Lie group manifold analysis: an unsupervised domain adaptation approach for image classification

Hongwei Yang[1] · Hui He[1] · Weizhe Zhang[1,2] · Yawen Bai[1] · Tao Li[1]

Abstract

Domain adaptation aims to minimize the mismatch between the source domain in which models are trained and the target domain to which those models are applied. Most existing works focus on instance reweighting, feature representation, and classifier learning independently, which are ineffective when the domain discrepancy is substantially large. In this study, we propose a new unified hybrid approach that takes advantage of Lie group theory, weighted distribution alignment, and manifold alignment, which are referred to as Lie Group Manifold Analysis (LGMA). LGMA mainly finds a one-parameter sub-group decided by the Lie algebra elements of the intrinsic mean of all samples, and this one-parameter sub-group is a geodesic on the original Lie group. Moreover, the Lie group samples are projected onto the geodesics to maximize the separability of the projected samples for realizing discrimination in the nonlinear Lie group manifold space. As far as we know, LGMA is the first attempt to perform Lie algebra transformation to project the original features in the Lie group space onto Lie algebra manifold space for domain adaptation. Comprehensive experiments validate that our approach considerably outperforms competitive methods on real-world datasets.

Keywords Domain adaptation · Transfer learning · Lie algebra transformation · Image classification

1 Introduction

The fields of machine learning [1] and pattern recognition have been widely and successfully applied to many practical applications, in which patterns can be extracted from training data to predict future results [2, 3]. Traditional machine learning methodologies assume that the training and test data come from the same domain, such that the input feature space and data distribution are the same. The performance of the predictive classifier can be degraded when data distribution between the training and the test data differs. In some scenarios, obtaining training data that matches the feature space and predicted data distribution of the test data can be exhausting and costly. Therefore, adaptive classifiers need to be created for target domains trained from related domains. This objective is the motivation of transfer learning.

Transfer learning is used to solve the problem in one domain (i.e., target domain) by using the information from a related domain (i.e., source domain). Domain adaptation is a subtopic of transfer learning, which constructs knowledge transfer from the labeled source domain to the unlabeled target domain by learning domain-invariant and label-discriminative knowledge representations that manifest similarities between domains under significant differences. To date, domain adaptation has been successfully applied in various fields, such as text sentiment classification [4, 5], image classification [6–8], human activity classification [9], and multi-language text classification [10].

✉ Hongwei Yang
yanghongwei@hit.edu.cn

Hui He
hehui@hit.edu.cn

Weizhe Zhang
wzzhang@hit.edu.cn

Yawen Bai
baiyawen@hit.edu.cn

Tao Li
taolee@hit.edu.cn

1 School of Cyberspace Science,
 Harbin Institute of Technology, Harbin 150001, China

2 Cyberspace Security Research Center, Pengcheng Laboratory,
 Shenzhen 518055, China

Domain divergence poses a major obstacle for adapting predictive models across domains.

The main problem of domain adaptation is the reduction of distribution divergence between domains. To this end, existing approaches can be categorized into four main groups [2, 3]: (a) *instance-based adaptation*, which reweights samples in the source domain or in both domains to reduce domain discrepancy [11, 12], (b) *feature representation-based adaptation*, which learns feature representations to minimize domain shift or learning task error or both [13, 14], (c) *classifier-based adaptation*, which aims to learn a new model that minimizes the generalization error in the target domain via training data from both domains [15, 16], and (d) *hybrid knowledge-based adaptation*, which transfers more than one kind of knowledge, such as joint instance and feature representation-based adaptation [17–20], joint instance and classifier-based adaptation [21, 22], or joint feature representation and classifer-based adaptation [23–25].

Among the abovementioned classical approaches, the hybrid methods perform better than the single methods in reducing the cross-domain discrepancy. Most existing hybrid methods follow a two-step procedure: first, either instance reweighting or feature representation is performed independently, finally, the cross-domain classifier is trained separately, but these methods do not perform well in practical applications, and many factors cannot to be considered. For example, some methods are significantly influenced by feature representations or irrelevant instances, some ignore the importance of evaluating data distributions, and some fail to exploit hidden knowledge structures in data labels of the source and target domains. Therefore, a new hybrid method for robust unsupervised domain adaptation needs to be developed. Knowledge that can be successfully transferred across domains should be (1) invariant to feature representations and unbiased to irrelevant instances, (2) quantitatively estimated in terms of the importance of distributions, and (3) able to exploit the potential manifold structural features behind the data.

As far as we know, no research has optimized all the three challenges together in a unified learning machine for unsupervised domain adaptation. In this paper, we complete this challenge and propose a new Lie Group Manifold Analysis (LGMA) method based on FLDA [26], which learns a domain-invariant and label-discriminative classifier in Lie algebra manifold space by extracting invariant representations, estimating unbiased instance weights, performing evaluated distribution alignment and graph Laplacian regularization that jointly minimize the cross-domain distribution discrepancy. To the best of our knowledge, LGMA is the first attempt to minimize the cross-domain discrepancy in Lie algebra manifold space for domain adaptation. Extensive experiments on five real-world benchmark datasets validate that LGMA can outperform competitive state-of-the-art methods.

The rest of the paper is organized as follows. Section 2 introduces related works of domain adaptation. Section 3 presents the LGMA algorithm based on Lie algebra transformation. Section 4 provides experiments to illustrate the effectiveness and efficiency of the proposed method. Section 5 draws the conclusions of this paper.

2 Related work

According to a recent survey [2], existing domain adaptation problems can be roughly divided into four categories according to research methods: *instance*, *feature representation*, *classifier*, and *hybrid knowledge-based* adaptations.

Instance-based adaptation methods aim to minimize the cross-domain distribution discrepancy by reweighting the source samples according to the related samples in the target domain. Baktashmotlagh et al. [27] introduced a sample selection method and a subspace-based method by using the structure of Riemannian manifold to compare the source and target distributions. Transfer component analysis (TCA) [28] learns transfer components across domains in a reproducing kernel Hilbert space (RKHS) using maximum mean discrepancy (MMD) [29].

Feature representation-based adaptation methods aim to reduce distribution differences by learning a new feature representation. Fernando et al. [30] proposed a subspace alignment (SA) algorithm by learning a mapping function that aligns the source subspace with the target one. Geodesic flow kernel (GFK) [31] extends the concept of sampling points in manifold [32], and a method for learning the GFK between domains is proposed. Generalized unsupervised manifold alignment (GUMA) [33] is proposed as a method to build the connections between domains without any known correspondences by using manifold alignment. Low-rank transfer subspace learning (LTSL) [34] is proposed as a novel framework to solve transfer learning problem through subspace learning and low-rank representation constraints. Zhai et al. [35] proposed a novel manifold alignment method by learning the underlying common manifold with supervision from the corresponding data pairs of different observation sets.

Classifier-based adaptation methods aim to learn a new domain-invariant classifier that minimizes the generalization error in the target domain via training data from both domains. The works of distribution matching machine (DMM) [36] and adaptation regularization transfer learning (ARTL) [6] aim to learn a unified domain-invariant classifier based on structural risk minimization (SRM) [37].

Hybrid knowledge-based adaptation methods aim to learn domain-invariant knowledge by jointly utilizing

multiple kinds of adaptations. Locality preserving joint transfer (LPJT) [19], domain invariant and class discriminative feature learning (DICD) [17], and transfer independently together (TIT) [20] jointly leverage instance-based and feature representation-based adaptations to learn domain-invariant and label-discriminative vector representations. Qin et al. [21] proposed a novel generatively inferential co-training (GICT) framework based on instance-based and classifier-based adaptations. In [25], three unsupervised transfer learning methods, i.e., discriminative subspace learning (DSL), joint geometrical and statistical distribution adaptation (GSDA), and joint subspace and distribution adaptation (DSL-GSDA) are proposed to transfer the common domain-invariant knowledge from the source domain to the target domain by jointly adapting feature representation and classifier.

Formally, the single adaptation methods explore instance reweighting, feature representation, or classifier learning independently, which are ineffective when the domain difference is substantially large. While hybrid knowledge-based adaptation methods perform better than single adaptation methods when the domain differences are large or some outlier source instances are unrelated to the target domain or the two conditions hold. Almost all the existing adaptation methods on image classification tasks proceed by linearizing the images, which makes an implicit Euclidean space assumption [38, 39]. However, when the domain divergence is extremely large, the classification performance of the adaptation method based on the assumption of Euclidean space will be degraded significantly. In general, most of the transformations used in image classification tasks have matrix Lie group structure. Thus, we first devise a nonlinear transformation to project samples in the original Lie group manifold space onto a corresponding Lie algebra manifold space, where the samples are more discriminative and can be classified more easily. Finally, we perform hybrid knowledge-based adaptation to further minimize the domain discrepancy between domains for higher cross-domain classification accuracy.

The most similar approaches to the proposed hybrid method LGMA are scatter component analysis (SCA) [40] and joint geometrical and statistical alignment (JGSA) [41]. However, LGMA differs significantly from SCA and JGSA in two key aspects: (a) LGMA jointly learns the invariant cross-domain classifier and transferable knowledge (invariant to feature representations) in a learning paradigm in a linear Lie algebra manifold space, whereas SCA and JGSA learn the transferable knowledge and transfer classifier in a nonlinear Lie group manifold space (reproducing kernel Hilbert space). (b) LGMA learns unbiased instance reweighting and unbiased to irrelevant instances not only by using the domain scatters but also

by exploiting the weighted distribution alignment and the graph Laplacian regularization, whereas SCA and JGSA learn reweighting by scatters or unweighted distribution alignment. In summary, the proposed LGMA approach can jointly learn the cross-domain classifier and transferable knowledge with statistical and geometrical guarantees.

3 LGMA

In this section, we provide the LGMA approach in detail.

3.1 Problem definition

We begin with the formalized definition of domain adaptation [6, 42, 43]. For clarity, the frequently used notations are summarized in Table 1.

Definition 1 (Domain adaptation). A labeled source domain $\mathcal{D}_s = \{x_{s_i}, y_{s_i}\}_{i=1}^n$ and an unlabeled target domain $\mathcal{D}_t = \{x_{t_j}\}_{j=n+1}^{n+m}$, we assume the feature space $\mathcal{X}_s = \mathcal{X}_t$ and the label space $\mathcal{Y}_s = \mathcal{Y}_t$. However, the marginal probability distribution $P_s(x_s) \neq P_t(x_t)$ with the conditional probability distribution $Q_s(y_s|x_s) \neq Q_t(y_t|x_t)$. The purpose of unsupervised domain adaptation is to learn a classifier $f : x_t \mapsto y_t, \ y_t \in \mathcal{Y}_t$ to classify the samples for the target domain $\mathcal{D}_t$ using the related label information in the source domain $\mathcal{D}_s$. Data in the source and target domains can be denoted as $X_s \in \mathbb{R}^{D \times n}, X_t \in \mathbb{R}^{D \times m}$, respectively.

Classical Fisher's linear discriminant analysis (FLDA) [26] can be represented as

$$\arg \max_v J(v) = \frac{v^{\mathrm{T}} S_b v}{v^{\mathrm{T}} S_w v} \tag{1}$$

where S_b and S_w are the matrices of between-class and within-class scatter, respectively. Maximizing FLDA increases the separation of samples with respect to the class cluster. However, the classification accuracy will be affected due to the different distributions between $\mathcal{D}_s$ and $\mathcal{D}_t$. Thus,

Table 1 Notations and corresponding descriptions used in this paper

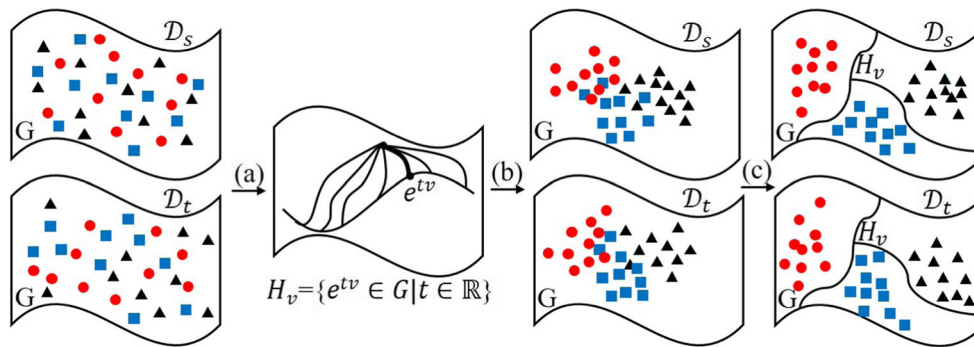| Notations	Descriptions
$\mathcal{D}_s, \mathcal{D}_t$	source/target domain
X_s, X_t	source/target samples
A, B	source/target projection matrix
$\mathfrak{g}, G$	Lie algebra/group space
n, m	the number of samples in source/target domain
β, δ, μ	regularization parameters
C	the number of shared classes
k	the number of subspaces

Fig. 1 The main idea of LGMA. (a) Features in the Lie group manifold space are mapped to Lie algebra manifold space (A projection point is the intersection of the black bold curve and the geodesic e^{tv}). Data with similar manifold properties are aggregated together after Lie algebra transformation. (b) LGMA finds a paired transformation (one for source domain, and another one for target domain) to obtain new representations of respective domains. (**c**) Weighted distribution alignment and manifold alignment are performed in Lie algebra manifold space to learn the cross-domain invariant classifier f

minimizing the domain distribution discrepancy is the only way to improve classification performance when learning a cross-domain classifier f.

3.2 Main idea

LGMA mainly includes three steps. First, LGMA performs Lie algebra transformation to project features in Lie group manifold space onto the corresponding Lie algebra manifold space. Second, LGMA finds a paired transformation (i.e., A and B, A and B for source domain and target domain, respectively) to obtain new representations of respective domains. Third, LGMA performs weighted distribution alignment and manifold alignment to learn a cross-domain invariant classifier in the linear Lie algebra space. Figure 1 shows the main idea of the proposed LGMA method.

We first obtain the transformed features by means of Lie algebra transformation. Then, on the basis of FLDA and weighted distribution alignment and manifold alignment [41], the domain-invariant classifier f can be represented as

$$\max_{A,B} \frac{\alpha S_f(\mathcal{D}_t) + \beta S_{bf}(\mathcal{D}_s)}{\bar{D}_f(\mathcal{D}_s, \mathcal{D}_t) + \delta R_f(\mathcal{D}_s, \mathcal{D}_t) + \lambda D_f(S_A, S_B) + \beta S_{wf}(\mathcal{D}_s)} \tag{2}$$

where the terms $S_f(\cdot)$, $S_{bf}(\cdot)$, $\bar{D}_f(\cdot, \cdot)$, $R_f(\cdot, \cdot)$, $D_f(\cdot, \cdot)$, and $S_{wf}(\cdot)$ represent the domain variance, the between-class variance, the weighted distribution alignment, the graph Laplacian regularization, the subspace divergence, and the within-class variance, respectively. α, β, δ, and λ are the regularization parameters.

3.3 Lie algebra transformation

Lie algebra transformation serves as the preprocessing step that aims to find a geodesic on the Lie group manifold and

project all features onto this geodesic and perform weighted distribution and manifold alignment thereafter to maximize the ratio of (2).

Before Lie algebra transformation is introduced, we first elaborate the definition of Lie group and Lie algebra [44, 45].

Definition 2 (Lie group). A real Lie group [44] is a group that is also a finite-dimensional real smooth manifold, in which the group operations of multiplication and inversion are smooth maps. Smoothness of the group multiplication $\mu : G \times G \rightarrow G$ $\mu(x, y) = xy$ means that μ is a smooth mapping of the product manifold $G \times G$ into G. These two requirements can be combined to the single requirement that the mapping $(x, y) \mapsto x^{-1}y$ be a smooth mapping of the product manifold into G.

Definition 3 (Lie algebra). A Lie algebra [45] is a vector space $\mathfrak{g}$ over some field $\mathbb{F}$ togeter with a binary operation $[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}$ called the Lie bracket that satisfies the following axioms:

- Bilinearity: $[ax + by, z] = a[x, z] + b[y, z]$, $[z, ax + by] = a[z, x] + b[z, y]$ for all scalars a, b in $\mathbb{F}$ and all elements x, y, z in $\mathfrak{g}$.
- Alternativity: $[x, x] = 0$ for all x in $\mathfrak{g}$.
- The Jacobi identity: $[x, [y, z]] + [z, [x, y]] + [y, [z, x]] = 0$ for all x, y, z in $\mathfrak{g}$.
- Anticommutativity: $[x, y] = -[y, x]$ for all x, y in $\mathfrak{g}$.

Exponential and logarithmic transformations [45] are important theories in Lie group, and exponential transformation can be defined as

$$\exp : \mathfrak{g} \rightarrow G, \quad \exp(x) = \sum_{i=0}^{\infty} \frac{x^i}{i!} \tag{3}$$

Elements in Lie algebra manifold space can be transformed into Lie group manifold space through this transformation. Similarly, logarithmic transformation can also be represented as

$$\log : G \rightarrow \mathfrak{g}, \quad \log(x) = \sum_{i=0}^{\infty} \frac{(-1)^{i-1}}{i}(x-e)^i \tag{4}$$

Features in Lie group manifold space can be transformed into Lie algebra manifold space through this transformation.

We denote $g(\cdot)$ as the Lie algebra transformation. Thus, the feature in the Lie group manifold space can be transformed into Lie algebra manifold space through $z = g(x)$.

3.4 Target domain variance maximization

The variance of the target domain can be maximized in the corresponding subspace to avoid projecting features onto some irrelevant dimensions. Therefore, the variance maximization term can be generalized as

$$\max_{B} S_f(\mathcal{D}_t) = \max_{B} \mathrm{tr}(B^T S_t B) \tag{5}$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix and

$$S_t = Z_t H_t Z_t^T \tag{6}$$

is the scatter matrix of the target domain, Z_t is the set of projected target samples, $H_t = I_t - \frac{1}{m} 1_t 1_t^T$ is the centering matrix, and $1_t \in \mathbb{R}^m$ is the column vector with all elements equal to 1.

3.5 Source domain discriminative feature preservation

We use the rich label information in the source domain to make the new representation of samples in the source domain discriminative as follows:

$$\max_{A} S_{bf}(\mathcal{D}_s) = \max_{A} \mathrm{tr}\left(A^T S_b A\right) \tag{7}$$

$$\min_{A} S_{wf}(\mathcal{D}_s) = \min_{A} \mathrm{tr}\left(A^T S_w A\right) \tag{8}$$

where S_b and S_w are the between-class and within-class scatter matrices, respectively, and are defined as follows:

$$S_w = \sum_{c=1}^{C} Z_s^{(c)} H_s^{(c)} \left(Z_s^{(c)}\right)^T \tag{9}$$

$$S_b = \sum_{c=1}^{C} n^{(c)} \left(m_s^{(c)} - \bar{m}_s\right) \left(m_s^{(c)} - \bar{m}_s\right)^T \tag{10}$$

where $Z_s^{(c)}$ indicates the set of transformed source samples that belong to class c, $m_s^{(c)} = \frac{1}{n^{(c)}} \sum_{i=1}^{n^{(c)}} z_{s_i}^{(c)}$, $\bar{m}_s = \frac{1}{n} \sum_{i=1}^{n} z_{s_i}$, and $H_s^{(c)} = I_s^{(c)} - \frac{1}{n^{(c)}} 1_s^{(c)} \left(1_s^{(c)}\right)^T$ is the

centering matrix of samples within class c, $I_s^{(c)} \in \mathbb{R}^{n^{(c)} \times n^{(c)}}$ is the identity matrix, $1_s \in \mathbb{R}^{n^{(c)}}$ is a column vector with all ones, and $n^{(c)}$ is the number of source samples in class c.

3.6 Weighted distribution alignment

Weighted distribution alignment is devised to minimize the distribution divergence between the source and target domains by quantitatively assessing the importance of the marginal distribution (i.e., P) and the conditional distribution (i.e., Q). Formally, the weighted distribution alignment $\bar{D}_f(\mathcal{D}_s, \mathcal{D}_t)$ can be defined as follows:

$$\bar{D}_f(\mathcal{D}_s, \mathcal{D}_t) = (1 - \mu)D(P_s, P_t) + \mu D(Q_s, Q_t) \tag{11}$$

with $\mu \in [0, 1]$ as the adaptive parameter. The projected MMD [6, 46, 47] methods can be adopted to compute the marginal and conditional distributions, which compare the different distributions on the basis of distance between the sample means of the two domains in the low-dimensional smooth manifold. The marginal distribution divergence $D(P_s, P_t)$ can be detailed as

$$\| \frac{1}{n} \sum_{z_{s_i} \in Z_s} A^T z_{s_i} - \frac{1}{m} \sum_{z_{t_j} \in Z_t} B^T z_{t_j} \|_F^2 \tag{12}$$

Correspondingly, the conditional distribution divergence $D(Q_s, Q_t)$ can be expressed as

$$\sum_{c=1}^{C} \| \frac{1}{n^{(c)}} \sum_{z_{s_i} \in Z_s^{(c)}} A^T z_{s_i} - \frac{1}{m^{(c)}} \sum_{z_{t_j} \in Z_t^{(c)}} B^T z_{t_j} \|_F^2 \tag{13}$$

where $Z_s^{(c)} = \left\{ z_{s_i} : z_{s_i} \in Z_s \wedge y(z_{s_i}) = c \right\}$ is the projected source samples that belong to class c and $y(z_{s_i})$ is the true label of z_{s_i}. $Z_t^{(c)} = \left\{ z_{t_j} : z_{t_j} \in Z_t \wedge \hat{y}(z_{t_j}) = c \right\}$ is the set of projected target samples that belong to class c, $\hat{y}(z_{t_j})$ is the true label of z_{t_j}, and $n^{(c)} = |Z_s^{(c)}|$, $m^{(c)} = |Z_t^{(c)}|$ are the number of samples in class c in respective projected manifold spaces of the source and target domains. The evaluation of conditional distribution divergence $D(Q_s, Q_t)$ is relative difficult because there is no labeled data are in the target domain. Long et al. [6] proposed to utilize the pseudo labels of the target domain which predicted by some supervised approaches (e.g., KNN) trained on the data in the source domain. The pseudo labels can be refined iteratively to minimize the difference in conditional distributions between the source and target domains. Thus, we follow this idea to further reduce the conditional MMD between domains.

Thus, combining the marginal and conditional MMDs together, the final weighted distribution alignment optimization can be stated in the following matrix form

$$\min_{A,B} \bar{D}_f(\mathcal{D}_s, \mathcal{D}_t) = \min_{A,B} \mathrm{tr}\left(\begin{bmatrix} A^T & B^T \end{bmatrix} \begin{bmatrix} M_{ss} & M_{st} \\ M_{ts} & M_{tt} \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix}\right)$$
(14)

where

$$M_{ss} = Z_s\left((1-\mu)N_{ss} + \mu\sum_{c=1}^{C} N_{ss}^{(c)}\right)Z_s^T, \ N_{ss} = \frac{1}{n^2}1_n 1_n^T,$$
$$\left(N_{ss}^{(c)}\right)_{ij} = \begin{cases} \frac{1}{(n^{(c)})^2}, & z_i, z_j \in Z_s^{(c)} \\ 0, & \text{otherwise} \end{cases}$$
(15)

$$M_{tt} = Z_t((1-\mu)N_{tt} + \mu\sum_{c=1}^{C} N_{tt}^{(c)})Z_t^T, \ N_{tt} = \frac{1}{m^2}1_m 1_m^T,$$
$$\left(N_{tt}^{(c)}\right)_{ij} = \begin{cases} \frac{1}{(m^{(c)})^2}, & z_i, z_j \in Z_t^{(c)} \\ 0, & \text{otherwise} \end{cases}$$
(16)

$$M_{st} = Z_s((1-\mu)N_{st} + \mu\sum_{c=1}^{C} N_{st}^{(c)})Z_t^T, \ N_{st} = -\frac{1}{nm}1_n 1_m^T,$$
$$(N_{st}^{(c)})_{ij} = \begin{cases} -\frac{1}{n^{(c)}m^{(c)}}, & z_i \in Z_s^{(c)}, z_j \in Z_t^{(c)} \\ 0, & \text{otherwise} \end{cases}$$
(17)

$$M_{ts} = Z_t((1-\mu)N_{ts} + \mu\sum_{c=1}^{C} N_{ts}^{(c)})Z_s^T, \ N_{ts} = -\frac{1}{nm}1_m 1_n^T,$$
$$(N_{ts}^{(c)})_{ij} = \begin{cases} -\frac{1}{n^{(c)}m^{(c)}}, & z_j \in Z_s^{(c)}, z_i \in Z_t^{(c)} \\ 0, & \text{otherwise} \end{cases}$$
(18)

3.7 Graph Laplacian regularization

In this section, we use graph Laplacian regularization to guarantee the unbiased problem of irrelevant instances.

In domain adaptation, labeled and unlabeled data are used. It is expected that knowledge of marginal distributions (i.e., P_s and P_t) can be further exploited to improve the performance of function learning. Thus, the unlabeled samples may often reveal the underlying facts of the target domain, such as sample variances. The idea of manifold assumption [48] can be expressed as follows. If two points, namely, $z_i, z_j \in g$ are close in the geometry of marginal distributions $P_s(z_s)$ and $P_t(z_t)$, then the conditional distributions $Q_s(y_s|z_s)$ and $Q_t(y_t|z_t)$ are similar. Under the hypothesis of the smooth properties of geodesics, Laplacian regularization can be used for further exploiting the similar geometrical properties of nearest

points in Lie algebra manifold space g. Thus, the final optimization of graph Laplacian regularization $R_f(\mathcal{D}_s, \mathcal{D}_t)$ can be computed as

$$\min_{A,B} R_f(\mathcal{D}_s, \mathcal{D}_t) = \min_{A,B} \mathrm{tr}\left(\begin{bmatrix} A^T & 0 \\ 0 & B^T \end{bmatrix} \begin{bmatrix} Z_s L_{ss} Z_s^T & Z_s L_{st} Z_t^T \\ Z_t L_{ts} Z_s^T & Z_t L_{tt} Z_t^T \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}\right)$$
(19)

where $L = I - D^{-1/2}WD^{-1/2}$ is the graph Laplacian matrix and D is a diagonal matrix with its ith diagonal element calculated as the sum of ith row of W, i.e., $D_{ii} = \sum_{j=1}^{n} W_{ij}$. W is defined by

$$W_{ij} = \begin{cases} \cos(z_i, z_j), & z_i \in \mathcal{N}_p(z_j) \vee z_j \in \mathcal{N}_p(z_i) \\ 0, & \text{otherwise,} \end{cases}$$
(20)

where $\mathcal{N}_p(z_i)$ is z_i's p nearest neighbors which are from the same class with z_i.

3.8 Subspace divergence minimization

In this section, we further mitigate the domain divergence by moving the source and target subspaces closer together, which is similar to the aforementioned methods, such as transfer component analysis (TCA) [28] or joint distribution alignment (JDA) [42]. The differences in the two domains will be reduced but cannot be completely removed through this transformation. By contrast, we obtain the idea from [30] to minimize A and B simultaneously. In this way, the statistical and geometrical features can be preserved. Formally, we use the following minimization form of Frobenius-norm to move the two subspaces closer.

$$\min_{A,B} D_f(S_A, S_B) = \min_{A,B} \|A - B\|_F^2$$
(21)

3.9 Optimization

To control the scale of solution B, we follow [40, 41] to impose a constraint that $\mathrm{tr}(B^T B)$ is sufficiently small. We formulate the LGMA method by incorporating (5), (7), (8), (14), (19), and (21). Then, our objective function (2), therefore, can be formulated as follows:

$$\arg\max_{A,B} \frac{\mathrm{tr}\left(\begin{bmatrix} A^T & B^T \end{bmatrix} \begin{bmatrix} \beta S_b & 0 \\ 0 & \alpha S_t \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix}\right)}{\mathrm{tr}\left(\begin{bmatrix} A^T & B^T \end{bmatrix} \begin{bmatrix} M_{ss} + \delta L_{ss} + \lambda I + \beta S_w & M_{st} + \delta L_{st} - \lambda I \\ M_{ts} + \delta L_{ts} - \lambda I & M_{tt} + \delta L_{tt} + (\lambda + \alpha)I \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix}\right)}$$
(22)

where α, β, δ, and λ are penalty parameters, and $I \in \mathbb{R}^{d \times d}$ is the identity matrix.

LGMA aims to find a paired transformation A and B by solving the generalized eigendecomposition problem in the

projected Lie algebra manifold space. To optimize (22), we define $[A^T \ B^T]$ to be equal to U^T. Thus, we get

$$
\begin{aligned}
\arg\max_{U} \ & \mathrm{tr}\left(U^T \begin{bmatrix} \beta S_b & 0 \\ 0 & \alpha S_t \end{bmatrix} U\right) \\
\text{s.t.} \ & \mathrm{tr}\left(U^T \begin{bmatrix} M_{ss}+\delta L_{ss}+\lambda I+\beta S_w & M_{st}+\delta L_{st}-\lambda I \\ M_{ts}+\delta L_{ts}-\lambda I & M_{tt}+\delta L_{tt}+(\lambda+\alpha)I \end{bmatrix} U\right) = 1
\end{aligned}
\tag{23}
$$

Equivalently, the constraint optimization of (23) can be written in the form of Lagrangian. Thus, we have

$$
\begin{aligned}
L(U) = & \ \mathrm{tr}\left(U^T \begin{bmatrix} \beta S_b & 0 \\ 0 & \alpha S_t \end{bmatrix} U\right) \\
& + \mathrm{tr}\left(\left(U^T \begin{bmatrix} M_{ss}+\delta L_{ss}+\lambda I+\beta S_w & M_{st}+\delta L_{st}-\lambda I \\ M_{ts}+\delta L_{ts}-\lambda I & M_{tt}+\delta L_{tt}+(\lambda+\alpha)I \end{bmatrix} U - I\right) \Lambda\right)
\end{aligned}
\tag{24}
$$

To solve (24), we set the first derivative $\frac{\partial L(U)}{\partial U} = 0$. Then, we obtain generalized eigendecomposition

$$
\begin{bmatrix} \beta S_b & 0 \\ 0 & \alpha S_t \end{bmatrix} U = \begin{bmatrix} M_{ss}+\delta L_{ss}+\lambda I+\beta S_w & M_{st}+\delta L_{st}-\lambda I \\ M_{ts}+\delta L_{ts}-\lambda I & M_{tt}+\delta L_{tt}+(\lambda+\alpha)I \end{bmatrix} U\Lambda
\tag{25}
$$

where $\Lambda = \mathrm{diag}(\lambda_1, ..., \lambda_k)$ is the k leading eigenvalue and $U = [U_1, ..., U_k]$ contains the corresponding eigenvectors. Finding the optimal adaptation matrix U is decreased to solving (25) for k eigenvectors. Algorithm 1 provides a complete summary of LGMA.

3.10 Computational complexity

The computational complexity of Algorithm 1 consists of four parts as follows.

(1) The computation of S_t, S_b, and S_w in step 2.
(2) The construction of M_{ss}, M_{tt}, M_{st}, and M_{ts} in step 2.
(3) The optimization of eigendecomposition problem in step 4.
(4) The computation for all other processes.

Generally, in terms of the big O notation. The computation of S_t, S_b, and S_w cost $O(m^2)$, $O(n^2)$, and $O(n^2)$. The construction of M_{ss}, M_{tt}, M_{st}, and M_{ts} cost $O(TCn^2)$, $O(TCm^2)$, $O(TCnm)$, and $O(TCmn)$. The optimization of eigendecomposition problem costs $O(Tkm^2)$. The computation for all other processes cost $O(Tmn)$. Denote T and k as the number of iterations and the subspace bases. The overall computational costs of Algorithm 1 would be $O(T(k+C)m^2 + TCn^2 + TCmn)$.

Algorithm 1 Lie group manifold analysis.

Data: Data sets and source labels: X_s, X_t, $\mathcal{Y}_s$; Set parameters $\alpha = 1$, $\lambda = 1$, $T = 10$, k, free parameters β, μ, δ, and p.

Result: Projection matrices: A and B; Embeddings: E_s and E_t; Adaptive classifier f.

1   Perform Lie algebra transformation and obtain the Lie algebra manifold feature Z;

2   Compute S_t, S_b, S_w, M_{ss}, M_{tt}, M_{st}, and M_{ts} according to (6), (10), (9), (15), (16), (17), and (18) in Lie group manifold space; Initialize pseudo labels $\hat{y}_t$ in the target domain by using a classifier trained on the source domain data;

3   **repeat**

4   Solve the generalized eigendecomposition problem in (25) and select the k corresponding eigenvectors of k leading engenvalues as the transformation U, then obtain the subspaces A, B;

5   Map the projected data to respective subspaces to obtain the embeddings: $\begin{bmatrix} E_s & 0 \\ 0 & E_t \end{bmatrix} = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} Z_s & 0 \\ 0 & Z_t \end{bmatrix}$;

6   Train a classifier f on $\{E_s, \mathcal{Y}_s\}$ to update pseudo labels in the target domain $\hat{y}_t = f(E_t)$;

7   Update M_{ss}, M_{tt}, M_{st}, and M_{ts} according to (15), (16), (17), and (18);

8   **until** *Convergence*;

9   **return** *Obtain the final adaptive classifier f on* $\{E_s, \mathcal{Y}_s\}$.

---

4 Experiments

In this section, we perform extensive experiments on real-world image recognition datasets to evaluate the proposed LGMA approach against the state-of-the-art methods. The experiments are divided into three parts. Section 4.1 visualizes performance on image classification tasks. Section 4.2 evaluates the performance on a range of cross-domain image classification tasks with a standard and realistic hyper-parameter tuning. Section 4.3 reports the results with a tuning protocol established in the literature for completeness.

4.1 Feature visualization

Figure 2a, b, e, f, c, d, g and h show the visualization of transfer tasks V→ I and A→W after performing SCA, JGSA, and LGMA algorithms, respectively. Some interesting conclusions can be drawn. (a) SCA can not learn the invariant cross-domain features well because the differences between the source domain and target domain are still large. (b) JGSA does not learn the weighted distribution alignment because the distribution of the source
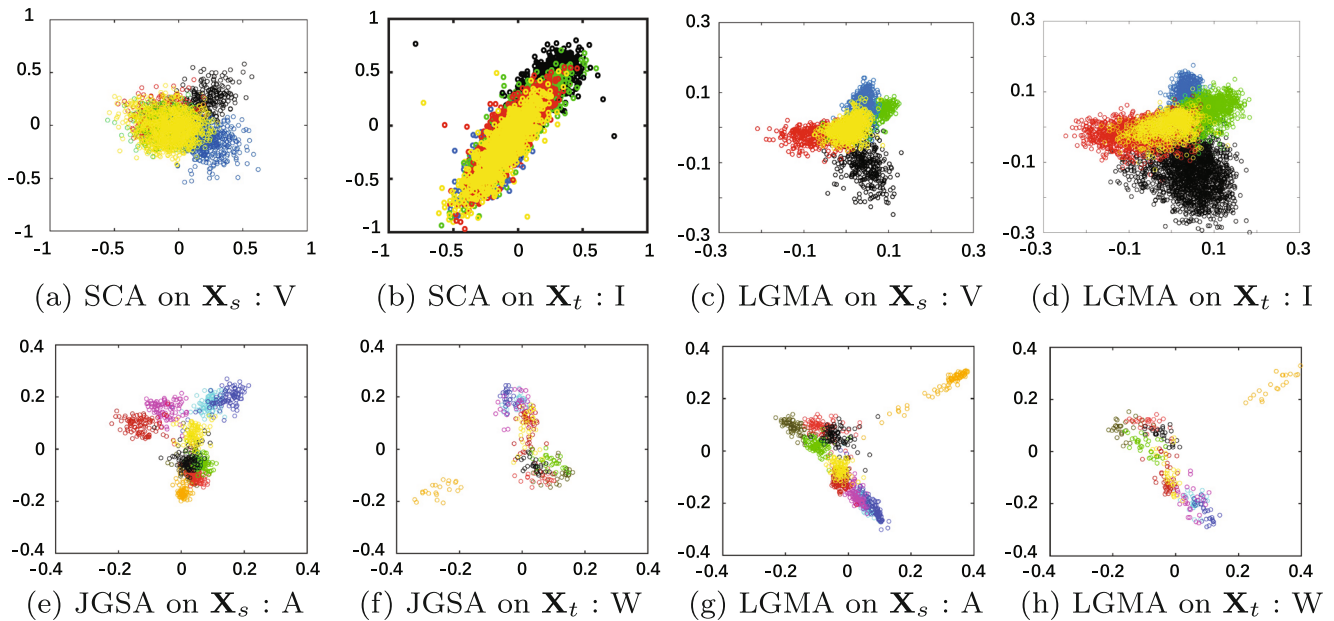
(a) SCA on $\mathbf{X}_s$: V (b) SCA on $\mathbf{X}_t$: I (c) LGMA on $\mathbf{X}_s$: V (d) LGMA on $\mathbf{X}_t$: I

(e) JGSA on $\mathbf{X}_s$: A (f) JGSA on $\mathbf{X}_t$: W (g) LGMA on $\mathbf{X}_s$: A (h) LGMA on $\mathbf{X}_t$: W

Fig. 2 Feature visualization of source and target domain data. (a) and (b) indicate the visualization of the source domain V and the target domain I after performing SCA, respectively. (c) and (d) indicate the visualization of the source domain V and the target domain I after performing LGMA, respectively. (e) and (f) indicate the visualization of

the source domain A and the target domain W after performing JGSA, respectively. (g) and (h) indicate the visualization of the source domain A and the target domain W after performing LGMA, respectively. Color makers denote different classes

domain are dissimilar to the target domain, thereby leading to large domain bias. The abovementioned conclusions show the inferior performance of SCA and JGSA and validate the superiority of LGMA.

4.2 Real world object recognition

4.2.1 Experimental setup

Five public large-scale image datasets are used, as shown in Table 2.

The public large-scale image recognition datasets in our experiments include Office+Caltech10, Office-31, and ImageNet+VOC2007, which are popular image classification datasets that are widely used for evaluating machine learning and data mining models, such as [6, 31, 41].

Office+Caltech10 [49] contains 2,533 images from 10 different subcategories. The dataset includes 4 image

domains, i.e., **Amazon** (A), **DSLR** (D), **Webcam** (W), and **Caltech** (C). Figure 3 depicts the sample images from the object monitor category in the four domains, namely, **Caltech**, **Amazon**, **DSLR**, and **Webcam** [31]. Features in Office and Caltech follow different distributions, domain adaptation can help the performance of cross-domain image classification. Formally, 10 classes are used in each dataset. Thus, 12 tasks are constructed, namely, A→C, A→D, A→W,..., D→W. In this study, A→B represents the transfer task from the source domain A to the target domain B.

Office-31 [49] is also a widely used dataset for transfer learning tasks in image recognition and multimedia analysis. It includes 4,652 images and 31 categories from three domains: **Amazon** (A), **Webcam** (W), and **DSLR** (D). Each of these two domains can construct a transfer learning task, thereby leading to 6 tasks: A→D, A→W ,..., and W→D, respectively.

ImageNet+VOC2007 (I, V) are another widely used image datasets. Because images from the same classes of both domains follow different distributions, each dataset can be considered one domain. In this paper, we use the datasets in [50] to perform transfer learning tasks. Both of the two datasets have five classes, namely, *bird*, *cat*, *chair*, *dog*, and *person*, respectively. Thus, another two transfer learning tasks, i.e., I→V and V→I, are constructed.

Table 2 Five benchmark datasets used in this paper

Datasets names	Data	Features	Classes	Domain(s)
Office-10	1,410	800 (4,096)	10	A, W, D
Caltech-10	1,123	800 (4,096)	10	C
Office-31	4,652	4,096	31	A, W, D
ImageNet	7,341	4,096	5	ImageNet(I)
VOC2007	3,376	4,096	5	VOC(V)

(a) **Caltech** (b) **Amazon** (c) **DSLR** (d) **Webcam**

Fig. 3 Sample images from object monitor category in the four domains **Caltech**, **Amazon**, **DSLR**, and **Webcam** [31]

For all the baseline approaches, we use the optimal parameters reported in the original papers. As for LGMA, we set $\lambda = 1$ and $\alpha = 1$, such that the inner subspace bias and the target variance are treated as equally important. The subspace dimension $k = 30$ in the tasks of Office+Caltech10 datasets with $DeCaf_6$ features and the tasks of ImageNet+VOC2007 datasets, and the subspace dimension $k = 100$ in the tasks of Office-31 datasets with $DeCaf_7$ features. We empirically validate that the fixed parameters can obtain promising performance on different types of tasks. Therefore, the weighted coefficient μ, the regularization parameter β, the number of iteration T, the number of nearest neighbors p, and the coefficient of the graph Laplacian regularization term δ are free parameters.

We also exploit classification *Accuracy* on test data as the evaluation metric, which is widely used in many studies [28, 31, 47]:

$$Accuracy = \frac{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_t \wedge \hat{y}(\mathbf{x}) = y(\mathbf{x})|}{|\mathbf{x} : \mathbf{x} \in \mathcal{D}_t|}, \tag{26}$$

where $y(\mathbf{x})$ and $\hat{y}(\mathbf{x})$ indicate the truth and predicted labels in the target domain, respectively.

4.2.2 Baselines

To evaluate the robustness of the proposed LGMA approach to different configurations of datasets, we conduct comprehensive evaluation on image recognition datasets and compare LGMA with competitive state-of-the-art domain adaptation methods as follows:

– 1-Nearest neighbor (**1NN**) classifier;
– Support vector machine (**SVM**) [51];
– Transfer component analysis (**TCA**) [28], which adapts marginal distribution;
– Transfer joint matching (**TJM**) [52], which performs marginal distribution with the sample selection of the source domain;
– Distribution matching machine (**DMM**) [36], which aims to learn an SVM classifier to adapt distributions alignment based on SRM;
– Scatter component analysis (**SCA**) [40], which leans a classifier through scatter component analysis;
– Joint geometrical and statistical alignment (**JGSA**) [41], which performs geometrical and statistical alignment with label propagation.

Table 3 Recognition accuracy(%) against other baseline methods on Office+Caltech10 (DeCaf6) datasets, the best results and the best baseline results are shown in boldface and italic, respectively

Tasks	SVM	TCA	TJM	SCA	JGSA	DMM	DICD	TIT	LPJT	LGMA
C→ A	91.6	89.8	88.8	89.5	91.4	92.4	91.0	90.9	*92.7*	**93.2**
C→W	80.7	78.3	81.4	85.4	86.8	87.5	**92.2**	*89.3*	87.8	89.2
C→ D	86.0	85.4	84.7	87.9	**93.6**	90.4	**93.6**	87.5	89.8	*91.1*
A→C	82.2	82.6	84.3	78.8	84.9	84.8	86.0	*86.7*	86.6	**88.9**
A→W	71.9	74.2	71.9	75.9	81.0	84.7	81.4	84.2	*87.5*	**88.8**
A→D	80.9	81.5	76.4	85.4	88.5	**92.4**	83.4	84.9	88.5	*91.7*
W→C	67.9	80.4	83.0	74.9	*85.0*	81.7	84.0	83.4	82.6	**87.1**
W→A	73.4	84.1	87.6	85.0	*90.7*	86.5	89.7	88.7	*90.7*	**92.6**
W→D	**100**	**100**	**100**	**100**	**100**	98.7	**100**	99.0	*99.4*	97.5
D→C	72.8	71.4	83.8	78.1	*86.2*	83.3	86.1	84.3	80.0	**88.1**
D→A	78.7	76.3	90.3	90.0	92.0	90.7	*92.2*	90.7	90.9	**92.9**
D→W	98.3	99.3	99.3	98.6	*99.7*	99.3	99.0	99.5	**100**	94.2
Average	82.0	85.6	86.0	85.9	*90.0*	89.4	89.9	89.1	89.7	**91.3**

Table 4 Recognition accuracy(%) against other baseline methods on Office-31 (DeCaf7) datasets, the best results and the best baseline results are shown in boldface and italic, respectively

Tasks	SVM	1NN	TCA	TJM	SCA	JGSA	DICD	TIT	LPJT	LGMA
A→D	55.7	59.2	45.4	55.6	60.2	67.1	65.8	*67.3*	66.9	**68.5**
A→W	50.6	55.6	40.5	51.2	55.7	63.1	64.6	**65.9**	65.1	*65.7*
D→W	93.1	93.1	78.2	88.9	95.0	**95.9**	92.5	95.3	*95.8*	91.8
D→A	46.5	44.9	36.5	42.7	48.6	52.6	50.4	51.1	*53.5*	**54.1**
W→D	97.4	98.2	84.0	95.6	**99.0**	**99.0**	*98.3*	94.4	**99.0**	97.4
W→A	43.0	41.7	34.1	43.2	46.6	51.5	*52.9*	50.2	52.3	**57.6**
Average	64.4	65.5	53.1	62.9	67.5	71.5	70.8	70.7	*72.1*	**72.5**

- Unsupervised transfer metric learning (**UTML**) [18], which decreases intra-class distance and increases inter-class distance;
- Locality preserving joint transfer (**LPJT**) [19], which jointly exploits feature adaptation with distribution matching and sample adaptation with landmark selection;
- Domain invariant and class discriminative feature learning (**DICD**) [17], which matches the marginal and conditional distributions, and maximizes the inter-class dispersion and minimizes the intra-class scatter;
- Transfer independently together (**TIT**) [20], which learns multiple transformations for each domain to map data onto a shared latent space where the domains are well aligned.

4.2.3 Experimental results and analysis

The classification performance of all comparison models on the 12 transfer tasks of Office+Caltech10 datasets with $DeCaf_6$ features, the 6 transfer tasks of Office-31 datasets with $DeCaf_7$ features, and the 2 transfer tasks of ImageNet+VOC2007 datasets are shown in Tables 3, 4, and 5, respectively. LGMA considerably outperforms the competitive baseline methods on most of the transfer tasks. Specifically, LGMA achieves the following performance gains compared with the best baselines: (1) 1.3% on the 12 transfer tasks of Office+Caltech10 datasets with $DeCaf_6$ features, (2) 0.4% on the 6 transfer tasks of Office-31 datasets with $DeCaf_7$ features, and (3) 6.1% on the 2 transfer tasks of ImageNet+VOC2007 datasets.

Although LGMA cannot perform the best on all tasks, if LGMA performs the best, then it usually performs considerably better than the best baseline approach; otherwise, it performs only slightly worse than the optimal baseline. This finding demonstrates that LGMA is robust to feature shift and instance bias for domain adaptation.

We can make more observations. (1) Domain adaptation methods (i.e., instance-based adaptation, feature representation-based adaptation, classifier-based adaptation, and hybrid knowledge-based adaptation methods) are generally superior to SVM and 1NN, which indicates that minimizing the distribution differences is the key to domain adaptation. (2) Classifier-based adaptation DMM method outperforms TCA, thereby showing the effectiveness of minimizing the distribution differences based on SRM in the infinite dimension reproducing kernel Hilbert space (DMM) rather than in the dimension reduced kernel PCA space (TCA). (3) Hybrid knowledge-based adaptation methods (i.e., SCA, JGSA, TIT, LPJT, UTML, DICD and LGMA) further outperform TCA and other single methods, whereas LGMA performs the best in most transfer tasks. Only single knowledge-based adaptation methods are insufficiently good for domain adaptation when the domain discrepancy is substantially large. The reason is that some source samples which are irrelevant to the target samples are not helpful for learning a unified classifier even when using the cross-domain invariant features or the high dimensional nonlinear features or both. LGMA addresses this limitation by reweighting the source instances according to their relevance to the target instances and performing weighted

Table 5 Recognition accuracy(%) against other baseline methods on ImageNet+VOC2007 datasets, the best results and the best baseline results are shown in boldface and italic, respectively

Tasks	SVM	TCA	TJM	SCA	JGSA	DICD	TIT	LPJT	UTML	LGMA
I→V	59.2	*63.7*	*63.7*	60.1	52.3	55.9	62.8	63.5	60.4	**65.1**
V→I	65.5	64.9	73.0	66.7	70.6	69.3	*76.3*	74.2	71.0	**86.2**
Average	64.4	64.3	68.4	63.4	61.5	62.6	*69.6*	68.9	65.7	**75.7**

Table 6 Recognition accuracy(%) against other baseline methods on Office+Caltech10 (SURF) datasets, the best results and the best baseline results are shown in boldface and italic, respectively

| Tasks	SVM	1NN	TJM	SCA	JGSA	UTML	DICD	LPJT	LGMA
C→ A	53.1	23.7	46.8	45.6	51.5	51.7	47.3	**60.8**	*58.3*
C→W	41.7	25.8	39.0	40.0	45.4	51.2	46.4	**55.9**	*52.7*
C→ D	47.8	25.5	44.6	47.1	45.9	**54.8**	*49.7*	46.5	48.2
A→C	41.7	26.0	39.5	39.7	41.5	*44.4*	42.4	**44.5**	43.7
A→W	31.9	29.8	42.0	34.9	45.8	*47.8*	45.1	44.4	**48.5**
A→D	44.6	25.5	45.2	39.5	47.1	42.0	38.9	**51.6**	*47.7*
W→C	28.8	19.9	30.2	31.1	33.2	**37.0**	33.6	34.2	*34.8*
W→A	27.6	23.0	30.0	30.0	39.9	*41.0*	34.1	40.4	**41.1**
W→D	78.3	59.2	89.2	87.3	90.5	**91.1**	89.8	88.5	*90.6*
D→C	26.4	26.3	31.4	30.7	29.9	**36.3**	34.6	34.3	*35.2*
D→A	26.2	28.5	32.8	31.6	38.0	36.7	34.5	**42.6**	*41.5*
D→W	52.5	63.4	85.4	84.4	**91.9**	90.5	*91.2*	86.8	91
Average	41.1	31.4	46.3	45.2	50.0	52.0	49.0	*52.5*	**52.8**

distribution alignment in the linear Lie algebra manifold space.

Although SCA, JGSA, LPJT, and DICD perform distribution matching by using hybrid knowledge based adaptation, the advantages of LGMA over these four methods are threefold. (1) LGMA corrects the domain mismatch by quantitatively evaluating the importance of the marginal and conditional distributions in the generalized FLDA framework. LGMA further performs feature matching to guarantee a large number of effective source instances for classifying the related target domain. In SCA, JGSA, LPJT, and DICD, the evaluation of distribution importance is ignored. (2) LGMA jointly learns the domain-invariant and label-discriminative transfer classifier and the transferable knowledge (invariant to feature representations and unbiased to irrelevant instances) in a learning paradigm in the nonlinear Lie group manifold space, whereas SCA, JGSA, LPJT, and DICD learn the transferable knowledge and cross-domain classifier in a linear manifold space. (3) LGMA aims to find a geodesic on the original Lie group and projects all the samples onto a Lie algebra manifold space along the geodesic direction, while ensuring the discrimination of the projected samples in a linear Lie

algebra manifold space. However, the other four methods (i.e., SCA, JGSA, LPJT, and DICD) cannot guarantee that the transformed samples are linear separable in the RKHS.

We further verify the performance of LGMA on another Office+Caltech10 datasets using SURF features, and the performance results are reported in Table 6. It is worth noting that LGMA outperforms other baselines range from traditional machine learning methods (i.e., 1NN and SVM) to state-of-the-art transfer learning models (i.e., TJM, SCA, JGSA, UTML, TIT, DICD, and LPJT), which demonstrates that LGMA is significantly superior to other baselines in minimizing the cross-domain discrepancy.

We also evaluate the importance of the Lie algebra transformation, the graph Laplacian regularization term (including the parameters p and δ), and the weighted distribution alignment factor μ, where we stand out from the baseline methods. We randomly select several tasks and show the results in Figs. 4, 5, and 6. In Fig. 4, the dotted lines represent the baseline method, the solid lines represent the proposed LGMA method. We can make additional observations. (1) The Lie algebra transformation (L), the graph Laplacian regularization (GLR), and the weighted

(a) neighbor p   (b) Parameter δ (c) Parameter μ (d) iteration T
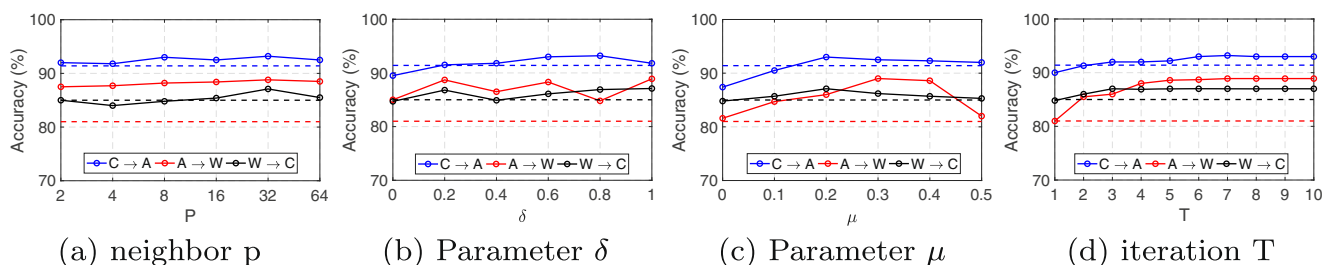
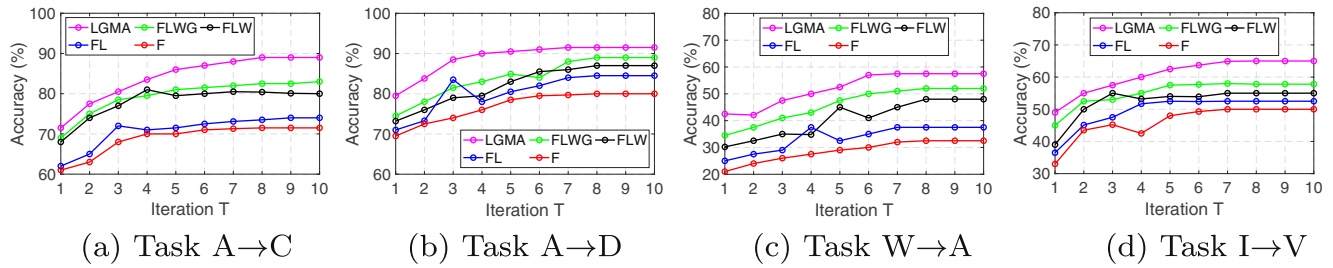**Fig. 4** The parameter sensitivity and convergence analysis of the proposed LGMA approach

Fig. 5 The recognition accuracy of methods F, FL, FLW, FLWG, and LGMA

distribution alignment (WDA) are highly important in dealing with the domain adaptation problems (Figs. 5 and 6). (2) Compared with other methods (FLDA, **F**LDA with **L**ie algebra transfermation (FL), **F**LDA with **L**ie algebra transfermation and **W**eighted distribution alignment (FLW), **F**LDA with **L**ie algebra transfermation, **W**eighted distribution alignment, and **G**raph Laplacian regularization (FLWG)), the performance of LGMA method is better, which validates the effectiveness of the proposed method. (3) LGMA can reach a steady performance in approximately $T \leqslant 10$ iterations (Fig. 4d and Fig. 5). (4) LGMA can reach a high performance using the wide range of parameters (Fig. 4a, b, and c).

The reasons for these results are presented as follows. First is that the instances in Lie group manifold space are projected onto the linear Lie algebra manifold space by Lie algebra transformation to realize the data discrimination in the nonlinear Lie group manifold space. Second is that the graph Laplacian regularization can further exploit the similar geometrical properties of the nearest points in domain adaptation. The third is that the weighted distribution alignment factor $\mu \in \{0, 0.01, ..., 0.99, 1\}$ can evaluate the importance of the marginal and conditional distributions. We do not perform experiments on the

$DeCaf_7$ features of Office-31 datasets because the results are satisfactory.

4.3 Results with parameter tuning on target domain

In this section, we analyze the parameter fluctuations of LGMA on different types of datasets to validate that a wide range of parameter values can be selected for improved performance.

We find the sensitivity of the number of the nearest neighbors p by experimenting with a large range of $p \in \{2, 4, 8, ..., 64\}$ on randomly selected tasks. From Fig. 4a and the experimental results, we can conclude that LGMA is robust in terms of $p = 32$. μ is a weight factor with the value range $\mu \in \{0, 0.01, ..., 0.99, 1\}$, and we can choose the value of μ from the analysis of Fig. 4c.

LGMA uses a wide range of values for regularization parameters β, δ, and some other necessary parameters k, T. We follow the same setup of [41] that $\beta \in [2^{-15}, 2^{-1}]$ and $k \in [20, 180]$. In this study, we set the number of iterations $T = 10$ (Fig. 4d). δ (Fig. 4b) is a factor with $\delta \in \{0, 0.01, ..., 0.99, 1\}$. We observe that LGMA can achieve robust performance for a wide range of parameter values.

In the experiment on Office+Caltech10 datasets using $DeCaf_6$ features, we set the free parameters $\beta = 0.08$, $\delta = 0.18$, and $\mu = 0.81$. In the experiment on Office-31 datasets using $DeCaf_7$ features, we set the free parameters $\beta = 0.1$, $\delta = 0.46$, and $\mu = 0.74$. In the experiment on ImageNet+VOC2007 datasets, we set the free parameters $\beta = 0.1$, $\delta = 0.11$, and $\mu = 0.81$.

5 Conclusions

In this paper, we proposed a new Lie group manifold analysis (LGMA) method for unsupervised domain adaptation. LGMA performs transformation using variances between subsets of data to suppress insignificant differences (within labels and between domains) and to amplify useful differences (between labels and overall variability) in a linear Lie algebra manifold space. In the meanwhile,
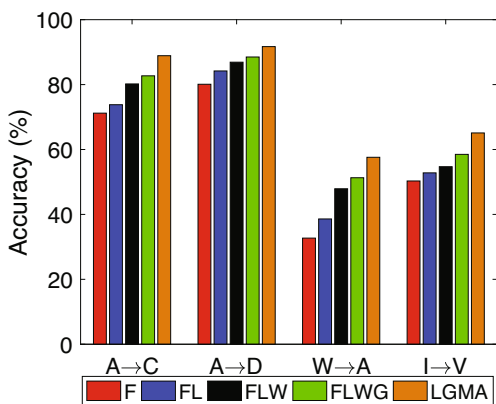

Fig. 6 Evaluate the impotance of Lie algebra transformation (L), weighted distribution alignment (W), and graph Laplacian regularization (G)

LGMA learns a invariant cross-domain classifier by extracting domain-invariant feature representations, evaluating the importance of distributions (marginal and conditional distributions), exploiting the similar geometrical properties of the nearest points, and estimating irrelevant instance weights that jointly reduce the cross-domain distribution difference. Extensive experiments on several cross-domain image datasets validate that LGMA considerably outperforms state-of-the-art domain adaptation methods.

In general, the problem of dataset bias in domain adaptation is far from being solved. The actual performance of existing approaches ($\geqslant 90\%$ accuracy) is only achieved in several cross-domain tasks, even using advanced feature extraction methods, such as $DeCaf_6$ and $DeCaf_7$ features. Using raw features is clearly not satisfactory. Therefore, it is critical to develop more robust algorithms that can significantly reduce data bias in all cases.

References

1. Kim S, Jeong M, Ko BC (2020) Energy efficient pupil tracking based on rule distillation of cascade regression forest. Sensors 20(18):5141
2. Weiss K, Khoshgoftaar TM, Wang D (2016) A survey of transfer learning. J Big Data 3(1):1–40
3. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
4. Xing FZ, Pallucchini F, Cambria E (2019) Cognitive-inspired domain adaptation of sentiment lexicons. Inf Process Manag 56(3):554–564
5. Zhao C, Wang S, Li D (2020) Multi-source domain adaptation with joint learning for cross-domain sentiment classification. Knowl-Based Syst 191:105254
6. Long M, Wang J, Ding G, Pan SJ, Yu PS (2014) Adaptation regularization: A general framework for transfer learning. IEEE Trans Knowl Data Eng 26(5):1076–1089
7. Chong Y, Peng C, Zhang C, Wang Y, Feng W, Pan S (2021) Learning domain invariant and specific representation for cross-domain person re-identification. Appl Intell:1–14
8. Lee C-Y, Batra T, Baig MH, Ulbricht D (2019) Sliced wasserstein discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 10285–10295
9. Khan MAAH, Roy N, Misra A (2018) Scaling human activity recognition via deep learning-based domain adaptation. In: 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom). IEEE, pp 1–9
10. Ziser Y, Reichart R (2018) Pivot based language modeling for improved neural domain adaptation. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp 1241–1251
11. Quanz B, Huan J, Mishra M (2012) Knowledge transfer with low-quality data: A feature extraction issue. IEEE Trans Knowl Data Eng 24(10):1789–1802
12. Si S, Tao D, Geng B (2010) Bregman divergence-based regularization for transfer subspace learning. IEEE Trans Knowl Data Eng 22(7):929–942
13. Wei P, Ke Y, Goh CK (2016) Deep nonlinear feature coding for unsupervised domain adaptation. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pp 2189–2195
14. Donahue J, Jia Y, Vinyals O, Hoffman J, Zhang N, Tzeng E, Darrell T (2014) Decaf: A deep convolutional activation feature for generic visual recognition. Int Conf Mach Learn:647–655
15. Bendavid S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW (2010) A theory of learning from different domains. Mach Learn 79(1):151–175
16. Ma Z, Yang Y, Nie F, Sebe N, Yan S, Hauptmann AG (2014) Harnessing lab knowledge for real-world action recognition. Int J Comput Vis 109(1):60–73
17. Li S, Song S, Huang G, Ding Z, Wu C (2018) Domain invariant and class discriminative feature learning for visual domain adaptation. IEEE Trans Image Process PP(99):1–1
18. Huang J, Zhou Z (2019) Transfer metric learning for unsupervised domain adaptation. IET Image Process 13(5):804–810
19. Li J, Jing M, Lu K, Zhu L, Shen HT (2019) Locality preserving joint transfer for domain adaptation. IEEE Trans Image Process 28(12):6103–6115
20. Li J, Lu K, Huang Z, Zhu L, Shen HT (2018) Transfer independently together: A generalized framework for domain adaptation. IEEE Trans Cybern 49(6):2144–2155
21. Qin C, Wang L, Zhang Y, Fu Y (2019) Generatively inferential co-training for unsupervised domain adaptation. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). IEEE, pp 1055–1064
22. Ahmadvand M, Tahmoresnezhad J (2021) Metric transfer learning via geometric knowledge embedding. Appl Intell 51:921–934
23. Hoffman J, Rodner E, Donahue J, Darrell T, Saenko K (2013) Efficient learning of domain-invariant image representations. International Conference on Learning Representations
24. Zhu F, Shao L (2014) Weakly-supervised cross-domain dictionary learning for visual recognition. Int J Comput Vis 109(1):42–59
25. Gholenji E, Tahmoresnezhad J (2020) Joint discriminative subspace and distribution adaptation for unsupervised domain adaptation. Appl Intell 50(7):2050–2066
26. Fisher RA (1936) The use of multiple measurements in taxonomic problems. Ann Eugen 7(2):179–188
27. Baktashmotlagh M, Harandi MT, Lovell BC, Salzmann M (2014) Domain adaptation on the statistical manifold. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp 2481–2488
28. Pan SJ, Tsang IW, Kwok JT, Yang Q (2010) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210
29. Ben-David S, Blitzer J, Crammer K, Pereira F (2006) Analysis of representations for domain adaptation. In: International Conference on Neural Information Processing Systems
30. Fernando B, Habrard A, Sebban M, Tuytelaars T (2014) Unsupervised visual domain adaptation using subspace alignment. In: IEEE International Conference on Computer Vision, pp 2960–2967
31. Gong B, Shi Y, Sha F, Grauman K (2012) Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp 2066–2073

32. Gopalan R, Li R, Chellappa R (2011) Domain adaptation for object recognition: An unsupervised approach. In: 2011 international conference on computer vision. IEEE, pp 999–1006

33. Cui Z, Chang H, Shan S, Chen X (2014) Generalized unsupervised manifold alignment. In: Advances in Neural Information Processing Systems, pp 2429–2437

34. Shao M, Castillo C, Gu Z, Fu Y (2012) Low-rank transfer subspace learning. In: 2012 IEEE 12th International Conference on Data Mining. IEEE, pp 1104–1109

35. Ben-David S, Blitzer J, Crammer K, Pereira F (2010) Manifold alignment via corresponding projections. In: British Machine Vision Conference, pp 1–11

36. Cao Y, Long M, Wang J (2018) Unsupervised domain adaptation with distribution matching machines. In: AAAI, pp 2795–2802

37. Vapnik VN (1999) An overview of statistical learning theory. IEEE Trans Neural Netw 10(5):988–999

38. Tuzel O, Porikli F, Meer P (2008) Learning on lie groups for invariant detection and tracking. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp 1–8

39. Li F, Zhang L, Zhang Z (2018) Lie group machine learning. Walter de Gruyter GmbH & Co KG

40. Ghifary M, Balduzzi D, Kleijn WB, Zhang M (2017) Scatter component analysis: A unified framework for domain adaptation and domain generalization. IEEE Trans Pattern Anal Mach Intell 39(7):1414–1430

41. Zhang J, Li W, Ogunbona P (2017) Joint geometrical and statistical alignment for visual domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1859–1867

42. Long M, Wang J, Ding G, Sun J, Yu PS (2014) Transfer feature learning with joint distribution adaptation. In: IEEE International Conference on Computer Vision, pp 2200–2207

43. Wang J, Feng W, Chen Y, Yu H, Huang M, Yu PS (2018) Visual domain adaptation with manifold embedded distribution alignment. In: Proceedings of the 26th ACM international conference on Multimedia, pp 402–410

44. Georgi H, Jagannathan K (1983) Lie algebras in particle physics. Phys Today 36(12):62–62

45. Hall BC (2015) Lie groups, lie algebras, and representations. Springer Berl 659(5):xiv,351

46. Quanz B, Huan J (2009) Large margin transductive transfer learning. In: Acm Conference on Information & Knowledge Management

47. Wang J, Chen Y, Hao S, Feng W, Shen Z (2017) Balanced distribution adaptation for transfer learning. In: IEEE International Conference on Data Mining

48. Belkin M, Niyogi P, Sindhwani V, Bartlett P (2006) Manifold regularization: a geometric framework for learning from examples. J Mach Learn Res 7(1):2399–2434

49. Saenko K, Kulis B, Fritz M, Darrell T (2010) Adapting visual category models to new domains. In: European Conference on Computer Vision, pp 213–226

50. Fang C, Xu Y, Rockmore DN (2014) Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In: IEEE International Conference on Computer Vision, pp 1657–1664

51. Suykens JAK, Vandewalle J (1999) Least squares support vector machine classifiers. Neural Process Lett 9(3):293–300

52. Long M, Wang J, Ding G, Sun J, Yu PS (2014) Transfer joint matching for unsupervised domain adaptation. In: Computer Vision and Pattern Recognition, pp 1410–1417

Hongwei Yang is currently working toward the PhD degree in the School of Cyberspace Science, Harbin Institute of Technology, China. His research interests include transfer learning, information security, and big data analysis.

Hui He is currently a Professor of Network Security Center in the School of Cyberspace Science, China. She received the Ph.D. in department of computer science from the Harbin Institute of Technology, China. Her research interests are mainly focused on distributed computing, IoT and big data analysis. She is a member of the IEEE.

Weizhe Zhang is currently a Professor in the School of Cyberspace Science at Harbin Institute of Technology, China. His research interests are primarily in cyberspace security, cloud computing, and parallel computing. He has published more than 100 academic papers in journals, books, and conference proceedings. He is a senior member of the IEEE.

Yawen Bai is currently working toward the MS degree in the School of Cyberspace Science, Harbin Institute of Technology, China. Her research interests include machine learning, deep learning, and federated learning.

Tao Li is currently working toward the MS degree in the School of Cyberspace Science, Harbin Institute of Technology, China. His research interests include transfer learning, information security, and big data analysis.