

# Structured Output-Associated Dictionary Learning for Haptic Understanding

Huaping Liu, Fuchun Sun, Di Guo, Bin Fang, and Zhengchun Peng

**Abstract**—Haptic sensing and feedback play extremely important roles for humans and robots to perceive, understand, and manipulate the world. Since many properties perceived by the haptic sensors can be characterized by adjectives, it is reasonable to develop a set of haptic adjectives for the haptic understanding. This formulates the haptic understanding as a multilabel classification problem. In this paper, we exploit the intrinsic relation between different adjective labels and develop a novel dictionary learning method which is improved by introducing the structured output association information. Such a method makes use of the label correlation information and is more suitable for the multilabel haptic understanding task. In addition, we develop two iterative algorithms to solve the dictionary learning and classifier design problems, respectively. Finally, we perform extensive experimental validations on the public available haptic sequence dataset Penn Haptic Adjective Corpus 2 and show the advantages of the proposed method.

**Index Terms**—Haptic understanding, intelligent robot perception, kernel dictionary learning, structured output prediction.

## I. INTRODUCTION

MANY physical properties of objects, such as *hard* or *soft*, are very difficult to visually ascertain, particularly without some kind of object manipulation [1]–[5]. For example, a cotton sheet and a white paper are difficult to be distinguished by their color, while their softness and elasticity work well. In practice, humans usually glean object properties through active manipulation of objects by hands, and therefore, haptic sensing and feedback play extremely important roles for humans to perceive, understand and manipulate the world [6]. Due to its importance, haptic understanding has now been extensively used for a wide variety of fields, such as manufacturing industry, robotics [7]–[9], and so on. See [10] for a detailed survey of the recent work.

Regarding haptic object recognition, a popular trend appears that advanced machine learning technology becomes the main

tool to distinguish objects with different physical properties. Schneider *et al.* [11] developed a naive Bayes method for bag-of-words feature which was obtained by grasping different parts of the rigid object. Such features were extended by introducing novel descriptors in [12]. The nearest neighborhood method, support vector machine [13], extreme learning machines [14], [15], Gaussian process [16], non-parametric Bayes learning method [17] had been developed for haptic material recognition. For the deformable object, Chitta *et al.* [18] developed a simple feature to identify the internal state of bottles. Drimus *et al.* [19] developed a new haptic sensor and contributed a haptic object recognition dataset. Soh and Demiris [20] utilized the spatio-temporal online recursive kernel Gaussian process to extract the haptic feature and developed an incremental recognition method. Navarro *et al.* [21] and Schmitz *et al.* [22] used the single-layer and deep learning method for object recognition. Recently, Madry *et al.* [23] developed a spatio-temporal unsupervised feature learning method for haptic object recognition and achieved very excellent results on tasks of grasp stability analysis and haptic object recognition. Spiers *et al.* [24] implemented object identification and feature extraction techniques on data acquired during a single, unplanned grasp with a simple, underactuated robot hand equipped with inexpensive barometric pressure sensors. In addition, one of our previous work utilized the sparse coding method to address the object recognition problem using multiple fingers [25].

On the other hand, the haptic classification is highly related to the material recognition problem. For example, classifying a surface as *foam* implies the presence of some haptic properties, such as *absorbent*, *bumpy*, *compressible*, and *soft*. However, many exceptions exist. For example, different bottle surfaces have vastly different hardness properties: a glass bottle is *hard* but an aluminum bottle is *soft*. Consequently, haptic understanding goes beyond simply identifying object materials and exhibits great challenges.

Since many properties perceived by the haptic sensors can be characterized by adjectives such as *hard*, *soft*, *smooth*, and so on, it is reasonable to develop a set of haptic adjectives for the haptic understanding. Griffith *et al.* [26] demonstrated that a rich and diverse haptic measurement system that measured temperature, compliance, roughness, and friction was key to accurately discerning between haptic adjectives such as *sticky* and *rough*. Chu *et al.* [27] detailed the collection of haptic classification datasets and concentrated on classifying objects with binary haptic adjectives. This paper relied on hand-crafted features for haptic classification. Recently, Gao *et al.* [28] proposed a deep learning method of classifying surfaces with haptic adjectives from both visual and physical interaction data. Orhan *et al.* [29] proposed a framework that conceptualized adjectives and nouns as separate categories that

Manuscript received June 3, 2016; revised September 7, 2016; accepted November 10, 2016. Date of publication January 25, 2017; date of current version June 22, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant U1613212, Grant 61673238, Grant 91420302, Grant 61327809, and Grant 61503212, and in part by the National High-Tech Research and Development Plan under Grant 2015AA042306. This paper was recommended by Associate Editor Z. Liu.

H. Liu, F. Sun, D. Guo, and B. Fang are with the Department of Computer Science and Technology, Tsinghua University, Beijing, China, and also with the State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, 100084 Beijing, China (e-mail: hpliu@tsinghua.edu.cn).

Z. Peng is with the Department of Optoelectronic Engineering, Shenzhen University, 518060 Shenzhen, China.

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/TSMC.2016.2635141

2168-2216 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See [http://www.ieee.org/publications\\_standards/publications/rights/index.html](http://www.ieee.org/publications_standards/publications/rights/index.html) for more information.



Fig. 1. Some representative examples which are adopted from PHAC-2 dataset [27]. The red text indicates the name of the object and the following black texts show the corresponding haptic adjectives. Please note that the images are just used for visualization illustration but not for algorithm development. From such examples we make two observations: 1) the adjective recognition is indeed a multilabel classification problem and 2) the *absorbent* usually co-occurs with *compressible*; *cool* usually co-occurs with *smooth* and *solid*. On the contrary, *hard* never co-occurs with *soft*. Such correlation will be further analyzed in this paper and we will exploit such relation to improve the haptic understanding performance.

were linked to and interact with each other. They demonstrated how those co-learned concepts might be useful for a cognitive robot.

As we have analyzed, objects usually exhibit multiple physical properties [30], [31]. Therefore, the haptic adjective classification can be formulated as a multilabel classification problem. See Fig. 1 for some examples. However, existing work such as [27] and [28] decomposed the problem into a set of independent binary classification problems. That is to say, each object is described by a set of multiple binary labels corresponding to the existence of absence of each of the haptic adjectives. Such a strategy totally neglected the interdependencies among multiple binary prediction tasks. For example, *absorbent* usually co-occurs with *compressible*; *cool* usually co-occurs with *smooth* and *solid*. On the contrary, *hard* never co-occurs with *soft*. Such relations cannot be exploited by the above simple strategy.

Different from the conventional multiclass problems where each sample should be mapped to a single class label, multilabel classification needs to map each sample to typically a few interdependent class labels in a relatively large output space. The goal of multilabel classification is therefore to discover the underlying label correlation structure to improve the classification performance. However, existing work on multilabel classification is mainly concentrated in the applications of image, video, and text. In [32], an image is typically associated with multiple labels, and its visual representation reflects the combination of the involved labels. It is observed that each label corresponds to certain local patch in the image. Therefore, the entire image representation can be decomposed into a set of local label representations corresponding to the labels associated with an image. However, the haptic signal does not exhibit such properties. That is to say, it is difficult, if impossible, to segment the haptic signal according to different labels.

In this paper, we exploit the intrinsic relation among different adjective labels and develop a novel dictionary learning method which is improved by introducing the structured output

association information. Such a method makes use of the label correlation information and is more suitable for the multilabel haptic understanding task. The main contributions are listed as follows.

- 1) We establish a novel structured output-associated dictionary learning (SO-DL) framework for multilabel haptic adjective classification.
- 2) We develop two iterative algorithms to solve the dictionary learning and classifier design problems, respectively.
- 3) We perform extensive experimental validations on the public available haptic sequence dataset and show the advantages of the proposed method.

The rest of this paper is organized as follows. In Section II, we review some related works and point out the uniqueness of this paper. Section III presents the problem formulation and the objective functions of the proposed SO-DL method. In Section IV, we introduce the optimization algorithm. The classifier design is presented in Section V and the experimental results are given in Section VI.

*Notations:* We use the capital  $X$  to represent the matrix and lowercase  $x$  to represent the vector. For a vector  $x$ , we use  $x(i)$  to denote its  $i$ th element. We will use various vector and matrix norms, here are the notations we use:  $\|x\|_1$  and  $\|x\|_2$  represent the 1-norm and 2-norm of the vector  $x$ , respectively.  $\|X\|_F$  is the Frobenius norm, and  $\|X\|_{1,1}$  is calculated by the absolute values of all elements in  $X$ .

## II. RELATED WORK

Since the core contribution of this paper is a new dictionary learning method for haptic understanding, we give a discussion to illustrate the difference between existing related work and ours.

As we have mentioned above, the haptic adjective understanding problem is intrinsically a multilabel classification problem. There exist some representative work which used sparse coding and dictionary learning method to solve multilabel classification problem. For example, Wang *et al.* [33] developed a multilabel sparse coding framework for feature extraction and classification within the context of automatic image annotation. Cai *et al.* [34] proposed a structured sparsity-inducing norm regularization to incorporate the relational graph information into multilabel classification model and imposed the correlated classes to share the common space, such that the input data relevant to both classes would be learned jointly. Jing *et al.* [35] developed a multilabel dictionary learning with label consistency regularization and partial-identical label embedding, which conducted multilabel dictionary learning and partial-identical label embedding simultaneously. However, to the best of our knowledge, the idea about SO-DL has never been addressed. Furthermore, we notice that there is extensive research on structured sparse coding and dictionary learning [36], [37]. However, such work focused on imposing structured information on the coding vectors or dictionary atoms, but did not consider the structured information in the output vectors. In this sense, the model proposed in this paper is novel.

The idea using sparse coding for haptic object recognition was reported in our previous work [25], [38]. However, there exist significant differences between them and this paper.

- 1) Both [25] and [38] regarded all of the training samples as the dictionary and did not investigate the dictionary

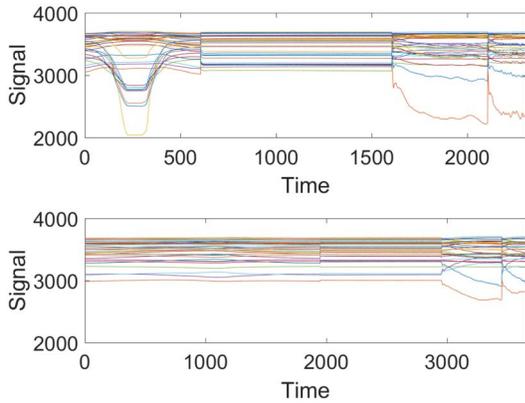


Fig. 2. Two representative haptic sequences which are adopted from PHAC-2 dataset [27]. It is obvious that they exhibit different characteristics and lengths.

learning problem. This limits the application in more practical scenarios. In this paper, we focus on the dictionary learning problem for haptic understanding.

- 2) Both [25] and [38] addressed the single-label object recognition problem, while this paper investigates the more practical multilabel haptic adjective classification problem. To tackle such a challenging problem, we propose a novel dictionary learning model which incorporates the structured output association information.

To summarize, the work in this paper differs with our previous work significantly in aspects of the theoretic model and classification applications.

In addition to the sparse coding and dictionary learning, we notice that deep learning technology has become an effective strategy to tackle haptic signals [28]. The most great advantages of deep learning is that the feature can be automatically learned. However, this requires more haptic training data which is difficult to collect in practice.

### III. PROBLEM FORMULATION

For conveniens, we denote the measurable space of interested haptic signals by  $\mathcal{S}$ . Some representative examples are shown in Fig. 2. Given a set of  $C$  adjective labels and a set of  $N$  training sequences  $\{\mathcal{S}_i\}_{i=1}^N \subset \mathcal{S}$ . The element of the label vector  $\mathbf{y}_i \in \mathbb{R}^C$  corresponding to the  $i$ th sample is defined as

$$\mathbf{y}_i(c) = \begin{cases} +1 & \text{If label } c \text{ is associated with sample } \mathcal{S}_i \\ -1 & \text{otherwise} \end{cases} \quad (1)$$

for  $c = 1, 2, \dots, C$ . The goal in multilabel haptic adjective classification is to label an unseen haptic sample  $\mathcal{S} \in \mathcal{S}$  with the subset of relevant adjective labels from the prespecified adjective set. A haptic sample can be labeled with any of the  $2^C$  possible subsets. The main challenge, therefore, lies in optimizing over this exponentially large label space subject to label correlations.

Since the haptic sequences do not lie in the Euclidean space [25], [39], we cannot adopt the conventional sparse coding method which requires the linear reconstruction assumption [40]. A popular method to deal with this problem is to adopt the kernel trick on the dictionary learning. By using a suitably designed kernel, the linear sparse coding can be extended to the nonlinear case [41], [42]. To this end, we denote the reproducing Kernel Hilbert space of functions on  $\mathcal{S}$  as  $\mathcal{H}_{\mathcal{S}}$ , whose dimension is denoted as  $D$ ,

which may be infinite. Any positive-definite kernel function  $\kappa(\cdot, \cdot)$  corresponds to an inner product in a latent Hilbert space and induces a latent feature map which is denoted by  $\varphi(\cdot) : \mathcal{S} \rightarrow \mathcal{H}_{\mathcal{S}}$  from the original input space to this feature space. Please note that  $\varphi(\cdot)$  is not required to be explicitly represented and we only concern the concrete form about the kernel function  $\kappa(\mathcal{S}_i, \mathcal{S}_j) = \varphi^T(\mathcal{S}_i)\varphi(\mathcal{S}_j)$ , which will be detailed in experiment section.

The task of haptic dictionary learning aims to obtain a compact dictionary  $\mathcal{D} = [\mathbf{d}_1, \dots, \mathbf{d}_K] \in \mathbb{R}^{D \times K}$ , where  $K < N$  is the size of the dictionary, and  $\mathbf{d}_k \in \mathcal{H}_{\mathcal{S}}$  for  $k = 1, 2, \dots, K$ . Such a dictionary should effectively summarize the whole training set  $\{\mathcal{S}_i\}_{i=1}^N$  and provides a good basis for sparse coding in the implicit feature space.

By denoting  $\varphi(\mathcal{S}) = [\varphi(\mathcal{S}_1), \dots, \varphi(\mathcal{S}_N)] \in \mathbb{R}^{D \times N}$ , we formulate the following kernel dictionary learning problem:

$$\begin{aligned} \min_{\mathcal{D}, \mathbf{X}} \quad & \|\varphi(\mathcal{S}) - \mathcal{D}\mathbf{X}\|_F^2 + \alpha\|\mathbf{X}\|_{1,1} \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2 = 1 \text{ for } k = 1, 2, \dots, K \end{aligned} \quad (2)$$

where  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{K \times N}$  is the sparse coding matrix and  $\alpha$  is used to control the sparsity.

However, the optimization problem in (2) cannot be directly solved since the mapping  $\varphi(\cdot)$  is not explicitly represented. Fortunately, the kernel trick provides an effective method to avoid dealing with the mapping  $\varphi(\cdot)$  [43], [44]. However, the dictionary learning problem is still nontrivial since we have to deal with the dictionary atoms  $\mathbf{d}_k$  which lies in the implicit feature space. For general cases of kernels, Harandi and Salzmann [44] and Van Nguyen *et al.* [41] utilized the *Representer Theorem*, which indicates that the dictionary  $\mathcal{D}$  can be represented by

$$\mathcal{D} = \varphi(\mathcal{S})\mathbf{A} \quad (3)$$

where  $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_K] \in \mathbb{R}^{N \times K}$  is the reconstruction matrix. This means that the dictionary atoms can be linearly reconstructed by the training samples in the feature space. Furthermore, the constraint condition on the dictionary atoms  $\|\mathbf{d}_k\|_2 = 1$  is in fact  $\|\varphi(\mathcal{S})\mathbf{a}_k\|_2 = 1$ , which can be equivalently transformed as  $\mathbf{a}_k^T \mathbf{K}_{\mathcal{S}\mathcal{S}} \mathbf{a}_k = 1$ , where  $\mathbf{K}_{\mathcal{S}\mathcal{S}} \in \mathbb{R}^{N \times N}$  is the kernel matrix over all of the training samples. The  $(i, j)$ th element of  $\mathbf{K}_{\mathcal{S}\mathcal{S}}$  is defined as  $\kappa(\mathcal{S}_i, \mathcal{S}_j) = \varphi^T(\mathcal{S}_i)\varphi(\mathcal{S}_j)$ . Therefore, we can reformulate the kernel dictionary learning problem as

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{X}} \quad & \|\varphi(\mathcal{S}) - \varphi(\mathcal{S})\mathbf{A}\mathbf{X}\|_F^2 + \alpha\|\mathbf{X}\|_{1,1} \\ \text{s.t.} \quad & \mathbf{a}_k^T \mathbf{K}_{\mathcal{S}\mathcal{S}} \mathbf{a}_k = 1. \end{aligned} \quad (4)$$

By this formulation we observe that the original data matrix  $\varphi(\mathcal{S}) \in \mathbb{R}^{D \times N}$  is compressed as a reduced dictionary  $\mathcal{D} = \varphi(\mathcal{S})\mathbf{A} \in \mathbb{R}^{D \times K}$  and each sample  $\mathcal{S}_i \in \mathcal{S}$  is represented as  $K$ -dimensional sparse vector  $\mathbf{x}_i \in \mathbb{R}^K$ . The optimization model in (4) provides a great advantage since it does not search the dictionary atoms in the feature space, but only calculates the coefficient matrix  $\mathbf{A}$ . Therefore, this formulation can be used for any type of kernel functions.

After getting the sparse code  $\mathbf{x}_i$  for the original sample  $\mathcal{S}_i$ , we can use conventional classifier for classification. However, such a method exists an obvious disadvantage that the dictionary learning stage and the classifier design stage are independent. This limits the performance of the dictionary. A

better solution is to jointly learn the dictionary and the classifier, i.e., introducing the discriminative capability into the dictionary learning.

One straightforward approach for multilabel classification is to decompose the multilabel learning problem into a set of independent binary classification problems. According to this idea, we define a new label vector  $\mathbf{z}_c \in \mathbb{R}^N$  for each label  $c = 1, 2, \dots, C$ , as

$$\mathbf{z}_c(i) = \begin{cases} +1 & y_i(c) = +1 \\ -1 & y_i(c) = -1 \end{cases} \quad (5)$$

and formulate the following  $C$  independent supervised dictionary learning problem:

$$\begin{aligned} \min_{\mathbf{A}_c, \mathbf{X}_c, \mathbf{w}_c} & \|\varphi(\mathbb{S}) - \varphi(\mathbb{S})\mathbf{A}_c\mathbf{X}_c\|_F^2 + \alpha\|\mathbf{X}_c\|_{1,1} \\ & + \beta\|\mathbf{z}_c^T - \mathbf{w}_c^T\mathbf{X}_c\|_2^2 + \gamma\|\mathbf{w}_c\|_2^2 \\ \text{s.t.} & \mathbf{a}_{c,k}^T\mathbf{K}_{SS}\mathbf{a}_{c,k} = 1 \end{aligned} \quad (6)$$

for  $c = 1, 2, \dots, C$ . In the above equation,  $\mathbf{A}_c \in \mathbb{R}^{N \times K}$  and  $\mathbf{X}_c \in \mathbb{R}^{K \times N}$  are the dictionary coefficient matrix and coding matrix for the  $c$ th task, respectively, and  $\mathbf{w}_c \in \mathbb{R}^K$  is the classifier coefficient vector. The parameters  $\beta$  and  $\gamma$  are used to control the importance of the corresponding regularization terms.

However, the above approach considers each adjective as an independent class task and the multilabel correlations are neglected. It is well known that the task correlations are helpful for the prediction. Therefore we can require the shared common dictionary but design different classifiers for each adjective classification task. This leads to the following optimization problem:

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{X}, \mathbf{w}_c} & \|\varphi(\mathbb{S}) - \varphi(\mathbb{S})\mathbf{A}\mathbf{X}\|_F^2 + \alpha\|\mathbf{X}\|_{1,1} \\ & + \beta\sum_{c=1}^C\|\mathbf{z}_c^T - \mathbf{w}_c^T\mathbf{X}\|_2^2 + \gamma\sum_{c=1}^C\|\mathbf{w}_c\|_2^2 \\ \text{s.t.} & \mathbf{a}_k^T\mathbf{K}_{SS}\mathbf{a}_k = 1 \end{aligned} \quad (7)$$

for  $c = 1, 2, \dots, C$  and  $\mathbf{w}_c \in \mathbb{R}^K$  is the classifier vector.

Nevertheless, the above formulation does not explicitly incorporate the output association information. In fact, the output components can be considered as auxiliary features and used to complement the more standard input features. We assume that the output  $y_i(c)$  is related to all the other outputs  $y_{i,-c}$ , which is defined as

$$\mathbf{y}_{i,-c} = [y_i(1), \dots, y_i(c-1), y_i(c+1), \dots, y_i(C)]^T$$

and we may minimize the discriminative error with the linear form

$$\mathbf{y}_i(c) - \mathbf{u}_c^T\mathbf{x}_i - \mathbf{v}_c^T\mathbf{y}_{i,-c} \quad (8)$$

where  $\mathbf{u}_c \in \mathbb{R}^K$  and  $\mathbf{v}_c \in \mathbb{R}^{C-1}$  are the classifier coefficient vectors for the  $c$ th classification task.

By incorporating the above error term into the original dictionary learning objective function (4), we formulate the proposed SO-DL problem as

$$\begin{aligned} \min_{\mathbf{A}, \mathbf{X}, \mathbf{U}, \mathbf{V}} & \|\varphi(\mathbb{S}) - \varphi(\mathbb{S})\mathbf{A}\mathbf{X}\|_F^2 + \alpha\|\mathbf{X}\|_{1,1} \\ & + \beta\sum_{c=1}^C\sum_{i=1}^N(\mathbf{y}_i(c) - \mathbf{u}_c^T\mathbf{x}_i - \mathbf{v}_c^T\mathbf{y}_{i,-c})^2 \\ & + \gamma(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \\ \text{s.t.} & \mathbf{a}_k^T\mathbf{K}_{SS}\mathbf{a}_k = 1 \end{aligned} \quad (9)$$

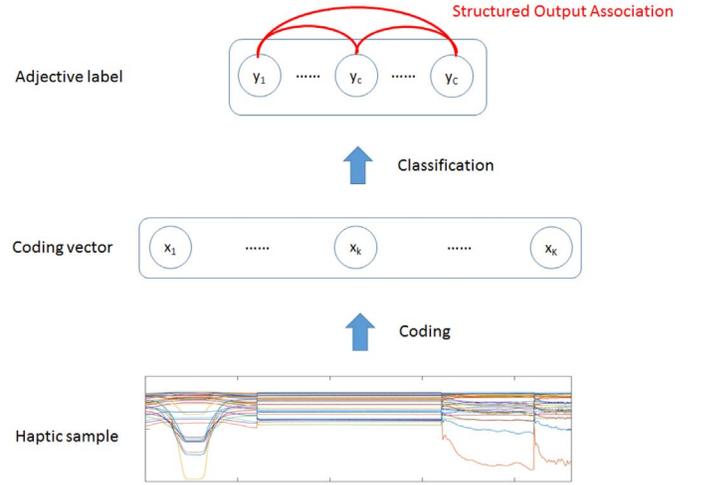


Fig. 3. Coding and labeling principle illustration. The haptic sample is coded as a  $K$ -dimensional vector and a classifier could be developed using such a coding vector. The red lines indicate that the structured output association information can be exploited to improve the multilabel classification performance.

where  $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_C] \in \mathbb{R}^{K \times C}$  and  $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_C] \in \mathbb{R}^{(C-1) \times C}$  are classifier parameter matrices which represent the relationship between inputs and outputs.

In Fig. 3, we give a schematic plot to show the whole procedure of the joint coding and structured output labeling.

*Remark 1:* In (8), we only consider the linear dependence relation between the inputs and outputs. To better characterize the complicated relation between them, we can resort to the kernel function to design the nonlinear discriminative regularization term

$$\mathbf{y}_i(c) - \mathbf{u}_c^T\phi(\mathbf{x}_i) - \mathbf{v}_c^T\psi(\mathbf{y}_{i,-c}) \quad (10)$$

or even

$$\mathbf{y}_i(c) - \mathbf{u}_c^T\phi(\mathbf{x}_i, \mathbf{y}_{i,-c}) \quad (11)$$

where  $\phi(\cdot)$  and  $\psi(\cdot)$  are some nonlinear mapping functions which can be explicitly designed by the user or be implicitly represented by the suitably defined kernel functions. Adopting this strategy helps us to obtain better representation of the input-output relation but introduces more tuning parameters and complicates the solving procedure. In this paper, we find the simple linear dependence relation in (8) can work well in our multilabel haptic understanding tasks.

#### IV. OPTIMIZATION ALGORITHM

The optimization problem in (9) is obviously nonconvex and nonsmooth. We adopt the alternative optimization method to solve it. The algorithm can be divided into the following stages. For conveniences, we use the superscript  $t$  to represent the solutions at the  $t$ th iteration.

##### A. Calculating the Sparse Coding Vectors

This step updates the coding vectors  $\mathbf{X}^{(t+1)}$ , given the values of  $\mathbf{A}^{(t)}$ ,  $\mathbf{U}^{(t)}$ , and  $\mathbf{V}^{(t)}$ . Please note that each column of  $\mathbf{X}$  can

be calculated separately, and therefore, the problem reduces to

$$\min_{\mathbf{x}_i} \left\| \varphi(\mathcal{S}_i) - \varphi(\mathcal{S})\mathbf{A}^{(t)}\mathbf{x}_i \right\|_2^2 + \alpha \|\mathbf{x}_i\|_1 + \beta \sum_{c=1}^C \left( \hat{y}_{ci}^{(t)} - \mathbf{u}_c^{(t)T}\mathbf{x}_i \right)^2 \quad (12)$$

where  $\hat{y}_{ci}^{(t)} = \mathbf{y}_i(c) - \mathbf{v}_c^{(t)T}\mathbf{y}_{i,-c}$ . By denoting  $\hat{\mathbf{y}}_i^{(t)} = [\hat{y}_{1i}^{(t)}, \hat{y}_{2i}^{(t)}, \dots, \hat{y}_{Ci}^{(t)}]^T$ , we can rewrite the third term in the above equation as  $\beta \|\hat{\mathbf{y}}_i^{(t)} - \mathbf{U}^{(t)}\mathbf{x}_i\|_2^2$ .

As we have  $\|\varphi(\mathcal{S}_i) - \varphi(\mathcal{S})\mathbf{A}^{(t)}\mathbf{x}_i\|_2^2 = \varphi^T(\mathcal{S}_i)\varphi(\mathcal{S}_i) - 2\mathbf{k}_{S_i}^T\mathbf{A}^{(t)}\mathbf{x}_i + \mathbf{x}_i^T\mathbf{A}^{(t)T}\mathbf{K}_{SS}\mathbf{A}^{(t)}\mathbf{x}_i$  and  $\|\hat{\mathbf{y}}_i^{(t)} - \mathbf{U}^{(t)}\mathbf{x}_i\|_2^2 = \hat{\mathbf{y}}_i^{(t)T}\hat{\mathbf{y}}_i^{(t)} - 2\hat{\mathbf{y}}_i^{(t)T}\mathbf{U}^{(t)}\mathbf{x}_i + \mathbf{x}_i^T\mathbf{U}^{(t)T}\mathbf{U}^{(t)}\mathbf{x}_i$ , we can expand (12) as

$$\min_{\mathbf{x}_i} -2\left(\mathbf{k}_{S_i}^T\mathbf{A}^{(t)} + \beta\hat{\mathbf{y}}_i^{(t)T}\mathbf{U}^{(t)}\right)\mathbf{x}_i + \mathbf{x}_i^T\left(\mathbf{A}^{(t)T}\mathbf{K}_{SS}\mathbf{A}^{(t)} + \beta\mathbf{U}^{(t)T}\mathbf{U}^{(t)}\right)\mathbf{x}_i + \alpha\|\mathbf{x}_i\|_1 \quad (13)$$

where  $\mathbf{k}_{S_i} = \varphi^T(\mathcal{S}_i)\varphi(\mathcal{S})$  is the  $i$ th column of  $\mathbf{K}_{SS}$ .

This problem can be easily solved using any efficient  $l_1$  optimization algorithm such as feature-sign search [43] or alternating direction method of multipliers [45]. Also, we can employ popular sparse solver software such as SPAMS [46] to solve this problem.

Using the above procedure for  $i = 1, 2, \dots, N$ , we can get the updated values of  $\bar{\mathbf{X}}^{(t+1)} = [\bar{\mathbf{x}}_1^{(t+1)}, \dots, \bar{\mathbf{x}}_N^{(t+1)}]$ .

### B. Calculating the Dictionary Atoms

This step update the dictionary coefficient matrix. At the  $(t+1)$ th iteration, we are given  $\bar{\mathbf{X}}^{(t+1)}$ ,  $\mathbf{U}^{(t)}$  and  $\mathbf{V}^{(t)}$ . The optimization problem reduces to

$$\min_{\mathbf{A}} \left\| \varphi(\mathcal{S}) - \varphi(\mathcal{S})\mathbf{A}\bar{\mathbf{X}}^{(t+1)} \right\|_F^2 \quad \text{s.t. } \mathbf{a}_k^T\mathbf{K}_{SS}\mathbf{a}_k = 1. \quad (14)$$

By defining  $\bar{\mathbf{A}} = \bar{\mathbf{X}}^{(t+1)T}(\bar{\mathbf{X}}^{(t+1)}\bar{\mathbf{X}}^{(t+1)T})^{-1}$  and

$$\bar{\mathbf{a}}_k^{(t+1)} = \frac{\bar{\mathbf{a}}_k}{\sqrt{\bar{\mathbf{a}}_k^T\mathbf{K}_{SS}\bar{\mathbf{a}}_k}}$$

where  $\bar{\mathbf{a}}_k$  is the  $k$ th column of  $\bar{\mathbf{A}}$ , we can get the updated values of  $\mathbf{A}^{(t+1)} = [\mathbf{a}_1^{(t+1)}, \dots, \mathbf{a}_K^{(t+1)}]$ .

### C. Calculating the Classifier Parameters

This step updates the classifiers parameters  $\mathbf{U}$  and  $\mathbf{V}$ . At the  $(t+1)$ th iteration, we are given  $\bar{\mathbf{X}}^{(t+1)}$  and  $\mathbf{A}^{(t+1)}$ . Please note that each column of  $\mathbf{U}$  and  $\mathbf{V}$  can be calculated separately, and therefore, the problem reduces to

$$\min_{\mathbf{u}_c, \mathbf{v}_c} \beta \sum_{i=1}^N \left( \mathbf{y}_i(c) - \mathbf{u}_c^T\mathbf{x}_i^{(t+1)} - \mathbf{v}_c^T\mathbf{y}_{i,-c} \right)^2 + \gamma \left( \|\mathbf{u}_c\|_2^2 + \|\mathbf{v}_c\|_2^2 \right). \quad (15)$$

Denoting  $\bar{\mathbf{u}}_c = [\mathbf{u}_c^T \ \mathbf{v}_c^T]^T$  and  $\bar{\mathbf{x}}_i^{(t+1)} = [\mathbf{x}_i^{(t+1)T} \ \mathbf{y}_{i,-c}^T]^T$ , we can get

$$\min_{\bar{\mathbf{u}}_c} \beta \sum_{i=1}^N \left( \mathbf{y}_i(c) - \bar{\mathbf{u}}_c^T\bar{\mathbf{x}}_i^{(t+1)} \right)^2 + \gamma \|\bar{\mathbf{u}}_c\|_2^2 \quad (16)$$

which admits the following solution:

$$\bar{\mathbf{u}}_c = \left\{ \beta \bar{\mathbf{X}}^{(t+1)}\bar{\mathbf{X}}^{(t+1)T} + \gamma\mathbf{I}_{K+C-1} \right\}^{-1} \left( \beta \bar{\mathbf{X}}^{(t+1)}(\mathbf{y}^c)^T \right)$$

### Algorithm 1 SO-DL

**Input:** Data set  $\{\mathcal{S}_i\}$  for  $i = 1, 2, \dots, N$ , the size of dictionary  $K$ , the parameters  $\alpha, \beta, \gamma$   
**Output:** Solutions  $\mathbf{X} \in \mathbb{R}^{K \times N}$ ,  $\mathbf{A} \in \mathbb{R}^{N \times K}$ ,  $\mathbf{U} \in \mathbb{R}^{K \times C}$  and  $\mathbf{V} \in \mathbb{R}^{(C-1) \times C}$ .

- 1: **while** Not convergent **do**
- 2: Fix  $\mathbf{A}$ ,  $\mathbf{U}$ ,  $\mathbf{V}$  and update  $\mathbf{X}$  according to Section IV.A.
- 3: Fix  $\mathbf{X}$ ,  $\mathbf{U}$ ,  $\mathbf{V}$  and update  $\mathbf{A}$  according to Section IV.B.
- 4: Fix  $\mathbf{A}$ ,  $\mathbf{X}$  and update  $\mathbf{U}$ ,  $\mathbf{V}$  according to Section IV.C.
- 5: **end while**

where  $\bar{\mathbf{X}}^{(t+1)} = [\bar{\mathbf{x}}_1^{(t+1)}, \bar{\mathbf{x}}_2^{(t+1)}, \dots, \bar{\mathbf{x}}_N^{(t+1)}]$ ;  $\mathbf{y}^c = [\mathbf{y}_1(c), \dots, \mathbf{y}_N(c)]$ ; and  $\mathbf{I}_{K+C-1}$  is the  $(K+C-1) \times (K+C-1)$  identity matrix.

Furthermore, we can easily extract the first  $K$  rows of  $\bar{\mathbf{u}}_c$  as  $\mathbf{u}_c^{(t+1)}$  and other  $C-1$  rows of  $\bar{\mathbf{u}}_c$  as  $\mathbf{v}_c^{(t+1)}$ .

Using the above procedure for  $c = 1, 2, \dots, C$ , we can get the updated values of  $\mathbf{U}^{(t+1)} = [\mathbf{u}_1^{(t+1)}, \dots, \mathbf{u}_C^{(t+1)}]$  and  $\mathbf{V}^{(t+1)} = [\mathbf{v}_1^{(t+1)}, \dots, \mathbf{v}_C^{(t+1)}]$ .

### D. Algorithm Summarization

With the above updating rules, the proposed algorithm is summarized in Algorithm 1. The convergent condition can be triggered when the change of the objective function is smaller than a prescribed tolerant error, or the prescribed maximum iteration number is achieved. In this paper, we adopt the latter strategy and the maximum iteration number is set to 30. The initial value of  $\mathbf{A}$  is set as  $[\mathbf{I}_K, \mathbf{O}_{(N-K) \times K}]^T$ . This implies that we use the first  $K$  samples in the training sample set to construct the initial dictionary.

## V. CLASSIFIER DESIGN

The above learning procedure provides us the solutions which are denoted as  $\mathbf{A}^*$ ,  $\mathbf{U}^*$ , and  $\mathbf{V}^*$ . Then we should use them to design the classifier. Differently from previous supervised dictionary learning method which did not consider the output association, we should carefully design an algorithm for the classifier to preserve the output-associative information.

For a test sample  $\mathcal{S} \in \mathcal{S}$ , we denote its label vector as  $\mathbf{l} \in \{-1, +1\}^C$  which should be determined. We use  $\mathbf{y} \in \mathbb{R}^C$  as the relaxed label vector and solve the following joint coding and labeling problem:

$$\min_{\mathbf{x}, \mathbf{y}} \left\| \varphi(\mathcal{S}) - \varphi(\mathcal{S})\mathbf{A}^*\mathbf{x} \right\|_F^2 + \alpha \|\mathbf{x}\|_1 + \beta \sum_{c=1}^C \left( \hat{y}_c - \mathbf{u}_c^{*T}\mathbf{x} \right)^2 \quad (17)$$

where  $\hat{y}_c = \mathbf{y}(c) - \mathbf{v}_c^{*T}\mathbf{y}_{-c}$ .

This problem is also nonconvex, and therefore, we resort the alternative optimization method. The iterations are divided into the following stages. For convenience, we also use the superscript  $t$  to indicate the iteration number.

- 1) Update  $\mathbf{x}$  as

$$\mathbf{x}^{(t+1)} = \operatorname{argmin}_{\mathbf{x}} \left\| \varphi(\mathcal{S}) - \varphi(\mathcal{S})\mathbf{A}^*\mathbf{x} \right\|_F^2$$

$$+ \alpha \|\mathbf{x}\|_1 + \beta \sum_{c=1}^C \left( \hat{y}_c^{(t)} - \mathbf{u}_c^{*T}\mathbf{x} \right)^2 \quad (18)$$

where  $\hat{y}_c^{(t)} = \mathbf{y}^{(t)}(c) - \mathbf{v}_c^{*T} \mathbf{y}_{-c}^{(t)}$ . This problem is similar to the coding problem in Section IV-A and can be solved efficiently.

2) Update  $\mathbf{y}$  as

$$\mathbf{y}^{(t+1)} = \underset{\mathbf{y}}{\operatorname{argmin}} \sum_{c=1}^C \left( \hat{y}_c - \mathbf{u}_c^{*T} \mathbf{x}^{(t+1)} \right)^2. \quad (19)$$

Note  $\hat{y}_c$  can be represented as  $\bar{\mathbf{v}}_c^{*T} \mathbf{y}$ , where  $\bar{\mathbf{v}}_c^* = [\mathbf{v}_c^*(1), \dots, \mathbf{v}_c^*(c-1), 1, \mathbf{v}_c^*(c+1), \dots, \mathbf{v}_c^*(C)]^T$ . Therefore the objective function in (19) can be represented as

$$\left\| \bar{\mathbf{V}}^{*T} \mathbf{y} - \bar{\mathbf{U}}^{*T} \mathbf{x}^{(t+1)} \right\|_2^2$$

where  $\bar{\mathbf{V}}^* = [\bar{\mathbf{v}}_1^*, \dots, \bar{\mathbf{v}}_C^*]$  and  $\bar{\mathbf{U}}^* = [\bar{\mathbf{u}}_1^*, \dots, \bar{\mathbf{u}}_C^*]$ . The solution can thus be obtained as

$$\mathbf{y}^{(t+1)} = \left( \bar{\mathbf{V}}^* \bar{\mathbf{V}}^{*T} \right)^{-1} \bar{\mathbf{V}}^* \bar{\mathbf{U}}^{*T} \mathbf{x}^{(t+1)}.$$

In practice, we follow the suggestions given by [47] to initialize the above optimization problem using prediction given by classifier trained on independent outputs. This strategy significantly improves the convergence speed compared to random initialization. In our experiment, after at most four iterations we can get the converged solution  $\mathbf{y}^*$ . Finally, we set the refined final label vector  $\mathbf{l}$  according to

$$\mathbf{l}(c) = \operatorname{sign}(\mathbf{y}^*(c))$$

for  $c = 1, 2, \dots, C$ .

## VI. EXPERIMENTAL RESULTS

### A. Data Description and Experimental Setting

We demonstrate our approach on the Penn Haptic Adjective Corpus 2 (PHAC-2) dataset, which was originally developed in [27]. The PHAC-2 dataset contains haptic signals of 60 household objects. Each object is explored by a pair of SynTouch biomimetic haptic sensors (BioTacs), which were mounted to the gripper of a Willow Garage Personal Robot2. Each object was felt with four exploratory procedures. The BioTac sensor generates five types of signals: 1) low-frequency fluid pressure; 2) high-frequency fluid vibrations; 3) core temperature; 4) core temperature change; and 5) 19 electrode impedance which are spatially distributed across the sensor. Although the joint positions and gripper velocity and acceleration are available, we concentrate on classifying the haptic signals using the electrode impedances. Therefore, we concatenate the electrode impedances data from the two BioTac sensors to get  $d = 19 \times 2 = 38$ -D haptic sequence signals (see Fig. 2 for some representative haptic sequences). For each object, ten trials of each exploratory procedures were performed and 600 haptic sample sequences were obtained.

Each object is described with a set of 24 binary labels, corresponding to the existence or absence of each of the 24 haptic adjectives (e.g., *hard* or *soft*). The link information is shown in Fig. 4 and the calculated correlation information is detailed in Fig. 5, which shows that there indeed exists obvious correlation between some adjectives, such as *solid-hard*, *compressible-squishy*, and *soft-compressible*. For the three adjectives *nice*, *sticky*, and *unpleasant*, which exhibit only one positive object instance, it is impossible to construct

the positive training and testing sets. Therefore, we simply delete such three ones. As a result, there are 21 possible adjective labels and the average number of adjective labels for each sample is 4.2. In Fig. 6, we show the label distribution of haptic adjectives. It shows that *smooth*, *solid*, and *squishy* are the most three popular adjectives.

The performance evaluation of multilabel learning is very different from that of single-label classification and we use the popular Hamming loss to evaluate the classification score [48]. For the  $i$ th testing sample, we use  $\mathcal{G}_i$  as the set of ground-truth labels and  $\mathcal{R}_i$  as the adjective label set of the algorithm output. Then the score is calculated as

$$\operatorname{score}(i) = 1 - \frac{1}{C} |\mathcal{G}_i \ominus \mathcal{R}_i|$$

where  $\ominus$  stands for the symmetric difference between two sets and  $|\cdot|$  calculates the number of elements in the set. The symmetric difference is the set of elements which are in either of the sets and not in their intersection. For example, if  $\mathcal{G}_i = \{\textit{smooth}, \textit{rough}, \textit{cool}, \textit{squishy}\}$  and  $\mathcal{R}_i = \{\textit{smooth}, \textit{solid}, \textit{squishy}\}$ , then we have  $\mathcal{G}_i \ominus \mathcal{R}_i = \{\textit{rough}, \textit{cool}, \textit{solid}\}$ . If  $\mathcal{G}_i = \mathcal{R}_i$ , then we have  $\mathcal{G}_i \ominus \mathcal{R}_i = \emptyset$  and therefore the score is 1. Otherwise, only when one of  $\mathcal{G}_i$  and  $\mathcal{R}_i$  is a full set and the other one is an empty set, we can obtain the zero score. The overall performance is evaluated by averaged score over all of the testing samples.

### B. Haptic Sequence Representation

At each time step, we collect the haptic data from the  $d$  sensor cells. Since we want to utilize the dynamic features which seek to model the way the entire haptic signal changes over time, we represent the dynamic haptic sequence  $\mathcal{S}_i \in \mathcal{S}$  as

$$\mathcal{S}_i = [\mathbf{s}_{i,1}, \mathbf{s}_{i,2}, \dots, \mathbf{s}_{i,t_i}] \quad (20)$$

where  $\mathbf{s}_{i,t} \in \mathbb{R}^d$  for  $t = 1, 2, \dots, t_i$ , and  $t_i$  is the number of the sampled time instants for this sequence. The lengths are different from sequence to sequence.

A popular comparison measurement of time sequences is dynamic time warping (DTW) distance, which was also used for haptic sequence comparison [19], [25]. However, it is well known that DTW is not a true metric and it is difficult to construct the positive definite kernel from DTW distance [49]. In some previous work such as [25] and [49], some extra modifications were always required to guarantee the positive-definiteness of the DTW kernel matrix. Such a processing introduced unexpected effects for the final performance. To avoid this problem, we resort the global alignment (GA) kernel which was proposed in [50]. The details are introduced as follows.

Consider two haptic sequences  $\mathcal{S}_i = (\mathbf{s}_{i,1}, \dots, \mathbf{s}_{i,t_i})$  and  $\mathcal{S}_j = (\mathbf{s}_{j,1}, \dots, \mathbf{s}_{j,t_j})$  of lengths  $t_i$  and  $t_j$ , respectively. We denote an alignment  $\pi$  to be a pair of increasing integral vectors  $(\pi_i, \pi_j)$  of length  $|\pi|$  such that  $1 = \pi_i(1) \leq \dots \leq \pi_i(|\pi|) = t_i$  and  $1 = \pi_j(1) \leq \dots \leq \pi_j(|\pi|) = t_j$ , with unitary increments and no simultaneous repetitions. Since the two haptic sequences have  $t_i + t_j$  points and they are matched at least at one point of time, we have  $|\pi| \leq t_i + t_j - 1$ .

We define alignment cost of  $\mathbf{s}_i$  and  $\mathbf{s}_j$  under the alignment strategy  $\pi$  as  $C_{ij} = \sum_{t=1}^{|\pi|} \psi(\mathbf{s}_{i,\pi_i(t)}, \mathbf{s}_{j,\pi_j(t)})$ , where  $\psi(\cdot, \cdot)$  is used to denote the local divergence that measures the discrepancy between the two vectors. According to the suggestions in [50], we use the following local divergence

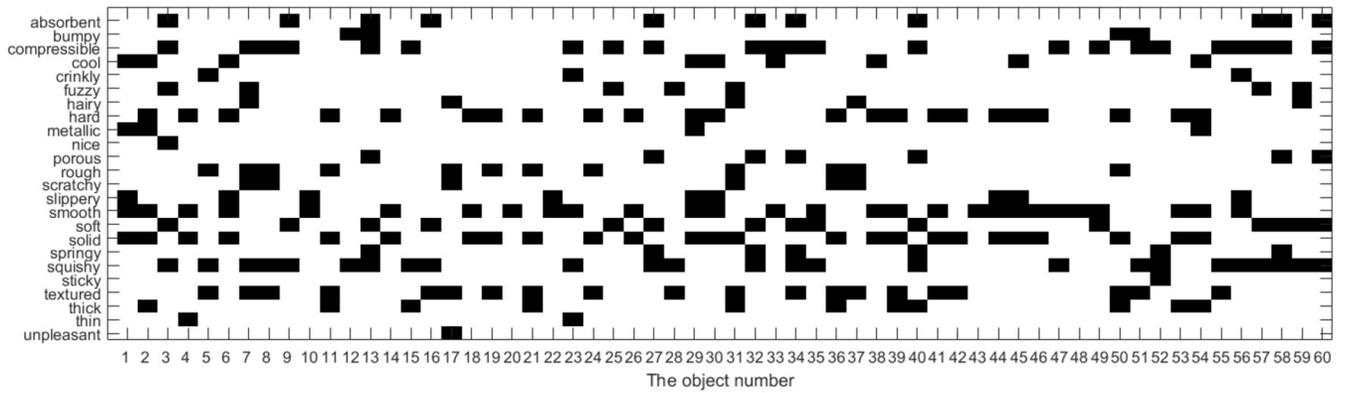


Fig. 4. Link information between the adjective and the object number. The occupied box shows that the corresponding object has the exhibited adjective property.

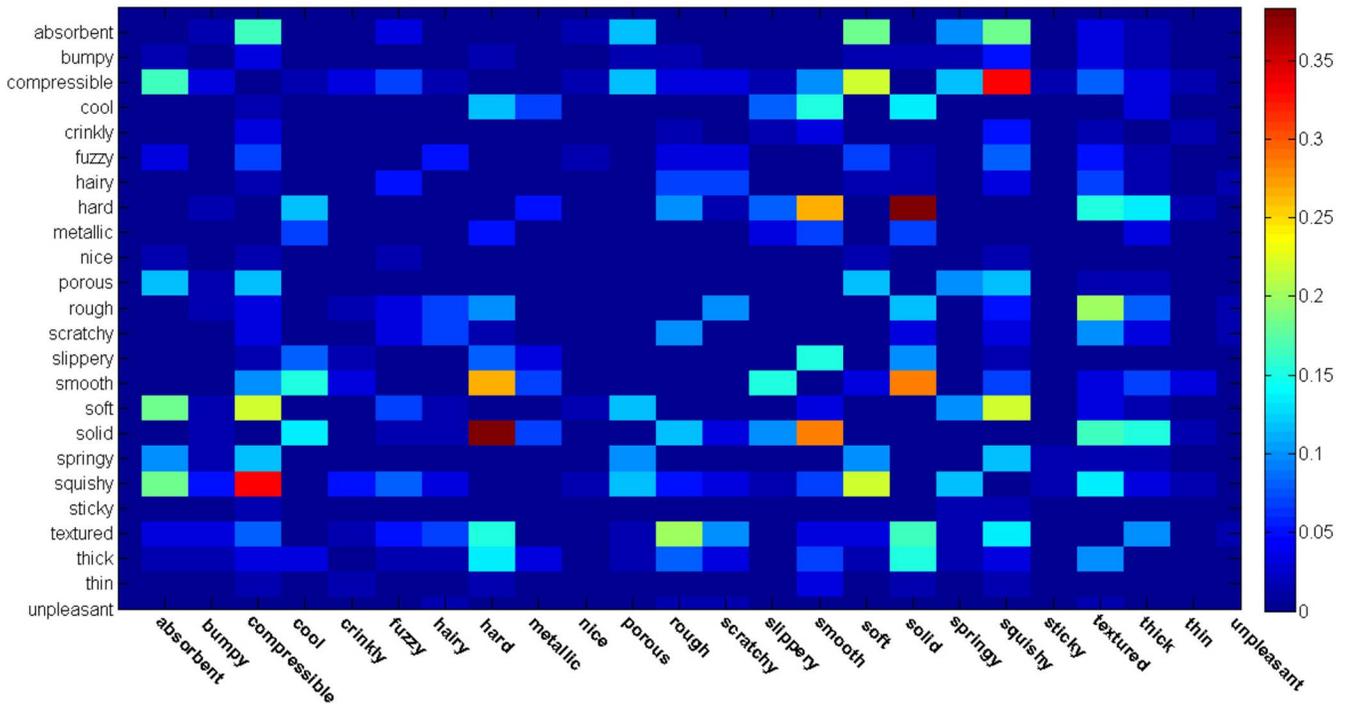


Fig. 5. Correlation of multilabel adjectives. From this figure we clearly find that *solid* is highly related to *hard*; *compressible* is rather related to *squishy* and *soft*. All of them are consistent to our intuitive knowledge.

$\psi(\mathbf{u}, \mathbf{v}) = (1/2\sigma^2)\|\mathbf{u} - \mathbf{v}\|^2 + \log(2 - e^{-[(\|\mathbf{u} - \mathbf{v}\|^2)/2\sigma^2]})$ , where  $\sigma$  is a parameter which is empirically set to 10.

The GA kernel assumes that the minimum value of alignments may be sensitive to peculiarities of the time sequences and intends to take advantage of all alignments weighted exponentially. It is defined as the sum of exponentiated and sign changed costs of the individual alignments

$$\kappa(S_i, S_j) = \sum_{\pi \in \Pi(t_i, t_j)} e^{-C_{ij}(\pi)} \quad (21)$$

where  $\Pi(t_i, t_j)$  denotes the set of all alignments between two time sequences of length  $t_i$  and  $t_j$ . It has been argued that  $\kappa$  runs over the whole spectrum of the costs and gives rise to a smoother measure than the minimum of the costs, i.e., the DTW distance. It has also been shown that this kernel is positive definite.

### C. Performance Comparison

To show the advantages of the proposed method, we make a fair comparison with the following classifiers.

- 1) *Separate k-Nearest Neighbor Method*: It decomposes the original problem into 21 separate binary classification problems (one per adjective) and uses the conventional  $k$ -nearest neighbor ( $k$ -NN) method with the GA kernel-induced distance to determine whether the adjective emerges or not. This method does not consider the correlations of labels and serves as a baseline. In our experiment we set  $k = 1$  because we find increasing the value of  $k$  may deteriorate the performance.
- 2) *Multilabel k-NN Method [48]*: For each test sample, it first identifies its  $k$ -NNs using the GA kernel-induced distance in the training set. After that, based on statistical information gained from the label sets of these neighboring instances, i.e., the number of neighboring

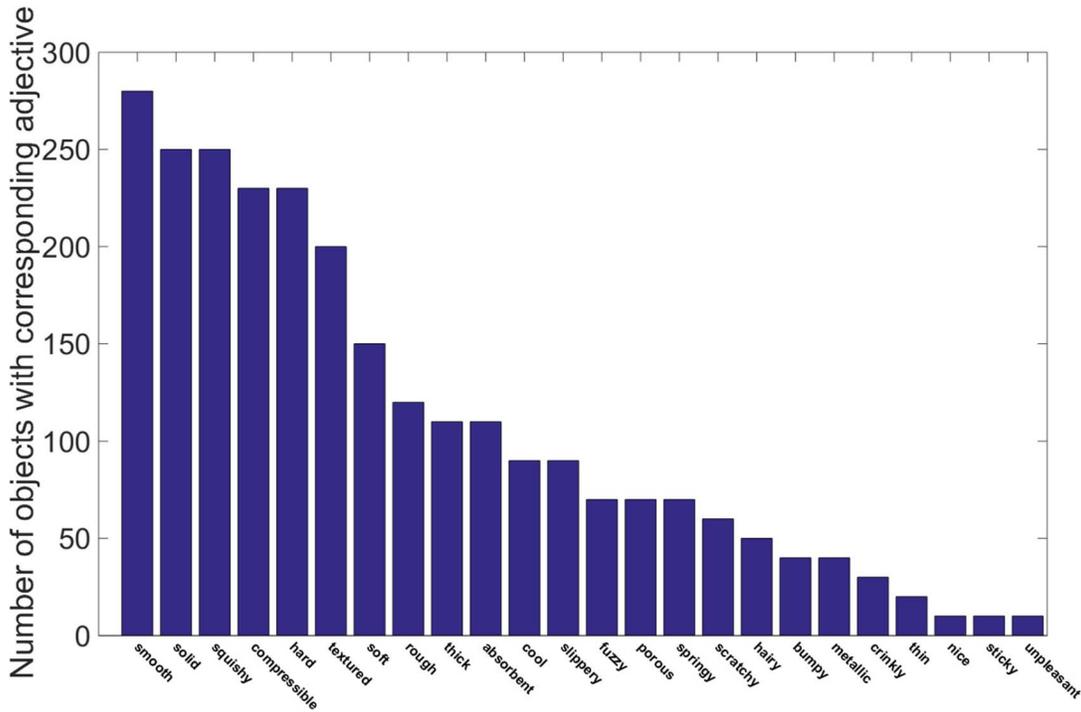


Fig. 6. Label distribution of the PHAC-2 dataset. The adjectives *nice*, *sticky* and *unpleasant* are not used in the experimental validation.

TABLE I  
METHOD COMPARISON

Method	S-kNN	ML-kNN	S-DL	C-DL	SO-DL
Training	×	×	✓	✓	✓
Task correlation	×	×	×	✓	✓
Output correlation	×	✓	×	×	✓

instances belonging to each possible class, maximum *a posteriori* principle is utilized to determine the label set for the unseen instance. In this paper, we set  $k = 5$  because the experimental results show that larger  $k$  does not result in better performance.

- 3) *Separate Dictionary Learning Method*: This method solves 21 separate optimization problems in (6) to obtain the dictionaries and classifiers. Obviously, the label correlation is not considered in this method.
- 4) *Common Dictionary Learning Method*: This method solves the optimization problem in (7) which learns a common dictionary and separate classifier for each adjective. Please note this method uses a shared dictionary to construct the connections of all the classification tasks but still does not consider the label correlation.
- 5) *Structured Output-Associated Dictionary Learning Method*: This method is developed in this paper and solves the optimization problem in (9) and explicitly incorporates the effect of label correlations.

In Table I we give the information to indicate whether some method requires training stage, whether it considers the task correlation and whether it exploits the label correlation. The task correlation of common dictionary learning (C-DL) and SO-DL is exploited by utilizing the common shared dictionary, and the label correlation of SO-DL is exploited by the introduced output association information.

For separate dictionary learning (S-DL), C-DL, and SO-DL, we set the dictionary size  $K$  as the 60% of the number of

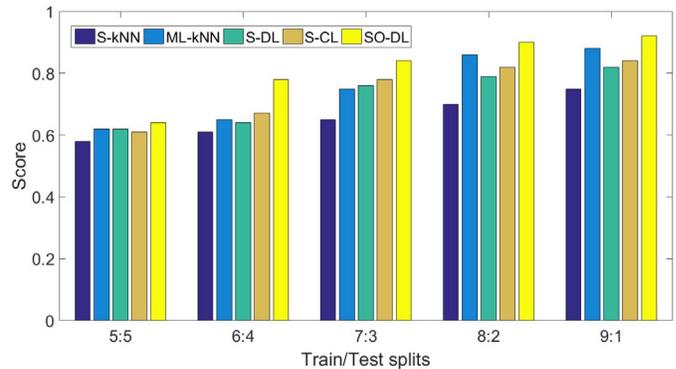


Fig. 7. Performance score for the five train/test splits.

training samples and extensively tune other parameters to get the best results. In Section VI-D, we will show the parameters' effect on the performance.

We partition the 60 objects into five split cases: training/testing = 5/5, 6/4, 7/3, 8/2, and 9/1. For each split case, the objects are randomly split as training set and testing set for ten trials, and the averaged recognition scores are reported. Similar to [27], we do not allow the same object to appear in both the train and test split. This permits us to investigate the capability of the algorithm to deal with new or unseen objects. Fig. 7 shows the performance of the five cases. The proposed SO-DL method consistently outperforms the other methods. The multilabel  $k$ -NN (ML- $k$ -NN) method, which also incorporates the label correlation information, performs better than separate  $k$ -NN (S- $k$ -NN) and is competitive to S-DL and C-DL, but worse than SO-DL. From those results, we make the following observations.

- 1) S- $k$ -NN performs worse than all of the other methods. The reason is obvious: without considering the

TABLE II  
REPRESENTATIVE RESULTS

Method	white foam	toilet paper	glass bottle	aluminum block
Ground-Truth	{absorbent, porous, compressible, soft, springy, squishy}	{absorbent, compressible, fuzzy soft, squishy}	{cool, hard, metallic slippery, smooth, solid}	{cool, metallic, slippery smooth, solid}
S-kNN	{compressible, fuzzy soft, scratchy, squishy}	{absorbent, compressible, soft, textured}	{hard, smooth, solid}	{metallic, smooth solid, thick}
ML-kNN	{absorbent, compressible, soft, solid, springy, squishy}	{absorbent, compressible, soft, squishy}	{hard, metallic slippery, smooth, solid}	{cool, hard, metallic, smooth, solid}
S-DL	{compressible, fuzzy, soft, springy, squishy, textured}	{absorbent, compressible, fuzzy soft, squishy, textured}	{hard, slippery, smooth, solid}	{metallic, slippery smooth, solid}
C-DL	{absorbent, compressible, springy, squishy, thick}	{absorbent, compressible, fuzzy soft}	{hard, slippery smooth, solid, squishy}	{metallic, slippery smooth, solid}
SO-DL	{absorbent, porous, compressible, soft, springy, squishy}	{absorbent, compressible, soft, squishy}	{cool, hard, metallic slippery, smooth, solid}	{cool, hard, metallic, slippery smooth, solid}

correlations of multiple adjectives, it is very difficult for such a method to get satisfactory performance over all of the adjectives. Similarly, S-DL performs better than S-k-NN by a very small margin.

- 2) ML-k-NN, which explicitly incorporates the correlation information of adjectives, performs better than S-k-NN and S-DL. On the other hand, we find C-DL, which just uses the shared common dictionary but does not incorporate the output correlation, obtains similar results with ML-k-NN in most cases. The possible reason is that the sparse coding strategy introduces more discriminative information. Please note ML-k-NN is just a lazy classifier and does not exploit the discriminative information. From such a comparison we can observe the power of dictionary learning and sparse coding.
- 3) The proposed SO-DL method performs better than all of the other methods for all cases. The gained improvement originates from two aspects: a) the discriminative capability introduced by the dictionary learning and sparse coding and b) the correlation information of the adjective labels.
- 4) For the case of 5/5 split, the performance gained by SO-DL is small, while for other cases, the improvement is significant. This is partially due to the fact that the correlation information can be exploited from more training samples.

In Table II we list some representative results using different methods. In the following, we give some explanations.

For the sample of *white foam*. The adjective *porous* is difficult to be perceived. However, since *porous* exhibits some correlations with *absorbent*, *compressible*, *soft*, and *squishy*, our method successful finds it. ML-k-NN, though adopts the multilabel correlation information, fails to find it. The possible reason is that the correlation of *porous* with other adjectives are not strong enough for ML-k-NN to work. This example also shows that the proposed method exploits more information than ML-k-NN.

For the sample of *toilet paper*, S-k-NN and S-DL falsely find *textured*. Such mistakes do not occur with ML-k-NN and SO-DL since the label correlation is fully exploited. However, the adjective *fuzzy*, though can be found by S-DL and S-CL, cannot be found by ML-k-NN and SO-DL. The reason is that *fuzzy* exhibits weak correlation with other adjectives and therefore the correlation information in ML-k-NN and SO-DL plays very little role.

Both *glass bottle* and *aluminum block* exhibit the property *cool*, which cannot found by S-k-NN, S-DL, and S-CL. Using

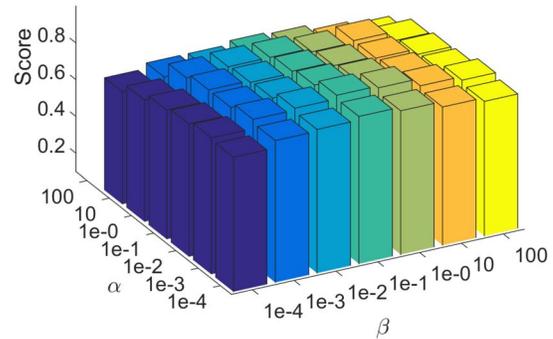


Fig. 8. Sensitivity of  $\alpha$  and  $\beta$ .

our method, though we do not use the temperature information, we can still accurately find *cool*. The main reason is that *cool* is related to the adjectives *hard*, *solid*, and *smooth*, and our method can successful exploit such relations to get correct results.

It is embarrassed to admit that the label correlation does not always play positive role. One example is that both ML-k-NN and SO-DL falsely find *hard* with *aluminium block*. The reason is that *hard* is closely related with *smooth* and *solid*, therefore the algorithm would like to assign *hard* adjective to the *aluminium block*, which exhibits the properties of *smooth* and *solid*. It is a pity that *hard* is not included in the ground-truth label set because the annotators feel it is not hard enough.

#### D. Parameter Sensitivity Analysis

In our model (9), there are several regularization parameters  $\alpha$ ,  $\beta$ , and  $\gamma$ . All of them are of physical meanings, and therefore, it is not difficult to tune them for better performance. To analyze the effects of  $\alpha$  and  $\beta$ , we set  $\gamma = 10^{-3}$  and vary the values of  $\alpha$  and  $\beta$  from  $10^{-4}$  to  $10^2$ . The results are shown in Fig. 8. Those results show that the proposed algorithm works well when the parameter  $\alpha$  is in the interval  $[10^{-4}, 10^{-3}]$ . When  $\beta$  is too large, the reconstruction error term is attenuated and the obtained coding vector cannot reflect the characteristics of the original haptic samples. On the contrary, when  $\beta$  is too small, the role of discriminative classifier becomes weak. Therefore, a properly designed classifier term indeed plays an important role in haptic dictionary learning.

Finally, we fix  $\alpha = 0.001$  and  $\beta = 0.01$  and vary the ratio  $K/N$  for the five split cases, where  $N$  is the number of the training samples, and record the scores in Fig. 9, which shows

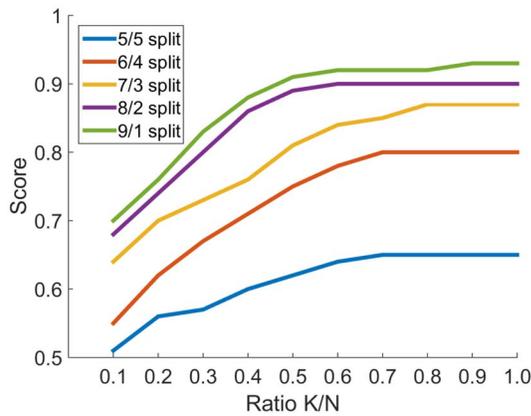


Fig. 9. Sensitivity of  $K$  for five train/test splits.

that increasing the value of  $K$  improves the performance, but the curves almost reach the plateau when the ratio is larger than 0.6. This means that the performance is not sensible to the dictionary size, partially due to the merits of the introduced discriminative learning term.

## VII. CONCLUSION

Developing a set of haptic adjectives provides a bridge to understand haptic sense since many properties perceived by the haptic sensors can be characterized by adjectives. However, the adjectives perceived by the haptic sensors exhibit strong and complicated correlations, which provide us challenge and opportunities for cognitive understanding of haptic information. In this paper, we formulate the haptic understanding as a multilabel classification problem and exploit the intrinsic relation between different adjective labels by developing a novel dictionary learning method which is improved by introducing the structured output association information. Such a method makes use of the label correlation information and is more suitable for the multilabel haptic understanding task. To solve this problem, we develop two iterative algorithms for dictionary learning and classifier design, respectively. Finally, we analyze the adjective correlation on the public available haptic sequence dataset PHAC-2 and perform extensive experimental validations to show the advantages of the proposed method.

By this paper we make the following findings.

- 1) Although haptic sensing is complicated and subtle, it can be well characterized by many adjectives. In practice, haptic adjectives serve as powerful tools to represent the semantic haptic information.
- 2) By a detailed analysis on the adjectives in the extensive PHAC-2 dataset, we confirm that there exists a lot of correlations among the haptic adjectives. This is also consistent with the human's intuition.
- 3) Exploiting the correlation of haptic adjectives indeed improves the haptic understanding performance. A representative example is that we can find the *cool* property without the temperature information. Such a synesthesia can be obtained by correlations of haptic adjectives.

This paper provides an effective strategy for cognitive haptic understanding. In addition, although our focus is on haptic understanding, the described problem framework is common in the automation community. The algorithm described in this paper can therefore work with other multilabel classification problems. In practical scenarios, the actuator saturation

often occurs and produces extra effects on the haptic signals [51]–[53]. This provides great challenges on the haptic understanding and remains our future work.

## REFERENCES

- [1] Y. Hu *et al.*, "Development of sensory-motor fusion-based manipulation and grasping control for a robotic hand-eye system," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1169–1180, Jul. 2017.
- [2] C. Chen, Z. Liu, Y. Zhang, and S. Xie, "Coordinated motion/force control of multiarm robot with unknown sensor nonlinearity and manipulated object's uncertainty," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1123–1134, Jul. 2017.
- [3] G. Lai, Z. Liu, Y. Zhang, and C. L. P. Chen, "Adaptive fuzzy tracking control of nonlinear systems with asymmetric actuator backlash based on a new smooth inverse," *IEEE Trans. Cybern.*, vol. 46, no. 6, pp. 1250–1262, Jun. 2016.
- [4] G. Lin, Z. Li, L. Liu, H. Su, and W. Ye, "Development of multi-fingered dexterous hand for grasping manipulation," *Sci. China Inf. Sci.*, vol. 57, no. 12, 2014, Art. no. 120208.
- [5] X. Hu *et al.*, "High precision intelligent flexible grasping front-end with CMOS interface for robots application," *Sci. China Inf. Sci.*, vol. 59, no. 3, 2016, Art. no. 032203.
- [6] C. Liu, H. Qiao, J. Su, and P. Zhang, "Vision-based 3-D grasping of 3-D objects with a simple 2-D gripper," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 5, pp. 605–620, May 2014.
- [7] D. Wang, X. Zhang, Y. Zhang, and J. Xiao, "Configuration-based optimization for six degree-of-freedom haptic rendering for fine manipulation," *IEEE Trans. Haptics*, vol. 6, no. 2, pp. 167–180, Apr./Jun. 2013.
- [8] X. Hou and R. Mahony, "Dynamic kinesthetic boundary for haptic teleoperation of VTOL aerial robots in complex environments," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 694–705, May 2016.
- [9] C. Chen, Z. Liu, Y. Zhang, C. L. P. Chen, and S. Xie, "Saturated Nussbaum function based approach for robotic systems with unknown actuator dynamics," *IEEE Trans. Cybern.*, vol. 46, no. 10, pp. 2311–2322, Oct. 2016.
- [10] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Tactile sensing in dexterous robot hands—Review," *Robot. Auton. Syst.*, vol. 74, no. 1, pp. 195–220, 2015.
- [11] A. Schneider *et al.*, "Object identification with tactile sensors using bag-of-features," in *Proc. IEEE/RISJ Int. Conf. Intell. Robots Syst. (IROS)*, St. Louis, MO, USA, 2009, pp. 243–248.
- [12] S. Luo, W. Mou, K. Althoefer, and H. Liu, "Novel tactile-SIFT descriptor for object shape recognition," *IEEE Sensors J.*, vol. 15, no. 9, pp. 5001–5009, Sep. 2015.
- [13] S. Decherchi, P. Gastaldo, R. S. Dahiya, M. Valle, and R. Zunino, "Tactile-data classification of contact materials using computational intelligence," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 635–639, Jun. 2011.
- [14] Y. Yang and Q. M. J. Wu, "Multilayer extreme learning machine with subnetwork nodes for representation learning," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2570–2583, Nov. 2016.
- [15] J. Cao *et al.*, "Landmark recognition with sparse representation classification and extreme learning machine," *J. Frankl. Inst.*, vol. 352, no. 10, pp. 4528–4545, 2015.
- [16] M. Strese *et al.*, "A haptic texture database for tool-mediated texture recognition and classification," in *Proc. IEEE Int. Symp. Haptic Audio Vis. Environ. Games (HAVE)*, Richardson, TX, USA, 2014, pp. 118–123.
- [17] P. Dallaire, P. Giguère, D. Émond, and B. Chaib-draa, "Autonomous tactile perception: A combined improved sensing and Bayesian non-parametric approach," *Robot. Auton. Syst.*, vol. 62, no. 4, pp. 422–435, 2014.
- [18] S. Chitta, J. Sturm, M. Piccoli, and W. Burgard, "Tactile sensing for mobile manipulation," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 558–568, Jun. 2011.
- [19] A. Drimus, G. Kootstra, A. Bilberg, and D. Kragic, "Design of a flexible tactile sensor for classification of rigid and deformable objects," *Robot. Auton. Syst.*, vol. 62, no. 1, pp. 3–15, 2014.
- [20] H. Soh and Y. Demiris, "Incrementally learning objects by touch: Online discriminative and generative models for tactile-based recognition," *IEEE Trans. Haptics*, vol. 7, no. 4, pp. 512–525, Oct./Dec. 2014.
- [21] S. E. Navarro *et al.*, "Haptic object recognition for multi-fingered robot hands," in *Proc. IEEE Haptics Symp. (HAPTICS)*, Vancouver, BC, Canada, 2012, pp. 497–502.
- [22] A. Schmitz *et al.*, "Tactile object recognition using deep learning and dropout," in *Proc. IEEE RAS Int. Conf. Humanoid Robots (Humanoids)*, Madrid, Spain, 2014, pp. 1044–1050.

- [23] M. Madry, L. Bo, D. Kragic, and D. Fox, "ST-HMP: Unsupervised spatio-temporal feature learning for tactile data," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Hong Kong, 2014, pp. 2262–2269.
- [24] A. J. Spiers, M. V. Liarokapis, B. Calli, and A. M. Dollar, "Single-grasp object classification and feature extraction with simple robot hands and tactile sensors," *IEEE Trans. Haptics*, vol. 9, no. 2, pp. 207–220, Apr./Jun. 2016.
- [25] H. Liu, D. Guo, and F. Sun, "Object recognition using tactile measurements: Kernel sparse coding methods," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 3, pp. 656–665, Mar. 2016.
- [26] S. Griffith, J. Sinapov, V. Sukhoy, and A. Stoytchev, "A behavior-grounded approach to forming object categories: Separating containers from noncontainers," *IEEE Trans. Auton. Mental Develop.*, vol. 4, no. 1, pp. 54–69, Mar. 2012.
- [27] V. Chu *et al.*, "Robotic learning of haptic adjectives through physical interaction," *Robot. Auton. Syst.*, vol. 63, no. 3, pp. 279–292, 2015.
- [28] Y. Gao, L. A. Hendricks, K. J. Kuchenbecker, and T. Darrell, "Deep learning for tactile understanding from visual and haptic data," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Stockholm, Sweden, 2016, pp. 536–543.
- [29] G. Orhan, S. Olgunsoylu, E. Sahin, and S. Kalkan, "Co-learning nouns and adjectives," in *Proc. IEEE Int. Conf. Develop. Learn. Epigenet. Robot.*, Osaka, Japan, 2013, pp. 1–6.
- [30] S. Ḃak, M. S. Biagio, R. Kumar, V. Murino, and F. Bṙemond, "Exploiting feature correlations by Brownian statistics for people detection and recognition," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.
- [31] J. Cao, W. Wang, J. Wang, and R. Wang, "Excavation equipment recognition based on novel acoustic statistical features," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2016.2609999.
- [32] T. Li, T. Mei, S. Yan, I.-S. Kweon, and C. Lee, "Contextual decomposition of multi-label images," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, 2009, pp. 2270–2277.
- [33] C. Wang, S. Yan, L. Zhang, and H.-J. Zhang, "Multi-label sparse coding for automatic image annotation," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, 2009, pp. 1643–1650.
- [34] X. Cai, F. Nie, W. Cai, and H. Huang, "New graph structured sparsity model for multi-label image annotations," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Sydney, NSW, Australia, 2013, pp. 801–808.
- [35] X.-Y. Jing, F. Wu, Z. Li, R. Hu, and D. Zhang, "Multi-label dictionary learning for image annotation," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2712–2725, Jun. 2016.
- [36] H. Cheng, Z. Liu, L. Yang, and X. Chen, "Sparse representation and learning in visual recognition: Theory and applications," *Signal Process.*, vol. 93, no. 6, pp. 1408–1425, 2013.
- [37] Y. Suo, M. Dao, U. Srinivas, V. Monga, and T. D. Tran, "Structured dictionary learning for classification," unpublished paper, 2014. [Online]. Available: <https://arxiv.org/abs/1406.1943>
- [38] H. Liu, Y. Yu, F. Sun, and J. Gu, "Visual-tactile fusion for object recognition," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 996–1008, Apr. 2017.
- [39] T.-H. S. Li, M.-C. Kao, and P.-H. Kuo, "Recognition system for home-service-related sign language using entropy-based K-means algorithm and ABC-based HMM," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 1, pp. 150–162, Jan. 2016.
- [40] J. Tang, C. Deng, and G.-B. Huang, "Extreme learning machine for multilayer perceptron," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 4, pp. 809–821, Apr. 2016.
- [41] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Design of non-linear kernel dictionaries for object recognition," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 5123–5135, Dec. 2013.
- [42] B. Chen, S. Zhao, P. Zhu, and J. C. Principe, "Quantized kernel recursive least squares algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 9, pp. 1484–1491, Sep. 2013.
- [43] S. Gao, I. W. Tsang, and L.-T. Chia, "Sparse representation with kernels," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 423–434, Feb. 2013.
- [44] M. Harandi and M. Salzmann, "Riemannian coding and dictionary learning: Kernels to the rescue," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 3926–3935.
- [45] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
- [46] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Jan. 2010.
- [47] L. Bo and C. Sminchisescu, "Structured output-associative regression," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, 2009, pp. 2403–2410.
- [48] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, 2007.
- [49] Z. Chen, W. Zuo, Q. Hu, and L. Lin, "Kernel sparse representation for time series classification," *Inf. Sci.*, vol. 292, pp. 15–26, Jan. 2015.
- [50] M. Cuturi, "Fast global alignment kernels," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 929–936.
- [51] C. Chen, Z. Liu, Y. Zhang, and C. L. P. Chen, "Modeling and adaptive compensation of unknown multiple frequency vibrations for the stabilization and control of an active isolation system," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 900–911, May 2016.
- [52] C. Chen *et al.*, "Adaptive fuzzy asymptotic control of MIMO systems with unknown input coefficients via a robust Nussbaum gain based approach," *IEEE Trans. Fuzzy Syst.*, to be published, doi: 10.1109/TFUZZ.2016.2604848.
- [53] G. Lai, Z. Liu, Y. Zhang, C. L. P. Chen, and S. Xie, "Asymmetric actuator backlash compensation in quantized adaptive control of uncertain networked nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 294–307, Feb. 2017.

**Huaping Liu** received the Ph.D. degree from Tsinghua University, Beijing, China, in 2004.

He currently is an Associate Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interest includes robot perception and learning.

Dr. Liu serves as an Associate Editor of several journals including the IEEE ROBOTICS AND AUTOMATION LETTERS, *Neurocomputing*, *Cognitive Computation*, and some conferences including the International Conference on Robotics and Automation and the International Conference on Intelligent Robots and Systems. He also served as a Program Committee Member of RSS2016 and IJCAI2016.

**Fuchun Sun** received the Ph.D. degree from Tsinghua University, Beijing, China, in 1997.

He currently is a Full Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include intelligent control and robotics.

Dr. Sun was a recipient of the National Science Fund for Distinguished Young Scholars. He serves as an Associate Editor of a series of international journals including the IEEE TRANSACTIONS ON FUZZY SYSTEMS, the IEEE TRANSACTIONS ON MECHATRONICS, and the IEEE TRANSACTIONS ON ROBOTICS AND AUTONOMOUS SYSTEMS.

**Di Guo** is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Tsinghua University, Beijing, China.

Her research interests include robotic manipulation and sensor fusion.

**Bin Fang** received the Ph.D. degree from Beihang University, Beijing, China, in 2014.

He is currently a Research Assistant with the Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include robotic sensor fusion and human-robot interaction.

**Zhengchun Peng** received the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA.

He currently is an Endowed Professor with the Department of Optoelectronic Engineering, Shenzhen University, Shenzhen, China. His research interests include electronic skins for robots, stretchable electronics, and sensors in general.

Dr. Peng was a recipient of the China 1000 Plan Young Scholars.