We report both the previously published results in [3] (**Metric** rept.) and our own implementation (**Metric** impl.).

SGF Gopalan et al.'s method [1], which takes advantage of a series of subspaces by sampling the geodesic flow. We report results on three variants: i) the results of **SGF** reported in [1] (**SGF** rept.); ii) our implementation using the recommended parameters in [1] (**SGF** impl.); iii) our implementation using the optimal dimensionality automatically selected by our algorithm (**SGF** opti.). We have found that the dimensionality of the subspaces is one of the most important parameters to be tuned; for other parameters, we still use what is recommended in [1].

GFK our geodesic flow kernel method.

Note that **PLS**$_{\mathcal{T}}$, **PLS**$_{\mathcal{S}+\mathcal{T}}$, and **Metric** methods can only be used in the semi-supervised case since they require access to the labeled data in the target domain.

C.2. Adaptation results

Table 1 and 2 summarize the averaged classification accuracies as well as their standard errors in the unsupervised and the semi-supervised tasks, respectively. The best performing group of methods whose performances are within one standard error of the highest accuracy are in red and bold font. The second-best performing group of methods are in blue color, in italics font, and underlined.

Across the board, our **GFK** based methods **GFK(PCA, PCA)** and **GFK(PLS, PCA)** perform the best. We contrast them to other methods in more details in the following:

Table 2. Recognition accuracies on target domains with *semi-supervised* adaptation (A: Amazon, W: Webcam, and D: DSLR).

Method	A → W	D → W	W → D
OrigFeat	34.9± 0.6	38.6±0.4	48.9±0.5
PCA$_{\mathcal{S}}$	43.3±0.6	56.8±0.4	60.9±0.4
PCA$_{\mathcal{T}}$	*44.4*±0.6	**62.9**±0.5	63.4±0.4
PCA$_{\mathcal{S}+\mathcal{T}}$	**45.5**±0.5	*61.8*±0.4	*64.3*±0.4
PLS$_{\mathcal{S}}$	40.3±0.5	52.2±0.4	54.8±0.5
PLS$_{\mathcal{T}}$	18.3±0.4	39.5±0.5	47.0±0.5
PLS$_{\mathcal{S}+\mathcal{T}}$	38.4±0.6	49.6±0.4	54.2±0.6
Metric (rept.) [3]	44	31	27
Metric (impl.)	34.5±0.7	36.9±0.8	48.1±0.6
SGF (rept.) [1]	57±3.5	36±1.1	37±2.3
SGF (impl.)	37.4±0.5	55.2±0.6	61.0±0.5
SGF (opti.)	*45.1*±0.6	*61.4*±0.4	*63.4*±0.5
GFK(PCA, PCA)	**46.0**±0.6	*61.1*±0.4	*63.8*±0.4
GFK(PLS, PCA)	**46.4**±0.5	*61.3*±0.4	**66.3**±0.4
GFK(PLS, PLS)	31.7±0.5	54.5±0.6	59.6±0.6

Table 3. ROD values between Amazon/DSLR/Webcam.

ROD →	Amazon	DSLR	Webcam
Amazon	0	0.18	0.06
DSLR	0.18	0	**0.03**
Webcam	**0.06**	**0.03**	0

- Comparison between **SGF** and **GFK**. While the previously proposed method of **SGF** (impl.) outperforms **OrigFeat** significantly, our **GFK(PCA, PCA)** and **GFK(PLS, PCA)** outperform **SGF**(impl.) most of the time. Note that the results from our own implementation are different from what were previously reported (**SGF**(rept.)), though in most parts, ours attain *better* accuracies. We believe that the differences are most likely due to feature preparation and data split. Nevertheless, our results corroborate previous findings that it is beneficial to use subspaces to model domain shifts [1].

  SGF(opti.) is our improved version of **SGF**(impl.): instead of using the recommended subspace dimensionality in the published work [1], we used the optimal subspace dimensionality selected by our methods. Despite that, our methods still outperform **SGF**(opti.). We attribute this advantage to two factors: i) we integrate an infinite number of subspaces, thus model domain shift better; ii) we have used PLS, a subspace of discriminative nature as it takes label information into consideration, to characterize the source domain.

  However, using PLS on the target domain does not seem to be beneficial. **GFK(PLS, PLS)** in the semi-supervised learning performs worse than other **GFK** methods. This is likely due to the lack of sufficient labeled target data in estimating its PLS subspace.

Table 4. Recognition accuracies on target domains with *unsupervised* adaptation (C: Caltech, A: Amazon, W: Webcam, and D: DSLR).

Method	C→A	C→W	C→D	A→C	A→W	A→D	W→C	W→A	W→D	D→C	D→A	D→W
OrigFeat	20.8±0.4	19.4±0.7	22.0±0.6	22.6±0.3	23.5±0.6	22.2±0.4	16.1±0.4	20.7±0.6	37.3±1.2	24.8±0.4	27.7±0.4	53.1±0.6
PCA$_{\mathcal{S}}$	34.7±0.5	*31.3*±0.6	33.6±1.2	34.0±0.3	31.3±0.5	29.4±0.8	23.4±0.6	28.0±0.5	68.2±1.0	26.8±0.3	28.1±0.3	61.7±0.7
PCA$_{\mathcal{T}}$	*37.5*±0.4	**33.9**±1.1	*37.8*±0.9	*35.4*±0.4	**34.9**±1.0	*33.3*±0.8	**29.6**±0.5	*32.5*±0.8	67.4±0.7	*31.2*±0.3	*34.4*±0.3	**79.4**±0.5
PCA$_{\mathcal{S}+\mathcal{T}}$	*36.6*±0.5	*32.1*±1.2	34.9±1.4	*35.8*±0.4	*32.8*±0.7	31.5±0.9	*28.1*±0.5	*31.6*±0.7	**74.1**±0.8	*30.8*±0.2	33.3±0.3	**79.7**±0.6
PLS$_{\mathcal{S}}$	26.7±0.9	26.0±0.6	28.2±1.3	31.1±0.5	29.3±0.9	28.0±1.0	18.3±0.5	21.1±0.9	42.8±1.4	21.4±0.6	26.5±0.6	41.9±1.4
SGF(impl.)	*36.8*±0.5	*30.6*±0.8	32.6±0.8	*35.3*±0.5	31.0±0.7	30.7±0.8	21.7±0.4	27.5±0.5	54.3±1.2	29.4±0.5	32.0±0.4	66.0±0.5
SGF(opti.)	*36.9*±0.5	**33.9±1.2**	35.2±1.0	*35.6*±0.4	**34.4**±0.9	**34.9**±0.9	*27.3*±0.5	*31.3*±0.7	*70.7*±0.9	30.0±0.2	32.6±0.5	*74.9*±0.6
GFK(A,A)	*36.9*±0.4	**33.7**±1.1	35.2±1.0	*35.6*±0.4	**34.4**±0.9	**35.2**±0.9	*27.2*±0.5	*31.1*±0.8	*70.6*±0.9	29.8±0.3	32.5±0.5	*74.9*±0.6
GFK(S,A)	**40.4**±0.7	**35.8±1.0**	**41.1**±1.3	**37.9**±0.4	**35.7**±0.9	**35.1**±0.8	**29.3**±0.4	**35.5**±0.7	*71.2*±0.9	**32.7**±0.4	**36.2**±0.4	**79.1**±0.7

Table 5. Recognition accuracies on target domains with *semi-supervised* adaptation (C: Caltech, A: Amazon, W: Webcam, and D: DSLR).

Method	C→D	C→W	C→A	A→C	A→W	A→D	W→C	W→A	W→D	D→C	D→A	D→W
OrigFeat	26.5±0.7	25.2±0.8	23.1±0.4	24.0±0.3	31.6±0.6	28.1±0.6	20.8±0.5	30.8±0.6	44.3±1.0	22.4±0.5	31.3±0.7	55.5±0.7
PCA$_{\mathcal{S}}$	*48.9*±1.0	*54.2*±0.9	40.3±0.4	35.5±0.5	47.3±0.7	*47.8*±1.0	28.1±0.8	38.2±0.6	72.1±0.8	27.0±0.5	36.8±0.5	64.4±0.7
PCA$_{\mathcal{T}}$	*49.9*±0.8	52.1±0.8	*41.7*±0.4	*37.6*±0.4	51.8±0.8	44.1±1.0	**33.9**±0.6	41.5±0.5	70.0±0.7	**34.1**±0.4	*42.1*±0.4	*81.3*±0.4
PCA$_{\mathcal{S}+\mathcal{T}}$	*48.7*±1.2	*55.8*±0.9	*42.0*±0.6	*37.7*±0.4	49.8±1.0	*47.5*±1.2	**33.6**±0.7	*42.9*±0.6	**77.1**±0.6	**34.0**±0.4	*42.9*±0.5	**83.0**±0.4
PLS$_{\mathcal{S}}$	43.1±1.0	45.9±1.0	36.8±0.5	31.4±0.6	41.4±0.9	45.5±1.1	24.7±0.7	32.2±0.9	49.1±0.9	26.0±0.8	34.5±0.4	49.4±1.2
PLS$_{\mathcal{T}}$	27.3±1.1	25.3±0.4	28.9±0.6	26.3±0.3	23.6±0.9	28.0±1.0	22.2±0.4	25.2±0.9	47.0±1.2	25.8±0.4	27.9±0.4	47.1±0.9
PLS$_{\mathcal{S}+\mathcal{T}}$	36.9±0.9	37.0±0.9	33.5±0.5	32.4±0.4	35.6±1.1	36.9±1.2	25.4±0.8	31.6±0.6	52.1±1.2	27.5±0.7	32.9±0.6	53.1±1.2
Metric (impl.)	35.0±1.1	34.7±1.0	33.7±0.8	27.3±0.7	36.0±1.0	33.7±0.9	21.7±0.5	32.3±0.8	51.3±0.9	22.5±0.6	30.3±0.8	55.6±0.7
SGF(impl.)	36.6±0.8	37.2±0.9	40.2±0.7	*37.7*±0.5	37.9±0.7	34.5±1.1	29.2±0.7	38.2±06	60.6±1.0	30.2±0.7	39.2±0.7	69.5±0.9
SGF(opti.)	*50.2*±0.8	*54.2*±0.9	*42.0*±0.5	*37.5*±0.4	*54.2*±0.8	*46.9*±1.1	**32.9**±0.7	*43.0*±0.7	*75.2*±0.7	*32.9*±0.4	**44.9**±0.7	78.6±0.4
GFK(A,A)	*49.5*±0.9	*54.2*±0.9	*42.0*±0.5	*37.8*±0.4	*53.7*±0.8	*47.0*±1.2	**32.8**±0.7	*42.8*±0.7	*75.0*±0.7	*32.7*±0.4	**45.0**±0.7	78.7±0.5
GFK(S,A)	**55.0**±0.9	**57.0**±0.9	**46.1**±0.6	**39.6**±0.4	**56.9**±1.0	**50.9**±0.9	*32.3*±0.6	**46.2**±0.7	*74.1*±0.9	**33.9**±0.6	**46.2**±0.6	80.2±0.4
GFK(S,S)	38.6±1.4	34.0±0.9	38.7±0.6	36.6±0.4	36.3±0.9	34.1±1.0	28.6±0.6	36.3±0.5	68.6±1.0	*32.6*±0.4	35.0±0.4	74.6±0.5

- Metric learning for domain adaptation. The **Metric** methods in Table 2 use the correspondence between source and target labeled data to learn a Mahalanobis metric to map data into a new feature space for classification. Probably due to the lack of enough labeled data in the target domains to give a reliable estimation, it does not perform as well as subspace-based methods.

- PCA baselines. It is also interesting and surprising to note that PCA based baselines, especially **PCA**$_{\mathcal{S}+\mathcal{T}}$ and **PCA**$_{\mathcal{T}}$, perform quite well. They are often in the second-best performing group, and are even better than the **SGF** methods on DLSR → Webcam and Webcam → DSLR.

  We suspect that because the domain difference between DSLR and Webcam is small, either **PCA**$_{\mathcal{T}}$ or **PCA**$_{\mathcal{S}+\mathcal{T}}$ is already able to capture the commonness of the two domains well. For instance, both DSLR and Webcam contain similar office images though with different resolutions (see Fig. 2 in the main text for an example).

  The similarity between Webcam and DSLR is also confirmed by our ROD metric, which we will describe next.

C.3. Analysis based on the ROD Metric

Table 3 shows the ROD values of the three domains when paired with other domains for domain adaptation. For each column (the target domain), we use the corresponding values in rows to rank the source domains. For example, when Amazon is used as the target domain, Webcam is ranked higher than DSLR. This suggests that, Webcam would lead to better accuracies if it is used as the source domain. Our results in Table 1 and 2 corroborate this ranking. We have found rankings in other columns are also validated by the adaptation results. In particular, we have identified that DSLR and Webcam are "pals" as they prefer to use the other as source domains.

D. Results on Caltech-256, Amazon, Webcam & DSLR

The setup of this set of empirical studies was described in the main text (section 4.1). We have followed similar feature extraction and experiment protocols for domain adaptation among 3 domains (Amazon, Webcam and DSLR) [1, 3]. Specifically, for the "new" domain Caltech-256, when it is used as the source domain, we used 20 images per category and when it is used as the target domain, we used 3 images per category in the setting of semi-supervised domain adaptation. For all other datasets, we follow the previously published practice.

In the main text, we have shown domain adaptation results for 8 pairs of source and target domains. In this section, we provide details on the remaining 4 pairs, other methods that we have compared to, as well as experiment details.

Table 4 shows the averaged accuracies and their standard errors for the unsupervised tasks and Table 5 shows the results for the semi-supervised tasks. Note that, to fit the table within the width of the page, we have shortened **GFK(PCA, PCA)** with **GFK**(A,A), **GFK(PLS, PCA)** with **GFK**(S,A)

Table 6. ROD values between PASCAL/ImageNet/Caltech-101. *Lower* values signify *stronger* adaptability of the corresponding source domain.

ROD $\rightarrow$	PASCAL	ImageNet	Caltech101
PASCAL	0	**1.9E-3**	8E-3
ImageNet	**1.9E-3**	0	**6.6E-3**
Caltech101	8E-3	6.6E-3	0

and **GFK(PLS, PLS)** with **GFK**(S,S).

In general, our **GFK(PLS, PCA)** performs the best, followed by **GFK(PCA, PCA)**, **SGF** with the optimal dimensionalities, **PCA**$_{\mathcal{T}}$ and/or **PCA**$_{\mathcal{S}+\mathcal{T}}$.

E. ROD of PASCAL/ImageNet/Caltech-101

In Section 4.6 of the main text, we have compared the datasets of PASCAL, ImageNet, and Caltech-101 on cross-dataset classification accuracies. We had also calculated their values of ROD and summarize them in Table 6. Lower values indicate stronger adaptability of the corresponding source domain. We can see that the ROD metric is consistent with what we observe in Section 4.6 of the main text. Take the PASCAL column for instance. ImageNet has smaller ROD value, and is better than Caltech-101 to serve as the source domain to adapt a classifier to the target domain PASCAL.

References

[1] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *Proc. of ICCV*, pages 999–1006, 2011. 2, 3, 4

[2] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *Proc. of CVPR*, pages 1785–1792, 2011. 2

[3] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *Proc. of ECCV*, pages 213–226, 2010. 2, 3, 4