

# A Grassmann Manifold-based Domain Adaptation Approach

Jingjing Zheng<sup>§</sup>      Ming-Yu Liu<sup>§\*</sup>  
University of Maryland<sup>§</sup>  
{zjngjng, mingyliu, rama}@umiacs.umd.edu

Rama Chellappa<sup>§</sup>      P. Jonathon Phillips<sup>†</sup>  
National Institute of Standards and Technology<sup>†</sup>  
jonathon.phillips@nist.gov

## Abstract

*Domain adaptation algorithms that handle shifts in the distribution between training and testing data are receiving much attention in computer vision. Recently, a Grassmann manifold-based domain adaptation algorithm that models the domain shift using intermediate subspaces along the geodesic connecting the source and target domains was presented in [6]. We build upon this work and propose replacing the step of concatenating feature projections on a very few sampled intermediate subspaces by directly integrating the distance between feature projections along the geodesic. The proposed approach considers all the intermediate subspaces along the geodesic. Thus, it is a more principled way of quantifying the cross-domain distance. We present the results of experiments on two standard datasets and show that the proposed algorithm yields favorable performance over previous approaches.*

## 1 Introduction

Traditional visual object recognition methods assume that the testing and training data are sampled from the same distribution. However, in practice, the training and testing data are captured under different conditions and exhibit different distributions. Failing to model this shift often leads to inferior results. Methods that can handle domain shift are essential for improving the recognition performance. This is referred to as the domain adaptation problem.

Several methods have been proposed to handle domain shift for support vector machines [13, 7, 3]. In the field of visual object recognition, [9, 8] computed domain-invariant metrics to quantify the similarity between objects of different domains. Recently, Gopalan

et al. modeled the domain shift using the geodesic connecting the source and target domains on a Grassmann manifold [6]. The key idea was to synthesize intermediate domains using intermediate subspaces along the geodesic and represent an object by concatenating its projections on these subspaces.

In this paper, we propose an alternative Grassmann manifold-based approach to address the domain adaptation problem. Specifically, we replace the step that concatenates a few intermediate subspace projections as done in [6] by integrating the distance between feature projections on all the intermediate subspaces along the geodesic. Our approach has two major advantages.

- It avoids ad-hoc sampling of intermediate subspaces in [6].
- It is more expressive because it implicitly projects data onto all the subspaces along the geodesic and smoothly accumulates the distance between data projections along the geodesic. In addition, it does not suffer from information loss that occurs in [6] due to discrete sampling.

Fig. 1 illustrates the difference between our approach and that of [6].

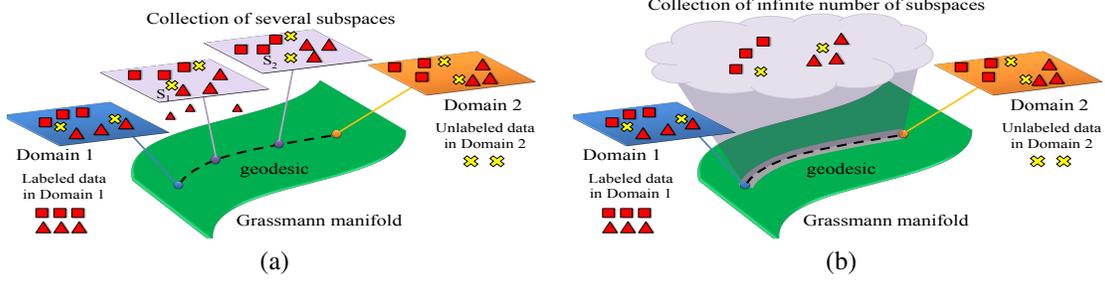
The paper is organized as follows. We review the calculation of geodesics on the Grassmann manifold in § 2 and present our approach in § 3. Experimental results are discussed in § 4. The paper is concluded in § 5.

## 2 Geodesic on the Grassman Manifold

A Grassmann manifold  $G_{n,d}$  is the set of all the  $d$ -dimensional subspaces of the vector space  $\mathbb{R}^n$ . In this paper, we denote a subspace  $S \in G_{n,d}$  using a matrix  $S$  in  $\mathbb{R}^{n \times d}$  whose columns are orthogonal and form a basis for this subspace. Note that if  $S$  is right-multiplied by a  $d$ -dimensional orthogonal matrix, it still denotes  $S$  because the subspace spanned by the columns of  $S$  remains the same.

<sup>§</sup>Ming-Yu Liu is now at Mitsubishi Electric Research Laboratories.

<sup>†</sup>This work was partially supported by a Cooperative Grant from the National Institute of Standards and Technology, Gaithersburg, MD. The identification of any commercial product or trade name does not imply endorsement or recommendation by NIST.



**Figure 1. Finite sampling versus continuous integration.** (a) Gopalan et al. [6] samples several intermediate subspaces along the geodesic connecting two domains on the Grassmann manifold. Here samples are represented by concatenating data projections on the sampled subspaces, and cross-domain class analysis is performed on the concatenated representation. (b) In our approach, we measure the similarity between two data points from different domains by integrating the distance of their projections through all the intermediate subspaces along the geodesic. Since we consider all the intermediate subspaces, this produces a more accurate metric.

Let  $S_0$  and  $S_1$  be two matrices in  $\mathbb{R}^{n \times d}$  whose columns are orthogonal bases for the  $d$ -dimensional subspaces  $\mathcal{S}_0$  and  $\mathcal{S}_1$  respectively. Let  $U_1 \Gamma V_1^T$  be a singular value decomposition (SVD) of the  $d \times d$  matrix  $S_0^T S_1$ . The geodesic  $\psi(t)$  on the Grassmann manifold  $\mathcal{G}_{n,d}$  starting from  $\mathcal{S}_0$  to  $\mathcal{S}_1$  is given by

$$\psi(t) = Q \exp(tB) J \quad \text{s.t.} \quad \begin{cases} \psi(0) = S_0 \\ \psi(1) = S_1 \end{cases} \quad (1)$$

where  $J = \begin{bmatrix} I_d \\ O_{n-d,d} \end{bmatrix}$ ,  $I_d$  is a  $d \times d$  identity matrix, and  $O_{n-d,d}$  is a matrix with all zeros [11]. Here,  $Q$  is an orthogonal matrix with determinant +1 and is given by

$$Q = \begin{bmatrix} S_{01} - I_d \\ S_{02} \end{bmatrix} [I_d - S_{01}^T]^{-1} [S_{01}^T - I_d S_{02}^T]. \quad (2)$$

The matrices  $S_{01} \in \mathbb{R}^{d \times d}$  and  $S_{02} \in \mathbb{R}^{(n-d) \times d}$  are the upper and lower parts of  $S_0$  respectively, i.e.,  $S_0 = \begin{bmatrix} S_{01} \\ S_{02} \end{bmatrix}$ , and the matrix  $B$  in Eq. (1) is asymmetric and block-diagonal given by  $B = \begin{bmatrix} O_{d,d} & A^T \\ -A & O_{d,d} \end{bmatrix}$  where  $A \in \mathbb{R}^{(n-d) \times d}$ .

Instead of directly calculating  $\psi(t)$ , we use the approach proposed by Gallivan et al. [4] which calculates the equivalent geodesic  $\bar{\psi}(t) = \psi(t)U_1$  connecting  $\mathcal{S}_0$  and  $\mathcal{S}_1$  such that  $\bar{\psi}(0) = S_0 U_1$  and  $\bar{\psi}(1) = S_1 V_1$ . The intuition behind this is that the subspaces represented by  $\psi(t)$ ,  $S_0$ , and  $S_1$  remains the same when these matrices are right multiplied by an orthogonal matrix.

Now the geodesic  $\bar{\psi}(t)$  connecting  $\mathcal{S}_0$  and  $\mathcal{S}_1$  is given by

$$\bar{\psi}(t) = Q \exp(tB) J U_1 \quad \text{s.t.} \quad \begin{cases} \bar{\psi}(0) = S_0 U_1 \\ \bar{\psi}(1) = S_1 V_1 \end{cases} \quad (3)$$

Using the results pertaining to the geodesic on Grassmann manifold [4], the geodesic  $\bar{\psi}(t)$  can be further simplified as

$$\bar{\psi}(t) = Q \begin{bmatrix} U_1 \Gamma(t) \\ -\tilde{U}_2 \Sigma(t) \end{bmatrix}, \quad (4)$$

where  $\tilde{U}_2 \in \mathbb{R}^{(n-d) \times d}$  is made up of  $d$  orthogonal columns. The derivation of  $\tilde{U}_2$  makes use of the boundary condition  $\bar{\psi}(1) = S_1 V_1$  and will be given in § 3. The matrices  $\Gamma(t), \Sigma(t) \in \mathbb{R}^{d \times d}$  are diagonal with diagonal elements being  $\gamma_i = \cos(t\theta_i)$  and  $\sigma_i = \sin(t\theta_i)$  respectively where  $0 \leq \theta_1 \leq \dots \leq \theta_d \leq \pi/2$ . Note that  $\{\theta_i\}_{i=1}^d$  form the rotation angles from  $\mathcal{S}_0$  to  $\mathcal{S}_1$ . We use  $\Theta$  to denote the diagonal matrix with diagonal elements given by  $\{\theta_i\}_{i=1}^d$ . Further details of the derivation can be found in [4].

Later on in the paper, we use the derived geodesic form to construct a measure that quantifies the distance between samples of different domains.

### 3 Domain Adaptation

Let  $X^s \in \mathbb{R}^{n \times m_s}$  and  $X^t \in \mathbb{R}^{n \times m_t}$  denote the feature representation of  $m_s$  and  $m_t$  samples in source and target domains respectively where each column  $\mathbf{x}_i \in \mathbb{R}^n$  denotes a sample and  $n$  is the feature dimension.

In [6], Gopalan et al. propose an approach which performs cross-domain class analysis using intermediate subspace along the geodesic on the Grassmann manifold. Specifically, they first apply the principle component analysis (PCA) on  $X^s$  and  $X^t$  respectively, which generates two  $d$ -dimensional subspaces denoted by two matrices  $S_0, S_1 \in \mathbb{R}^{n \times d}$ . The geodesic path  $\bar{\psi}(t)$  from  $\mathcal{S}_0$  to  $\mathcal{S}_1$  is then given by Eq. (3). Since each point on the geodesic is a subspace, the intermediate subspaces

can be obtained by sampling the geodesic  $\bar{\psi}(t)$  at different time points  $t_i$ . Let  $\hat{S} = \{S_t\}_{t=t_1}^{t_k}$  denote the collection of the  $k$  sampled intermediate subspaces, where  $0 = t_1 \leq \dots \leq t_k = 1$ . They then project each sample from both domains onto  $k$  subspaces in  $\hat{S}$  and concatenate all the  $k$  projections to form a long vector of size  $d \times k$ . A discriminative classifier is then trained to classify samples of unknown labels based on the high-dimensional vector representation using the samples whose labels are known. Note that in the semi-supervised classification task, labels of some samples in the target domain are also known.

The sampling based approach of [6] has two main disadvantages. First, it is not clear which sampling method should be used since different sampling methods result in different intermediate subspace representations and the final classification recognition degrades if an inferior sampling method is used. Second, the number of sampled points is limited because a large number of sampled points along the geodesic results in a very high dimensional feature vector which increases computational complexity. In order to overcome these disadvantages, we propose an alternative approach. Instead of sampling some points along the geodesic, we integrate the distance of data projections onto the subspaces along the geodesic. This yields a cross-domain distance metric which can be used for cross-domain class analysis. Our approach consists of the following three steps.

**Calculate the  $\Theta$ :** Given  $S_0$  and  $S_1$ , the matrix  $Q$  in Eq. (4) can be computed according to Eq. (2) and  $Q^T S_1$  is given by  $Q^T S_1 = \begin{bmatrix} S_0^T S_1 \\ S_{12} - S_{02} Z^T \end{bmatrix}$  where  $Z \in \mathbb{R}^{d \times d}$  satisfies  $Z(I_d - S_{01}^T) = (S_1^T S_0 - S_{11}^T)$ . Since

$$\psi(1) = Q \begin{bmatrix} U_1 \Gamma(t) \\ -\tilde{U}_2 \Sigma(t) \end{bmatrix} = \bar{S}_1 = S_1 V_1, \quad (5)$$

we have

$$\begin{aligned} Q^T S_1 &= \begin{bmatrix} U_1 \Gamma(1) V_1^T \\ -\tilde{U}_2 \Sigma(1) V_1^T \end{bmatrix} \\ &= \begin{bmatrix} U_1 & 0 \\ 0 & \tilde{U}_2 \end{bmatrix} \begin{bmatrix} \Gamma(1) \\ -\Sigma(1) \end{bmatrix} V_1^T \end{aligned} \quad (6)$$

Note that  $\tilde{U}_2$  and  $\Theta$  can be obtained by computing the thin CS decomposition of  $Q^T S_1$  [4].

**Calculate geodesic  $\bar{\psi}(t)$ :** With the matrix  $\Theta$  and  $\tilde{U}_2$ , one can obtain  $\Gamma(t)$  and  $\Sigma(t)$  using their definitions. By substituting  $\Gamma(t)$  and  $\Sigma(t)$  in (4), we obtain the geodesic starting from the source domain  $S_0$  to the target domain  $S_1$ :

$$\bar{\psi}(t) = Q \begin{bmatrix} U_1 \Gamma(t) \\ -\tilde{U}_2 \Sigma(t) \end{bmatrix} \quad (7)$$

**Calculate domain-invariant distances:** For a given pair of examples  $(\mathbf{x}_1, \mathbf{x}_2)$  where  $\mathbf{x}_1$  and  $\mathbf{x}_2$  come from the source and target domain respectively, we project them onto the subspace  $\bar{\psi}(t)$  indexed by  $t$  on the geodesic to obtain  $\tilde{\mathbf{x}}_1 = \bar{\psi}(t)^T \mathbf{x}_1$  and  $\tilde{\mathbf{x}}_2 = \bar{\psi}(t)^T \mathbf{x}_2$ . The final distance between  $\tilde{\mathbf{x}}_1$  and  $\tilde{\mathbf{x}}_2$  is calculated by integration given by

$$d(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2) = \int_0^1 (\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_2)^T (\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_2) dt \quad (8)$$

$$= (\mathbf{x}_1 - \mathbf{x}_2)^T \left( \int_0^1 \bar{\psi}(t) \bar{\psi}(t)^T dt \right) (\mathbf{x}_1 - \mathbf{x}_2) \quad (9)$$

$$= Q \begin{bmatrix} U_1 & 0 \\ 0 & \tilde{U}_2 \end{bmatrix} P \begin{bmatrix} U_1^T & 0 \\ 0 & \tilde{U}_2^T \end{bmatrix} Q^T. \quad (10)$$

where the matrix  $P$  can be easily determined using the subspace angles between  $S_0$  and  $S_1$ . Note that (10) is an analytical form and can be computed in constant time. Finally, we calculate the distance between a test sample and all the labeled samples from both domains and use a nearest neighbor algorithm for classification.

## 4 Experiments

We conducted experiments on cross-domain object category recognition and face recognition. We compared our approach with the state-of-the-art algorithms in [6] and [9].

### 4.1 Object Recognition

We evaluated the proposed algorithm on the cross-domain object category classification task using the benchmark dataset from [9], which contains images from 31 object categories. Based on the acquisition condition, the dataset images are divided into three domains: *amazon*, *dslr* and *webcam*. The *amazon* domain includes an average of 90 product images for each category downloaded from Amazon's website. Both the *dslr* and *webcam* domains have about 30 images per category; they are captured by a digital single lens reflex camera and a webcam respectively. We show example images in Fig 2. One can see that domain shift in the dataset is primarily due to changes in image resolution, object pose, background clutters, and scene lighting.

Image representation is based on SURF [1] features that are similar to those in [9, 6]. Specifically, we extracted SURF features for all the images in the *amazon* domain and used a random subset of the features to learn a codebook of 800 codewords. The codebook was used to encode the SURF features and each image in the dataset was presented by a histogram of the 800 codewords. We further normalized the histograms so that it



**Figure 2. Sample images from the benchmark dataset [9].** We show several images from the object categories of bike helmet, keyboard, and mug in the three domains of *amazon*, *dslr*, and *webcam*. Domain shift in the dataset is mainly due to changes in image resolution, object pose, and scene lighting.

Settings	source domain	target domain	[9] (asymm)	[9] (symm)	[6]	proposed
same-category	<i>webcam</i>	<i>dslr</i>	25	27	37	<b>66</b>
	<i>dslr</i>	<i>webcam</i>	30	31	36	<b>61</b>
	<i>amazon</i>	<i>webcam</i>	48	44	<b>57</b>	45
new-category	<i>webcam</i>	<i>dslr</i>	53	49	59	<b>66</b>

**Table 1. Comparison of classification accuracy.** We reported classification accuracy (in percentage) for the state-of-the-art algorithms in [9] (asymm and symm variants), [6] and our approach.

sums up to one. To obtain the final representation, the histograms of images in the same domain were further normalized so that each dimension have a zero mean and unit deviation. Then PCA is performed on the final representation.

There were two evaluation settings on the benchmark: same-category and new-category. In the same-category setting, there were labeled images for all the categories for both domains. In the new-category setting, there were labeled images for all the categories in the source domain, but only half of the categories in the target domain contained labeled images.

We compared the classification accuracy of the proposed algorithm with the state-of-the-art in Table 1. For our results, accuracy was obtained by averaging over 20 experiment trials; each trail contained a random set of labeled images in both the source and target domains. We observed that the proposed algorithm yields a better performance for two out of three tasks in the same-category setting. In addition, the proposed algorithm significantly improved the performance for the task in the new-category setting—by a margin of more than 10%. This showed the benefit of the integration-based approach which accumulated the distance along the geodesic over the previous approach [6]. However, we noted that the proposed algorithm is not effective for the adaptation from *amazon* to *webcam*. We believe this may be due to the fact that the proposed algorithm uses a simple nearest neighbor classification technique, while [9] and [6] classifiers are based on powerful machine learning algorithms of information theoretic met-

ric learning [2] and partial least squares method [12].

## 4.2 Face recognition across blur and illumination

We conducted face recognition experiments using the CMU-PIE dataset [10]. This dataset consists of images from 68 subjects captured under 21 different illumination conditions. We randomly selected 11 illumination conditions. All the images captured under these 11 conditions constituted the source domain data, while the remaining ones formed the target domain data. The images in the source domain were labeled, but not those in the target domain.

We synthesized domain shifts by applying two different types of blur kernels to the target domain data: 1) Gaussian blur kernel, and 2) motion blur kernel. For the two types of kernels, we gradually increased the kernel sizes to synthesize different degrees of domain shifts. For the Gaussian blur, we varied the standard deviation from 1 to 5. For the motion blur, we varied the motion speed, from 1 to 17 pixels. (The motion angle was set to 30 degrees.) Some of the target images were presented in Fig. 3. In summary, the domain shift consisted of two components. The first was a change in illumination direction, 11 illumination directs in the source domain and 10 directs in the target. The second component was a blur.

In Tables 2 and 3, we compared the proposed algorithm to [6] (without applying the partial least square analysis) for the Gaussian and motion blurs respectively. It can be seen that the closed-set identifica-



**Figure 3. Target image samples.** We illustrate several synthesized target images under different Gaussian blur and motion blur. The variable  $\sigma$  denotes the standard deviation, while  $L$  denotes the motion speed.

Gaussian blur	$\sigma=1.0$	$\sigma=1.5$	$\sigma=2.0$	$\sigma=2.5$	$\sigma=3.0$	$\sigma=3.5$	$\sigma=4.0$	$\sigma=4.5$	$\sigma=5.0$
[6]	93.8	86.8	86.3	70.9	57.5	43.7	28.4	21.5	17.7
Proposed	<b>94.9</b>	<b>88.7</b>	<b>88.2</b>	<b>74.6</b>	<b>62.7</b>	<b>47.4</b>	<b>31.9</b>	<b>24.3</b>	<b>19.4</b>

**Table 2. Comparison of identification accuracy in percent under different Gaussian blur in the target domain.** We vary the standard deviation of the Gaussian blur in the target domain from 1 to 5 and compare our recognition performance with [6].

Motion blur	$L=1$	$L=3$	$L=5$	$L=7$	$L=9$	$L=11$	$L=13$	$L=15$	$L=17$
[6]	95.0	93.4	90.4	86.0	77.5	65.2	53.2	43.4	36.8
Proposed	<b>95.2</b>	<b>94.0</b>	<b>92.1</b>	<b>88.2</b>	<b>82.8</b>	<b>70.3</b>	<b>58.8</b>	<b>53.8</b>	<b>42.7</b>

**Table 3. Comparison of identification accuracy under different motion blur in the target domain.** We vary the motion speed in the target domain from 1 to 17 pixels per sensor integration time and compare our recognition performance with [6].

tion accuracy of both methods decreases as the amount of blur in the target domain increases. However, our proposed algorithm consistently yielded better performance than [6].

## 5 Conclusion

We presented a cross-domain classification approach based on integrating the distance between data projections on the subspaces along the geodesic on a Grassmann manifold. We showed that the integration-based approach yields a better performance as compared to the previous approach that only samples few intermediate subspaces along the geodesic. In future, we plan to extend proposed approach by incorporating powerful machine learning methods.

## References

- [1] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *ECCV*, 2006.
- [2] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, 2007.
- [3] L. Duan, I. W.-H. Tsang, D. Xu, and S. J. Maybank. Domain transfer SVM for video concept detection. In *CVPR*, 2009.
- [4] K. Gallivan, A. Srivastava, X. Liu, and P. Van Dooren. Efficient algorithms for inferences on grassmann manifolds. In *IEEE Workshop on Statistical Signal Processing*, 2003.
- [5] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2012.
- [6] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011.
- [7] W. Jiang, E. Zavesky, S.-F. Chang, and A. C. Loui. Cross-domain learning methods for high-level visual concept classification. In *ICIP*, 2008.
- [8] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011.
- [9] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010.
- [10] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *TPAMI*, 2003.
- [11] P. K. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa. Statistical computations on grassmann and stiefel manifolds for image and video-based recognition. *TPAMI*, 2011.
- [12] H. Wold. Partial least squares. *International Journal of Cardiology*, 147(2):581–591, 1985.
- [13] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive SVMs. In *ACM Multimedia*, 2007.