Discriminative Exemplar Coding for Sign Language Recognition with Kinect

Chao Sun, Tianzhu Zhang, Bing-Kun Bao, Changsheng Xu, *Senior Member, IEEE,* and Tao Mei, *Senior Member, IEEE*

***Abstract*—Sign language recognition is a growing research area in the field of computer vision. A challenge within it is to model various signs, varying with time resolution, visual manual appearance, and so on. In this paper, we propose a discriminative exemplar coding (DEC) approach, as well as utilizing Kinect sensor, to model various signs. The proposed DEC method can be summarized as three steps. First, a quantity of class-specific candidate exemplars are learned from sign language videos in each sign category by considering their discrimination. Then, every video of all signs is described as a set of similarities between frames within it and the candidate exemplars. Instead of simply using a heuristic distance measure, the similarities are decided by a set of exemplar-based classifiers through the multiple instance learning, in which a positive (or negative) video is treated as a positive (or negative) bag and those frames similar to the given exemplar in Euclidean space as instances. Finally, we formulate the selection of the most discriminative exemplars into a framework and simultaneously produce a sign video classifier to recognize sign. To evaluate our method, we collect an American sign language dataset, which includes approximately 2000 phrases, while each phrase is captured by Kinect sensor with color, depth, and skeleton information. Experimental results on our dataset demonstrate the feasibility and effectiveness of the proposed approach for sign language recognition.**

***Index Terms*—Discriminative exemplar coding, Kinect sensor, sign language recognition.**

I. Introduction

SIGN LANGUAGE is one of the most natural means of exchanging information for deaf and hearing impaired person. It is a kind of visual language via hands and arm movements accompanying facial expressions and lip motions. Sign language recognition aims to efficiently and accurately translate sign language into text or speech. Generally, there are two important components in sign language recognition. One is how to extract useful information from raw video data, and the other is how to model different signs and measure their similarities for recognition. We deal with both two in this paper. Meanwhile, no one universal sign language is spread all over the world. Regionally different sign languages have been evolved, such as American sign language (ASL) [1], German sign language (GSL) [2], Chinese sign language (CSL) [3]. For simplification, we focus on the ASL in this paper, and the proposed method can also be utilized to other sign languages. Currently, automatic sign language recognition is still in its infancy, roughly decades behind automatic speech recognition. It corresponds to a gradual transition from isolated to continuous recognition for small vocabulary task. Many approaches to sign language recognition treated the problem as gesture recognition and mainly focused on robust extraction of manual features or statistical modeling of signs. Recent works [4]–[9] have proved the feasibility of action recognition based on key poses from single video frame. This kind of methods attempt to describe an action video with a set of representative frames called exemplars and then model various actions into a space defined by distances (or similarities) to these exemplars. However, varying from actors, environments or cameras, etc., videos of the same sign may contain dissimilar frames as well as different lengths or time resolutions [sequences (1) and (2) of Fig. 1]. Furthermore, videos from different signs may also include similar frames [sequences (1)–(4) of Fig. 1]. All these issues, but not limited to them, will increase the difficulties to recognize various signs from videos. Inspired by the exemplar-based approach, we attempt to build a generic method to recognize sign language.

The frames in videos collected by the Kinect sensor contain a huge range of variability, which include images of signs performed by people who have varying body sizes and different clothing. Although those noises included, there exist some representative key poses for others to recognize the signs. In many cases, the background clutter impedes good exemplar-based recognition using existing algorithms. It is observed that, how to select the key-pose frames as the representative exemplars and how to learn a suitable distance metric between the key-pose frames are the two important and challenging issues in the exemplar-based model.

In exemplar selection aspect, many research efforts have been conducted. For instance, some methods proposed to subsample or cluster the space of exemplars [10], [11]. Such

Manuscript received November 1, 2012; revised April 2, 2013; accepted May 13, 2013. Date of publication June 19, 2013; date of current version September 11, 2013. This work was supported in part by the National Basic Research Program of China (No. 2012CB316304), National Natural Science Foundation of China (No. 61225009), the Microsoft Research Asia UR Project, the Singapore National Research Foundation under its International Research Centre@Singapore Funding Initiative, and administered by the IDM Programme Office. This paper was recommended by Associate Editor L. Shao.

C. Sun, T. Zhang, and C. Xu are with the Institute of Automation, Chinese Academy of Sciences, Beijing 100191, China (e-mail: csun@nlpr.ia.ac.cn; tzzhang10@gmail.com; csxu@nlpr.ia.ac.cn).

B.-K. Bao is with the China-Singapore Institute of Digital Media, Singapore (e-mail: bingkunbao@gmail.com).

T. Mei is with the Internet Media Group, Microsoft Research Asia, Beijing 100190, China (e-mail: tmei@microsoft.com).

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.2013.2265337

Fig. 1. Some examples of different signs. Each row indicates frames from the video with the shown sign name.

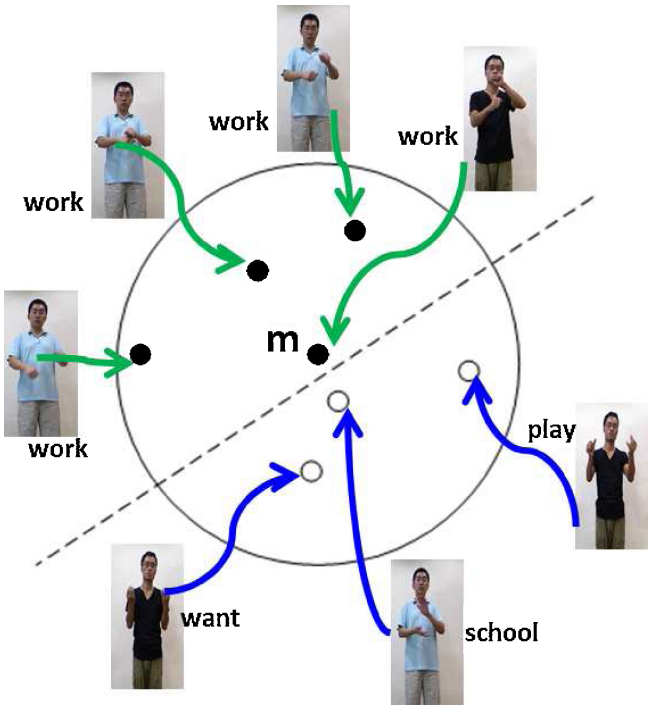

Fig. 2. Learning a classifier to describe the similarity of other frames to a frame. Note that the heuristic distance may lead to a wrong result.

methods required nevertheless a large mount of exemplars. What worse is, the clustering might miss some important exemplars [5]. In the works of Daniel and Edmond [8] and Weinland *et al.* [5], some discriminative exemplars are obtained with forward selection, which was particularly robust against over-fitting. However, the forward selection algorithm was slow calculation caused by the iterative learning and evaluation cycles.

In distance metric learning aspect, generally, heuristic distance metrics or specified matching approaches, such as squared Euclidean distance [8] or HMM-based (Hidden Markov Models) matching [5], were conducted to measure the similarity between exemplars. However, these approaches ignore the distribution of frames in the feature space and fail to achieve the best discrimination for recognition. As shown in Fig. 2, for one frame, not all its similar frames evaluated by a heuristic distance, such as Euclidean distance, belong to the same sign with it. Therefore, it is necessary to learn the similar frames for every frame across all samples within the same sign, instead of using a predefined and heuristic distance.

To overcome these two issues, in this paper, we propose a discriminative exemplar coding (DEC) approach (Fig. 3) to recognize various signs. First, amount of class-specific candidate exemplars are obtained by using of K-means for simplicity. Second, for each candidate exemplar, we employ the multiple instance learning (MIL) to learn the exemplar-based classifier to measure the similarity. For the MIL problem, each video of a sign is considered as a bag, and frames of the video are deemed as instances. Hence, if we obtain E candidate exemplars, each bag is then described as a E-dimensional vector of similarities between the E exemplars and the bag. Third, we apply AdaBoost algorithm to integrate the further selection of representative exemplars and sign modeling together. Specifically, the most discriminative exemplars are selected through the boosting learning, and simultaneously the similarities based on the selected exemplars as the weak classifier are combined to obtain a bag-based sign classifier. Experimental results on our self-built dataset demonstrate the feasibility and effectiveness of the proposed approach.

Beyond theoretical field, a difficulty in sign language recognition is to capture hand movement by sensor. In computer vision techniques, sensor is typically a camera which can generate depth maps for sign language recognition. Different kinds of sensors are explored varying from tracking systems, like data gloves, to computer vision techniques using camera [12] and motion capture systems [13]. As for now, commercially available depth camera systems are expensive, and only a few researchers use depth information to recognize hand pose.

Fortunately, the release of Microsoft Kinect sensor has provided a low cost and off-the-shelf choice for depth sensors. The Kinect sensor involves an infra-red(IR) light projector, standard CMOS camera, color camera, and a standard USB interface. The distortion of IR pattern is used to calculate depth maps, which have a per-pixel depth resolution of one centimeter while camera is two meters far away. The images are 640x480 and transferred at 30 frames per second [14]. The complementary nature of the depth and visual information provided by the Kinect sensor opens up new opportunities to some solve problems in computer vision [15], for example, object tracking, human activity analysis, hand gesture analysis, and sign language recognition in this paper.

The conjunction of depth map and color image from Kinect sensor could produce great contribution for sign language recognition in three aspects. First, with the depth map, background modeling becomes more simple and robust. we can easily and accurately extract human body part from color images. Second, in previous 2-D solutions, how to track hands is a difficulty task. However, the skeleton information, developed from depth map, can be utilized to locate the

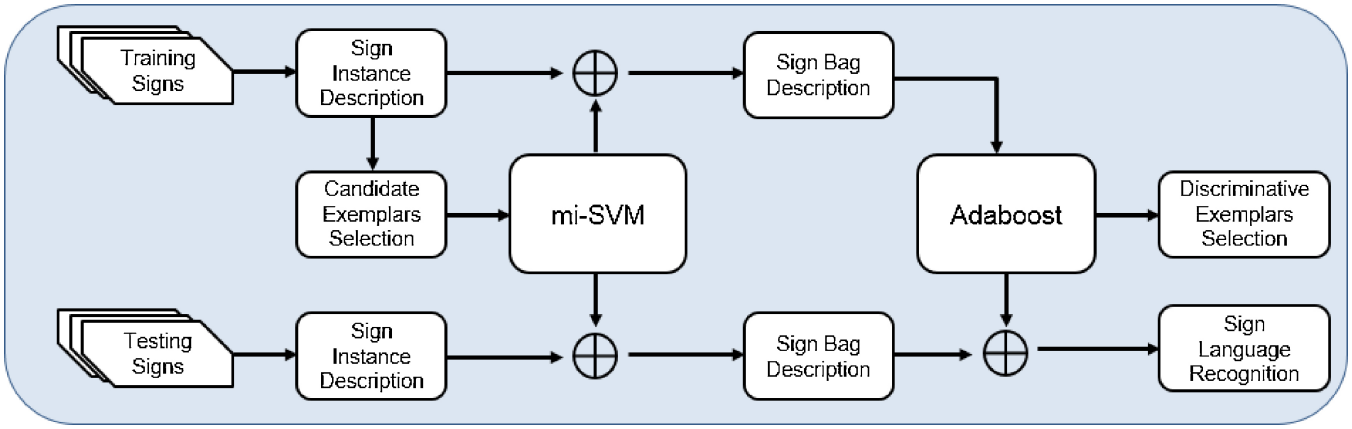

Fig. 3. Framework of our approach.

positions of hands robustly and in real time. Third, beyond the traditional 2-D features, Kinect senor can provide some novel 3-D features, which are quite useful and hence and improve the performance of sign language recognition. These advantages will emerge in our following experimental results.

The paper is organized as follows. In Section II we review the related work. The detailed implementation of the proposed DEC method is introduced in Section III. In Section IV we evaluate our approach on our self-built dataset. We conclude the paper with future work in Section V.

II. Related Work

Acquiring data is the first step in a sign language recognition. Many early methods used datagloves and accelerometers to acquire data of the hands. However, due to the high costs of such approaches, the use of vision has become more popular. In the case of vision input, a sequence of images are captured from a combination of cameras (e.g., monocular, stereo, orthogonal) or other noninvasive sensors. Most recently Microsoft Kinect has offered an affordable depth camera, which has made depth a viable option for more researchers. However, at present there are no datasets available and as such the results are limited.

Once data have been acquired, they are described via features. The features selection often depends on the elements of sign being detected. Exemplar-based embedding methods have already been proposed in computer vision field [10], [16], [17]. Athitsos and Sclaroff [10] presented an approach for hand pose estimation based on Lipschitz embeddings. In these approaches complex distances between signals were approximated in a Euclidean embedding space that was spanned by a set of distances to exemplar measures.

Recently, such exemplar-based approaches have been applied in many methods on action recognition. Dedeoglu *et al*. [18] proposed a real-time system for action recognition based on key poses and histograms. In order to group similar human poses together, Wang *et al*. [19] utilized deformable template matching for computing the distance between poses. In the work of Carlsson and Sullivan [4], class representative silhouettes were matched against video frames to recognize forehand and backhand strokes in tennis recordings. Thurau and Hlavac [7] approached the problem by using non-negative matrix factorization on pose primitives. In the work of Daniel and Edmond [8], exemplar-based approach was adopted to transform length-variant orderless feature set of action videos into matching distances to exemplars. A classifier was then trained based on this fixed length representation. In simple terms, the pose primitives were learned from non cluttered videos and applied on images to find the closest pose.

In terms of the discriminative exemplars selection, the forward selection method was conducted by Weinland *et al*. [5] and Daniel *et al*. [8]. Some other exemplar-based approaches, like [5], [11], [20], [21], attempted to learn HMMs with observation probabilities based on matching distances to exemplars. However, the similarities between frames and exemplars were measured using heuristic distance, which might not be accurate. Therefore, adapting an efficient approach to select the discriminative exemplars is essential and necessary.

To overcome this defect, we learn the similarity metric by MIL [22] instead of using heuristic distance. MIL allows of training classifiers with only labeling at the bag level, instead of labeling at the instance level. It has been developed into many different MIL approaches [23]–[25]. For example, Chen *et al*. [23] proposed a learning method, named multiple-instance learning via embedded instance selection (MILES), which converts the multiple-instance learning problem to a standard supervised learning problem that does not impose the assumption relating instance labels to bag labels. Zha *et al*. [24] proposed an integrated multilabel multiinstance learning (MIML) approach, which simultaneously models both the relation between semantic labels and instances, and the correlations among the labels in an integrated formulation. In our work, we employed multiinstance support vector machines (mi-SVM) [26] to learn exemplar-based classifiers for sign bag description.

III. Discriminative Exemplar Coding for Sign Language Modeling

In this section, we elaborate our DEC approach for sign language recognition. To recognize sign videos, our discriminative exemplar-based sign model should overcome the following three challenges: 1) to efficiently describe each sign

instance; 2) to accurately describe sign bag based on the candidate exemplars when only sign bag label is given; and 3) to effectively explore sign bags into an overall classifier on the given descriptions of sign bags. In our DEC approach, a unified and effective solution against these three challenges is presented.

Fig. 3 illustrates the framework of our DEC approach. Specifically, each sign instance is described by features, its details are introduced in Section IV-C. Based on this description, for each kind of sign, some candidate exemplars are selected firstly. Then, corresponding classifiers are trained for each candidate exemplar via mi-SVM. Next, based on the classifier of each exemplar, similarities between the exemplar and instances in a sign bag can be obtained. Then, the sign bag can be described using the similarities as its features. Finally, sign level classifiers are trained based on those sign bag descriptions. Considering the large intraclass variation of different sign, AdaBoost is employed to form a strong classifier to classify signs, as well as to select the most discriminative features, which corresponds to the most discriminative exemplars.

In the following, we present the candidate exemplars selection in Section III-A, the multiple instance learning for sign bag description in Section III-B, and the AdaBoost based sign classifier in Section III-C.

A. Candidate Exemplars Selection

Before elaborating our method, we first introduce the notations in this paper. A sign bag is a video of this sign and a sign instance is a frame of a video. Denote v_i as the ith sign bag and $I_{v_i,j}$ is the feature of the jth instance of bag v_i. Based on this description, for a sign bag v_i, it can be described as a set of histogram features. The formal definition of sign v_i is as follows: $v_i = \{I_{v_i,j} | j = 1, 2, \ldots, n_i\}$, where n_i is the number of instances from sign bag v_i. Let v_i^+ denote a positive sign bag and v_i^- denote a negative sign bag. v_{ij}^+ is the jth instance of a positive sign bag v_i^+ and v_{ij}^- denotes the j^{th} instance of a negative sign bag v_i^-. Let $\{v_1^+, v_2^+, \ldots, v_s^+, v_1^-, v_2^-, \ldots v_t^-\}$ denote the set of s positive and t negative training sign bags. $l(v_i) \in \{+1, -1\}$ is the bag label of v_i and $l(v_{ij}) \in \{+1, -1\}$ is the instance label of v_{ij}. For each negative sign bags, all its instances are negative. For each positive sign bags, all its instances should contain at least one true positive instance.

Intuitively, each sign instance could be treated as an exemplar. However, it will produce a huge number of candidate exemplars and leads to a very high computational cost when training each exemplar-based classifier. One possible remedy is to select a representative subset of instances (called candidate exemplars). For simplicity, we use k-means to create an initial vocabulary by grouping similar sign instances based on their features for each sign category, and select instances nearest to each cluster as initial exemplar set. As a result, for each category c, we obtain $E_c^{'}$ exemplars. In this way, we can obtain candidate exemplars for each kind of sign, and the total number of exemplars is $E = E_1^{'} + \ldots + E_c^{'} + \ldots + E_C^{'}$. Based on these candidate exemplars, we then adapt discriminative exemplar coding to model the discrimination of exemplars for sign language recognition.

B. Sign Bag Description

After candidate exemplars are selected from each class c in Section III-A, we then describe each sign bag as a set of similarities between its instances. Assume that $M_c^{'}$ exemplars from positive sign bags are obtained: $\{I_m | m = 1 \ldots E_c^{'}\}$, where I_e represents the feature for the e^{th} exemplar. For exemplar e in the sign bag of category c, it is possible that some instances from the same sign to e are less similar than the ones from other signs when a uniform distance metric is adopted (Fig. 2). To tackle this problem, we propose a discriminative solution to obtain semantic similarity by learning exemplar-based classifiers. Here, we formulate the similarity measure learning as a problem of MIL [22] and mi-SVM [26] is employed to solve the problem.

To obtain the exemplar-based classifiers, for an exemplar $e\,(e = 1, \ldots, E_c^{'})$ from the sign of category c, a corresponding mi-SVM classifier is trained and denoted by $mi - SVM_e$. After that, This process is conducted for different kinds of signs to train all exemplar-based classifiers. The training samples are the bags in c denoted as positive bags and those in other categories are denoted as negative ones. Some sign bags may contain a large number of instances. If we train the mi-SVM classifier by using all the instances, the computational and storage requirements may become too large. To reduce the computational burden and learn efficient classifier, we adapt an efficient KNN-based strategy to obtain sign bags by filtering out the instances which are very different from e for each sign bag v_i. This strategy enables the classifier to be learned only in the local feature space. Specifically, mi-SVM classifier is trained in the hyper-sphere centered at I_e with radius of r_e in the feature space (as shown in Fig. 2).

Denote $y_{v_i,j}$ as the instance label of $I_{v_i,j}$ and Y_{v_i} as the label of sign bag v_i, where $I_{v_i,j}$ indicates the feature of the instance j in the sign bag v_i. Then mi-SVM is formulated as follows:

$$\begin{aligned} &\min_{\{y_{v_i,j}\}} \min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{v_i,j}\xi_{v_i,j} \\ &s.t.\ \sum_j \frac{y_{v_i,j}+1}{2} \geqslant 1,\ \forall v_i\ s.t.\ Y_{v_i} = 1, \\ &\mathrm{y}_{v_i,j} = -1,\ \forall v_i\ s.t.\ Y_{v_i} = -1, \\ &\forall j : \mathrm{y}_{v_i,j}(\langle w, I_{v_i,j}\rangle + b) \geqslant 1 - \xi_{v_i,j}, \xi_{v_i,j} \geqslant 0, \\ &\mathrm{y}_{v_i,j} \in \{-1, 1\} \end{aligned} \tag{1}$$

where $\xi_{v_i,j}$ is a slack variable.

For the eth classifier $mi - SVM_e$, a sign bag v_i can be projected to real value with a function. For simplicity, The projection function is defined as follows:

$$g_e(v_i) = \begin{cases} \max\limits_j mi\_SVM_e(I_{v_i,j})\ \exists I_{v_i,j}, s.t.\ \left\|I_{v_i,j} - I_e\right\| \leqslant r_e \\ -1 \quad otherwise \end{cases} \tag{2}$$

where $mi\_SVM_e(I_{v_i,j}) \in \mathbb{R}$ is the output of $mi - SVM_e$ with the input $I_{v_i,j}$. Consequently, we can obtain $g_e(v_i)$ as the similarity for sign bag v_i with the learned classifier of exemplar e. If we have selected $E = E_1^{'} + \ldots + E_c^{'} + \ldots + E_C^{'}$ exemplars for all kinds of sign in training dataset, we then have E classifiers trained according to mi-SVM, respectively.

Algorithm 1 Proposed DEC algorithm. We first select candidate exemplars from all sign videos, then train mi-SVM for each candidate to get sign bag description, and finally adapt boosting to select the discriminative exemplars and classify sign language videos.

1: Given: N labeled training examples (v_i, y_i) with $y_i \in \{-1, 1\}$ and $v_i = \{I_{v_i,1}, \ldots, I_{v_i,n}\}$, and initial distribution of weights $w_i = \frac{1}{N}, i = 1, \ldots, N$.

2: Select E candidate exemplars for all kinds of sign using the proposed method in Section III-A and train their corresponding classifiers using mi-SVM to obtain $\{g_e(v_i)|i = 1, \ldots, N, e = 1, \ldots, E\}$ for sign bag description. The $g_e(v_i)$ can be viewed as the e_{th} feature of sign bag v_i.

3: For $t = 1, \ldots, T$:, Do

1) Train: Find E hypotheses h_e, by training the base learner on each feature g_e of the given training set, using current weighting w_i, and calculate the weighted training error for each hypothesis h_e

$$\varepsilon_e = \sum_{i=1}^{N} w_{t,i} 1(y_i \neq h_e(g_e(v_i)))$$

2) Select: hypothesis h_e with the lowest ε_e, set $h_t = h_e$ and $\varepsilon_t = \varepsilon_e$.

3) Calculate: hypothesis coefficient $\alpha_t = \frac{1}{2}\log(\frac{1-\varepsilon_t}{\varepsilon_t})$.

4) Update: sample weights $w_{t+1,i} = \frac{1}{Z_t} w_{t,i} \exp(-\alpha_t y_i h_t(v_i))$, where Z_t is a normalization coefficient such that $\sum_i (w_{t+1,i}) = 1$.

4: Output: the DEC classifier $H(v) = sgn(\sum_{t=1}^{T} \alpha_t h_t(v))$.

Eventually, each sign bag v_i can be described as E dimensional features $(g_1(v_i), \ldots, g_e(v_i), \ldots, g_E(v_i))^T$ based on these E classifiers, which are then utilized to train Adaboost classifier for sign recognition.

C. Sign-level AdaBoost Classifier

Based on the obtained sign bag descriptions, we turn to learn a sign-level classifier. In our paper, we utilize the boosting [27] method. Practically, boosting method is ideally suitable for combining diverse classifiers into an overall classifier. It combines multiple weak learners into a single strong classifier to achieve a low overall error of classification, while each of those weak learners may suffer from the high classifying error. In boosting, weak classifiers are trained sequentially with adjusting the weights of the training samples, which will optimize the weight of the incorrectly classified examples and improve the final classifying performance.

The discrete version of AdaBoost [27] defines a strong binary classifier H

$$H(v_i) = \text{sgn}(\sum_{t=1}^{T} \alpha_t h_t(g_e(v_i))) \tag{3}$$

using a weighted combination of T weak learners h_t with weights α_t. Each weak learner

$$h_t(g_e(v_i)) = \begin{cases} 1 & if\ g_e(v_i) > threshold \\ -1 & otherwise \end{cases} \tag{4}$$

may explore any feature $g_e(v_i)$ of the sign bag v_i.

Based on the features of sign bags, the optimal threshold in (4) is determined and weak learners are trained and combined to get a strong classifier for sign recognition.

In addition, as each dimension of one feature vector corresponds to an exemplar, the discriminative exemplars can be selected from the candidate frames during the AdaBoost learning process.

The procedure of proposed DEC method can be summarized as Algorithm 1.

IV. Experimental Results

In this section, we first introduce our self-built sign language dataset collected by Kinect sensor, then conduct recognition on this dataset to validate the effectiveness of our proposed method. In experiments, for the multiclass classification problem, we deal with it as a series of two-class problems, for which one-against-all strategy is adopted.

A. Kinect Sign Language Dataset

Currently, these is no public kinect sign language dataset. The existing public sign language datasets are totally based on 2-D camera, which lack the depth information and thus can not be used to evaluate the proposed method. In this situation, we built the Kinect sign language dataset by ourselves.

Our dataset includes 73 ASL signs, while each sign corresponds to a vocabulary, as shown in Fig. 4. These signs came from a hundred basic ASL signs that are frequently used by the beginners of signers. We discarded those look like too similar in vision, and finally selected 73 signs of them. We recruited nine participants, each of which stood in front of Kinect sensor and performed all the signs three times. A total of 1971 phrases were collected, each of which includes a set of color image, a set of depth map, and a set of skeleton information.

B. Baselines

To evaluate our discriminative exemplar coding method for sign language recognition, we compare with several state-of-the-art coding methods, such as hard-assignment coding (HC) [28], soft-assignment coding (LSC) [29], and locality-constrained linear coding (LLC) [30]. Our sign language model is based on exemplars, and ignores the temporal information. To compensate this disadvantage, we make use of the basic idea of spatial pyramid matching (SPM) [31] to model the temporal information for representation. In our paper, the SPM with two levels 1×1 and 1×3 is adopted.

For these coding methods [28]–[30], the basic ideas are the following. Let b_i ($b_i \in R^d$) denote a visual word or an exemplar, where d is the dimensionality of a local feature or a frame representation. The total number of exemplars is n. A matrix $B = [b_1, b_2, \cdots, b_n]$ denotes a visual codebook or a set of basis vectors. Let x_i ($x_i \in R^d$) be the ith local feature in an image. Let z_i ($z_i \in R^n$) be the coding coefficient vector of x_i, with z_{ij} being the coefficient with respect to word b_j.

Hard-assignment coding [28]: For a local feature x_i, there is one and only one nonzero coding coefficient. It corresponds to

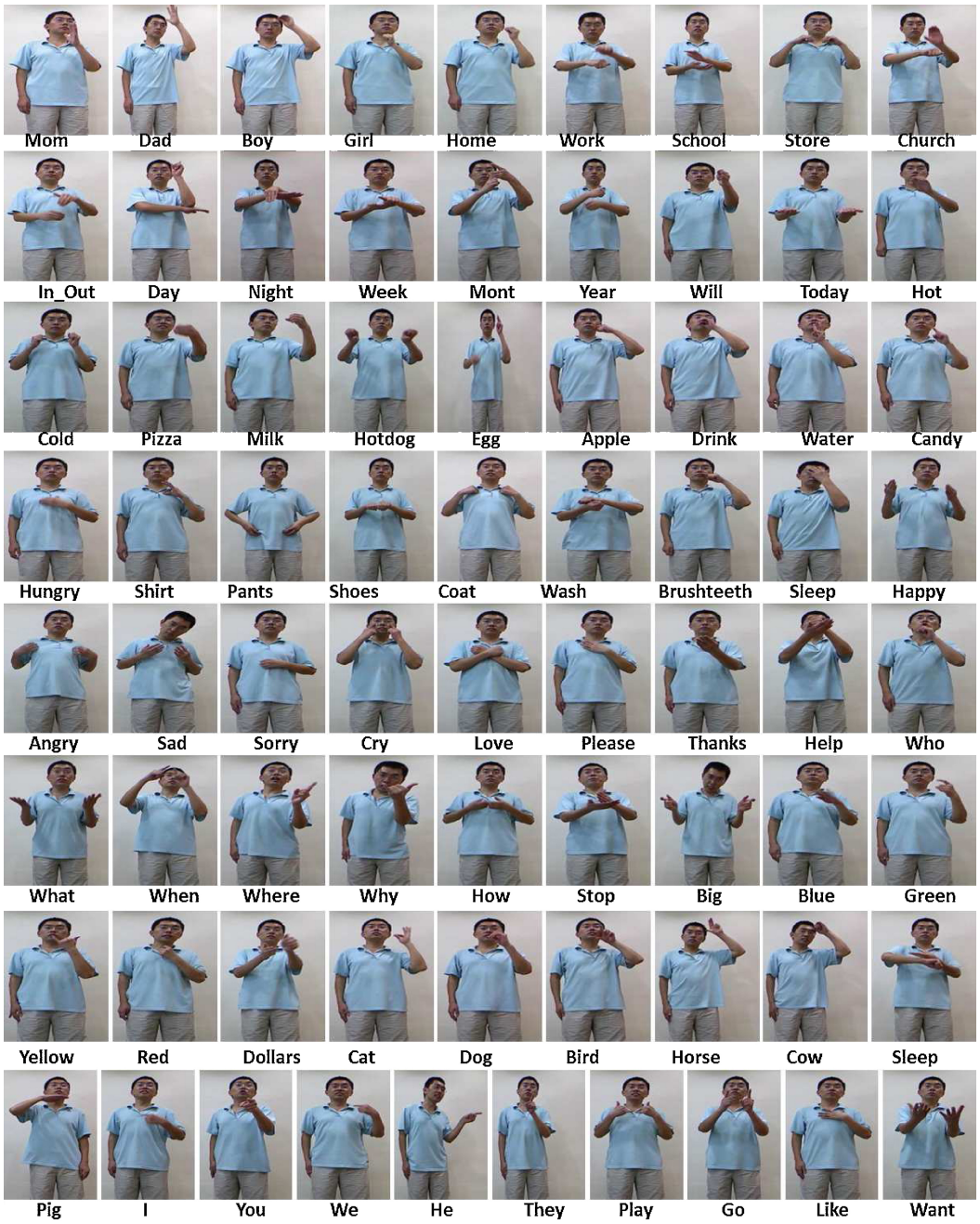

Fig. 4. Some examples of collected images for sign language recognition. These show the cropped images based on the mask, and each example is one kind of sign language. In total, there are 73 classes.

the nearest visual word subject to a predefined distance. When Euclidean distance is used

$$z_{ij} = \begin{cases} 1 & if\ j = \arg\min\limits_{j=1,\dots,n} \left\| x_i - b_j \right\|_2^2 \\ 0 & otherwise. \end{cases}$$

Soft-assignment coding [29]: The jth coding coefficient represents the degree of membership of a local feature x_i to the jth visual word

$$z_{ij} = \frac{\exp(-\alpha \left\| x_i - b_j \right\|_2^2)}{\sum_{k=1}^{n} \exp(-\alpha \left\| x_i - b_k \right\|_2^2)}$$

where α is the smoothing factor controlling the softness of the assignment. Note that all the n visual words are used in computing $\mathbf{z_{ij}}$.

Locality-constrained linear coding (LLC) [30]: Unlike the sparse coding, LLC enforces locality instead of sparsity. This leads to smaller coefficient for the basis vectors farther away from a local feature x_i. The coding coefficient is obtained by solving the following optimization:

$$\begin{aligned} & z_i = \arg\min_{z \in \Re^n} \left\| x_i - Bz \right\|_2^2 + \lambda \left\| d_i \odot z \right\|_2^2 \\ & s.t.\ 1^T z_i = 1 \end{aligned} \tag{5}$$

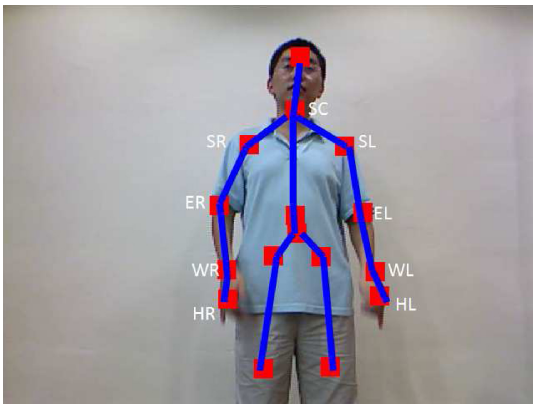

Fig. 5. Skeleton joints' names.

TABLE I
BODY POSE FEATURES

3D Vectors	Angles	Distance
SR → ER (3)	∠ SC-SR-ER (1)	HR → HL (1)
ER → WR (3)	∠ SR-ER-WR (1)	
WR → HR (3)	∠ ER-WR-HR (1)	
SL → DEC (3)	∠ SC-SL-DEC (1)	
DEC → WL (3)	∠ SL-DEC-WL (1)	
WL → HL (3)	∠ DEC-WL-HL (1)	
HR → HL (3)		

where $d_i = \exp(dist(x_i, B)/\delta)$ and $dist(x_i, B) = (dist(x_i, b_1), dist(x_i, b_2), \ldots, dist(x_i, b_n))^T$, $dist(x_i, b_j)$ denotes the Euclidean distance between x_i and each b_j. δ is a parameter controlling the weighting vector d_i. In [30], a smart approximation is proposed to improve its computational efficiency in practice. Ignoring the second term in (5), it directly selects the k nearest basis vectors of x_i to minimize the first term by solving a much smaller linear system. This gives the coding coefficient for the selected k basis vectors and other coefficient are simply set to zero.

After coding, Libnear SVM [32] is adopted for classification as in [28]–[30].

C. Features and Sign Instance Description

For sign language recognition, our algorithm was conducted on the self-built dataset. We first introduce features to describe sign instance. In this paper, we adopt two different features including HOG features and Kinect features. The HOG feature can describe the appearance information. Based on the output of Kinect, we can know the position of hands, and obtain their shape information and motion features. In addition, we can also estimate body pose by using of Kinect features.

HOG Features: Based on the depth map from Kinect, it is easy to obtain the mask image and crop the foreground. Once the humans are centralized, we extract HOG descriptor for each detected area. In human detection, the HOG has been shown to be successful [33]. We follow the construction in [33] to define a dense representation of an image at a particular resolution. The image is first divided into 8×8 non-overlapping pixel regions, or cells. For each cell we accumulate a 1-D histogram of gradient orientations over pixels in that cell.

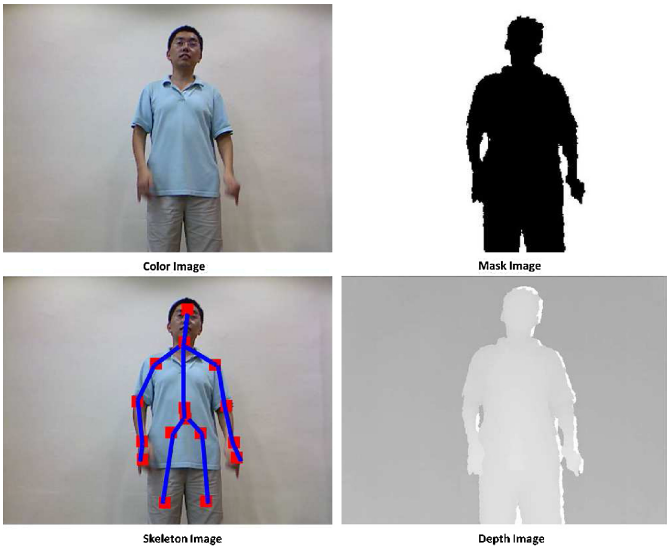

Fig. 6. Four kinds of output from Kinect.

These histograms capture local shape properties but are also somewhat invariant to small deformations.

The gradient at each pixel is discretized into one of nine orientation bins, and each pixel votes for the orientation of its gradient, with a strength that depends on the gradient magnitude at that pixel. For color images, we compute the gradient of each color channel and pick the channel with the highest gradient magnitude at each pixel. Finally, the histogram of each cell is normalized with respect to the gradient energy in a neighborhood around it. We look at the four 2 × 2 blocks of cells that contain a particular cell and normalize the histogram of the given cell with respect to the total energy in each of these blocks. This leads to a 9 × 4 dimensional vector representing the local gradient information inside a cell. In our implementation, we resize each image to 256 × 128 and then extract HOGs in 8 × 8 cells. Our final feature vector is the 2340-dimensional normalized HOG cell vector. After PCA [34], the dimension of the feature is further reduced to 750 to obtain compact description and efficient computation.

Kinect Features: Kinect sensor has four kinds of output: color image, depth image, mask image, and skeleton image, as shown in Fig 6. The Kinect features include body pose, hand shape, and hand motion features. The body pose features are extracted using skeleton information. By using Microsoft KinectSDK 1.5, we can obtain the positions of shoulder, elbow, wrist and hand, both in right and left side of the body. The body pose features are the combination of three parts.

1) The unit vectors of the elbows with respect to the shoulders, the wrists with respect to the elbows, the hands with respect to the wrists, and the left hand with respect to the right hand.
2) The joint angles of shoulders, elbows, and wrists.
3) The distance between the right hand and the left hand, normalized by being divided by twice shoulder width. In total, the body pose feature has 28 dimensions. Fig. 5 and Table I show the details of body pose feature.

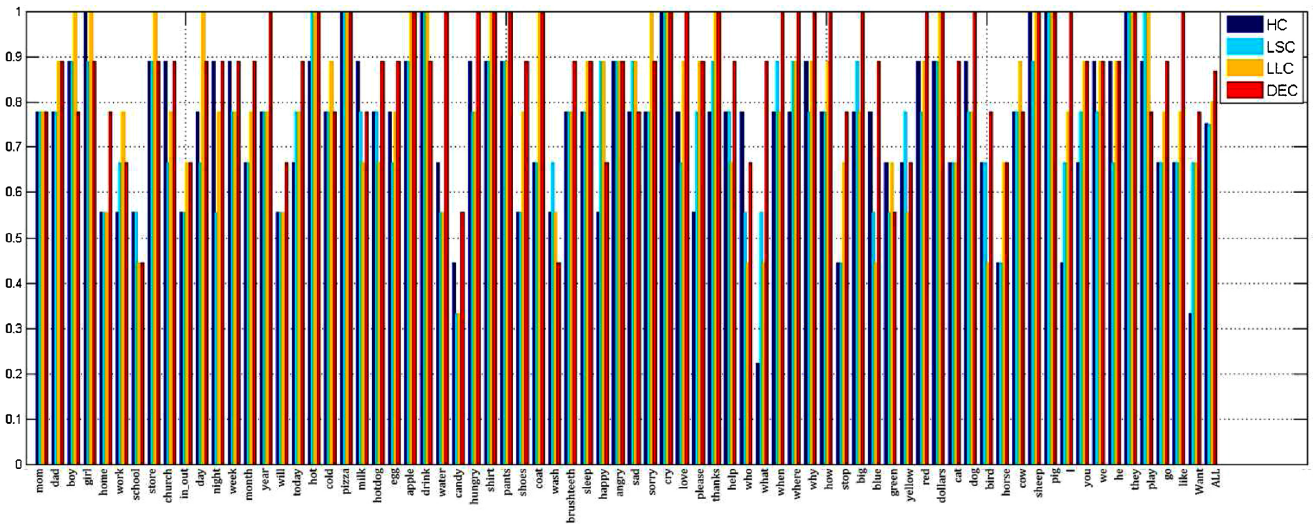

Fig. 7. Comparison of different methods on each sign class.

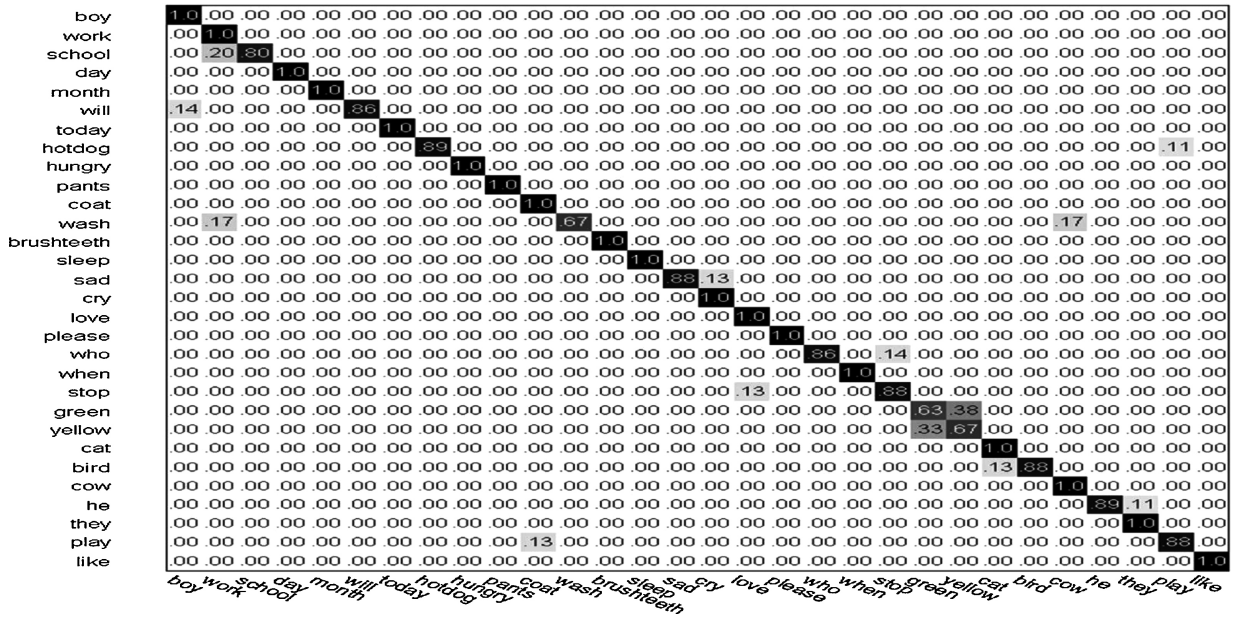

Fig. 8. Confuse matrix on the randomly selected 30 different kinds of signs.

To generate the hand shape feature, we first crop a 48×48 pixels patch in the position of hand point on every color frame. Then we extract the HOG feature on every patch and treat this feature as hand shape feature. This hand shape feature has 288 dimensions.

For generating the hand motion feature, we re-use the patch mentioned above. Optic flow (OP) feature is calculated between one patch on a color frame and the patch in the same position on the previous frame. This feature is treated as hand motion feature and has 2304 dimensions.

To obtain compact description and efficient computation, the combined 2592 dimensions feature is the reduced to about 300 dimensions using PCA [34].

D. Sign Language Recognition Results

As the training dataset contains tens of thousands of cropped frames, we use K-means to cluster the cropped frames and the cluster number is manually set to 100 for each kind of sign. Then, the frames closest to the cluster are viewed as exemplars. For each exemplar, $k_p = 20$ and $k_n = 2$ are adopted to obtain some training samples for training an mi-SVM classifier. After obtaining all mi-SVM classifiers, each sign bag is described and the discriminative exemplars are selected by AdaBoost classifier.

1) *Comparison of different methods:* The recognition rates for individual classes, as well as the average recognition rate, are shown in Fig. 7. Comparison results of different baselines and our method are also shown in Table II. From Fig. 7, we can see that the recognition rate of our method on each class is much better than other state-of-the-art methods. In addition, as shown in Table II, the average recognition rate is about 85.5% with all 73 classes when using HOG and Kinect features without temporal information, which has about 15% improvement compared with HC [28]. Even compared with the

Fig. 9. Illustrations of the discriminative exemplars for different signs selected by the final AdaBoost classifiers, (a) mom, (b) dad, (c) boy, (d) girl, (e) milk, (f) work, (g) school, (h) store, (i) holding, (j) in-out, (k) egg, (l) night, (m) week, (n) month, (o) year, (p) will, (q) today, (r) apple, (s) cold, and (t) pizza.

TABLE II
COMPARISON OF DIFFERENT METHODS ON OUR DATASET WITHOUT TEMPORAL INFORMATION. PERCENTAGES ARE THE AVERAGE ACCURACIES OVER ALL SIGNS

Methods	Mean Accuracy	Feature
HC [28]	65.6%	HOG
LSC [29]	74.1%	HOG
LLC [30]	73.4%	HOG
DEC	78.1%	HOG
HC [28]	70.2%	HOG + Kinect
LSC [29]	73.1%	HOG + Kinect
LLC [30]	77.6%	HOG + Kinect
DEC	**85.5%**	HOG + Kinect

TABLE III
COMPARISON OF DIFFERENT METHODS ON OUR DATASET CONSIDERING TEMPORAL INFORMATION FOR SIGN LANGUAGE REPRESENTATION. PERCENTAGES ARE THE AVERAGE ACCURACIES OVER ALL SIGNS

Methods	Mean Accuracy	Feature
HC [28]	67.9%	HOG
LSC [29]	74.4%	HOG
LLC [30]	73.7%	HOG
DEC	79.1%	HOG
HC [28]	75.2%	HOG + Kinect
LSC [29]	75.0%	HOG + Kinect
LLC [30]	80.1%	HOG + Kinect
DEC	**86.8%**	HOG + Kinect

best baseline [30], our method has about 8% improvement. We believe that the improvement attributes to the efficient similarity measure learning and the effectively selected and combined discriminative exemplars via DEC approach. These results demonstrate that our DEC method outperforms all other state-of-the-art methods on sign language recognition.

2) *Comparison with different features:* To demonstrate the effectiveness of Kinect features, we conduct comparing experiments between recognition with Kinect features and recognition without Kinect features. The results are reported in Tables II and III. The first four lines in each table show recognition accuracy using only HOG feature, while the last

four lines in each table show recognition accuracy using Kinect features together with HOG feature. We can observe that, each method, no matter our DEC or other baseline, get a higher recognition accuracy when using Kinect features together with HOG feature than the one without Kinect features. The highest improvement is about 8%. These results prove the effectiveness of Kinect features in sign language recognition. The 3-D features provided by Kinect, such as depth information, skeleton information, are useful and could improve the performance of recognition.

3) *Comparison of temporal information:* We believe that the temporal information in sign videos could contribute to sign language recognition. To prove this, we conduct comparing experiments between recognition with considering temporal information and the one without considering temporal information. Experimental results are shown in Tables II and III. Experiments shown in Table III have considered temporal information by using SPM and the ones shown in Table II have not. It is observed that, for every method, recognition accuracy with considering temporal information is higher than the one without considering temporal information. The average improvement is about 2%. These results prove that, by considering temporal information, recognition accuracy of sign language could improve. Consideration of temporal information is necessary in sign language recognition.

4) *Conclusive results:* Besides above comparisons, we also give out the confusion matrix. Due to the large number of classes (73), it is difficult to show all of them in a confusion matrix. To show it clearly, we randomly select 30 different kinds of signs. The confusion matrix is shown in Fig. 8. From the confusion matrix, we can see that our DEC method could distinguish signs very well. Almost every sign is distinct from each other and is recognized correctly. The most of confusion occurs between "green" and "yellow" signs. This is not surprising, as these two signs look very similar and have minor differences in finger movements. Humans can also sometimes confuse these two signs. This similarity leads to some mistaken recognition results between these two signs by our method.

All the above results could demonstrate the validity and effectiveness of our proposed method for sign language recognition. In addition, our DEC method could also find out the most discriminative exemplars during sign recognition, as described in Section III-C. Fig. 9 shows the selected discriminative exemplars in the first four iterations of some signs. From the results, we can see that our method has ability to find out the most discriminative exemplars to represent each kind of signs.

V. Conclusion

We presented a DEC approach for ASL recognition with Kinect sensor. On one hand, we efficiently conducted background modeling to extract human body and locate the hand position in frames. On the other hand, we obtained more discriminative features, as well as sign description. Based on this description, the MIL was employed to learn the similarities between frames. Based on the learned exemplar-based classifiers, each sign bag were described, then AdaBoost was employed to select the most discriminative features to form a strong classifier, which was used to classify signs. Experimental results demonstrated the effectiveness and efficiency of the proposed method for sign language recognition. In future, we will extend our work to: 1) continuous sign language recognition, beyond the current isolated one, and 2) recognition under complex scene.

References

[1] H. Brashear, T. Starner, P. Lukowicz, and H. Junker., "Using multiple sensors for mobile sign language recognition," in *Proc. 7th IEEE Int. Symp. Wearable Comput.*, 2003, pp. 45–52.

[2] B. Bauer, H. Hienz, and K. Kraiss., "Video-based continuous sign language recognition using statistical methods," in *Proc. 15th Int. Conf. Pattern Recognit.*, vol. 2, Sep. 2000, pp. 463–466.

[3] G. Fang, W. Gao, and D. Zhao., "Large vocabulary sign language recognition based on hierarchical decision trees," in *Proc. Int. Conf. Multimodal Interfaces*, Nov. 2003, pp. 125–131.

[4] S. Carlsson and J. Sullivan, "Action recognition by shape matching to key frames," in *Proc. Workshop Models versus Exemplars Comput. Vision*, vol. 1. 2001, pp. 1–8.

[5] D. Weinland, E. Boyer, and R. Ronfard, "Action recognition from arbitrary views using 3d exemplars," in *Proc. ICCV*, 2007, pp. 1–7.

[6] K. Schindler and L. van Gool, "Action snippets: How many frames does human action recognition require?" in *Proc. CVPR*, 2008, pp. 1–8.

[7] C. Thurau and V. Hlavac, "Pose primitive based human action recognition in videos or still images," in *Proc. CVPR*, 2008, pp. 1–8.

[8] D. Weinland and E. Boyer, "Action recognition using exemplar-based embedding," in *Proc. CVPR*, 2008, pp. 1–7.

[9] N. I. Cinbis, R. G. Cinbis, and S. Sclaroff, "Learning actions from the web," in *Proc. ICCV*, 2009, pp. 995–1002.

[10] V. Athitsos and S. Sclaroff, "Estimating 3d hand pose from a cluttered image," in *Proc. CVPR*, vol. 2. 2003, pp. II–432.

[11] K. Toyama and A. Blake, "Probabilistic tracking in a metric space," in *Proc. ICCV*, vol. 2. 2001, pp. 50–57.

[12] T. E Starner, "Visual recognition of american sign language using hidden markov models," DTIC Document, Tech. Rep., 1995.

[13] J. L. Hernandez-Rebollar, N. Kyriakopoulos, and R. W. Lindeman, "A new instrumented approach for translating american sign language into sound and text," in *Proc. 6th IEEE Int. Conf. Automat. Face Gesture Recognit.*, 2004, pp. 547–552.

[14] Z. Y. Zhang, "Microsoft Kinect sensor and its effect," *IEEE Multimedia*, vol. 19, no. 2, pp. 4–10, Feb. 2012.

[15] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1318–1334, Oct. 2013.

[16] Y. Guo, Y. Shan, H. Sawhney, and R. Kumar, "Peet: Prototype embedding and embedding transition for matching vehicles over disparate viewpoints," in *Proc. CVPR*, 2007, pp. 1–8.

[17] T. Zhang, J. Liu, S. Liu, C. Xu, and H. Lu, "Boosted exemplar learning for action recognition and annotation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 7, pp. 853–866, Jul. 2011.

[18] Y. Dedeoglu, B. Toreyin, U. Gudukbay, and A. Cetin, "Silhouette-based method for object classification and human action recognition in video," in *Computer Vision in Human-Computer Interaction*. Springer, 2006, pp. 64–77.

[19] Y.Wang, H. Jiang, M. S. Drew, Z.-N. Li, and G. Mori, "Unsupervised discovery of action classes," in *Proc. CVPR*, vol. 2. 2006, pp. 1654–1661.

[20] A. M. Elgammal, V. D. Shet, Y. Yacoob, and L. S. Davis, "Learning dynamics for exemplar-based gesture recognition,"in *Proc. CVPR*, vol. 1. 2003, pp. I-571.

[21] A. Fathi and G. Mori, "Human pose estimation using motion exemplars," in *Proc. ICCV*, 2007, pp. 1–8.

[22] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez, "Solving the multiple-instance problem with axis parallel rectangles," *Artif. Intell.*, vol. 89, no. 1, pp. 31–71, 1997.

[23] Y. Chen, J. Bi, and J. Z. Wang, "Miles: Multiple-instance learning via embedded instance selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 1931–1947, 2006.

[24] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang, "Joint multi-label multi-instance learning for image classification," in *Proc. IEEE CVPR*, 2008, pp. 1–8.
[25] T. Zhang, C. Xu, G. Zhu, S. Liu, and H. Lu, "A generic framework for event detection in various video domains," in *Proc. Int. Conf. Multimedia*, 2010, pp. 103–112.
[26] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. NIPS*, vol. 15, 2002, pp. 561–568.
[27] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," Stanford Univ., Stanford, CA, USA, Tech. Rep., 1998.
[28] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, 2003, pp. 1470–1477.
[29] L. Liu, L. Wang, and X. Liu, "In defense of soft-assignment coding," in *Proc. ICCV*, 2011, pp. 2486–2493.
[30] J.Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. CVPR*, 2010, pp. 3360–3367.
[31] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. CVPR*, vol.2, 2006, pp. 2169–2178.
[32] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[33] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, pp. 886–893.
[34] B.-K. Bao, G. Liu, C. Xu, and S. Yan, "Inductive robust principal component analysis," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3794–3800, Aug. 2012.

Chao Sun received the B.E. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2006. He is currently pursuing the Ph.D. degree at the Multimedia Computing Group, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. In 2012, he was an intern student in China-Singapore Institute of Digital Media, Singapore.

His current research interests include multimedia and computer vision.

Tianzhu Zhang received the bachelor's degree in communications and information technology from Beijing Institute of Technology, Beijing, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011.

Currently, he is an Assistant Professor at the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision and multimedia, including action recognition, object classification and object tracking.

Bing-Kun Bao received the Ph.D. degree in control theory and control application from the Department of Automation, University of Science and Technology of China, China, in 2009.

From 2009 to 2011, she was a Research Engineer in electrical and computer engineering, National University of Singapore, Singapore. She is currently an Assistant Researcher at the Institute of Automation, Chinese Academy of Science, Beijing, China, and also a Researcher at the China-Singapore Institute of Digital Media, Singapore.

Changsheng Xu (M'97-SM'99) is currently a Professor at the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, and the Executive Director, China-Singapore Institute of Digital Media, Singapore. He holds 30 patents and pending patents and has published over 200 refereed research papers. His current research interests include multimedia content analysis/indexing/retrieval, pattern recognition and computer vision.

Dr. Xu is an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA and *ACM Transactions on Multimedia Computing, Communications*. He served as Program Chair of ACM Multimedia in 2009, and as an Associate Editor, Guest Editor, General Chair, Program Chair, Area/Track Chair, Special Session Organizer, Session Chair and TPC member for over 20 IEEE and ACM prestigious multimedia journals, conferences and workshops. He is an ACM Distinguished Scientist.

Tao Mei (M'07-SM'11) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively.

Currently, he is a Researcher with Microsoft Research Asia, Beijing, China. He has authored or coauthored over 140 papers in journals and conferences, eight book chapters, and edited two books. He holds six U.S. patents and more than 30 pending patents. His current research interests include multimedia information retrieval and computer vision.

Dr. Mei was the recipient of several Best Paper Awards from prestigious multimedia conferences, including the Best Paper Awards and the Best Demonstration Award at ACM Multimedia in 2007, the Best Poster Paper Award at the IEEE MMSP in 2008, the Best Paper Award at ACM Multimedia in 2009, the Top 10% Paper Award at the IEEE MMSP in 2012, the Best Paper Award at ACM ICIMCS in 2012, the Best Student Paper Award at the IEEE VCIP in 2012, the Best Paper Finalist at ACM Multimedia in 2012, and the IEEE Transactions on Multimedia Best Paper Award 2013. He received Microsoft Gold Star Award in 2010, and Microsoft Technology Transfer Awards in 2010 and 2012. He is an Associate Editor of *Neurocomputing* and the *Journal of Multimedia*, a Guest Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE MULTIMEDIA MAGAZINE, the *ACM/Springer Multimedia Systems*, and the *Journal of Visual Communication and Image Representation*, and so on. He is the Program Co-Chair of MMM 2013, and the General Co-Chair of ACM ICIMCS 2013. He is a Senior member of the ACM.