*Int. J. Appl. Math. Comput. Sci., 2019, Vol. 29, No. 3, 611–621*
DOI: 10.2478/amcs-2019-0045

# UTILIZING RELEVANT RGB–D DATA TO HELP RECOGNIZE RGB IMAGES IN THE TARGET DOMAIN

DEPENG GAO [a], JIAFENG LIU [a], RUI WU [a], DANSONG CHENG [a], XIAOPENG FAN [a], XIANGLONG TANG [a,*]

[a]School of Computer Science and Technology
Harbin Institute of Technology, No. 92 Xidazhi Street, Harbin, China
e-mail: tangxl@hit.edu.cn

With the advent of 3D cameras, getting depth information along with RGB images has been facilitated, which is helpful in various computer vision tasks. However, there are two challenges in using these RGB-D images to help recognize RGB images captured by conventional cameras: one is that the depth images are missing at the testing stage, the other is that the training and test data are drawn from different distributions as they are captured using different equipment. To jointly address the two challenges, we propose an asymmetrical transfer learning framework, wherein three classifiers are trained using the RGB and depth images in the source domain and RGB images in the target domain with a structural risk minimization criterion and regularization theory. A cross-modality co-regularizer is used to restrict the two-source classifier in a consistent manner to increase accuracy. Moreover, an $L_{2,1}$ norm cross-domain co-regularizer is used to magnify significant visual features and inhibit insignificant ones in the weight vectors of the two RGB classifiers. Thus, using the cross-modality and cross-domain co-regularizer, the knowledge of RGB-D images in the source domain is transferred to the target domain to improve the target classifier. The results of the experiment show that the proposed method is one of the most effective ones.

**Keywords:** object recognition, RGB-D images, transfer learning, privileged information.

## 1. Introduction

Object recognition as an important application has been widely researched (Donahue *et al.*, 2013; LeCun *et al.*, 2015); typically, a classifier is trained using the labeled images given to recognize a query object in a novel image. With the development of RGB-D sensors such as the Microsoft Kinect camera, RGB-D datasets are emerging (Hadfield and Bowden, 2013; Huynh *et al.*, 2012; Janoch *et al.*, 2013) containing labeled images paired with depth information. The latter points to the distance between an object and the camera, and is unperturbed by changes in illumination, color and background. Many researchers have successfully used depth information for various computer vision tasks, such as action recognition (Yu and Fu, 2016), face recognition (Goswami *et al.*, 2014), and object recognition (Nuricumbo *et al.*, 2015). However, these methods are not suitable for recognizing images captured by a conventional camera for two main reasons.

The first problem is asymmetry, because depth information is missing at the testing stage. In this situation, most existing RGB-D object recognition methods may not work as depth information is required at both the training and testing stages. The other issue is that training RGB-D images and test RGB images have different visual characters as they are captured by different cameras, which cause a distribution mismatch between training and test data. Therefore, the performance of most existing visual recognition approaches based on the independent and identically distributed assumption will degrade significantly.

To deal with the asymmetry, learning using privileged information (LUPI) methods (Feyereisl and Aickelin, 2012; Sharmanska *et al.*, 2013; Vapnik and Vashist, 2009) were proposed. While these can achieve better performance by utilizing additional information (i.e., privileged information) that is not available at the testing stage, they assume that the training and test images are drawn from the same data distribution. On the

*Corresponding author

other hand, many transfer learning (TL) methods (Kulis *et al.*, 2011; Li *et al.*, 2014) were proposed to address the distribution mismatch problem and achieved promising results. However, TL methods do not use additional depth information, which is helpful for object recognition.

In this paper, we propose a framework called asymmetrical transfer learning (ATL) to jointly address the asymmetry and distribution mismatch problems, wherein the abundance in labeled RGB-D images in the source domain contrasts with the scarcity in labeled RGB images in the target domain (see Fig. 1). Our goal is to recognize the unlabeled RGB images in the target domain. Thus, we directly learn a target classifier using the labeled target images. It will not accurately classify the test images because of insufficient training data. Fortunately, there are relevant RGB-D images in the source domain that can be used to improve the performance of the target classifier. To make use of RGB-D images, we learn two classifiers with RGB images and depth images in the source domain. The predictions of the two classifiers are restricted in a consistent way to improve accuracy simultaneously. Moreover, to leverage the knowledge from the source domain, we borrow the significant visual features to assist us in constructing the weight vector of the target classifier. This is achieved by minimizing an $L_{2,1}$ norm co-regularizer on source and target RGB classifiers. The parameters of the three classifiers are all optimized in a unified objective function by alternating and iteration. The results of the object recognition experiment on the four datasets show the effectiveness of the proposed method. The contributions are summarized as follows:

- An asymmetrical transfer learning algorithm is proposed that can utilize the labeled RGB-D images from the source domain to learn a robust classifier to recognize the RGB images in the target domain. This algorithm can simultaneously address the asymmetry and the distribution mismatch problem between RGB-D and RGB images.

- The proposed algorithm only contains a unified objective function, and the optimal solution can be obtained through several iterations.

- Our comprehensive experimental results on several visual datasets show that our method can outperform other state-of-the-art techniques in most cases.

The rest of this paper is organized as follows. Related work is introduced in Section 2. In Section 3, we describe the proposed framework and a concrete learning algorithm. Comprehensive experiments are conducted to evaluate effectiveness in Section 4. Finally, we conclude this work and discuss potential future research.

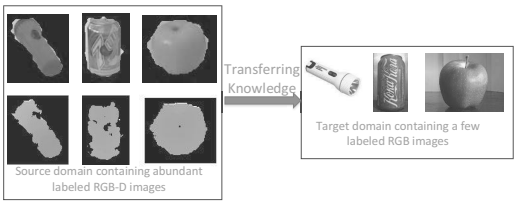



Fig. 1. Asymmetrical transfer learning: there are abundant labeled RGB images in the source domain while only a few labeled RGB images in the target domain.

## 2. Related work

Our work is related to transfer learning, also named domain adaptation, where the performance of a target classifier is improved by exploiting knowledge from a relevant domain (Weiss *et al.*, 2016). In particular, inductive transfer learning (Pan and Yang, 2010) requires some labeled data in the target domain to induce an objective predictive model, and is the closest to our work. Based on "what needs to be transferred", inductive transfer learning methods can be divided into four categories (Pan and Yang, 2010): instance transfer (Dai *et al.*, 2007; Jiang and Zhai, 2007), feature representation transfer (Saenko *et al.*, 2010; Kulis *et al.*, 2011; Li *et al.*, 2014), parameter transfer (Yang *et al.*, 2007; Evgeniou and Pontil, 2004) and relational knowledge transfer (Mihalkova *et al.*, 2007). However, most existing transfer learning methods only handle one type of information, while we have to contend with RGB and depth images in the source domain.

Our work is also related to the paradigm of learning using privileged information (Feyereisl and Aickelin, 2012; Fouad *et al.*, 2013; Motiian *et al.*, 2016; Sharmanska *et al.*, 2013), in which the privileged information provided by a supervisor at the training stage is used to train a more discriminative prediction model. However, these LUPI methods assume that both the training and testing data are independent and identically distributed (IID). In contrast, our work releases this IID assumption and can exploit a relevant domain to improve the learning in another, different, domain.

To the best of our knowledge, the works of Chen *et al.* (2014), Li *et al.* (2018; 2017), as well as Motiian and Doretto (2016) are the only ones addressing the same problem as ours. Chen *et al.* (2014) proposed the domain adaptation from multi-view to signal-view (DA-M2S) method to recognize RGB images by learning from RGB-D data and then extended it (Li *et al.*, 2018). Motiian and Doretto (2016) proposed the information bottleneck domain adaptation with privileged information (IBDAPI) method by extending the information bottleneck method and combining it with risk minimization. Both the DA-M2S and IBDAPI methods represent unsupervised domain adaptation and

do not use the labels in the target domain that are helpful for learning the target classifier. Li *et al.* (2017) proposed the domain adaptation from RGB-D images to RGB images (DARDR) method, which can use abundant labeled RGB-D images and scarce labeled RGB images simultaneously. However, the target classifier learned in DARDR is linear, which may not work well for non-linear classification problems. In contrast, our method can learn a more discriminative non-linear classifier using the kernel trick.

## 3. Asymmetrical transfer learning for object recognition

For ease of presentation, vectors and matrices are denoted by bold lowercase and uppercase letters, respectively. The transpose of a vector or matrix is denoted by the superscript 'T'.

**3.1. Problem statement.** We are given a source domain containing many triplets $(\mathbf{x}_{sv}^i, \mathbf{x}_{sd}^i, y_s^i), i = 1, \ldots, n_s$, drawn from the joint probability distribution $p_s(\mathbf{X}_{sv}, \mathbf{X}_{sd}, \mathbf{Y}_s)$, and a target domain containing scarce labeled data $(\mathbf{x}_{tv}^i, y_t^i), i = 1, \ldots, n_t$, and some test data drawn from the joint probability distribution $p_t(\mathbf{X}_{tv}, \mathbf{Y}_t)$. Here $\mathbf{x}_{sv}^i, \mathbf{x}_{tv}^i \in \mathcal{X}_v$ are visual features in the same space, but they have a different marginal distribution (i.e., $p(\mathbf{X}_{sv}) \neq p(\mathbf{X}_{tv})$). Furthermore, $\mathbf{x}_{sd}^i \in \mathcal{X}_d$ are the depth features in a space separate from the visual features. The quantities $y_s^i, y_t^i \in \{0, 1\}$ are the corresponding labels. Under these settings, our goal is to learn a prediction function $f_t : \mathbf{x}_{tv} \mapsto y_t$ to correctly classify the test images by utilizing the knowledge of the source domain. Table 1 lists the definitions of the symbols used in this paper.

**3.2. General framework.** We design the asymmetrical transfer learning (ATL) framework based on structural risk minimization and regularization theory. This framework directly learns the target visual classifier $f_{tv}$ with the labeled visual features in the target domain, and learns the source visual classifier $f_{sv}$ and the source depth classifier $f_{sd}$ with the labeled source visual and depth features, respectively. The proposed framework optimizes the following complementary objective functions simultaneously:

- minimizing the structural risk function on all labeled data from both domains,
- transferring knowledge from the source visual classifier to the target visual classifier,
- boosting the performance of the source visual classifier with additional information.

Thus, the objective function of ATL can be formulated as follows:

$$\begin{aligned} \min C_s \sum_{i=1}^{n_s} \left( l(f_{sv}, \mathbf{x}_{sv}^i) + l(f_{sd}, \mathbf{x}_{sd}^i) \right) \\ + C_t \sum_{i=1}^{n_t} l(f_{tv}, \mathbf{x}_{tv}^i) + \sigma(f_{sv}, f_{sd}, f_{tv}) \\ + \mu \Omega_{cm}(f_{sv}, f_{sd}) + \beta \Omega_{cd}(f_{sv}, f_{tv}), \end{aligned} \tag{1}$$

where $l$ is the loss function, $\Omega_{cm}$ is the cross-modality co-regularizer and $\Omega_{cd}$ is the cross-domain co-regularizer, $C_s$ and $C_t$ are the tradeoff parameters to balance the loss of source and target domain, $\sigma$, $\mu$ and $\beta$ are the regularization parameters.

**3.2.1. Structural risk minimization.** Our ultimate goal is to learn a prediction function $f_t$ for the target domain. For the sake of generality, the non-linear prediction function with the kernel trick $f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x})$ is used. Here $\mathbf{w}$ is the vector of classifier parameters and $\phi : \mathcal{X} \mapsto \mathcal{H}$ is the feature mapping function that projects the original feature vector to a Hilbert space $\mathcal{H}$. Thus, the structural risk of ATL can be formulated as

$$\begin{aligned} \underset{\mathbf{w}_{sv}, \mathbf{w}_{sd}, \mathbf{w}_{tv}}{\arg\min} \; C_s \sum_{i=1}^{n_s} \left( l(f_{sv}, \mathbf{x}_{sv}^i) + l(f_{sd}, \mathbf{x}_{sd}^i) \right) \\ + C_t \sum_{i=1}^{n_t} l(f_{tv}, \mathbf{x}_{tv}^i) \\ + \sigma \left( \|\mathbf{w}_{sv}\|_F^2 + \|\mathbf{w}_{sd}\|_F^2 + \|\mathbf{w}_{tv}\|_F^2 \right), \end{aligned} \tag{2}$$

where $\mathbf{w}_{sv}$, $\mathbf{w}_{sd}$ and $\mathbf{w}_{tv}$ are the weight vectors of the corresponding classifiers, $\|\mathbf{w}_{sv}\|_F^2$, $\|\mathbf{w}_{sd}\|_F^2$, $\|\mathbf{w}_{tv}\|_F^2$ are the regularizers and $\sigma$ is the regularization parameter.

**3.2.2. Knowledge transfer.** In this paper, we propose a parameter-based transfer learning method, which assumes that there are some shared parameters or prior distributions of the hyperparameters between the source and target models (Pan and Yang, 2010; Weiss *et al.*, 2016). By discovering the shared parameters or priors, knowledge can be transferred across domains (Pan and Yang, 2010). To achieve this goal, a revised representer theorem is used to construct the optimal solutions as in the works of Argyriou *et al.* (2008), Belkin *et al.* (2006) and Long *et al.* (2014).

**Theorem 1.** (Representer theorem) *The solutions of the two visual classifiers lie in the span*[1] *of all visual features*

[1]The definition of the span is as follows: The set of all linear combinations of a list of vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m$ in $\mathcal{V}$ is called the span of $\mathbf{v}_1, \ldots, \mathbf{v}_m$, denoted as $\mathrm{span}(\mathbf{v}_1, \ldots, \mathbf{v}_m)$. In other words, $\mathrm{span}(\mathbf{v}_1, \ldots, \mathbf{v}_m) = \{\alpha_1 \mathbf{v}_1 + \cdots + \alpha_m \mathbf{v}_m\}$, where $\alpha_i$ are the coefficients (Axler, 1997).

Table 1. Definitions of the symbols used.

| Symbols | Definitions |
|---|---|
| $\mathbf{X}_{sv}$ | visual features from the source domain |
| $\mathbf{X}_{sd}$ | depth features from the source domain |
| $\mathbf{X}_{tv}$ | visual features from the target domain |
| $\mathbf{Y}_s$, $\mathbf{Y}_t$ | source and target labels |
| $f_sv$ | source classifier with visual features |
| $f_sd$ | source classifier with depth features |
| $f_tv$ | target classifier with visual features |
| $\mathbf{w}_{sv}$,$\mathbf{w}_{sd}$,$\mathbf{w}_{tv}$ | weight vector of the corresponding classifier |
| $\boldsymbol{\alpha}_{sv}$, $\boldsymbol{\alpha}_{sd}$, $\boldsymbol{\alpha}_{tv}$ | coefficient vectors of $\mathbf{w}_{sv}$,$\mathbf{w}_{sd}$,$\mathbf{w}_{tv}$ |
| $\mathbf{K}_{sv}$, $\mathbf{K}_{sd}$, $\mathbf{K}_v$, $\mathbf{K}_{tv}$ | kernel matrices |
| $n_s$, $n_t$ | number of source and target samples |
| $C_s$, $C_t$ | tradeoff parameters |
| $\sigma$, $\mu$, $\beta$ | regularization parameters |

*from the source and target domain,*

$$\mathbf{w} = \sum_{i=1}^{n_s+n_t} \alpha_i \phi(\mathbf{x}_v^i) \tag{3}$$

*and*

$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}) = \sum_{i=1}^{n_s+n_t} \alpha_i k(\mathbf{x}_v^i, \mathbf{x}), \tag{4}$$

*where $\phi$ is the feature mapping function and $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ is the associated kernel function; $\alpha_i$ are coefficients that describe the significance of each sample in the weight vector $\mathbf{w}$ and $\boldsymbol{\alpha} = [\alpha_1; \ldots; \alpha_{n_s+n_t}] \in \mathbb{R}^{(n_s+n_t)}$ is the coefficient vector; $\mathbf{x}_i$ is the $i$-th column of $\mathbf{X}_v = [\mathbf{X}_{sv}, \mathbf{X}_{tv}] \in \mathbb{R}^{d \times (n_s+n_t)}$, denoting all the visual features in the source and target domains.*

With the revised representer theorem, the classifier parameters $\mathbf{w}_s$ and $\mathbf{w}_t$ can be denoted by a linear combination of all the visual features in the Hilbert space $\mathcal{H}$ and the corresponding coefficient vectors are $\boldsymbol{\alpha}_s \in \mathbb{R}^{n_s+n_t}$ and $\boldsymbol{\alpha}_t \in R^{n_s+n_t}$. Specifically, we assume that, if a sample is significant in the weight vector of a source classifier, it is also significant in one of the target classifiers, and vice versa. To exploit the relevance between the two visual classifiers, the coefficient vectors are combined as $\mathbf{A} = [\boldsymbol{\alpha}_s, \boldsymbol{\alpha}_t] \in \mathbb{R}^{(n_s+n_t)\times 2}$. Each row of $\mathbf{A}$ reflects the importance of the corresponding sample in both classifiers. To boost significant samples and inhibit insignificant ones, an $L_{2,1}$ norm[2] regularizer is used to restrict $\mathbf{A}$, making it row sparse. Thus, the cross-domain co-regularizer can be formulated as

$$\Omega_{cd}(f_{sv}, f_{tv}) = \|\mathbf{A}\|_{2,1}. \tag{5}$$

By minimizing the $L_{2,1}$ norm, each row of $\mathbf{A}$ will consist of all zeros or non-zeros simultaneously. In this process, the important samples are selected to construct the target classifier with the revised representer theorem.

[2]The definition of the $L_{2,1}$ norm as follows: For a matrix $\mathbf{W} \in \mathbb{R}^{m\times n}$, $\|\mathbf{W}\|_{2,1} = \sum_{i=1}^{m} \|\mathbf{W}_i\|_2 = \sum_{i=1}^{m} \sqrt{\sum_{j=1}^{n} w_{ij}^2}$.

**3.2.3. Exploiting depth information.** Motivated by multi-view learning (Sun, 2013), in which different types of features collaborate to learn more robust classifiers, we aim to exploit the depth images in the source domain to further improve the performance of the target classifier. In fact, the depth and visual features are two different types of representation of one object, so the two-source classifiers should offer identical predictions. Therefore, a cross-modality co-regularizer $\Omega_{cm}$ is used as follows to bring consistency to the two-source classifier and boost the performance of the sources simultaneously (Li *et al.*, 2018):

$$\Omega_{cm}(f_{sv}, f_{sd}) = \sum_{i=1}^{n_s} \left| f_{sv}(\mathbf{x}_{sv}^i) - f_{sd}(\mathbf{x}_{sd}^i) \right|^2. \tag{6}$$

By minimizing the cross-modality co-regularizer, the source depth classifier is connected to the source RGB classifier. In addition, the source RGB classifier is connected to the target RGB classifier with the cross-domain co-regularizer described in Section 3.2.2. Thus, the source depth classifier is connected to the target RGB classifier indirectly. Namely, the depth images in source domain are used to help learn the target RGB classifier.

**3.3. Concrete learning algorithm.** Our proposed ATL is a general framework. It may take various forms based on various loss functions, such as the least square loss $l = (y_i - f(\mathbf{x}_i))^2$ and hinge loss $l = \max(0, 1 - y_i f(\mathbf{x}_i))$. In the following, we introduce a specific learning algorithm using the least-squares loss function, named ATL-ls. Algorithms that use other loss functions will be investigated in future works.

**3.3.1. Objective function of ATL-ls.** According to the classical representer theorem, the optimal solution of the source depth classifier can be written as

$$\mathbf{w}_{sd} = \sum_{i=1}^{n_s} \alpha_{sd}^i \phi(\mathbf{x}_{sd}^i), \quad f_{sd}(\mathbf{x}) = \sum_{i=1}^{n_s} \alpha_{sd}^i k(\mathbf{x}_{sd}^i, \mathbf{x}), \tag{7}$$

where $\boldsymbol{\alpha}_{sd} = [\alpha_{sd}^1; \ldots; \alpha_{sd}^{n_s}] \in \mathbb{R}^{n_s}$ is the coefficient vector.

According to the definition of the $F$-norm, the regularizer in Eqn. (2) can be rewritten as $\|\mathbf{w}\|_F^2 = \text{tr}(\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{K}\boldsymbol{\alpha})$, where 'tr' is the trace function and $\mathbf{K}$ is the corresponding kernel matrix. Based on the least square loss function, the unified objective function of ATL-ls is obtained by integrating Eqns. (2)–(7),

$$\begin{aligned} \min_{\boldsymbol{\alpha}_{sv},\boldsymbol{\alpha}_{sd},\boldsymbol{\alpha}_{tv}} & C_s(\|\mathbf{y}_s - \mathbf{K}_{sv}\boldsymbol{\alpha}_{sv}\|_F^2 + \|\mathbf{y}_s - \mathbf{K}_{sd}\boldsymbol{\alpha}_{sd}\|_F^2) \\ & + C_t \|\mathbf{y}_t - \mathbf{K}_{tv}\boldsymbol{\alpha}_{tv}\|_F^2 + \sigma(\text{tr}(\boldsymbol{\alpha}_{sv}{}^{\mathrm{T}}\mathbf{K}_v\boldsymbol{\alpha}_{sv}) \\ & + \text{tr}(\boldsymbol{\alpha}_{sd}{}^{\mathrm{T}}\mathbf{K}_{sd}\boldsymbol{\alpha}_{sd}) \\ & + \text{tr}(\boldsymbol{\alpha}_{tv}{}^{\mathrm{T}}\mathbf{K}_v\boldsymbol{\alpha}_{tv})) + \mu\|\mathbf{A}\|_{2,1} \\ & + \beta \|\mathbf{K}_{sv}\boldsymbol{\alpha}_{sv} - \mathbf{K}_{sd}\boldsymbol{\alpha}_{sd}\|_F^2, \end{aligned} \tag{8}$$

where $\mathbf{K}_{sv} = k(\mathbf{X}_{sv}, \mathbf{X}_v) \in \mathbb{R}^{n_s \times (n_s+n_t)}$, $\mathbf{K}_{tv} = k(\mathbf{X}_{tv}, \mathbf{X}_v) \in \mathbb{R}^{n_t \times (n_s+n_t)}$, $\mathbf{K}_v = k(\mathbf{X}_v, \mathbf{X}_v) \in \mathbb{R}^{(n_s+n_t)\times(n_s+n_t)}$ and $\mathbf{K}_{sd} = k(\mathbf{X}_{sd}, \mathbf{X}_{sd}) \in \mathbb{R}^{n_s \times n_s}$ are kernel matrices constructed using different feature matrices.

**3.3.2. Optimization of ATL-ls.** In this section, we discuss how to solve the overall objective function in Eqn. (8). The main idea is to calculate $\boldsymbol{\alpha}_{sv}$, $\boldsymbol{\alpha}_{sd}$ and $\boldsymbol{\alpha}_{tv}$ alternatively and to repeat this process until convergence. However, there is a difficulty in that the $L_{2,1}$ norm minimization problem cannot be solved in closed form. Fortunately, some methods exist (Argyriou *et al.*, 2008; Liu *et al.*, 2009; Xiao *et al.*, 2013) to solve if. In this paper, the method of Argyriou *et al.* (2008) is adapted to transform the $L_{2,1}$ norm to an equivalent convex optimization problem as follows:

$$\|\mathbf{A}\|_{2,1} = \sum_{i=1}^{n_s+n_t} \|\mathbf{A}_i\|_2 = \text{tr}(\mathbf{A}^{\mathrm{T}}\mathbf{R}\mathbf{A}), \tag{9}$$

where $\mathbf{A}_i$ is the $i$-th row vector of $\mathbf{A}$ and $\mathbf{R}$ is a diagonal matrix with $\mathbf{R}_{ii} = 1/\|\mathbf{A}_i\|_2$. Thus, $\mathbf{A}$ and $\mathbf{R}$ can be sought alternatively. In each iteration, $\mathbf{A}$ is calculated based on the current $\mathbf{R}$, then $\mathbf{A}$ is updated, and then $\mathbf{R}$ is calculated based on the new $\mathbf{A}$.

In each iteration of the optimization of ATL-ls, the coefficient vectors $\boldsymbol{\alpha}_{sv}$, $\boldsymbol{\alpha}_{sd}$ and $\boldsymbol{\alpha}_{tv}$ are calculated alternatively by fixing two of them and setting the derivative of the objective function with respect to another to zero, thus yielding the following solutions:

$$\boldsymbol{\alpha}_{sv} = \mathbf{P}^{-1}\left(C_s\mathbf{K}_{sv}{}^{\mathrm{T}}\mathbf{y}_s + \beta\mathbf{K}_{sv}{}^{\mathrm{T}}\mathbf{K}_{sd}\boldsymbol{\alpha}_{sd}\right), \tag{10}$$

where

$$\mathbf{P} = (C_s + \beta)\mathbf{K}_{sv}{}^{\mathrm{T}}\mathbf{K}_{sv} + \sigma\mathbf{K}_v + \mu\mathbf{R},$$

$$\boldsymbol{\alpha}_{tv} = \left(C_t\mathbf{K}_{tv}{}^{\mathrm{T}}\mathbf{K}_{tv} + \sigma\mathbf{K}_v + \mu R\right)^{-1} C_t\mathbf{K}_{tv}{}^{\mathrm{T}}\mathbf{y}_t, \tag{11}$$

$$\boldsymbol{\alpha}_{sd} = \mathbf{Q}^{-1}\left(C_s\mathbf{K}_{sd}{}^{\mathrm{T}}\mathbf{y}_s + \beta\mathbf{K}_{sd}{}^{\mathrm{T}}\mathbf{K}_{sv}\boldsymbol{\alpha}_{sv}\right), \tag{12}$$

with $\mathbf{Q} = (C_s + \beta)\mathbf{K}_{sd}{}^{\mathrm{T}}\mathbf{K}_{sd} + \sigma\mathbf{K}_{sd}$.

The corresponding ATL-ls algorithm is summarized in Algorithm 1.

**Algorithm 1.** ATL-ls algorithm.

**Require:** Labeled source domain visual features $\mathbf{X}_{sv}$, depth features $\mathbf{X}_{sd}$, target domain visual features $\mathbf{X}_{tv}$, and parameters $\sigma, \mu, \beta, C_s, C_t$.

1: Initialize $\boldsymbol{\alpha}_{sv}$,$\boldsymbol{\alpha}_{sd}$ and $\boldsymbol{\alpha}_{tv}$ randomly.
2: Update $\mathbf{A} = [\boldsymbol{\alpha}_{sv}, \boldsymbol{\alpha}_{tv}]$ and calculate $\mathbf{R}$.
3: **repeat**
4: Calculate $\boldsymbol{\alpha}_{sv}$ via Eqn. (10).
5: Calculate $\boldsymbol{\alpha}_{tv}$ via Eqn. (11).
6: Update $\mathbf{A}$ and calculate $\mathbf{R}$.
7: Calculate $\boldsymbol{\alpha}_{sd}$ via Eqn. (12).
8: **until** convergence
9: **return** The optimal coefficient matrix $\boldsymbol{\alpha}_{tv}$ of the target classifier $f_t$.

**Multi-class extension** $\mathbf{y} \in \mathbb{R}^c$ is a row label vector such that $y_i = 1$ if it belongs to the $i$-th category, and $y_i = 0$ otherwise. Thus, the label matrices of the source and target domains are of the form $\mathbf{Y} = [\mathbf{y}_1; , \ldots, ; \mathbf{y}_n] \in \mathbb{R}^{n\times c}$. Moreover, the coefficient matrices of each classifier are $\mathbf{A}_{sv} = [\boldsymbol{\alpha}_{sv_1}, \ldots, \boldsymbol{\alpha}_{sv_c}] \in \mathbb{R}^{(n_s+n_t)\times c}$, $\mathbf{A}_{sd} = [\boldsymbol{\alpha}_{sd_1}, \ldots, \boldsymbol{\alpha}_{sd_c}] \in \mathbb{R}^{n_s\times c}$ and $\mathbf{A}_{tv} = [\boldsymbol{\alpha}_{tv_1}, \ldots, \boldsymbol{\alpha}_{tv_c}] \in \mathbb{R}^{(n_s+n_t)\times c}$, and $\mathbf{A} = [\mathbf{A}_{sv}, \mathbf{A}_{tv}] \in \mathbb{R}^{(n_s+n_t)\times 2c}$. In this way, the learning algorithm ATL-ls can be extended to deal with a $c$ categories classification problem.

## 4. Experiments and performance analysis

In this section, we perform extensive experiments to evaluate and analyze the ATL-ls algorithm for an object recognition task on four cross-domain dataset pairs.

**4.1. Baseline approaches.** In this paper, the proposed ATL-ls is compared with the following four types of methods.

**Naive approach.** The standard regularization least-squares (RLS) algorithm is used to train two classifiers (named RLS-t and RLS-s) using labeled RGB images in the target and source domains, respectively.

**Learning using privileged information.** The proposed method is compared with the LUPI method SVM+ (Vapnik and Vashist, 2009) and rank transfer (RT) (Sharmanska *et al.*, 2013), in which the depth images in the source domain are used to train an RGB classifier.

**Transfer learning.** The compared transfer learning methods include heterogeneous feature augmentation (HFA) (Li *et al.*, 2014) and the metric transfer learning framework (MTLF) (Xu *et al.*, 2017). Both HFA and the MTLF can learn a target classifier with scarce labeled target images by transferring knowledge from the source RGB images.

**Domain adaptation with depth images.** The DA-M2S (Chen *et al.*, 2014; Li *et al.*, 2018), IBDAPI (Motiian and Doretto, 2016) and DARDR (Li *et al.*, 2017) approaches address the same problem as ours; they can reduce the distribution mismatch and exploit the depth images in the source domain simultaneously. In particular, the DA-M2S and IBDAPI methods do not use the labels in the target domain.

**4.2. Dataset description.** We evaluate the proposed method on the following four cross-domain pairs; the details are listed in Table.2.

**RGB-D(R)→Caltech-256(C).** The RGB-D dataset (Lai *et al.*, 2011) is used as the source domain containing RGB and depth images in 51 categories. These images are cropped using the video sequence captured with the Microsoft Kinect camera. The Caltech-256 dataset (Griffin *et al.*, 2007) containing 29,780 RGB images from 256 categories is used as the target domain. In the experiments, ten categories shared across the two datasets are used, namely, *ball*, *calculator*, *cereal-box*, *cup*, *flash-light*, *keyboard*, *light-bulb*, *mushroom*, *soda-can*, and *tomato*. As the RGB-D dataset is recorded in the form of video sequences, we uniformly subsample the frames with an interval of four, leading to a total of 1824 pairs of RGB and depth images in the source domain.

**RGB-D(R)→ImageNet(I).** ImageNet (Deng *et al.*, 2009) is a large-scale dataset including more than 100,000 categories of RGB images organized in accordance with the WordNet hierarchy structure. In this paper, only ten common categories across RGB-D and ImageNet, namely, *apple, banana, calculator, cereal-box, coffee-mug, keyboard, light-bulb, plate, soda-can, water-bottle*, are used to demonstrate the proposed algorithm.

Table 2. Details of the dataset pairs used.

| Dataset pair | #Instances | #Classes |
|---|---|---|
| RGB-D/Caltech-256 | 1824/1132 | 10 |
| RGB-D/ImageNet | 1823/1000 | 10 |
| B3DO/Caltech-256 | 1129/776 | 8 |
| B3DO/ImageNet | 1135/800 | 8 |

Following the work of Li *et al.* (2017), we randomly select 100 images per category in ImageNet as the target domain. Meanwhile, 1823 pairs of RGB and depth images subsampled from RGB-D are used as the source domain.

**B3DO(B)→Caltech-256(C).** The Berkeley 3D Object (B3DO) dataset (Janoch *et al.*, 2013) contains 849 color and depth image pairs gathered by a Microsoft Kinect camera in actual domestic and office environments. Over 50 different object classes are represented in the dataset. We crop these objects with the provided bounding box and select eight classes shared across B3DO and Caltech-256. The images belonging to the eight classes, *cup, keyboard, monitor, mouse, phone, soda-can, spoon, water-bottle*, are used in our experiments. B3DO is as the source domain and Caltech-256 is used as the target domain.

**B3DO(B)→ImageNet(I).** To demonstrate the proposed algorithm, we also conduct experiments on B3DO and the ImageNet dataset pair, in which the same eight categories, *cup, keyboard, monitor, mouse, phone, plate, spoon, water-bottle*, are used. We use the B3DO as the source domain and randomly select 100 images per category in ImageNet as the target domain, similarly to Li *et al.* (2017).

**4.3. Experimental setup.** In the experiments, a joint cross validation parameter selection approach is applied to choose the three regularization parameters $\sigma$, $\mu$ and $\beta$ from $\{0.001, 0.01, 0.1, 1, 10, 100\}$. The Gauss kernel function[3] is used and its bandwidth $\delta$ is set to the average distance between all the samples, as in the works of Gehler and Nowozin (2009) and Kovashka and Grauman (2010). Moreover, because the number of labeled datapoints in the source domain is much larger than in the target domain, we set $C_s = n_t/2(n_s + n_t)$ and $C_t = n_s/(n_s + n_t)$ to balance the loss terms for the source and target domains in the objective function.

The multipath hierarchical matching pursuit (M-HMP) method (Bo *et al.*, 2013), which can capture multiple aspects of discriminative structures by combining a collection of hierarchical sparse features, is used to extract visual and depth features from the RGB and depth images, respectively. Furthermore, the PCA method is applied to reduce the features' dimensionality to 500.

[3]The Gauss kernel function has the form $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 \Big/ 2\delta^2\right)$.

The experiments are all run twenty times and multi-class classification accuracy is used as the evaluation metric. Each time, we randomly select $3\%$ of the samples from the target domain together with all the samples from the source domain as training data, and the remainder (i.e., $97\%$) in the target domain is used as testing data.

**4.4. Experimental results.** From the results reported in Table 3, we can see that the proposed ATL-ls method outperforms all the others. That proves the effectiveness of transferring the knowledge of relevant RGB-D images in the source domain to help recognize RGB images in the target domain.

Of all the compared methods, the naive approach RLS-s achieves the worst performance as it neither exploits the depth images nor considers the divergence in data distribution between domains. By exploiting the depth information in the source domain, the LUPI methods SVM+ and RT perform better than the RLS-s on all dataset pairs. Although SVM+ and RT can learn a more robust classifier by using privileged information (i.e., depth images), this is not suitable for the classification of images in the target domain because of the huge divergency in data distribution. Thus, the performance of SVM+ and RT is worse than that of the other methods except RLS-s. The transfer learning methods HFA and MTLF outperform the naive RLS-t approach as they use knowledge not only in the target domain but also in the source domain, and the labeled data in the target domain are not sufficient to train a robust RLS classifier. This demonstrates that exploiting knowledge from the source domain is particularly helpful in improving the performance for a target task.

All the methods that simultaneously consider the distribution mismatch and exploit the depth information perform better than the transfer learning methods. This again demonstrates the relevance of the depth images in the source domain. Moreover, DA-M2S and IBDAPI are worse than DARDR and FADALS. This may be because DA-M2S and IBDAPI do not use the labels in the target domain, which would have been helpful for learning a more robust target classifier. Therefore, it appears that exploiting the label information in the target domain is preferable. In addition, our method achieves better performance than DARDR on all dataset pairs. Indeed, the target classifier learned in DARDR is a linear least square classifier, which may not work well for non-linear classification problems. Our ATL-ls can learn a more discriminative non-linear classifier by using the kernel trick and sharing parameters between the source and target visual classifiers.

**4.5. Analysis of the ATL-ls algorithm.** Here we analyze the proposed ATL-ls algorithm in three aspects: the influence of depth information, the influence of transfer learning and the convergence.

**4.5.1. Influence of depth images.** To analyze the impact of the depth images, we execute a simplified algorithm based on the ATL framework called ATL-nodepth, which does not use depth images but keeps all other aspects of the objective function unchanged. The comparative results of the experiments on different dataset pairs are listed in Table 4. From the results we can see that the proposed FADALS method outperforms the ATL-nodepth technique on all dataset pairs. The main reason is that we can learn a more robust source visual classifier by using the depth images; thus, the performance of the target classifier is improved indirectly by sharing parameters with the source visual classifier.

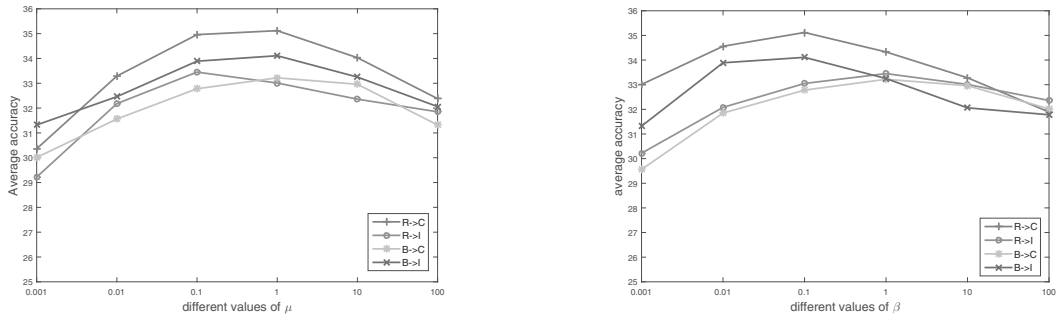
**4.5.2. Influence of transfer learning.** In this section, we will explore the influence of transfer learning for the ATL-ls algorithm. To achieve this goal, we construct another simplified algorithm based on the ATL framework called ATL-notransfer, which does not use any knowledge from the source domain. This can be achieved by setting the value of the cross-domain regularizer parameter $\beta = 0$ and keeping other terms in the objective function unchanged. The comparison with the experimental results on different dataset pairs is given in Table 5. ATL-ls performs better than ATL-notransfer on dataset pairs as it can construct the target classifier with the significant visual features in both domains by sharing the coefficient vectors.

**4.5.3. Parameter sensitivity.** In this section, we will investigate how regularization parameters $\mu$ and $\beta$ affect the accuracy for various object recognition tasks. We conducted experiments on four different problems (R→C, R→I, B→C, and B→I) and plotted the classification accuracy, as illustrated in Fig. 2. In these experiments, we tuned one parameter at a time over a given range while pinning the others to their optimal values.

As Fig. 4.5.1 shows, the accuracy degrades when the value of $\mu$ is too large or too small. If it is too large, the classification boundary of the target classifier becomes too close to that of the source classifier. Therefore, the effects of the target samples are inhibited, and this situation is referred to as overtransfer. Conversely, if the value is too small, knowledge from the source domain can adequately be transferred to the target domain, and this situation is referred to as undertransfer. Over and undertransfers both degrade accuracy. As shown in Fig. 4.5.1, the value of $\beta$ can also clearly affect performance. When $\beta$ is too small, depth information in the source domain is not fully

Table 3. Comparison of average accuracies (%) for object recognition on different dataset pairs.

| Dataset | RLS-s | RLS-t | SVM+ | RT | HFA | MTLF | DA-M2S | IBDAPI | DARDR | ATL-ls |
|---|---|---|---|---|---|---|---|---|---|---|
| R→C | 19.54 | 22.18 | 20.15 | 19.62 | 28.14 | 29.82 | 31.26 | 31.87 | 33.41 | **35.12** |
| R→I | 17.87 | 21.01 | 19.56 | 18.31 | 27.32 | 27.18 | 30.62 | 32.01 | 32.28 | **33.45** |
| B→C | 18.69 | 24.32 | 20.10 | 19.45 | 29.01 | 28.70 | 30.16 | 31.24 | 32.13 | **33.22** |
| B→I | 19.02 | 23.18 | 20.27 | 19.82 | 28.76 | 29.63 | 31.03 | 31.98 | 33.02 | **34.11** |




(a) (b)

Fig. 2. Influence of regularization parameters $\mu$ (a) and $\beta$ (b).

Table 4. Impact of depth images on the accuracy (%) of the ATL-ls algorithm.

| Dataset | ATL-nodepth | ATL-ls |
|---|---|---|
| R→C | 34.03 | 35.12 |
| R→I | 32.10 | 33.45 |
| B→C | 31.36 | 33.22 |
| B→I | 32.18 | 34.11 |

Table 5. Impact of transfer learning on the accuracy (%) of the ATL-ls algorithm.

| Dataset | ATL-notransfer | ATL-ls |
|---|---|---|
| R→C | 30.21 | 35.12 |
| R→I | 28.15 | 33.45 |
| B→C | 29.62 | 33.22 |
| B→I | 28.37 | 34.11 |

utilized. Conversely, when $\beta$ is too large, the effects of the other terms in the proposed method are inhibited. Therefore, to achieve the best performance, we need to select suitable values for $\mu$ and $\beta$.

**4.5.4. Convergence and running time.** Denote by $J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)})$ the value of the objective function at the $n$-th iteration. According to Eqns. (10) and (11), the value of the objective function will decrease at Steps 4 and 5 for each iteration of Algorithm 1, i.e.,

$$J(\mathbf{A}_s{}^{(n+1)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)}) \leq J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)}), \quad (13)$$

$$J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n+1)}) \leq J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)}). \quad (14)$$

x Moreover, the convergence for the alternating minimization algorithm computing the optimal solutions of the $L_{2,1}$ norm was proved by Argyriou *et al.* (2008). Thus, in Step 6 at the $n$-th iteration the value decreases further,

$$J(\mathbf{A}_s{}^{(n+1)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n+1)}) \leq J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)}). \quad (15)$$

According to Eqn. (12), after Step 7 at the $n$-th iteration we can get

$$J(\mathbf{A}_s{}^{(n+1)}, \mathbf{A}_s^{d(n+1)}, \mathbf{A}_t{}^{(n+1)}) \leq J(\mathbf{A}_s{}^{(n)}, \mathbf{A}_s^{d(n)}, \mathbf{A}_t{}^{(n)}). \quad (16)$$

Obviously, we can see that the total objective function decreases after each iteration, in other words, the algorithm is convergent.

To further demonstrate the convergence of our algorithm, the iteration and running time of ATL-ls on various dataset pairs are reported in Table 6, wherein

Table 6. Iteration times and running time (s) for ATL-ls on various dataset pairs.

| Dataset | Iterations | Running time |
|---|---|---|
| R→C | 8 | 64.2096 |
| R→I | 11 | 70.2388 |
| B→C | 8 | 22.3542 |
| B→I | 9 | 26.2247 |

the algorithm is running on Matlab Version R2015a and the convergence condition is $|obj_{t+1} - obj_t|/|obj_t| \leq 10^{-4}$. It appears that the algorithm converges after 8–11 iterations. The factors affecting running time are primarily the calculations of the four kernel matrices. As the number of samples increases, the running time will obviously increase. A practical solution to this problem involves using a manually assigned bandwidth for the Gauss kernel instead of a fixed value set to the average distance between samples.

## 5. Conclusion

In this paper, we proposed an asymmetrical transfer learning framework to utilize the relevant RGB-D images in the source domain to help recognize RGB images in the target domain, which contains scarce labeled data. Specifically, we jointly learn the source visual classifier, the source depth classifier and the target visual classifier with RGB and depth images from the source domain and RGB images from the target domain. To leverage depth information, we impose consistency in the predictions of the two-source classifiers, yielding performance gains. In addition, parameters are shared across the two visual classifiers so that knowledge can be transferred from the source domain to the target one. Model parameters can all be incorporated into a unified model, and the optimal solution can be obtained after a few iterations. The results of the experiments on different dataset pairs show that the proposed method can effectively exploit the relevant RGB-D images in the source domain to learn a robust target classifier and significantly outperform the state-of-the-art methods in various recognition tasks.

A limitation of our method is that it requires some labeled images in the target domain. Thus, in the future, we will explore ways to learn a classifier with labeled RGB-D images from the source domain when RGB images in the target domain are all unlabeled. Another important question is how to utilize RGB-D images in categories that are different to those of RGB images in the target domain.

## Acknowledgment

This research was supported by the National Natural Science Foundation of China (no. 61672190). We would like to thank Editage (https://www.editage.com/) for editing and reviewing this manuscript in English.

## References

Argyriou, A., Evgeniou, T. and Pontil, M. (2008). Convex multi-task feature learning, *Machine Learning* **73**(3): 243–272.

Axler, S. (1997). *Linear Algebra Done Right*, Undergraduate Texts in Mathematics, Vol. 2, Springer, New York, NY.

Belkin, M., Niyogi, P. and Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, *Journal of Machine Learning and Research* **7**: 2399–2434.

Bo, L., Ren, X. and Fox, D. (2013). Multipath sparse coding using hierarchical matching pursuit, *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA*, pp. 660–667.

Chen, L., Li, W. and Xu, D. (2014). Recognizing RGB images by learning from RGB-D data, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA*, pp. 1418–1425.

Dai, W., Yang, Q., Xue, G.R. and Yu, Y. (2007). Boosting for transfer learning, *International Conference on Machine Learning, Corvallis, FL, USA*, pp. 193–200.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database, *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, Miami, FL, USA*, pp. 248–255.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E. and Darrell, T. (2013). DeCAF: A deep convolutional activation feature for generic visual recognition, *Proceedings of the 31st International Conference on Machine Learning, Beijing, China*, pp. 647–655.

Evgeniou, T. and Pontil, M. (2004). Regularized multi-task learning, *10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA*, pp. 109–117.

Feyereisl, J. and Aickelin, U. (2012). Privileged information for data clustering, *Information Sciences* **194**: 4–23.

Fouad, S., Tino, P., Raychaudhury, S. and Schneider, P. (2013). Incorporating privileged information through metric learning, *IEEE Transactions on Neural Networks and Learning Systems* **24**(7): 1086–1098.

Gehler, P.V. and Nowozin, S. (2009). Let the kernel figure it out: Principled learning of pre-processing for kernel classifiers, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami, FL, USA*, pp. 2836–2843.

Goswami, G., Vatsa, M. and Singh, R. (2014). RGB-D face recognition with texture and attribute features, *IEEE Transactions on Information Forensics and Security* **9**(10): 1629–1640.

Griffin, G., Holub, A. and Perona, P. (2007). Caltech-256 object category dataset, California Institute of Technology, Pasadena, CA.

Hadfield, S. and Bowden, R. (2013). Hollywood 3D: Recognizing actions in 3D natural scenes, *IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA*, pp. 3398–3405.

Huynh, T., Min, R. and Dugelay, J.L. (2012). An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data, *Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan*, pp. 133–145.

Janoch, A., Karayev, S., Jia, Y., Barron, J.T., Fritz, M., Saenko, K. and Darrell, T. (2013). A category-level 3d object dataset: Putting the kinect to work, *in* A. Fossati *et al.* (Eds), *Consumer Depth Cameras for Computer Vision*, Springer, London, pp. 141–165.

Jiang, J. and Zhai, C.X. (2007). Instance weighting for domain adaptation in NLP, *Meeting of the Association of Computational Linguistics, Prague, Czech Republic*, pp. 264–271.

Kovashka, A. and Grauman, K. (2010). Learning a hierarchy of discriminative space-time neighborhood features for human action recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA*, pp. 2046–2053.

Kulis, B., Saenko, K. and Darrell, T. (2011). What you saw is not what you get: Domain adaptation using asymmetric kernel transforms, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA*, pp. 1785–1792.

Lai, K., Bo, L., Ren, X. and Fox, D. (2011). A large-scale hierarchical multi-view RGB-D object dataset, *2011 IEEE International Conference on Robotics and Automation, Shanghai, China*, pp. 1817–1824.

LeCun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning, *Nature* **521**(7553): 436–444.

Li, W., Chen, L., Xu, D. and Gool, L.V. (2018). Visual recognition in RGB images and videos by learning from RGB-D data, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PP**(99): 1–1.

Li, W., Duan, L., Xu, D. and Tsang, I.W. (2014). Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36**(6): 1134–1148.

Li, X., Fang, M., Zhang, J.-J. and Wu, J. (2017). Domain adaptation from RGB-D to RGB images, *Signal Processing* **131**: 27–35.

Liu, J., Ji, S. and Ye, J. (2009). Multi-task feature learning via efficient $l_{2,1}$-norm minimization, *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, Montreal, Canada*, pp. 339–348.

Long, M., Wang, J., Ding, G., Pan, S.J. and Yu, P.S. (2014). Adaptation regularization: A general framework for transfer learning, *IEEE Transactions on Knowledge and Data Engineering* **26**(5): 1076–1089.

Mihalkova, L., Huynh, T. and Mooney, R.J. (2007). Mapping and revising Markov logic networks for transfer learning, *Proceedings of the 22nd AAAI Conference on Artificial Intelligence, Vancouver, Canada*, pp. 608–614.

Motiian, S. and Doretto, G. (2016). Information bottleneck domain adaptation with privileged information for visual recognition, *Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands,* pp. 630–647.

Motiian, S., Piccirilli, M., Adjeroh, D.A. and Doretto, G. (2016). Information bottleneck learning using privileged information for visual recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA*, pp. 1496–1505.

Nuricumbo, J.R., Ali, H., Mrton, Z.C. and Grzegorzek, M. (2015). Improving object classification robustness in RGB-D using adaptive SVMS, *Multimedia Tools and Applications* **75**(12): 1–19.

Pan, S.J. and Yang, Q. (2010). A survey on transfer learning, *IEEE Transactions on Knowledge and Data Engineering* **22**(10): 1345–1359.

Saenko, K., Kulis, B., Fritz, M. and Darrell, T. (2010). *Adapting Visual Category Models to New Domains*, Springer, Berlin/Heidelberg.

Sharmanska, V., Quadrianto, N. and Lampert, C.H. (2013). Learning to rank using privileged information, *Proceedings of the IEEE International Conference on Computer Vision, Portland, OR, USA*, pp. 825–832.

Sun, S. (2013). A survey of multi-view machine learning, *Neural Computing and Applications* **23**(7–8): 2031–2038.

Vapnik, V. and Vashist, A. (2009). A new learning paradigm: Learning using privileged information, *Neural Networks* **22**(5): 544–557.

Weiss, K., Khoshgoftaar, T.M. and Wang, D. (2016). A survey of transfer learning, *Journal of Big Data* **3**(1): 9.

Xiao, Y., Wu, S.Y. and He, B.S. (2013). A proximal alternating direction method for $l_{2,1}$-norm least squares problem in multi-task feature learning, *Journal of Industrial and Management Optimization* **8**(4): 1057–1069.

Xu, Y., Pan, S.J., Xiong, H., Wu, Q., Luo, R., Min, H. and Song, H. (2017). A unified framework for metric transfer learning, *IEEE Transactions on Knowledge and Data Engineering* **29**(6): 1158–1171.

Yang, J., Yan, R. and Hauptmann, A.G. (2007). Cross-domain video concept detection using adaptive SVMS, *Proceedings of the 15th ACM International Conference on Multimedia, Augsburg, Germany*, pp. 188–197.

Yu, K. and Fu, Y. (2016). Discriminative relational representation learning for RGB-D action recognition, *IEEE Transactions on Image Processing* **25**(6): 2856–2865.

**Depeng Gao** is a PhD candidate at the Harbin Institute of Technology. He received his BS and MS degrees in computer science and technology from the Harbin Institute of Technology in 2011 and 2015, respectively. His current research interests include machine learning, domain adaptation learning, and computer vision.

**Jiafeng Liu** received his PhD degree from the Harbin Institute of Technology, China, in 1996. He is currently an associate professor at the School of Computer Science and Technology there. His research interests cover image and video analysis, optimal character recognition, pattern recognition, machine learning and artificial intelligence. He has published over 40 papers in refereed international journals.

**Rui Wu** is an associate professor. He received a PhD degree in computer application technology from the Harbin Institute of Technology in 2010. His research interests include computer vision, character recognition, robot intelligence, and embedded systems.

**Dansong Cheng** received the BS and PhD degrees from the Harbin Institute of Technology, China, in 1997 and 2009, respectively, and the MSc degree in communication engineering from the Chiba Institute of Technology, Chitanma, Japan, in 2001. Since 2002, he has been with the Harbin Institute of Technology, where he became an associate professor in 2012. His current research interests include machine learning, medical image processing, and pattern recognition.

**Xiaopeng Fan** received the BS and MS degrees from the Harbin Institute of Technology in 2001 and 2003, respectively, and the PhD degree from the Hong Kong University of Science and Technology in 2009. He was with the Intel China Software Laboratory, Shanghai, China, as a software engineer, from 2003 to 2005. He joined the School of Computer Science and Technology, HIT, in 2009, where he is currently a professor. He has authored or co-authored over 80 technical journal and conference papers. His research interests include image/video processing and wireless communication.

**Xianglong Tang** received his PhD degree from the Harbin Institute of Technology, China, in 1995. He is currently a professor at the School of Computer Science and Technology and the director of the Research Center of Pattern Recognition, both at the Harbin Institute of Technology. His main research interests are focused on Chinese character recognition, medical imaging and biometrics, computer vision and pattern recognition. He has published over 80 papers in refereed international journals.

Received: 30 November 2018
Revised: 29 March 2019
Accepted: 29 April 2019