

A Regularization Approach for Instance-Based Superset Label Learning

Chen Gong, Tongliang Liu, Yuanyan Tang, *Fellow, IEEE*, Jian Yang, Jie Yang, and Dacheng Tao, *Fellow, IEEE*

Abstract—Different from the traditional supervised learning in which each training example has only one explicit label, superset label learning (SLL) refers to the problem that a training example can be associated with a set of candidate labels, and only one of them is correct. Existing SLL methods are either regularization-based or instance-based, and the latter of which has achieved state-of-the-art performance. This is because the latest instance-based methods contain an explicit disambiguation operation that accurately picks up the groundtruth label of each training example from its ambiguous candidate labels. However, such disambiguation operation does not fully consider the mutually exclusive relationship among different candidate labels, so the disambiguated labels are usually generated in a nondiscriminative way, which is unfavorable for the instance-based methods to obtain satisfactory performance. To address this defect, we develop a novel regularization approach for instance-based superset label (RegISL) learning so that our instance-based method also inherits the good discriminative ability possessed by the regularization scheme. Specifically, we employ a graph to represent the training set, and require the examples that are adjacent on the graph to obtain similar labels. More importantly, a discrimination term is proposed to enlarge the gap of values between possible labels and unlikely labels for every training example. As a result, the intrinsic constraints among different candidate labels are deployed, and the

disambiguated labels generated by RegISL are more discriminative and accurate than those output by existing instance-based algorithms. The experimental results on various tasks convincingly demonstrate the superiority of our RegISL to other typical SLL methods in terms of both training accuracy and test accuracy.

Index Terms—Concave convex procedure (CCCP), disambiguation, regularization, superset label learning (SLL).

I. INTRODUCTION

IN SUPERSET label learning (SLL), one training example can be ambiguously labeled with multiple candidate labels, among which only one is correct. This is different from the conventional supervised classification which works on the training examples with each of them only has one explicit label.

SLL has a variety of applications. For example, an episode of a video or TV serial may contain several characters chatting with each other, and their faces may appear simultaneously in a screenshot. We also have access to the scripts and dialogues indicating the characters' names. However, these information only reveals who are in the given screenshot, but does not build the specific one-to-one correspondence between the characters' faces and the appeared names. Therefore, each face in the screenshot is ambiguously named, and our target is to determine the groundtruth name of each face in the screen shot [see Fig. 1(a)]. Another similar application is that in a photograph collection, such as newsletters or family album, each photo may be annotated with a description indicating who are in this photo. However, the detailed identity of each person in the photo is not specified, so matching the persons with their real names is useful [see Fig. 1(b)]. SLL problem also arises in crowdsourcing, in which each example (image or text) is probably assigned multiple labels by different annotators. Nevertheless, some of the labels may be incorrect or biased because of the difference among various annotators in terms of expertise or cultural background, so it is necessary to find the most suitable label of every example resided in the candidate labels [see Fig. 1(c)]. In above applications, manually labeling the groundtruth label of each example will incur unaffordable monetary or time cost, so SLL can be an ideal tool for tackling such problems with ambiguously labeled examples.

SLL [1] is also known as “partial label learning” [2]–[4] and “ambiguously label learning” [5], [6]. For the consistency of our presentation, we will use the term SLL throughout this paper. SLL is formally defined as follows. Suppose we have n training examples $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \in \mathbb{R}^d$ with

Manuscript received July 29, 2016; revised December 16, 2016; accepted February 11, 2017. Date of publication February 24, 2017; date of current version February 14, 2018. This work was supported in part by the NSFC of China under Grant 61602246, Grant 61572315, Grant 91420201, Grant 61472187, Grant 61502235, Grant 61233011, and Grant 61373063, in part by the 973 Plan of China under Grant 2014CB349303 and Grant 2015CB856004, in part by the Program for Changjiang Scholars, and in part by the Australian Research Council Projects under Grant DP-140102164, Grant FT-130101457, and Grant LP-150100671. This paper was recommended by Associate Editor S. Ventura.

C. Gong is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China, and also with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: chen.gong@njust.edu.cn).

T. Liu is with the School of Software, Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia (e-mail: tliang.liu@gmail.com).

Y. Tang is with the Faculty of Science and Technology, University of Macau, Macau 999078, China, and also with the College of Computer Science, Chongqing University, Chongqing 400000, China (e-mail: yytang@umac.mo).

J. Yang is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: csjyang@njust.edu.cn).

J. Yang is with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: jieyang@sjtu.edu.cn).

D. Tao is with the School of Information Technologies and the Faculty of Engineering and Information Technologies, University of Sydney, Sydney, NSW 2008, Australia (e-mail: dacheng.tao@sydney.edu.au).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCYB.2017.2669639



Fig. 1. Some example applications of SLL. (a) Screenshot of “Lost” TV serial (Season 1), in which three characters’ faces are detected. From the scripts provided below, we can infer that both Kate and Charlie appear in this screenshot. However, it still remains unclear to which face corresponds to Kate and which face belongs to Charlie. (b) News image and its description from the news website “<http://fox17online.com/>.” From the textual description we know that these two people are Vladimir Putin and Barack Obama. However, which face corresponds to Vladimir Putin or Barack Obama is not clearly indicated. (c) Image of elk, which is an animal very similar to both cow and deer. In the application of crowdsourcing, the involved annotators may have different levels of expertise, so different labels are possibly provided by the different annotators, which can be either correct or incorrect.

dimensionality d , and their candidate labels are recorded by n label sets S_1, S_2, \dots, S_n , respectively. Therefore, the entire candidate label space consisted of c possible class labels has the size 2^c . Besides, we assume that the groundtruth labels of these n training examples are y_1, y_2, \dots, y_n with $y_i \in S_i$ ($i = 1, 2, \dots, n$), whereas they are unknown to the learning algorithms. Therefore, given the output label set denoted by $\mathcal{Y} = \{1, 2, \dots, c\}$, the target of an SLL algorithm is to build a classifier f based on \mathcal{X} so that it can accurately predict the single unambiguous label $y_i \in \mathcal{Y}$ of an unseen test example \mathbf{x}_i .

A. Related Work

To the best of our knowledge, the concept of SLL was first proposed by Grandvalet [7], who elegantly adapts the traditional logistic regression to superset label cases. After that, there are mainly two threads for tackling the SLL problem: 1) regularization-based models and 2) instance-based models.

1) *Regularization-Based Models*: Regularization-based models try to achieve maximum margin effect by developing various loss functions. For example, Jin and Ghahramani [8] first assumed that every element in the candidate set S_i ($i = 1, 2, \dots, n$) has equal probability to be the correct label, and designs a “naive” superset label loss. Next, considering that it is inappropriate to treat all the candidate labels equally, they further propose to disambiguate the candidate labels, i.e., directly discovering each example’s groundtruth label from its multiple candidate labels, so that a discriminative loglinear model can be built. Besides, Cour *et al.* [2], [9] held that the above naive loss is loose compared to the real superset label 0-1 loss $\mathcal{L}_{01}(f(\mathbf{x}_i), S_i) = \mathbb{1}[f(\mathbf{x}_i) \notin S_i]$,¹ so they propose another novel surrogate loss that is a tighter approximation to the real 0-1 loss than the naive loss. To be specific, this loss function is formulated as $\mathcal{L}(f(\mathbf{x}_i), S_i) = \Psi[(1/|S_i|) \sum_{j \in S_i} f_j(\mathbf{x}_i)] + \sum_{j \notin S_i} \Psi[-f_j(\mathbf{x}_i)]$

where $\Psi[\cdot]$ can be hinge, exponential or logistic loss. Here, the first term computes the mean value of the scores $f_j(\mathbf{x}_i)$ of the labels in S_i . However, this averaging strategy has a critical shortcoming that its effectiveness can be largely decreased by the false positive label(s) $S_i - y_i$ in the candidate label set S_i . As a result, the training process will be dominated by these false positive labels and the final model output can be biased. Therefore, Nguyen and Caruana [10] developed the superset label hinge loss that maximizes the margin between the maximum model output among candidate labels and that among the remaining noncandidate labels, namely $\mathcal{L}(f(\mathbf{x}_i), S_i) = \max(0, 1 - [\max_{y_i \in S_i} f(\mathbf{x}_i, y_i; \omega) - \max_{y'_i \notin S_i} f(\mathbf{x}_i, y'_i; \omega)])$ where ω is the model parameter. Differently, Hüllermeier and Cheng [11] proposed a generalized loss with its expression $\mathcal{L}(f(\mathbf{x}_i), S_i) = \min_{y_i \in S_i} \Psi[y_i, f(\mathbf{x}_i)]$, where $\Psi[\cdot]$ represents the logistic loss. However, above two formulations do not discriminate the groundtruth label y_i from other candidate labels. Therefore, Yu and Zhang [4] devised a new SLL maximum margin formulation based on support vector machines (SVMs) which directly maximizes the margin between the groundtruth label and all other labels. The corresponding loss function is $\mathcal{L}(f(\mathbf{x}_i), S_i) = f(\mathbf{x}_i, y_i; \omega) - \max_{y'_i \neq y_i} f(\mathbf{x}_i, y'_i; \omega)$. Different from above methods that only assume that one example is associated with a set of candidate labels, Jie and Orabona [12] considered a generalized setting in which each training example is a bag containing multiple instances and is associated with a set of candidate label vectors. Each label vector encodes the possible labels for the instances in the bag, and only one of them is fully correct.

For the theoretical aspect, Cid-Sueiro [13] studied the general necessary and sufficient condition for designing an SLL loss function, and provide a detailed procedure to construct a proper SLL loss under practical situations. Cid-Sueiro *et al.* [14] also revealed that the consistency of loss functions depends on the mixing matrix, which refers to the transition matrix relating the candidate labels and the groundtruth label. More generally, Liu and Dietterich [15]

¹The operation “ $\mathbb{1}[\cdot]$ ” returns 1 if the argument within the bracket holds true, and 0 otherwise.

discussed the learnability of regularization-based SLL approaches, and reveal that the key to achieving learnability is that the expected classification error of any hypothesis in the space can be bounded by the superset label 0-1 loss averaged over the entire training set.

Other representative regularization-based SLL algorithms include [6], [16], [17] that utilize coding theory, [1] that employs the conditional multinomial mixture model, and [18] that leverages the low-rank assumption [19], [20] to capture the example-label correspondences.

2) *Instance-Based Models*: Instance-based models usually construct a nonparametric classifier on the training set, and the candidate label set of a training example can be either disambiguated or kept ambiguous as it originally presents. Hüllermeier and Beringer [5] proposed a series of nonparametric models, such as superset label K -nearest neighborhood classifier and decision tree. The models in [5] do not have a disambiguation operation and directly use the ambiguous label sets for training and testing. Differently, Zhang and Yu [3] proposed an iterative label propagation scheme to disambiguate the candidate labels of training examples. Furthermore, considering that the disambiguation process in current methods simply focuses on manipulating the label space, Zhang *et al.* [21] advocated making full use of the manifold information [22] embedded in the feature space, and propose a feature-aware disambiguation.

B. Our Motivation

Although the method proposed in [3] generally obtains the best performance among all existing SLL algorithms, it still suffers from several drawbacks. First, as an instance-based method, it falls short of discovering the mutually exclusive relationship among different candidate labels, and does not take specific measures to highlight the potential groundtruth label during the disambiguation process. Second, as an iterative algorithm, the convergence property of the propagation sequence is only empirically illustrated and does not have a theoretical guarantee.

To address above two shortcomings, we propose a regularization approach for instance-based superset label learning, and termed it as “RegISL.” The advantages of our RegISL are twofold: first, to make the disambiguated labels discriminative, we design a proper discrimination regularizer along with the related constraints to increase the gap of scores between possible candidate labels and unlikely candidate labels. As a result, the potential groundtruth labels will become prominent, whereas the unlikely labels will be suppressed. Second, to avoid the convergence problem of iterative algorithm like [3], we solve the designed optimization problem via the augmented Lagrangian multiplier (ALM) method [23], [24] which will always finds a stationary solution. Besides, due to the non-convexity of the augmented Lagrangian objective function, we show that it can be decomposed as the difference of two convex components and then minimized by the concave convex procedure (CCCP) [25].

We empirically test our RegISL and other representative SLL methodologies [1]–[5], [17] on various practical

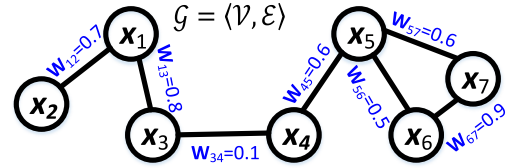


Fig. 2. Illustration of graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where in this example the seven circles represent the node set $\mathcal{V} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_7\}$, and the lines connecting them constitute the edge set $\mathcal{E} = \{\mathbf{W}_{12}, \mathbf{W}_{13}, \dots, \mathbf{W}_{67}\}$. The edge weights \mathbf{W}_{ik} ($i, k = 1, 2, \dots, 7$) are indicated besides the edges in blue color.

applications, such as character-name association in TV show, ambiguous image classification, automatic face naming in news images, and bird sound classification. The experimental results suggest that in most cases the proposed RegISL is able to outperform other competing baselines in terms of both training accuracy and test accuracy.

II. MODEL DESCRIPTION

This section introduces our nonparametric instance-based method RegISL. In the training stage (Section II-A), a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is established on the training set to capture the relationship between pairs of training examples, where \mathcal{V} is the node set representing all n training examples and \mathcal{E} is the edge set encoding the similarities between these nodes (see Fig. 2). In this paper, two examples \mathbf{x}_i and \mathbf{x}_k are linked by an edge in \mathcal{G} if one of them belongs to the K nearest neighbors of the other one, and the edge weight (i.e., the similarity between \mathbf{x}_i and \mathbf{x}_k) is computed by the Gaussian kernel function [26], [27]

$$\mathbf{W}_{ik} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_k\|^2}{2\theta^2}\right) \quad (1)$$

where θ denotes the kernel width. In contrast, \mathbf{W}_{ik} is set to 0 if there is no edge between \mathbf{x}_i and \mathbf{x}_k . After that, a regularized objective function is built on \mathcal{G} , which is able to disambiguate the candidate labels and discover the unique real label of every training example. In the test stage (Section II-B), the test example \mathbf{x}_t is assigned label y_t (y_t takes a value from $1, 2, \dots, c$ with c being the total number of classes) based on the disambiguated labels of its K nearest neighbors in the training set.

A. Training Stage

For our instance-based RegISL, the main target of training stage is to pick up the real label y_i of each training example \mathbf{x}_i from its candidate label set \mathcal{S}_i . The established graph \mathcal{G} can be quantified by the adjacency matrix \mathbf{W} , where its (i, k) th element is \mathbf{W}_{ik} if $i \neq k$ and 0 otherwise [28], [29].

Similar to [3], the candidate labels of a training example \mathbf{x}_i (i takes a value from $1, 2, \dots, n$) is represented by a c -dimensional label vector \mathbf{Y}_i , which is

$$\mathbf{Y}_{ij} := \begin{cases} 1/|\mathcal{S}_i|, & \mathbf{x}_i \text{ has the candidate label } j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where $|\mathcal{S}_i|$ denotes the size of set \mathcal{S}_i . Note that the sum of all the elements in every \mathbf{Y}_i is 1 according to the definition of (2).

Furthermore, we use the vectors $\mathbf{F}_1, \mathbf{F}_2, \dots, \mathbf{F}_n \in \mathbb{R}^{1 \times c}$ to record the obtained labels of training examples $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$, respectively, in which \mathbf{F}_{ij} can be understood as the probability of \mathbf{x}_i belonging to the class j , then our regularization model for RegISL can be expressed as

$$\begin{aligned} \min_{\mathbf{F}_1, \dots, \mathbf{F}_n} \quad & \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^n \mathbf{W}_{ik} \|\mathbf{F}_i - \mathbf{F}_k\|_2^2 + \alpha \sum_{i=1}^n \sum_{j \in \Omega_i} (\mathbf{F}_{ij} - \mathbf{Y}_{ij})^2 \\ & - \beta \sum_{i=1}^n \|\mathbf{F}_i\|_2^2 \\ \text{s.t.} \quad & \sum_{j=1}^c \mathbf{F}_{ij} = 1, \quad \mathbf{F}_{ij} \geq 0, \quad \forall i = 1, 2, \dots, n. \end{aligned} \quad (3)$$

In (3), the set Ω_i includes the subscripts of zero elements in \mathbf{Y}_i , “ $\|\cdot\|_2$ ” computes the l_2 norm of the vector, and α and β are non-negative tradeoff parameters controlling the relative weights of the three terms in the objective function.

The first term in the objective function of (3) is called *smoothness term*, which requires the two examples connected by a strong edge (i.e., the edge weight is large) in \mathcal{G} to obtain similar labels [28], [30], [31], so minimizing this smoothness term will force \mathbf{F}_i to get close to \mathbf{F}_k if \mathbf{W}_{ik} is large. The second term is called *fidelity term*, which suggests that if \mathbf{x}_i 's candidate label set \mathcal{S}_i does not contain the label j (i.e., $\mathbf{Y}_{ij} = 0$), then the j th element in the finally obtained label vector \mathbf{F}_i should also be zero. Although there are many other ways to character the difference between \mathbf{F}_{ij} and \mathbf{Y}_{ij} , here we simply adopt the quadratic form as it is perhaps the simplest way to compare \mathbf{F}_{ij} and \mathbf{Y}_{ij} . This form has also been widely used by many semi-supervised learning methodologies, such as [30], [32], and [33]. The third discrimination term along with the normalization constraint $\sum_{j=1}^c \mathbf{F}_{ij} = 1$ and non-negative constraint $\mathbf{F}_{ij} \geq 0$, critically makes the obtained \mathbf{F}_i to be discriminative. That is to say, by requiring the elements in \mathbf{F}_i non-negative and summing up to 1, minimizing $-\|\mathbf{F}_i\|_2^2$ (i.e., maximizing $\|\mathbf{F}_i\|_2^2$) will widen the gap of values between possible labels and unlikely labels of \mathbf{x}_i , and thus yielding discriminative and confident label vector \mathbf{F}_i . The detailed reasons are explained as follows.

Suppose that we are dealing with a binary classification problem (i.e., $c = 2$), and the label vector of example \mathbf{x}_i is $\mathbf{F}_i = [\mathbf{F}_{i1}, \mathbf{F}_{i2}]$. If \mathbf{x}_i is initially associated with the ambiguous candidate labels 1 and 2 (i.e., $\mathbf{Y}_{ij} = [0.5, 0.5]$), we hope that the finally obtained \mathbf{F}_i can approach to $[1, 0]$ or $[0, 1]$, which confidently implies that \mathbf{x}_i belongs to the first or second class. In contrast, the output close to $\mathbf{F}_i = [0.5, 0.5]$ is not encouraged because such \mathbf{F}_i does not convey any information for deciding \mathbf{x}_i 's real label. To this end, we impose the non-negative and normalization constraints on \mathbf{F}_i as in (3), then its elements \mathbf{F}_{i1} and \mathbf{F}_{i2} will only select the values along the red line in Fig. 3. Furthermore, we take the red line as x -axis and plot the squared l_2 norm of \mathbf{F}_i under different \mathbf{F}_{i1} and \mathbf{F}_{i2} (see the blue curve). It can be clearly observed that $\|\mathbf{F}_i\|_2^2$ hits the lowest value when both \mathbf{F}_{i1} and \mathbf{F}_{i2} are equal to 0.5, and $\|\mathbf{F}_i\|_2^2$ gradually increases when $[\mathbf{F}_{i1}, \mathbf{F}_{i2}]$ approaches to $[0, 1]$ or $[1, 0]$. Therefore, the label vector \mathbf{F}_i with large

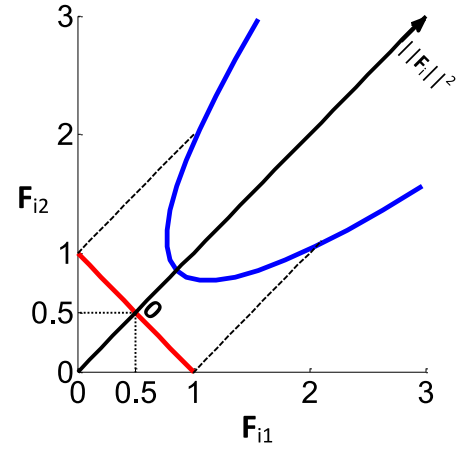


Fig. 3. Motivation of our introduced discrimination term along with the non-negative and normalization constraints. Suppose the example \mathbf{x}_i 's label vector is $\mathbf{F}_i = [\mathbf{F}_{i1}, \mathbf{F}_{i2}]$, then the valid outputs of \mathbf{F}_i satisfying the constraints in (3) are on the red line $\mathbf{F}_{i1} + \mathbf{F}_{i2} = 1$ ($\mathbf{F}_{i1} \geq 0, \mathbf{F}_{i2} \geq 0$). Taking this red curve as x -axis and $(0.5, 0.5)$ as original point, the value of $\|\mathbf{F}_i\|_2^2$ with varying \mathbf{F}_{i1} and \mathbf{F}_{i2} is recorded by the blue curve. We observe that the smallest $\|\mathbf{F}_i\|_2^2$ corresponds to the most ambiguous label vector $[0.5, 0.5]$, while $\|\mathbf{F}_i\|_2^2$ becomes large when $\mathbf{F}_i = [\mathbf{F}_{i1}, \mathbf{F}_{i2}]$ gets close to the discriminative results $[1, 0]$ and $[0, 1]$.

norm is encouraged by the discrimination term in (3), so that the obtained \mathbf{F}_i prefers definite results $[0, 1]$ or $[1, 0]$ and meanwhile avoids the ambiguous outputs that are close to $[0.5, 0.5]$.

For ease of optimizing (3), we may reformulate it into a compact formation. Based on \mathcal{G} 's adjacency matrix \mathbf{W} , we further define a diagonal degree matrix \mathbf{D} with the i th diagonal element representing \mathbf{x}_i 's degree computed by $\mathbf{D}_{ii} = \sum_{j=1}^n \mathbf{W}_{ij}$. Therefore, a positive semi-definite graph Laplacian matrix can be calculated as $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Besides, we stack the row vectors $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_n$ as $\mathbf{Y} = (\mathbf{Y}_1^\top, \mathbf{Y}_2^\top, \dots, \mathbf{Y}_n^\top)^\top$ to establish a $n \times c$ candidate label matrix \mathbf{Y} . Similarly, the label matrix \mathbf{F} to be optimized is established by $\mathbf{F} = (\mathbf{F}_1^\top, \mathbf{F}_2^\top, \dots, \mathbf{F}_n^\top)^\top$. Furthermore, by defining $\mathbf{1}_c, \mathbf{1}_n$ and $\mathbf{O}_{n \times c}$ as the c -dimensional all-one vector, n -dimensional all-one vector, and $n \times c$ -dimensional all-zero matrix, respectively, (3) can be rewritten as

$$\begin{aligned} \min_{\mathbf{F}} \quad & \text{tr}(\mathbf{F}^\top \mathbf{L} \mathbf{F}) + \alpha \|\mathbf{H} \odot (\mathbf{F} - \mathbf{Y})\|_F^2 - \beta \|\mathbf{F}\|_F^2 \\ \text{s.t.} \quad & \mathbf{F} \mathbf{1}_c = \mathbf{1}_n, \quad \mathbf{F} \geq \mathbf{O}_{n \times c}. \end{aligned} \quad (4)$$

In (4), “ $\|\cdot\|_F$ ” computes the Frobenius norm of corresponding matrix, and “ \odot ” refers to the elementwise product. \mathbf{H} is a $\{0, 1\}$ -binary matrix with the element $\mathbf{H}_{ij} = 1$ if $\mathbf{Y}_{ij} = 0$ and 0 otherwise.

Since (4) is a constrained optimization problem, we may use the method of ALM to find its solution. Compared to the traditional Lagrangian method, ALM adds an additional quadratic penalty function to the objective, which leads to faster convergence rate and lower computational cost [34]. Therefore, by introducing the multipliers Λ_1 and Λ_2 to deal with the non-negative constraint and normalization constraint, respectively,

the augmented Lagrangian function is expressed as

$$\begin{aligned} J(\mathbf{F}, \Lambda_1, \Lambda_2, \sigma) = & \text{tr}(\mathbf{F}^\top \mathbf{L} \mathbf{F}) + \alpha \|\mathbf{H} \odot (\mathbf{F} - \mathbf{Y})\|_F^2 \\ & - \beta \|\mathbf{F}\|_F^2 + \frac{1}{2\sigma} \text{tr}(\mathbf{M}^\top \mathbf{M} - \Lambda_1^\top \Lambda_1) \\ & - \Lambda_2^\top (\mathbf{F} \mathbf{1}_c - \mathbf{1}_n) + \frac{\sigma}{2} \|\mathbf{F} \mathbf{1}_c - \mathbf{1}_n\|_2^2 \end{aligned} \quad (5)$$

where $\mathbf{M} = \max\{\mathbf{O}_{n \times c}, \Lambda_1 - \sigma \mathbf{F}\}$ is an auxiliary variable that enforces the obtained optimal \mathbf{F} (i.e., \mathbf{F}^*) to be non-negative. The operation “ $\max(\mathbf{A}, \mathbf{B})$ ” returns a matrix with its (i, j) th element being the largest element between \mathbf{A}_{ij} and \mathbf{B}_{ij} . The variable $\sigma > 0$ is the penalty coefficient.

Based on (5), the optimal solution of (4) can be obtained by alternately updating \mathbf{F} , Λ_1 , Λ_2 , and σ , among which Λ_1 , Λ_2 , and σ can be easily updated via the conventional rules of ALM, namely

$$\begin{cases} \Lambda_1 := \max\{\mathbf{O}_{n \times c}, \Lambda_1 - \sigma \mathbf{F}\} & (6) \\ \Lambda_2 := \Lambda_2 - \sigma (\mathbf{F} \cdot \mathbf{1}_c - \mathbf{1}_n) & (7) \\ \sigma := \min\{\rho \sigma, 10^8\}. & (8) \end{cases}$$

In (8), the operation “min” selects the smallest value in the bracket, and $\rho = 1.1$ is the parameter that makes σ gradually increase in each loop so that the normalization constraint can be finally satisfied.

However, the updating of \mathbf{F} is difficult because (5) regarding \mathbf{F} is nonconvex due to the nonpositive term $-\beta \|\mathbf{F}\|_F^2$. Therefore, we use the method of CCCP proposed by Yuille and Rangarajan [25] to update \mathbf{F} . CCCP can be regarded as a majorization-minimization algorithm [35] that solves the original nonconvex problem as a sequence of convex programming. Specifically, the main idea of CCCP is to decompose the nonconvex objective function $J(\mathbf{F})$ as the difference of two convex functions $J_1(\mathbf{F})$ and $J_2(\mathbf{F})$, namely $J(\mathbf{F}) = J_1(\mathbf{F}) - J_2(\mathbf{F})$; and in each iteration $J_2(\mathbf{F})$ is replaced by its first order Taylor approximation $\tilde{J}_2(\mathbf{F})$, and the original objective function $J(\mathbf{F})$ is then approximated by the convex $\tilde{J}(\mathbf{F}) = J_1(\mathbf{F}) - \tilde{J}_2(\mathbf{F})$. Theoretical analyses suggest that CCCP is always able to converge to a local minima [36]. In our case, we choose the two convex functions $J_1(\mathbf{F})$ and $J_2(\mathbf{F})$ as

$$\begin{cases} J_1(\mathbf{F}) = \text{tr}(\mathbf{F}^\top \mathbf{L} \mathbf{F}) + \alpha \|\mathbf{H} \odot (\mathbf{F} - \mathbf{Y})\|_F^2 \\ \quad + \frac{1}{2\sigma} \text{tr}(\mathbf{M}^\top \mathbf{M} - \Lambda_1^\top \Lambda_1) - \Lambda_2^\top (\mathbf{F} \mathbf{1}_c - \mathbf{1}_n) \\ \quad + \frac{\sigma}{2} \|\mathbf{F} \mathbf{1}_c - \mathbf{1}_n\|_2^2 \\ J_2(\mathbf{F}) = \beta \|\mathbf{F}\|_F^2. \end{cases}$$

Therefore, in the t th iteration we may linearize $J_2(\mathbf{F})$ at $\mathbf{F}^{(t)}$ via Taylor approximation, and obtain $\tilde{J}_2(\mathbf{F}) = \beta[\|\mathbf{F}^{(t)}\|_F^2 + 2\text{tr}(\mathbf{F}^{(t)\top}(\mathbf{F} - \mathbf{F}^{(t)}))]$. As a result, the approximated objective function $\tilde{J}(\mathbf{F})$ for updating \mathbf{F} is

$$\begin{aligned} \tilde{J}(\mathbf{F}) = & J_1(\mathbf{F}) - \tilde{J}_2(\mathbf{F}) \\ = & \text{tr}(\mathbf{F}^\top \mathbf{L} \mathbf{F}) + \alpha \|\mathbf{H} \odot (\mathbf{F} - \mathbf{Y})\|_F^2 \\ & + \frac{1}{2\sigma} \text{tr}(\mathbf{M}^\top \mathbf{M} - \Lambda_1^\top \Lambda_1) \\ & - \Lambda_2^\top (\mathbf{F} \mathbf{1}_c - \mathbf{1}_n) + \frac{\sigma}{2} \|\mathbf{F} \mathbf{1}_c - \mathbf{1}_n\|_2^2 \\ & - \beta \left[\|\mathbf{F}^{(t)}\|_F^2 + 2\text{tr}(\mathbf{F}^{(t)\top}(\mathbf{F} - \mathbf{F}^{(t)})) \right]. \end{aligned} \quad (9)$$

Algorithm 1 CCCP for Minimizing (5)

- 1: **Input:** initial $\mathbf{F}^{(0)}$; stopping criteria $t_{\max} = 20$, $\epsilon_0 = 10^{-6}$
 - 2: Set $t = 0$;
 - 3: **repeat**
 - 4: Minimizing $\tilde{J}(\mathbf{F})$ in Eq. (9) via GD;
 - 5: $t := t + 1$;
 - 6: **until** $t = t_{\max}$ or $\|\mathbf{F}^{(t)} - \mathbf{F}^{(t-1)}\|_F \leq \epsilon_0$
 - 7: **Output:** \mathbf{F} that minimizes Eq. (5)
-

Algorithm 2 ALM for Optimizing (4)

- 1: **Input:** training examples $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ with ambiguous label sets $\mathcal{S}_1, \dots, \mathcal{S}_n$; tuning parameters α, β, K, θ ; stopping criteria $\text{loop}_{\max} = 40$, $\epsilon_1 = 10^{-4}$
 - 2: Construct KNN graph \mathcal{G} , compute the graph Laplacian matrix \mathbf{L} ;
 - 3: Compute \mathbf{Y} via Eq. (2);
 - 4: Set $\text{loop} = 0$;
 - 5: **repeat**
 - 6: Update \mathbf{F} via CCCP in Algorithm 1;
 - 7: Update Λ_1 via Eq. (6);
 - 8: Update Λ_2 via Eq. (7);
 - 9: Update σ via Eq. (8);
 - 10: $\text{loop} := \text{loop} + 1$;
 - 11: **until** $\text{loop} = \text{loop}_{\max}$ or $\|\mathbf{F}^{(\text{loop})} - \mathbf{F}^{(\text{loop}-1)}\|_F \leq \epsilon_1$
 - 12: **Output:** optimal \mathbf{F}^* that minimizes Eq. (4)
-

In this paper, we employ the well-known gradient descent (GD) method to find the optimal \mathbf{F} that minimizes (9), in which the gradient of $\tilde{J}(\mathbf{F})$ with respect to \mathbf{F} is computed as

$$\begin{aligned} \nabla \tilde{J}(\mathbf{F}) = & 2\mathbf{L}\mathbf{F} + 2\alpha[\mathbf{H} \odot (\mathbf{F} - \mathbf{Y})] - \mathbf{M} - \Lambda_2 \cdot \mathbf{1}_c^\top \\ & + \sigma(\mathbf{F} \mathbf{1}_c - \mathbf{1}_n) \cdot \mathbf{1}_c^\top - 2\beta \mathbf{F}^{(t)} \end{aligned} \quad (10)$$

and the updating rule for GD is subsequently $\mathbf{F} := \mathbf{F} - \tau \nabla \tilde{J}(\mathbf{F})$ with τ being the stepsize. The detailed CCCP for updating \mathbf{F} in each loop is provided in Algorithm 1, and the entire ALM optimization process for finding (4)’s solution \mathbf{F}^* is summarized in Algorithm 2. It can be easily verified that the objective function and constraints in (4) are twice continuously differentiable, therefore according to [37] the convergence of the ALM process is theoretically guaranteed. Based on \mathbf{F}^* , every training example \mathbf{x}_i ($i = 1, 2, \dots, n$) will receive its unique valid label as $y_i = \arg \max_{j=1,2,\dots,c} \mathbf{F}_{ij}^*$, and the corresponding disambiguated label vector is $\bar{\mathbf{F}}_i$ with $\bar{\mathbf{F}}_{ij} = 1$ if $j = y_i$ and 0 otherwise.

B. Test Stage

Given the disambiguated labels $\bar{\mathbf{F}}_1, \bar{\mathbf{F}}_2, \dots, \bar{\mathbf{F}}_n$ of the n training examples, we predict the label y_t of a test example \mathbf{x}_t via two steps. First, we find the \mathbf{x}_t ’s K nearest training examples $\{\mathbf{x}_{k_i}\}_{i=1}^K$ in the Euclidean space, and compute the similarity between \mathbf{x}_t and $\{\mathbf{x}_{k_i}\}_{i=1}^K$ (i.e., $\{\mathbf{W}_{tk_i}\}_{i=1}^K$) via (1). The disambiguated labels of these K training examples are denoted by $\{\bar{\mathbf{F}}_{k_i}\}_{i=1}^K$. Second, a soft label vector \mathbf{F}_t is calculated as the weighted sum of $\{\bar{\mathbf{F}}_{k_i}\}_{i=1}^K$ by $\mathbf{F}_t = \sum_{i=1}^K \mathbf{W}_{tk_i} \bar{\mathbf{F}}_{k_i}$, and \mathbf{x}_t ’s label is finally decided as $y_t = \arg \max_{j=1,2,\dots,c} \mathbf{F}_{tj}$.

TABLE I
SUMMARY OF THE ADOPTED DATASETS

	# Examples	# Features	# Classes	Average # labels	Application
<i>Lost</i>	1122	512	16	2.23	Character-name association in TV Serial
<i>MSRCv2</i>	591	512	23	1.71	Ambiguous image classification
<i>Soccer Player</i>	17472	279	171	2.09	Automatic face naming in news images
<i>Bird Song</i>	4998	38	13	2.18	Bird sound classification

III. EXPERIMENTAL RESULTS

In this section, we compare the performances of our proposed RegISL with several existing typical SLL methods on various practical applications, such as character-name association in TV show, ambiguous image classification, automatic face naming in news images, and bird sound classification.

A variety of methods belonging to different threads mentioned in the introduction (Section I-A) serve as baselines for our comparison, which include the following.

- 1) *Regularization-Based Methods*: The compared approaches include SVM-like methodologies maximum margin SLL (M3SL) [4], convex loss for superset labels (CLSL) [2] and its simplified version with the naive loss (CLSL_Naive) proposed in [8]; coding theory-based methodology error-correcting output codes (ECOC) [17]; and probability-based logistic stick-breaking conditional multinomial model (LSB-CMM) [1]. Note that another typical SVM-like method [10] is not compared because its accuracy is consistently lower than the latest M3SL with a considerable margin as reported in [4].
- 2) *Instance-Based Methods*: The compared approaches include the traditional superset label KNN (SLKNN) [5], and the state-of-the-art method instance-based SLL (ISL) [3].

For fair comparison, all the above baselines except SLKNN are implemented by using the codes directly provided by the authors. Although the code of SLKNN is not publicly available, it is very easy to reproduce and we implement this algorithm by ourselves.

In each of the experiments below, we randomly split the dataset into five nonoverlapped folds, and conduct the fivefold cross validation on all the compared methods. In each partition, 80% examples with their ambiguous labels constitute the training set, and the rest 20% examples are used for testing. Note that in each partition we keep the ratio of the number of examples from each class in the training set approximately identical to that in the test set. The partitions are also kept identical for all the compared methods. The mean training accuracy and test accuracy averaged over the five different partitions are calculated to assess the classification ability of all the competing algorithms. Besides, we also use the Friedman test [38] with 90% confidence level to investigate whether the proposed RegISL is significantly superior/inferior to the adopted baselines.

A. Character-Name Association in TV Serial

As mentioned in Section I, it is meaningful to study how to build the one-to-one correspondence between each character appeared in the video and the real name indicated by the script. To this end, we use the *Lost* dataset provided in [2] and [9] to associate the characters in the TV serial *Lost* with their groundtruth names. This dataset contains totally 1122 registered face images across 16 characters, and each character has 18–204 images. Given a scene, each of the appeared faces corresponds to an example and it is ambiguously labeled by all the names in the aligned script. The average amount of candidate labels for a single example in this dataset is 2.23. In our experiment, we resize every face image to 30×20 pixels which is further characterized by a 512-dimensional GIST feature [39]. Please refer to Table I for the details of the adopted datasets.

The regularization parameter C in both CLSL and CLSL_Naive is set to the default optimal value 1000. The maximum value for regularization parameter C_{\max} in M3SL is set to 0.01 as recommended by Yu and Zhang [4]. The optimization problems in M3SL are efficiently solved by utilizing the off-the-shelf solvers LIBLINEAR [40] and CVX [41]. In ECOC, the codeword length L is adaptively determined as $L = \lceil 100 \times \log_2(c) \rceil$, where “ $\lceil \cdot \rceil$ ” rounds up the inside value to the nearest integer, and c is the number of classes as defined in Section I. The inherited SVM utilizes the RBF kernel with the width $\gamma = 0.5$, and the regularization parameter is $C = 5$. In LSB-CMM, the number of mixture components is 10, and the parameter for the involved Dirichlet prior is $\alpha = 0.05$ [1]. The balancing parameter α in the iteration expression of ISL is set to 0.9 according to [3]. For fair comparison, the number of neighbors K in ISL, SLKNN and RegISL is set to the same value 5. In this paper, the two tradeoff parameters α and β in RegISL are tuned to 1000 and 0.01, respectively. In Section III-F, we will study the sensitivity of these two parameters and also explain why we set α and β to these values.

The training accuracy and test accuracy obtained by all the algorithms are presented in Table II, in which the highest and second highest records are highlighted in red and blue color, respectively. Because SLKNN is a lazy learning algorithm that does not have a training process, its training accuracy is incomputable and thus is not reported. From Table II, we have some interesting findings.

First, the disambiguation operation mentioned in Section I is critical to improve the performance. We observe that SLKNN

TABLE II

EXPERIMENTAL RESULTS ON *Lost* DATASET. EACH RECORD REPRESENTS “MEAN ACCURACY \pm STANDARD DEVIATION.” THE BEST AND SECOND BEST RECORDS ARE MARKED IN RED AND BLUE, RESPECTIVELY. “ $\sqrt{(\times)}$ ” INDICATES THAT REGISL IS SIGNIFICANTLY BETTER (WORSE) THAN THE CORRESPONDING METHOD

	Training Accuracy	Test Accuracy
CLSL [2]	$0.785 \pm 0.016 \checkmark$	$0.701 \pm 0.030 \checkmark$
CLSL_Naive [2]	$0.734 \pm 0.017 \checkmark$	$0.663 \pm 0.018 \checkmark$
ISL [3]	$0.821 \pm 0.018 \checkmark$	0.708 ± 0.032
M3SL [4]	$0.742 \pm 0.005 \checkmark$	$0.668 \pm 0.028 \checkmark$
ECOC [17]	$0.728 \pm 0.013 \checkmark$	$0.659 \pm 0.036 \checkmark$
LSB-CMM [1]	$0.782 \pm 0.024 \checkmark$	$0.692 \pm 0.021 \checkmark$
SLKNN [5]	-	$0.603 \pm 0.020 \checkmark$
RegISL	0.852 ± 0.011	0.726 ± 0.026

and ECOC obtain the lowest test accuracy because they do not contain such disambiguation operation, so the noisy candidate labels of the training examples may impair the training quality and also decrease the test accuracy. CLSL generates higher training accuracy and test accuracy than CLSL_Naive because CLSL improves CLSL_Naive by not equally treating all the candidate labels any more. Therefore, CLSL pays more attention to the true positive labels of training examples than CLSL_Naive during the training stage and produces more satisfactory performance.

Second, the regularization technique adopted by our RegISL enhances the quality of existing disambiguation operation. Table II clearly shows that the proposed RegISL achieves the best performance among all the comparators. The averaged training accuracy and test accuracy are 0.852 and 0.726, respectively. Comparatively, another state-of-the-art instance-based method ISL performs slightly worse than RegISL, which suggests that introducing regularization to ISL helps to boost the classification accuracy. We think that two factors considered by (3) contribute to the improved performance: one is the smoothness term that models the label similarity between different examples on the graph, and the other one is the discrimination term that highlights the most likely labels from all the possible candidate labels for every training example. These two factors make the entire disambiguation operation of RegISL more accurate than ISL, which further brings about higher training accuracy. Besides, it is straightforward that a better disambiguated training set containing less incorrect labels will lead to more encouraging test performance, that is why our RegISL also obtains the best test accuracy when compared with other baselines.

B. Ambiguous Image Classification

To test the classification ability of different methods on ambiguous image classification, we follow [1] and [17] and use the *MSRCv2* dataset for our comparison. This dataset contains 591 natural images with totally 23 classes. Every image is segmented into several compact regions with specific semantic information, and the labels of segmented regions form the candidate label set for the entire image. Among the segmented

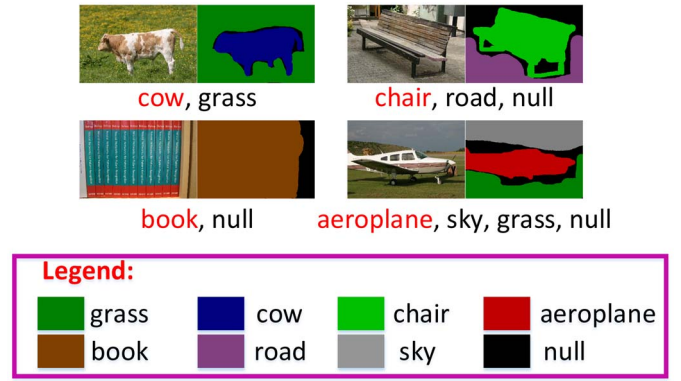


Fig. 4. Example images of the *MSRCv2* dataset. The labels of segmented regions are regarded as the candidate labels of the entire image (see the labels below the images), in which the label of the most dominant region is determined as the unique groundtruth label of the corresponding image, such as the labels “cow,” “chair,” “book,” and “airplane” that are marked in red.

TABLE III

EXPERIMENTAL RESULTS ON *MSRCv2* DATASET. EACH RECORD REPRESENTS MEAN ACCURACY \pm STANDARD DEVIATION. THE BEST AND SECOND BEST RECORDS ARE MARKED IN RED AND BLUE, RESPECTIVELY. $\sqrt{(\times)}$ INDICATES THAT REGISL IS SIGNIFICANTLY BETTER (WORSE) THAN THE CORRESPONDING METHOD

	Training Accuracy	Test Accuracy
CLSL [2]	$0.274 \pm 0.017 \checkmark$	$0.208 \pm 0.051 \checkmark$
CLSL_Naive [2]	$0.229 \pm 0.019 \checkmark$	$0.168 \pm 0.047 \checkmark$
ISL [3]	$0.634 \pm 0.015 \checkmark$	0.328 ± 0.043
M3SL [4]	$0.398 \pm 0.020 \checkmark$	$0.285 \pm 0.025 \checkmark$
ECOC [17]	$0.555 \pm 0.030 \checkmark$	$0.251 \pm 0.032 \checkmark$
LSB-CMM [1]	$0.369 \pm 0.007 \checkmark$	$0.292 \pm 0.027 \checkmark$
SLKNN [5]	-	$0.236 \pm 0.042 \checkmark$
RegISL	0.697 ± 0.019	0.333 ± 0.032

regions, the label of the most dominant region is taken as the single groundtruth label for the given image (see Fig. 4). Similar to the experiment on *Lost* dataset, we also adopt the 512-dimensional GIST feature to represent the images, and all feature vectors are normalized to unit length for all the competing methodologies.

The parameter settings of CLSL, CLSL_Naive, M3SL, ECOC, and LSB-CMM on *MSRCv2* are the same with those on *Lost* dataset, because they are directly suggested by the authors. The graph parameters K and θ for ISL, SLKNN and RegISL are, respectively, set to 10 and 0.1, where the optimal K is chosen from the set $\{5, 10, 15, 20\}$, and θ is selected from $\{0.01, 0.1, 1, 10\}$.

The experimental results are reported in Table III, which reveals that all the methods obtain relatively low accuracy. This is because *MSRCv2* dataset is quite challenging for SLL. First, this dataset is not large, but contains as many as 23 classes (see Table I), so the training examples belonging to every class are very sparse. Besides, the number of examples having a certain candidate label ranges from 24 to 184, therefore such insufficient and skewed training examples pose a great difficulty for training a reliable classifier. Second, Fig. 4

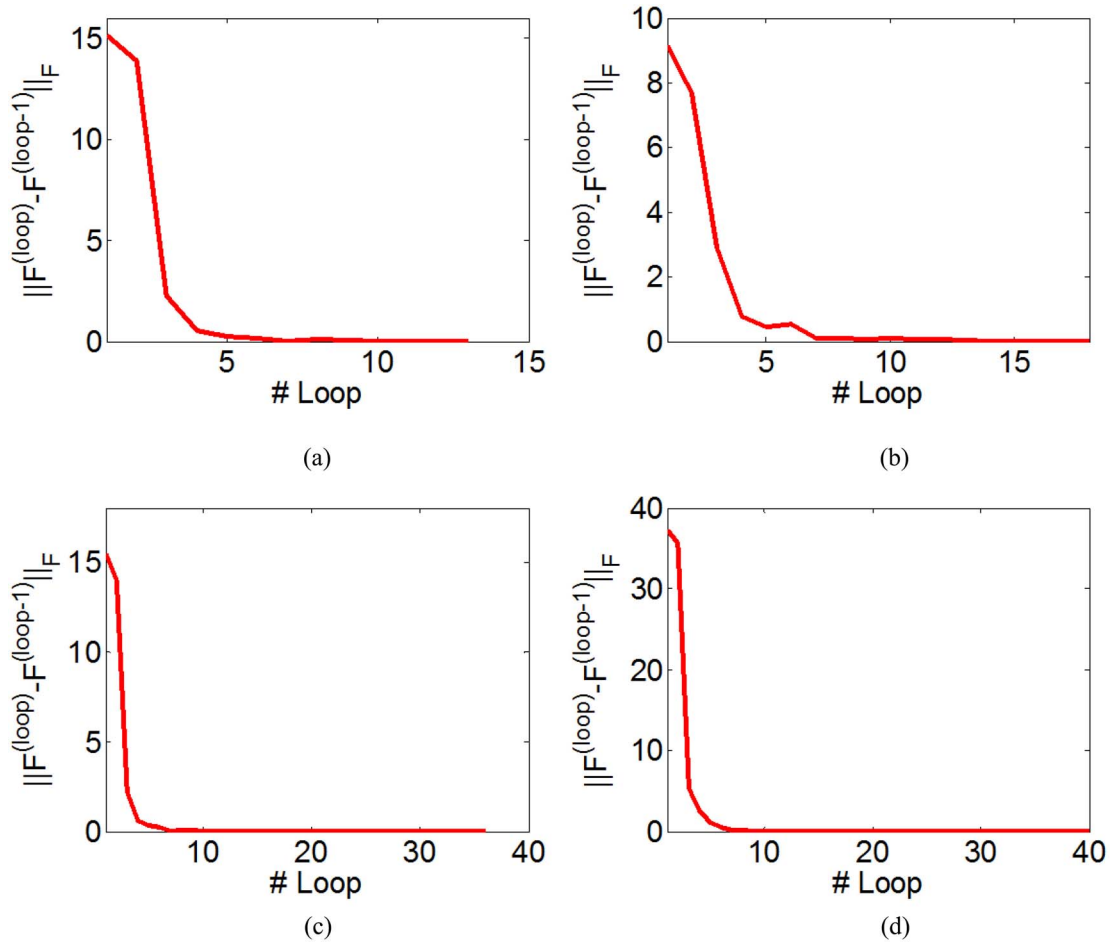


Fig. 5. The convergence curves of RegISL on the four adopted datasets. (a) *Lost*. (b) *MSRCv2*. (c) *Soccer Player*. (d) *Bird Song*.

reveals that the images in *MSRCv2* are very complex, and the dominant foreground is often surrounded by the background regions with false positive labels, which will mislead both the training and test stages. Although this dataset is quite challenging, Table III clearly indicates that the proposed RegISL still outperforms other methods with a noticeable margin in terms of either training accuracy or test accuracy. Specifically, it can be observed that RegISL leads the second best method ISL with the margins roughly 0.06 on training accuracy and 0.005 on test accuracy, which again demonstrate the superiority of our regularization strategy to the existing nonregularized instance-based model. In contrast, the training accuracy and test accuracy obtained by the remaining approaches like CLSL, CLSL_Naive, M3SL, ECOC, LSB-CMM, and SLKNN do not exceed 0.6 and 0.3, which are much worse than our RegISL.

C. Automatic Face Naming in News Images

It is often the case that in a news collection every image is accompanied by a short textual description to explain the content of this image. Such a news image may contain several faces and the associated description will indicate the names of the people appeared in this image. However, the further information about which face matches which name is not specified. Therefore, in this section, we use the *Soccer Player* [18], [42]

TABLE IV
EXPERIMENTAL RESULTS ON *Soccer Player* DATASET. EACH RECORD REPRESENTS MEAN ACCURACY \pm STANDARD DEVIATION. THE HIGHEST AND SECOND HIGHEST RECORDS ARE MARKED IN RED AND BLUE, RESPECTIVELY. \checkmark (\times) INDICATES THAT REGISL IS SIGNIFICANTLY BETTER (WORSE) THAN THE CORRESPONDING METHOD

	Training Accuracy	Test Accuracy
CLSL [2]	0.654 ± 0.005 \checkmark	0.371 ± 0.004 \checkmark
CLSL_Naive [2]	0.648 ± 0.003 \checkmark	0.366 ± 0.005 \checkmark
ISL [3]	0.676 ± 0.003	0.538 ± 0.007
M3SL [4]	0.648 ± 0.004 \checkmark	0.473 ± 0.005 \checkmark
ECOC [17]	0.681 ± 0.001	0.547 ± 0.004 \times
LSB-CMM [1]	0.672 ± 0.001 \checkmark	0.525 ± 0.003 \checkmark
SLKNN [5]	-	0.501 ± 0.003 \checkmark
RegISL	0.678 ± 0.002	0.538 ± 0.001

dataset to test the classification ability of various methods on dealing with news data.

The *Soccer Player* dataset is collected by Zeng *et al.* [18], which includes the names and images of soccer players from famous European football clubs downloaded from the “http://www.zimbio.com” website. There are totally 8640 images containing 17 472 faces across 1579 names. By following [18] and [42], we only retain 170 names that occur at least

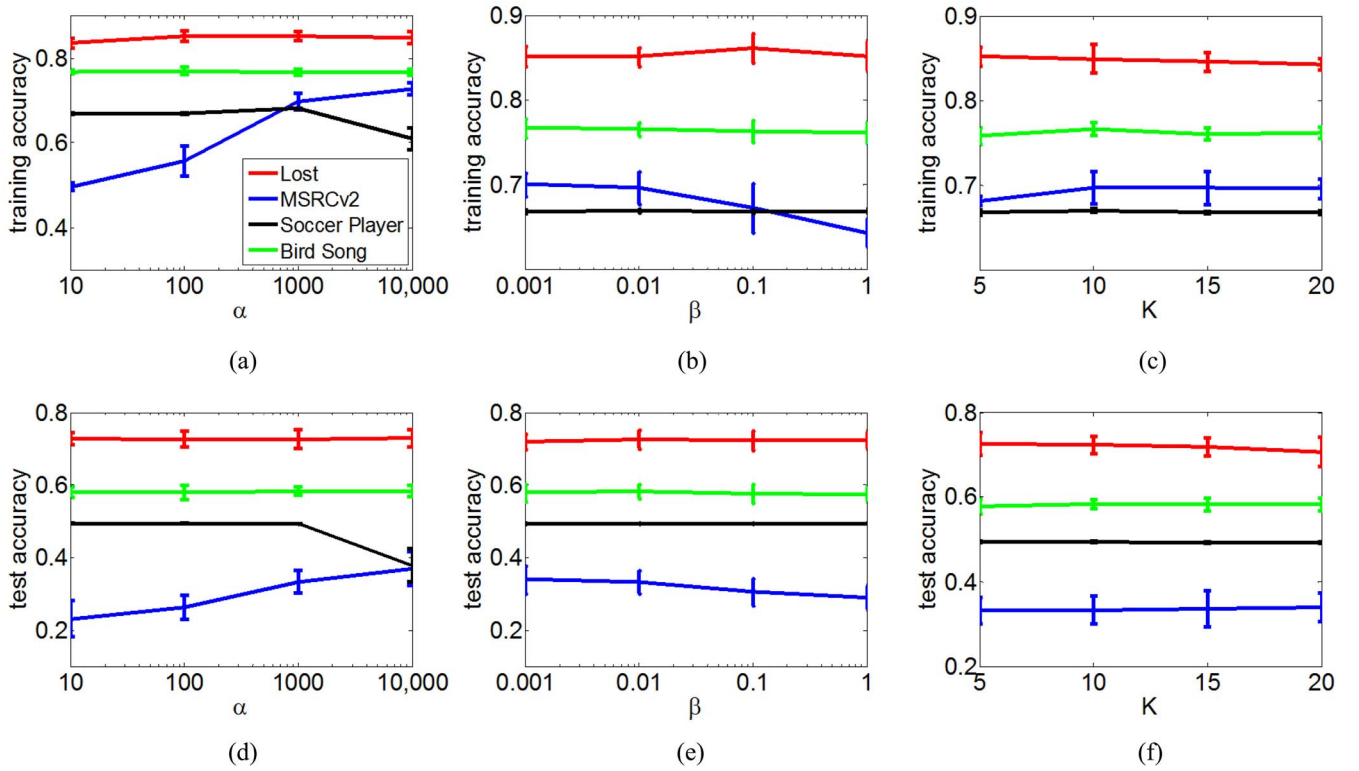


Fig. 6. Influence of tuning parameters α , β , and K to the final model output on the four datasets. The first column [(a) and (d)] shows the training accuracy and test accuracy obtained by RegISL under different choices of α . The second column [(b) and (e)] presents the variations of training accuracy and test accuracy with the increase of β . The third column [(c) and (f)] plots the training accuracy and test accuracy under different K .

20 times, and treat the remaining names as “Null” class. As a consequence, the faces appeared in every image are manually annotated using the real names from the descriptions or as Null class. Each face is represented by a 279-dimensional feature vector describing the 13 interest points (facial landmarks) detected by [43].

Table IV reports the experimental results, which reflect that ECOC achieves the best results on this dataset. Regarding the training accuracy, our RegISL is significantly better than CLSL, CLSL_Naive, M3SL, LSB-CMN, and comparable to ISL and ECOC. For test accuracy, RegISL performs favorably to CLSL, CLSL_Naive, M3SL, LSB-CMN, and SLKNN. However, it is inferior to the results generated by ECOC. Furthermore, we note that RegISL only falls behind ECOC by 0.003 in training accuracy and 0.009 in test accuracy, and it also generates the top level performance among the compared instance-based methods like SLKNN, ISL, and RegISL, so the performance of RegISL is still acceptable on this dataset.

D. Bird Sound Classification

Briggs et al. [44] established a dataset *Bird Song* which contains 548 bird sound recordings that last for ten seconds. Each recording is consisted of 1–40 syllables, leading to totally 4998 syllables included by the dataset. Each syllable is regarded as an example and is described by a 38-dimensional feature vector. Since every recording contains the songs produced by different species of birds, our target is to identify which example (i.e., syllable) corresponds to which kind of bird. In this

TABLE V
EXPERIMENTAL RESULTS ON *Bird Song* DATASET. EACH RECORD REPRESENTS MEAN ACCURACY \pm STANDARD DEVIATION. THE BEST AND SECOND BEST RESULTS ARE MARKED IN RED AND BLUE, RESPECTIVELY. \checkmark (\times) INDICATES THAT REGISL IS SIGNIFICANTLY BETTER (WORSE) THAN THE CORRESPONDING METHOD

	Training Accuracy	Test Accuracy
CLSL [2]	0.615 ± 0.003 \checkmark	0.414 ± 0.004 \checkmark
CLSL_Naive [2]	0.613 ± 0.001 \checkmark	0.414 ± 0.003 \checkmark
ISL [3]	0.736 ± 0.004 \checkmark	0.559 ± 0.011 \checkmark
M3SL [4]	0.658 ± 0.048 \checkmark	0.478 ± 0.036 \checkmark
ECOC [17]	0.361 ± 0.013 \checkmark	0.359 ± 0.015 \checkmark
LSB-CMM [1]	0.663 ± 0.006 \checkmark	0.482 ± 0.022 \checkmark
SLKNN [5]	-	0.552 ± 0.009 \checkmark
RegISL	0.766 ± 0.008	0.583 ± 0.002

dataset, the bird species appeared in every record are manually annotated, so they serve as the candidate labels for all the syllables inside this recording.

The number of neighbors K for ISL, SLKNN and our RegISL is set to 10, and the kernel width θ in (1) is tuned to 1 to achieve the best performance. The tradeoff parameters α and β are adjusted to 1000 and 0.01 as mentioned in Section III-A. We present the training accuracy and test accuracy of all the compared methods in Table V. A notable fact revealed by Table V is that the instance-based methods (e.g., SLKNN, ISL, and RegISL) generate better performance

than the regularization-based methodologies such as CLPL, CLPL_Naive, M3PL, LSB-CMM, and ECOC. Among the three instance-based methods, ISL and SLKNN have already achieved very encouraging performances. However, our proposed RegISL can still improve their performances with a noticeable margin regarding either training accuracy or test accuracy. Therefore, the effectiveness of RegISL is demonstrated, which again suggests that integrating the regularization technique with the instance-based framework is beneficial to achieving the improved performance.

E. Illustration of Convergence

In Section II, we explained that the iteration process of ALM in our algorithm will converge to a stationary point. Here, we present the convergence curves of RegISL on the adopted four datasets, including *Lost*, *MSRCv2*, *Soccer Player*, and *Bird Song*. In Fig. 5, the difference of the optimization variable \mathbf{F} between successive loops is particularly evaluated. We observe that the value of $\|\mathbf{F}^{(\text{loop})} - \mathbf{F}^{(\text{loop}-1)}\|_F$ gradually vanishes when the loops proceed, and the ALM process usually reaches the convergent point between the 13th–40th loop. Therefore, the convergence of the optimization process employed by our RegISL is demonstrated.

F. Effect of Tuning Parameters

The tradeoff parameters α and β in (3), and the number of neighbors K are three key tuning parameters in our RegISL model. They should be manually adjusted before implementing the proposed algorithm. Therefore, this section investigates how their variations influence the training accuracy and test accuracy produced by RegISL. The four datasets appeared in Sections III-A–III-D are used here for our experiments.

In every dataset, we investigate the effects of α , β , and K on the model output by fixing two of them and then examining the classification accuracy with respect to the change of the remaining one. From the experimental results shown in Fig. 6, we see that the performance of RegISL is generally not sensitive to the choices of these three parameters. In other words, the involved parameters can be easily tuned to achieve satisfactory performance. Specifically, we observe that in most cases RegISL hits the highest accuracy on the four datasets when $\alpha = 1000$ and $\beta = 0.01$, therefore we use this parameter setting for all the experiments in Sections III-A–III-D. Besides, it can be seen that RegISL obtains the best performance on *Lost*, *MSRCv2*, *Soccer Player*, and *Bird Song* datasets when $K = 5, 10, 10, 10$, respectively, and this provides us the foundation for choosing the optimal K on the four datasets.

IV. CONCLUSION

In this paper, we propose a novel RegISL, which is dubbed as RegISL. Based on the graph \mathcal{G} , RegISL disambiguates the candidate labels of training examples by considering both the label smoothness between different examples, and the label discriminative property for every single example. As a consequence, the possible groundtruth labels in the candidate set become manifest while the values of false

positive candidate labels are suppressed. Thorough experimental results on various practical datasets suggest that in most cases the proposed RegISL achieves better training and test performances than the existing representative SLL methods.

Considering that the classification accuracy of our developed RegISL depends on the quality of constructed graph \mathcal{G} , in the future we plan to find a way to build a more accurate graph for conducting RegISL. Besides, due to the prevalence of label noise problem [45] today, it would be valuable to extend RegISL to the situation when the groundtruth labels of a small fraction of training examples are not included by their candidate label sets.

REFERENCES

- [1] L.-P. Liu and T. G. Dietterich, "A conditional multinomial mixture model for superset label learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 557–565.
- [2] T. Cour, B. Sapp, and B. Taskar, "Learning from partial labels," *J. Mach. Learn. Res.*, vol. 12, pp. 1501–1536, May 2011.
- [3] M.-L. Zhang and F. Yu, "Solving the partial label learning problem: An instance-based approach," in *Proc. Int. Joint Conf. Artif. Intell.*, Buenos Aires, Argentina, 2015, pp. 4048–4054.
- [4] F. Yu and M.-L. Zhang, "Maximum margin partial label learning," in *Proc. Asian Conf. Mach. Learn.*, Hong Kong, 2015, pp. 96–111.
- [5] E. Hüllermeier and J. Beringer, "Learning from ambiguously labeled examples," *Intell. Data Anal.*, vol. 10, no. 5, pp. 419–439, 2006.
- [6] Y.-C. Chen, V. M. Patel, R. Chellappa, and P. J. Phillips, "Ambiguously labeled learning using dictionaries," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2076–2088, Dec. 2014.
- [7] Y. Grandvalet, "Logistic regression for partial labels," in *Proc. Inf. Process. Manag. Uncertainty*, Annecy, France, 2002, pp. 1935–1941.
- [8] R. Jin and Z. Ghahramani, "Learning with multiple labels," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2002, pp. 897–904.
- [9] T. Cour, B. Sapp, C. Jordan, and B. Taskar, "Learning from ambiguously labeled images," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, 2009, pp. 919–926.
- [10] N. Nguyen and R. Caruana, "Classification with partial labels," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, Las Vegas, NV, USA, 2008, pp. 551–559.
- [11] E. Hüllermeier and W. Cheng, "Superset learning based on generalized loss minimization," in *Proc. Eur. Conf. Mach. Learn. Principles Pract. Knowl. Disc.*, Porto, Portugal, 2015, pp. 260–275.
- [12] L. Jie and F. Orabona, "Learning from candidate labeling sets," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2010, pp. 1504–1512.
- [13] J. Cid-Sueiro, "Proper losses for learning from partial labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1565–1573.
- [14] J. Cid-Sueiro, D. García-García, and R. Santos-Rodríguez, "Consistency of losses for learning from weak labels," in *Proc. Eur. Conf. Mach. Learn. Principles Pract. Knowl. Disc.*, Nancy, France, 2014, pp. 197–210.
- [15] L.-P. Liu and T. G. Dietterich, "Learnability of the superset label learning problem," in *Proc. Int. Conf. Mach. Learn.*, Beijing, China, 2014, pp. 1629–1637.
- [16] A. Shrivastava, J. K. Pillai, V. M. Patel, and R. Chellappa, "Learning discriminative dictionaries with partially labeled data," in *Proc. IEEE Int. Conf. Image Process.*, Orlando, FL, USA, 2012, pp. 3113–3116.
- [17] M.-L. Zhang, "Disambiguation-free partial label learning," in *Proc. SIAM Int. Conf. Data Min.*, Philadelphia, PA, USA, 2014, pp. 37–45.
- [18] Z. Zeng et al., "Learning by associating ambiguously labeled images," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, 2013, pp. 708–715.
- [19] C. Xu, T. Liu, D. Tao, and C. Xu, "Local rademacher complexity for multi-label learning," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1495–1507, Mar. 2016.
- [20] C. Xu, D. Tao, and C. Xu, "Multi-view learning with incomplete views," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5812–5825, Dec. 2015.

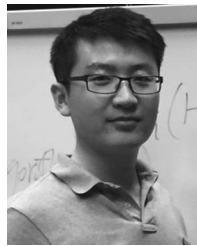
- [21] M.-L. Zhang, B.-B. Zhou, and X.-Y. Liu, "Partial label learning via feature-aware disambiguation," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, San Francisco, CA, USA, 2016, pp. 1335–1344.
- [22] C. Gong *et al.*, "Deformed graph Laplacian for semisupervised learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2261–2274, Oct. 2015.
- [23] C. Xu, D. Tao, and C. Xu, "Multi-view intact space learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 12, pp. 2531–2544, Dec. 2015.
- [24] C. Gong, D. Tao, W. Liu, L. Liu, and J. Yang, "Label propagation via teaching-to-learn and learning-to-teach," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2514360.
- [25] A. L. Yuille and A. Rangarajan, "The concave-convex procedure," *Neural Comput.*, vol. 15, no. 4, pp. 915–936, 2003.
- [26] Y. Xiao, H. Wang, and W. Xu, "Parameter selection of Gaussian kernel for one-class SVM," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 941–953, May 2015.
- [27] C. Gong *et al.*, "Multi-modal curriculum learning for semi-supervised image classification," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3249–3260, Jul. 2016.
- [28] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Proc. Int. Conf. Mach. Learn.*, vol. 3. Washington, DC, USA, 2003, pp. 912–919.
- [29] D. Wang, X. Gao, and X. Wang, "Semi-supervised nonnegative matrix factorization via constraint propagation," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 233–244, Jan. 2016.
- [30] C. Gong, D. Tao, K. Fu, and J. Yang, "Fick's law assisted propagation for semisupervised learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 2148–2162, Sep. 2015.
- [31] X. Pei, Z. Lyu, C. Chen, and C. Chen, "Manifold adaptive label propagation for face clustering," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1681–1691, Aug. 2015.
- [32] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2003, pp. 321–328.
- [33] J. Wang, F. Wang, C. Zhang, H. C. Shen, and L. Quan, "Linear neighborhood propagation and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 9, pp. 1600–1615, Sep. 2009.
- [34] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York, NY, USA: Academic press, 2014.
- [35] B. K. Sriperumbudur, D. A. Torres, and G. R. Lanckriet, "A majorization-minimization approach to the sparse generalized eigenvalue problem," *Mach. Learn.*, vol. 85, nos. 1–2, pp. 3–39, 2011.
- [36] G. R. Lanckriet and B. K. Sriperumbudur, "On the convergence of the concave-convex procedure," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2009, pp. 1759–1767.
- [37] D. Fernández and M. Solodov, "Local convergence of exact and inexact augmented Lagrangian methods under the second-order sufficient optimality condition," *SIAM J. Optim.*, vol. 22, no. 2, pp. 384–407, 2012.
- [38] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Stat. Assoc.*, vol. 32, no. 200, pp. 675–701, 1937.
- [39] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.
- [40] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, no. 9, pp. 1871–1874, 2008.
- [41] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: <http://cvxr.com/cvx>
- [42] S. Xiao, D. Xu, and J. Wu, "Automatic face naming by learning discriminative affinity matrices from weakly labeled images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2440–2452, Oct. 2015.
- [43] J. Sivic, M. Everingham, and A. Zisserman, "Who are you?"—Learning person specific classifiers from video," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, 2009, pp. 1145–1152.
- [44] F. Briggs, X. Z. Fern, and R. Raich, "Rank-loss support instance machines for MIML instance annotation," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, Beijing, China, 2012, pp. 534–542.
- [45] B. Frénay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 845–869, May 2014.



Chen Gong received the B.E. degree from the East China University of Science and Technology, Shanghai, China, in 2010, and the dual Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, and the University of Technology Sydney, Ultimo, NSW, Australia, in 2016, under the supervision of Prof. Jie Yang and Prof. Dacheng Tao, respectively.

He is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His current research interests include machine learning, data mining, and learning-based vision problems. He has published over 30 technical papers at prominent journals and conferences, such as the IEEE T-NNLS, IEEE T-IP, IEEE T-CYB CVPR, AAAI, and IJCAI.

Dr. Gong was a recipient of the National Scholarship by the Ministry of Education in 2013 and 2014, the Excellent Self-Financed Overseas Student Scholarship by China Scholarship Council in 2015, and the IBM Excellent Student Scholarship in 2015.



Tongliang Liu received the B.E. degree from the University of Science and Technology of China, Hefei, China, in 2012, and the Ph.D. degree in computer science from the University of Technology Sydney, Ultimo, NSW, Australia, in 2016.

His current research interests include statistical learning theory, computer vision, and optimization.

Dr. Liu was a recipient of the Best Paper Award in the IEEE International Conference on Information Science and Technology 2014.



Yuanyan Tang (F'04) received the B.S. degree in electrical and computer engineering from Chongqing University, Chongqing, China, the M.S. degree in electrical engineering from the Beijing University of Post and Telecommunications, Beijing, China, and the Ph.D. degree in computer science from Concordia University, Montreal, QC, Canada.

He is currently a Chair Professor with the Faculty of Science and Technology, University of Macau, Macau, China, and a Professor with Chongqing University, an Adjunct Professor with Concordia

University, and a Honorary Professor with Hong Kong Baptist University, Hong Kong. He has authored over 400 academic papers, and authored or co-authored over 25 monographs/books/book chapters. His current research interests include wavelets, pattern recognition, and image processing.

Dr. Tang is a fellow of the International Association for Pattern Recognition (IAPR). He is the Founder and the Chair of the Pattern Recognition Committee of the IEEE International Conference on Systems, Man, and Cybernetics. He has served as the General Chair, the Program Chair, and a Committee Member of many international conferences. He is the Founder and the General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition. He is the Founder and the Chair of the Macau Branch of IAPR. He is the Founder and an Editor-in-Chief of the *International Journal of Wavelets, Multiresolution, and Information Processing*, and an Associate Editor of several international journals.



Jian Yang received the B.S. degree in mathematics from Xuzhou Normal University, Xuzhou, China, in 1995, the M.S. degree in applied mathematics from Changsha Railway University, Changsha, China, in 1998, and the Ph.D. degree in pattern recognition and intelligence systems from the Nanjing University of Science and Technology (NUST), Nanjing, China, in 2002.

In 2003 to 2007, he was a Post-Doctoral Fellow with the University of Zaragoza, Zaragoza, Spain, Hong Kong Polytechnic University, Hong Kong, and the New Jersey Institute of Technology, Newark, NJ, USA. He is currently a Professor with the School of Computer Science and Technology, NUST. He has authored over 100 scientific papers in pattern recognition and computer vision. He has over 4000 ISI Web of Science and 8000 Google Scholar citations. His current research interests include pattern recognition, computer vision, and machine learning.

Dr. Yang is currently an Associate Editor of *Pattern Recognition Letters* and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.



Jie Yang received the Ph.D. degree from the Department of Computer Science, Hamburg University, Hamburg, Germany, in 1994.

He is currently a Professor with the Institute of Image Processing and Pattern recognition, Shanghai Jiao Tong University, Shanghai, China. He has led many research projects (such as National Science Foundation and 863 National High Tech. Plan), had one book published in Germany, and authored over 200 journal papers. His current research interests include object detection and recognition, data fusion

and data mining, and medical image processing.



Dacheng Tao (F'15) is a Professor of Computer Science with the School of Information Technologies, the Faculty of Engineering and Information Technologies, University of Sydney, Sydney, NSW, Australia. He was a Professor of Computer Science and the Director of the Centre for Artificial Intelligence, University of Technology Sydney, Ultimo, NSW, Australia. He mainly applies statistics and mathematics to Artificial Intelligence and Data Science. His current research interests include computer vision, data science, image

processing, machine learning, and video surveillance. His research results have expounded in one monograph and over 200 publications at prestigious journals and prominent conferences, such as the IEEE T-PAMI, IEEE T-NNLS, IEEE T-IP, JMLR, IJCV, NIPS, ICML, CVPR, ICCV, ECCV, AISTATS, ICDM, and ACM SIGKDD.

Mr. Tao was a recipient of the Best Theory/Algorithm Paper Runner Up Award in IEEE ICDM'07, the Best Student Paper Award in IEEE ICDM'13, the 2014 ICDM 10-year Highest-Impact Paper Award, the 2015 Australian Scopus-Eureka Prize, the 2015 ACS Gold Disruptor Award, and the 2015 UTS Vice-Chancellor's Medal for Exceptional Research. He is a fellow of the OSA, IAPR, and SPIE.