J. Vis. Commun. Image R. 58 (2019) 138–148

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Hybrid incremental learning of new data and new classes for hand-held object recognition ☆

Chengpeng Chen [a,d], Weiqing Min [b,c], Xue Li [b], Shuqiang Jiang [b,d,*]

[a] State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[c] State key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China
[d] University of Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 29 March 2018
Revised 28 September 2018
Accepted 5 November 2018
Available online 22 November 2018

Keywords:
Incremental learning
Object recognition
SVM
Human-machine interaction

ABSTRACT

Intelligence technology is an important research area. As a very special yet important case of object recognition, hand-held object recognition plays an important role in intelligence technology for its many applications such as visual question-answering and reasoning. In real-world scenarios, the datasets are open-ended and dynamic: new object samples and new object classes increase continuously. This requires the intelligence technology to enable hybrid incremental learning, which supports both data-incremental and class-incremental learning to efficiently learn the new information. However, existing work mainly focuses on one side of incremental learning, either data-incremental or class-incremental learning while do not handle two sides of incremental learning in a unified framework. To solve the problem, we present a Hybrid Incremental Learning (HIL) method based on Support Vector Machine (SVM), which can incrementally improve its recognition ability by learning new object samples and new object concepts during the interaction with humans. In order to integrate data-incremental and class-incremental learning into one unified framework, HIL adds the new classification-planes and adjusts existing classification-planes under the setting of SVM. As a result, our system can simultaneously improve the recognition quality of known concepts by minimizing the prediction error and transfer the previous model to recognize unknown objects. We apply the proposed method into hand-held object recognition and the experimental results demonstrated its advantage of HIL. In addition, we conducted extensive experiments on the subset of ImageNet and the experimental results further validated the effectiveness of the proposed method.

1. Introduction

For the importance of the vision in the intelligence technology, object recognition has been a widely studied problem. It describes the task of finding and identifying objects in an image or video sequence. As manipulating objects with hands is a straight way for the intelligence technology, hand-held object recognition [1–3] is a special and important case in object recognition. The hand-held object recognition can not only help the system obtain a better understanding of user's intention but also a more comprehensive perception about surrounding environment. The real-world environment for the intelligence technology is open-ended and dynamic: new samples of existing object classes and new object classes become available all the time and the semantics of existing classes evolve [4]. This requires the hand-held object recognition with incremental learning, which makes the system improve the recognition performance continuously through the learning with self-growth.

For vision-based interaction systems, most work focused on two types of incremental learning: data-incremental learning [5–9] or class-incremental learning [10–12]. The data-incremental learning is to improve the recognition quality of known objects based on newly available training samples. For example, Ruping et al. [9] proposed a framework with SVM for supervised learning. Noboru et al. [8] further proposed an on-line framework for both supervised and unsupervised learning. In contrast, class-incremental learning is to learn for recognizing new objects with the emergence of unseen concepts. For example, Ristin et al. [12] proposed the variants of random forest to incorporate new classes for class-incremental learning. However, like the

☆ This article is part of the Special Issue on Multimodal_Cooperation.

* Corresponding author.
E-mail address: sqjiang@ict.ac.cn (S. Jiang).

https://doi.org/10.1016/j.jvcir.2018.11.009
1047-3203/

two aspects of human learning, these two types of incremental learning are indispensable for recognizing new concepts and enhancing the recognition ability of known concepts, and thus should work jointly. Therefore, in this work, we focus on integrating these two kinds of incremental ways into a unified framework, namely Hybrid Incremental Learning (HIL). Fig. 1 shows an example to illustrate these kinds of incremental learning.

However, it is non-trivial to incoprate HIL into a unified framework. One challenging problem is how to efficiently adjust classification-planes in HIL. Data incremental learning and class-incremental learning often encounter the stability-plasticity dilemma [13]: a completely stable classifier will preserve existing knowledge, but will not accommodate any new information while a completely plastic classifier will learn new information, but will not conserve prior knowledge [14]. Data incremental learning is to adjust existing classification-planes to find a balance between stability and plasticity. Class-incremental learning is to add new class classification-planes to find the balance. However, hybrid incremental learning should simultaneously adjust existing classification-planes and add new classification planes, and thus is more complex and difficult to find such the balance.

In order to solve the problem, we propose a Hybrid Incremental Learning (HIL) method under the setting of SVM, which is capable of simultaneous data-incremental and class-incremental learning to handle both new data and new classes. In order to enable simultaneous adjustment of existing classification planes and new ones, HIL enforces the target model to keep close to the source model, and meanwhile enforces the new added classification plane to keep close to the linear combination of other existed classification-planes. Based on the proposed HIL, we propose a hybrid incremental learning framework for hand-held object recognition. As shown in Fig. 2, the framework includes the following three modules: data preprocessing, hybrid incremental learning and hand-held object recognition, Firstly, data preprocessing is to collect the images including RGB images and depth information using the Kinect, where the additional depth and skeletal information is beneficial to eliminate background noise and make the following object segmentation easier and more precise. The RGB and depth CNN features are extracted from segmented RGB and depth images respectively, and are then fused. Secondly, the proposed HIL is to learn new data of learned concepts by adjusting existing classification-planes and new data of unknown concepts by adding new classification-planes. Finally, we applied the hybrid incremental learning method into hand-held object recognition. The contributions of our work can be summarized as follows:

(1) We propose a hybrid incremental learning framework to enable both data-incremental and class-incremental learning for hand-held object recognition.
(2) We propose a new incremental learning method to automatically learn new concepts by adding new classification-planes to the model and improve the recognition quality of learned concepts by adjusting existing classification-planes of the model under the setting of SVM.
(3) We conduct the experiment in the hand-held dataset and demonstrated its advantage of hybrid incremental learning. Furthermore, we conduct extensive experiments on the ImageNet and the experimental results further validate the effectiveness of the proposed method.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents the hybrid incremental learning method in detail. Experimental results are reported in Section 5. Finally, we conclude the paper in Section 6.

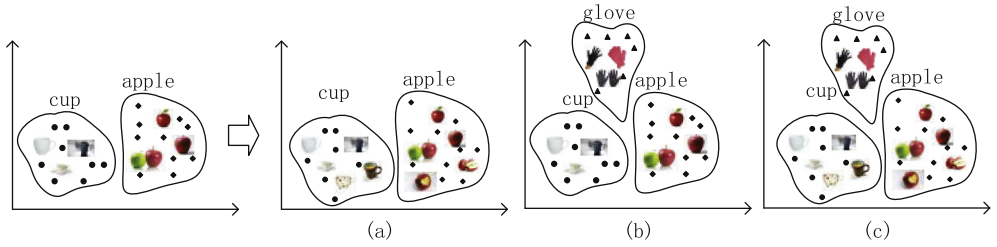

Fig. 1. The toy example for the incremental learning. (a): data-incremental learning. (b): class-incremental learning. (c): hybrid incremental learning.

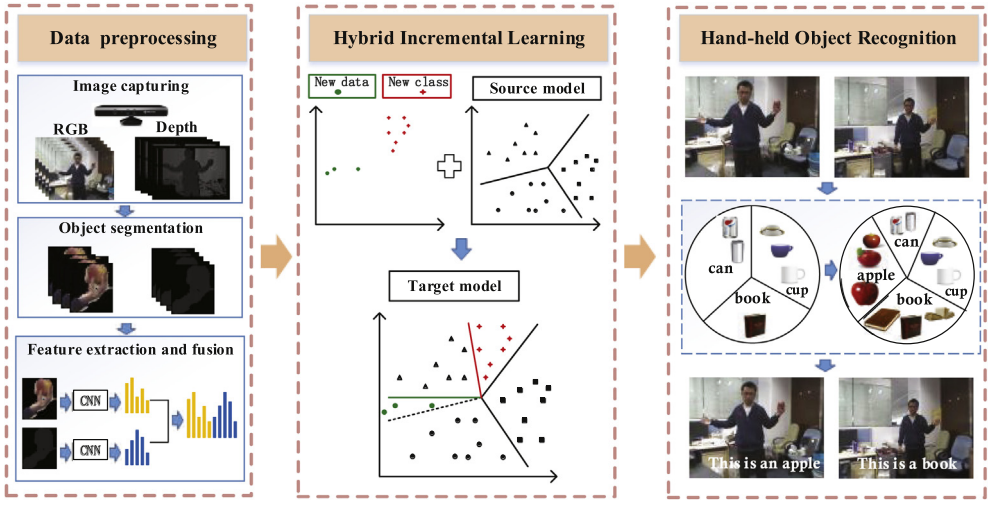

Fig. 2. The procedure of the hybrid incremental learning framework.

2. Related work

Our work is closely related to the following two research areas: (1) object recognition and (2) incremental learning.

2.1. Object recognition

Object recognition is an important and challenging task in multimedia analysis. Its goal is to find a given object in an image or video sequence. The first step of object recognition is to extract discriminative features, which include the following two types: hand-crafted features and deep features. The hand-crafted features, such as SIFT [15], fast point features histogram [16] and ensembles of shape features [17], have been proved to be robust to transformations, e.g, rotation and scale. However, sometimes they can only capture a subset of the cues that are useful for recognition [18]. In contrast, the recent progress in deep learning models, such as Convolutional Neural Networks (CNNs) [19] offer more discriminative features, and have been widely used in various tasks, such as video understanding [20], action recognition [21] attribute learning [22] and recommendation [23,24]. They can learn higher order features that describe the semantics of images. Therefore, the deep learning methods have been widely used for feature extraction in object recognition [1]. For example, Lv et al. [1] exploited deep features that are separately extracted from the RGB image and the depth image for hand-held object recognition. Different from them, we apply the deep features into incremental learning for hand-held object recognition.

Besides extracting discriminative features, object recognition includes a broad range of decision-theoretic classification algorithms for image recognition, such as the decision tree [25,26] and SVM [27]. For example, Friedl et al. [25] used the the decision tree for remote sensing image classification. Doan et al. [27] used a linear SVM classifier trained in a one-versus-rest manner to deal with large-scale image datasets. Different from their work, we extend SVM to enable hybrid incremental learning for hand-held object recognition.

As a special case in object recognition, hand-held object recognition has been paid more and more attentions recently [2,1]. In this application, the additional depth information and the skeletal information from the RGB-D devices (e.g., Kinect) can provide an important clue to characterize the visual object [28], and thus improve the performance of the hand-held object recognition [1,2]. For example, Ren et al.[3] developed a bottom-up motion-based approach to robustly segment out foreground objects for hand-held object recognition. Our work is different from them in that we pay more attention to the self growth of the system to improve the interaction performance and propose a hybrid incremental learning method in hand-held object recognition.

2.2. Incremental learning

In many applications, such as hand-held object recognition in the intelligence technology, new samples of existing classes and new classes become available all the time [4]. Acquisition of sufficient representative training data is expensive and time consuming. Fixed models for the intelligence technology are unable to cope with the changes of the dynamic environments. Therefore, it is necessary to update the existing classifier in an incremental fashion to accommodate the new data without compromising classification performance on the old data, namely incremental learning. According to the type of new data, incremental learning is divided into three types: data-incremental learning, class-incremental learning and hybrid-incremental learning.

The data-incremental learning aims to utilize newly available training samples to improve the recognition quality of known objects [9,8]. For example, Ruping et al. [9] proposed a SVM-based framework for data-incremental learning. Wen et al. [29] presented a tracking method for the significant variation of the object's appearance or surrounding illumination, which is based on incremental algorithms for PCA. This work is suitable for visual tracking, which differs from our method on the classification scenario. Jose et al. [30] proposed a visual data classification methodology that supports evolving data sets. It based on the boosting method, and the difference is that our method is under the setting of SVM. Different from data-incremental learning, class-incremental learning is to recognize new objects with the emergence of unseen concepts. For example, Ristin et al. [12] proposed different variants of random forest to incorporate new classes for class-incremental learning. However, as pointed out in[14], a good incremental learning algorithm should be able to both learn additional information from new data and accommodate new classes that may be introduced with new data. Therefore, there should be another kind of incremental learning method, namely hybrid incremental learning. Like Learn++, we also developed a hybrid incremental learning method based on the SVM and applied our method into hand-held object recognition. Learn++ method [14] claimed that it can accommodate new data including previously unseen classes. However, there are some differences between our method HIL and Learn++. Firstly, the method of Learn++ is based on adaboosting while our method is based on SVM. Secondly, for Learn ++, the number of the base model increases with the incremental learning. Our method always has one base model. Thirdly, these two methods do not require access the original data. However, for class-incremental learning, Learn++ needs the partial data of every learned class besides the new class as the training data for a better performance. Our method HIL does not need this.

In addition, some incremental methods learn the new information based on the transfer learning and domain adaption [31–34]. For example, Qi et al. [31] developed an efficient online learner to adapt the existing model with the new one by minimizing their model distance under a set of multilabel constraints. This work is a related work solves the problem of class-incremental learning. They [32] then proposed a new algorithm which mined both context and content links in social media networks to discover the underlying latent semantic space and this work sheds some light on automatic learning from social media networks rather than human teachers.

Recent progress on deep learning also promotes neural network based incremental learning. For example, Xiao et al. [35] developed a training algorithm that grows a network incrementally and hierarchically. In this work, classes are grouped according to similarities, and self-organized into levels and the learning process is guided by a pragmatic, error-driven preview process. Pratama et al. [36] developed a novel Evolving Type-2 Recurrent Fuzzy Neural Network (eT2RFNN), which can initiate its learning process from scratch with an empty rule base and automatically add, prune, merge its fuzzy rules from data streams afterwards. The different incremental tuning strategies employing CNNs, can be conveniently framed in three main strategies [37]. Comparing with these strategies for incremental learning on the neural network, we adopted SVM based method for hybrid incremental learning.

Incremental learning can also be divided into the following three types according to the type of models: (1) Learning new information by growing or pruning of classifier architectures [38]. For example, Ristin et al. [12] combined the proposed nearest class mean forest classifier and random forest to learn new information. This algorithm used the hierarchical concepts to make the model be adapted into new data. (2) Learning new information by selecting the most informative training samples. For example,

Engelbrecht and Brits [39] first clustered the candidate training set and then selected the most informative pattern of each cluster at each subset selection interval. Learn++ [14] utilized the adaboosting method to learn new information. 3) learning new information by modifying the classifier weights [40–42]. For example, Kuzborskij et al. [10] added a new classification-plane to the source model to learn a new concept. Cauwenberghs et al. [6] proposed a solution to adjust the existing classification-planes by trying to make all the previous seen data subject to Kuhn-Tucker conditions.Our proposed HIL belongs to the third category and uses SVM based method to modify classifier weights to learn new information.

3. Hybrid incremental learning

In this section, we first briefly review SVM [43] and Lagrange multipliers, and then propose a hybrid incremental learning method under the setting of SVM and parameter learning algorithm. We use lowercase letters for vectors and capital letters for matrices. $A_{k,n}$ is the (k,n) entry of matrix A. b_k is the k-th element of vector b. (x_n, y_n) denotes the visual feature vector $\mathbf{x}$ and the corresponding label y for the n-th sample.N represents the number of training samples.

3.1. Preliminaries

SVM solves the quadratic programming problem by minimizing the following objective function.

$$\min J = Complexity + Risk_{expected}$$

For the multi-class Least Squares Support Vector Machine (LSSVM) [44], it can be turned into a problem of solving linear equations by transferring the inequality constraints to equality constraints, which greatly facilitates the solution of Lagrangian multiplier method. We can minimize the following objective function with respect to W, b :

$$\min_{W,b,e} J(W,b) = \frac{1}{2}\|W\|_F^2 + \frac{1}{2}C\sum_{n=1}^{N}\sum_{m=1}^{M}(e_{mn})^2 \tag{1}$$

s.t.

$$y_n(w_m^T\varphi(x_n) + b_m) = 1 - e_{mn},\ n = 1,\ldots\ldots,N,\ m = 1,\ldots\ldots,M$$

where $W = [w_1, w_2, \cdots, w_M]$ and each column w_m represents the hyperplane that separates one of the M classes from the rest. $\varphi(x_n)$ is a kernel function, e_{mn} is the slack variable. C is the tradeoff between expected risk and the complexity. $\|\cdot\|_F$ is the Frobenius norm.

In mathematical optimization, the method of Lagrange multipliers [45] is a strategy for finding the local maxima and minima of a function subject to equality constraints. For the case of only one constraint and only two choice variables, consider the optimization problem:

$$\min f(x,y)$$

s.t.

$$g(x,y) = c$$

We assume that both f and g have continuous first partial derivatives. We introduce a new variable α called a Lagrange multiplier and study the Lagrange function defined by:

$$L(x,y,\alpha) = f(x,y) - \alpha[g(x,y) - c]$$

where the α term may be either added or subtracted.

3.2. Our proposed method

Two aspects of hybrid incremental learning should be considered. Firstly, for newly available data, we try to utilize previous support vectors to adjust current classification-planes $[w_1^{t-1}, w_2^{t-1}, \cdots, w_M^{t-1}]$ so they can describe all previous trained data at step t. Secondly, when there are new classes, we try to find a new classification-plane w_{M+1} to learn the unknown class. That is, at step t, find a new group of classification-planes $W^t = [w_1^t, w_2^t, \cdots, w_M^t]$ and w_{M+1}^t which is close to source classification-planes to make the M-class source classifier transfer to a $(M+1)$-class target classifier. In addition, we should modify the source classification-planes $W^{t-1} = [w_1^{t-1}, w_2^{t-1}, \cdots, w_M^{t-1}]$ to preserve the performance on known concepts.

Based on the above two considerations, we minimize the following objective function to realize the hybrid incremental learning when the new data is available at step t:

$$\min_{W^t,b^t,e^t} J(W^t,b^t) = \frac{1}{2}\left\|W^t - W^{t-1}\right\|_F^2 + \frac{1}{2}\left\|w_{M+1}^t - \beta^t W^{t-1}\right\|_F^2 + \frac{C^t}{2}\sum_{m=1}^{M+1}\Big(\sum_{i\in I^t}(e_{mi}^t)^2 + L^t\sum_{j\in S^{t-1}}(e_{mj}^t)^2\Big) \tag{2}$$

s.t.

where W^t means the new classification planes to learn and each column w_{M+1} represents the new hyperplane that separates one of the $M+1$ classes from the rest learned classes. W^{t-1} denotes the classification planes learned last time. $\varphi(x_n)$ is a kernel function, e_{mn} is one slack variable. b is the bias an C is the tradeoff between expected risk and the complexity. $\|\cdot\|_F$ is the Frobenius norm. I^t denotes the new data at step t, and S^{t-1} denotes the old support vectors at step t-1. β is a vector to describe the linear combination of existed classification-planes.

(1) The first two terms are regular terms which basically guarantees that the new information will not influence the learned information and the structure risk is minimal.

(2) The first term $\frac{1}{2}\left\|W^t - W^{t-1}\right\|_F^2$ aims at preserving previous knowledge and this term forces the target model to keep close to the source model.

(3) The second term $\frac{1}{2}\left\|w_{M+1}^t - W^{t-1}\beta\right\|_F^2$ incorporates new concepts into the current model. It enforces the added classification-plane to the linear combination of existed classification-planes, which is described by a vector β. This second term guarantees that our method can handle one new class at one time.

(4) The last two terms $\frac{C^t}{2}\sum_{m=1}^{M+1}(\sum_{i\in I^t}(e_{mi}^t)^2 + L\sum_{j\in S^{t-1}}(e_{mj}^t)^2)$ define the loss. The first one is to minimize the prediction error of the new information and the second is to minimize the prediction error of support vectors. The weight L makes support vectors not only represent classification-planes but also give a better description to previous data.C^t is the tradeoff.

3.3. Parameter optimization

Lagrangian multipliers α_i^t $(i = 1,\ldots\ldots,N)$ are introduced into optimization:

$$L(W^t,b^t,e^t;\alpha^t) = J - \sum_{m=1}^{M+1}\sum_{i\in I^t}\alpha_{im}^t\left\{y_i\left[(w_m^t)^T\varphi(x_i) + b^t\right] - 1 + e_{im}^t\right\} - \sum_{m=1}^{M+1}\sum_{j\in S^{t-1}}\alpha_{jm}^t\left\{y_j\left[(w_m^t)^T\varphi(x_j) + b_m^t\right] - 1 + e_{jm}^t\right\} \tag{3}$$

We take its partial derivative to W^t, b^t, e^t respectively:

$$W^t - W^{t-1} = \sum_{k\in(I^t\cup S^{t-1})} \alpha_k^t \varphi(x_k),$$

$$\alpha_k^t = [\alpha_{k1}^t, \alpha_{k2}^t, \cdots, \alpha_{km}^t]^T$$

$$\sum_{k\in(I^t\cup S^{t-1})} \alpha_k^t = \mathbf{0}$$

$$C^t \sum_{m=1}^{M+1}\left(\sum_{i\in I^t} e_{mi}^t + L^t \sum_{j\in S^{t-1}} e_{mj}^t\right) + \sum_{m=1}^{M+1}\sum_{i\in(I^t\cup S^{t-1})} \alpha_{im}^t = \mathbf{0}$$

The term $\varphi(X)^T\varphi(X)$ represents kernel matrix. The solution of the optimization is given by:

$$W_m^t = W_m^{t-1} + \sum_{k\in(I\cup S)} A_{k,m}^t \varphi(x_k)\ m = 1, 2, \ldots, M \tag{4}$$

$$w_{m+1}^t = \beta^t W^{t-1} + \sum_{k\in(I\cup S)} A_{k,M+1}^t \varphi(x_k) \tag{5}$$

$$b^t = (b^t)' - \left[(b^t)''(b^t)''^T \beta^t\right] \tag{6}$$

where:

$$A^t = (A^t)' - \left[(A^t)''(A^t)''\beta^t\right]$$

$$\begin{bmatrix}(A^t)'\\(b^t)'^T\end{bmatrix} = M^t\begin{bmatrix}Y\\0\end{bmatrix}$$

$$\begin{bmatrix}(A^t)''\\(b^t)''^T\end{bmatrix} = M^t\begin{bmatrix}\varphi(X)^T W^{t-1}\\0\end{bmatrix}$$

$$M^t = \begin{bmatrix}\varphi(X)^T\varphi(X) + \begin{bmatrix}\frac{1}{C^t} & \\ & \frac{1}{C^t L^t}\end{bmatrix} & \mathbf{1}\\ \mathbf{1}^T & \mathbf{0}\end{bmatrix}$$

$A_{k,m}^t$ is the (k, m) entry of matrix A^t and Y is the encoded One-versus-All (OVA) label matrix, where each label code is a column.

4. Hand-held object recognition

After the model training, we apply the proposed HIL method into hand-held object recognition. Compared with general object recognition, hand-held object recognition can exploit prior knowledge, take advantage of RGB-D devices, and use specific segmentation techniques to find hand-held objects. We also take advantage of these additional depth and skeletal information from RGB-D devices in our method. Specifically, we collect the object label and RGB-D information automatically by gathering user's voice and capturing object images when the object in user's hand is unidentified during interaction. We then follow the object segmentation method [1] and fuse the CNN features from RGB and depth features. After that, our hybrid incremental learning method learns the new data.

In order to achieve incremental learning, our hybrid incremental learning method can adjust existing classification-planes slightly to accommodate new information while preserve learned knowledge. The method makes use of the support vectors and gives them an appropriate weight to approximately replace all learned data so that it does not need the original data (such as the book class in Fig. 2). For unknown hand-held objects (such as the apple class in Fig. 2), our method learns a new classification-plane for the model to identify the new classes. Based on these considerations, our framework can handle both data-incremental learning and class-incremental learning.

5. Evaluation

In this section, we firstly describe the experimental setting including the dataset, implementation details and evaluation protocols and metrics. We then evaluate the performance of the proposed hybrid learning method in hand-held object recognition. Finally, we verify the effectiveness of our proposed method in general object recognition.

5.1. Dataset

5.1.1. HOD-20

HOD [1] dataset is designed for hand-held object recognition. We use 20 classes in our experiment: "apple", "ball", "bat", "book", "bottle", "box", "calculator", "can", "cup", "dish", "disk", "fan", "glove", "handbag", "hat", "keyboard", "medicine box", "tissue", "towel" and "trashcan" and name it HOD-20. HOD-20 contains a total of 16,000 images and 20 common object categories. There are 800 RGB and depth image pairs for each class. Fig. 3 shows some sample images. For every class, we randomly select 480 RGB and depth image pairs for training and the remaining 320 RGB and depth image pairs for testing. For a random sequence in class-incremental learning (Fig. 4(a)), we split the 20 classes into two groups: the former 10 classes are used for training source model, the other 10 classes for class-incremental learning. In data-incremental learning (Fig. 4(b)), for each class, 80 images are used for training a source model and 400 images for data-incremental learning. We also do experiments in a balance data partition and unbalance data partition way to obtain the difference in performance.

5.1.2. ImageNet-30

This data set is a subset of ImageNet [46], which contains about 3.2 million images in total with 1,000 categories. We randomly choose 30 common classes, including: "kite", "hammer", "water snake", "Eskimo dog", "umbrella", "table lamp", "maillot", "strawberry", "shopping basket", "tree frog", "keyboard", "traffic light", "jean", "honeycomb", "park bench", "mouse", "violin", "cheeseburger", "pillow", "sunglass", "ski mask", "coffee mug", "white shark", "running shoe", "school bus", "laptop", "lifeboat", "ashcan", "electric fan" and "wall clock", namely ImageNet-30. There are 1300 images per class and 39,000 images in total. Fig. 5 shows some sample images in ImageNet-30. For every class, we randomly select 700 images for training and the remaining 600 images for testing. Similar to Fig. 4, for a random sequence in

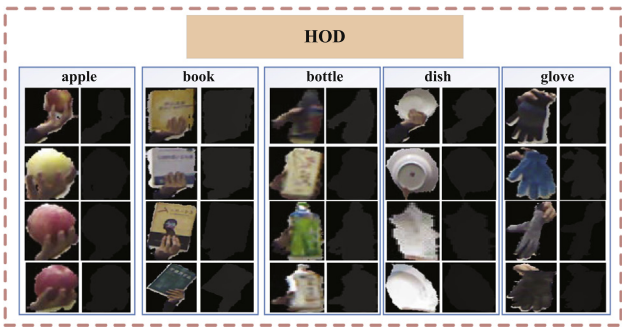

Fig. 3. Sample images in HOD-20.

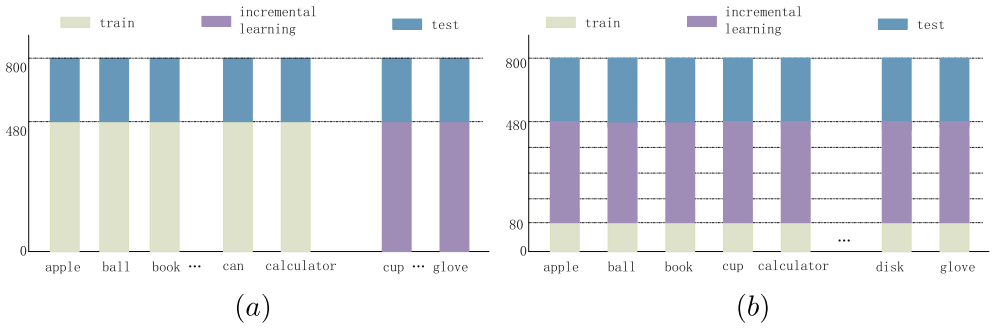

Fig. 4. Data partitioning for HOD-20: (a): class-incremental learning (b): data-incremental learning.

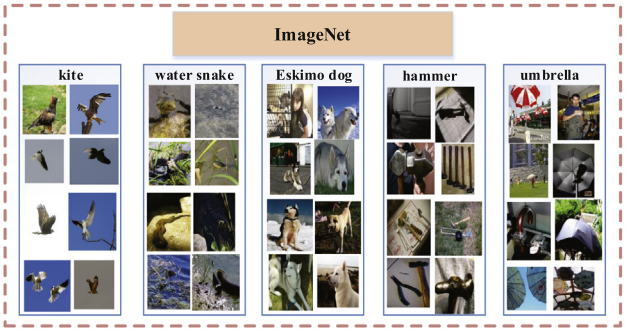

Fig. 5. Sample images in ImageNet-30.

class-incremental learning., we split these 30 classes into two parts: the former 10 classes are used for training and the other 20 classes for incremental learning. In data-incremental learning, for each class, 100 images are used for training a source model and 600 images for data-incremental learning.

5.2. Implementation Details

For the model parameters in our objective function, which includes C^t, L^t, β^t , and γ^t in the RBF kernel. We use 5-fold cross validation to find the best parameter. In HOD-20 and ImageNet-30, the range of C^t both is $\left[10^{-4}, 10^{5}\right]$, and they both use a RBF kernel, the difference is the range of parameter gamma. The range is $\left[2^{-20}, 2^{-15}\right]$ in HOD-20, while $\left[2^{-15}, 2^{-10}\right]$ in ImageNet-30. L^t is a weight assigned to support vectors so that support vectors can approximately respect all previous data. The value of L^t is related to the size of all previous data and new data, and is calculated from the following equation [9]:

$$L^t = 4 * \frac{Number\ of\ previous\ seen\ data\ until\ step\ t-1}{Number\ of\ new\ data\ at\ step\ t}$$

5.3. Evaluation protocols and metrics

The main lines of hand-object recognition framework consist of object segmentation, feature extraction, feature fusion and incremental learning. For object segmentation, we use the hand position located from Kinect API as reference and initial seed first, then we obtain the object mask using a region-growing algorithms [1]. Finally, we use the seed set to get the object region and its corresponding depth region. Feature extraction is to extract the CNN features from both the segmented RGB images and the depth images using AlexNet [47]. The architecture has eight layers. The first five layers are convolutional layers, while the sixth and the seventh layers are fully connected layers and the final layer is a softmax classifier. We discard the classifier and use the output of the seventh layer as a feature (4096-D). Then we cascade RGB features and depth features as the final 8192-D features. For incremental learning, we load a trained model as a source model, and use the new training data to learn new knowledge, transfer the source model to a target one. For data-incremental learning, the training data is new samples of learned classes, for class-incremental, the training data is a new class. For hybrid incremental learning, the training data includes new samples and new classes. In the test stage, for data-incremental learning, we care about whether the performance is improved with new samples or not. For class-incremental learning, we focus on whether the new class is learned successfully or not. For hybrid incremental learning, we pay attention to these two aspects, which can be reflected through the accuracy of the model during incremental learning.

For the process of recognition based on the imagenet. We directly extracted 4096-D deep features from RGB images without segmentation. The following training and test process is similar to the hand-object recognition framework.

We use the accuracy of the model as the metric to evaluate our method.

5.4. Experimental baselines

For the reason that our method is an extension of MULTIpLE [10] and SV-L-Inc. [9]. Therefore, we compare our models with these two baselines. In addition, we introduce corresponding offline HIL method:

(1) MULTIpLE [10]. This method is a class-incremental learning method. Every step the model learns a new classification-plane for the new class. When this model learns a new class, it needs additional training data from learned classes. Therefore the model may suffer from the data skew. However, our model don't need additional data. We use support vectors instead, which are parameters of SVM model. Besides, we define a global loss on both new data and the original data while only on the new data for MULTIpLE.
(2) SV-L-Inc. [9]. This method is a data-incremental learning method. The difference between SV-L-Inc. and ours is that the regularize in SV-L-Inc. only controlled the complexity of the model while ours not only controls the complexity of the model, but also better adjusts the classification-planes to guarantee the classification performance on both old data and new data.
(3) HIL-Offline. This baseline is the SVM retraining method, and the parameter setting is the same as the HIL method. However, this baseline is offline and not incremental.

Table 1
Qualitative comparison between baselines and our method.

	Class-incremental	Data-incremental	No aoriginal data
MULTIpLE [10]	✔		
SV-L-Inc.[9]		✔	✔
HIL-Offline	✔	✔	
HIL	✔	✔	✔

Different from our method, these two baselines only focused on one aspect of incremental learning while our method can handle two sides simultaneously, which is more comprehensive. Table 1 gives the comparison between the baselines and our method in details. Accuracy is used to evaluate the classification performance.

In all these baselines and our HIL method, we train model use a partial classes but test every model on all classes, in this way we can obviously see if the model learn new class successfully. And this way is also more similar to real environment in which the model will encounter some unknown objects.

5.5. Experimental results

5.5.1. HOD-20

Fig. 6 shows the results of different incremental learning methods in class-incremental way. For each incremental learning, we introduced a new class for class-incremental learning. From Fig. 6, we can see that (1) The validation on test dataset shows steadily increasing performance, indicating the algorithm is able to learn the new classes, successfully. (2) The accuracy of our online system HIL is higher than MULTIpLE. The reason is that MULTIpLE only considers the loss on new data while we consider the loss on new data and all previous learned data in a global optimum. Note that the larger improvements in the performance are obtained since the training sessions introduced new classes that were not available earlier. (3) The accuracy of our HIL system is slight lower than HIL-Offline. The possible reason is that incremental learning method learns new information based on previous model without retraining all previous images. However, our system adjusts what has been learned according to new examples instead of learning a new model, and thus the training time of incremental learning method is much shorter than HIL-Offline.

We further analyze the experimental results in more details in Fig. 6. These models learns one new concept each step. Every adjacent two points can be regarded as one process of class-incremental learning. The former point is the source model and the latter is the target model. For example, in our HIL method, the point corresponding to 10 classes is a 10-class source model with a total accuracy of 0.49016. After learning a new concept "cup", the source model turns into a 11-class target model with a total accuracy of 0.53. There are three cases in Fig. 7 for the qualitative analysis. The cup samples are not included in the 10 source

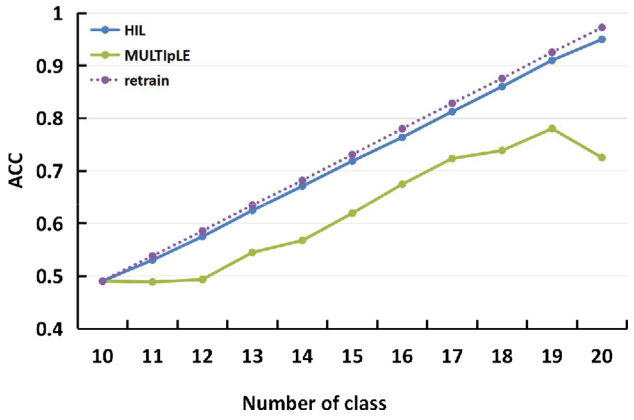

Fig. 6. Performance comparison of class-incremental learning methods in HOD-20.

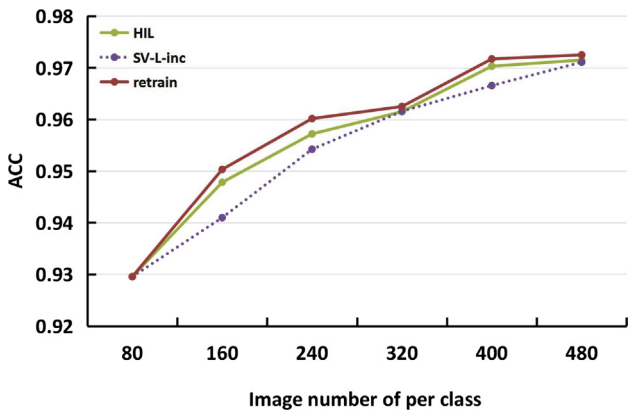

Fig. 8. Performance comparison of data-incremental learning methods in HOD-20.

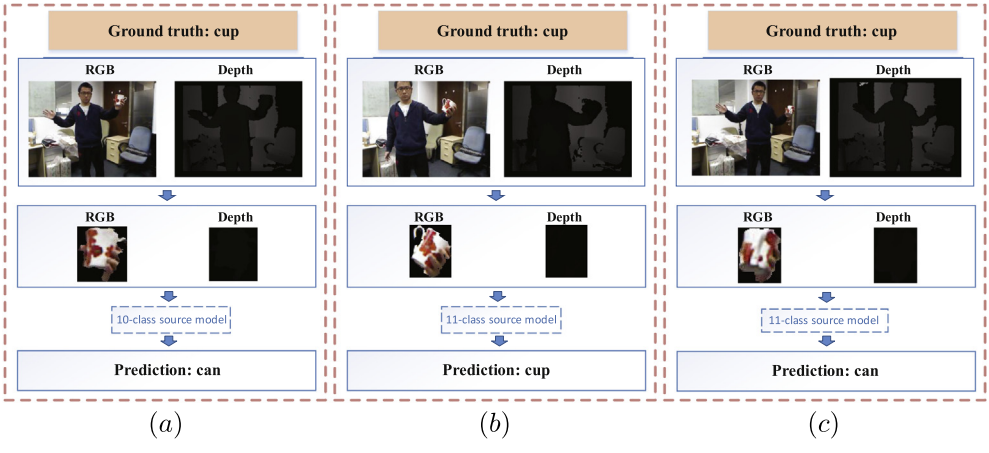

Fig. 7. Three cases of HIL for qualitative analysis in the HOD-20 dataset. (a) Cup is not included in the 10 source classes, therefore the source model can't recognize cup. (b) After learning a new concept "cup", the source model turns into a 11-class model, and it can recognize cup. (c) The handle of cup is not clear in this image, and the model recognize the image as a can.

classes, therefore the source model can't recognize the cup class (in Fig. 7(a)). After learning a new concept "cup", the source model turns into a 11-class model, and it can recognize cup (in Fig. 7(b)). In Fig. 7(c), the handle of the cup samples is not clear in this image, and thus the model recognize this image as a can. This is because the shape of cup without the handle is similar with can in this angle.

Besides class-incremental learning, we compare our method with other baselines in data-incremental learning. To avoid the influence of data imbalance, we added a fixed amount of new data to every source concept. Particularly, for each step, the model learned 80 test images of every source concepts. After five steps, the amount of data for each source class increases from 80 to 480 images. Fig. 8 shows the results of different methods. (1) The accuracy of our HIL is higher than SV-L-Inc. The reason is that the regularize in SV-L-Inc. only controlled the complexity of the model while ours not only controls the complexity of the model,

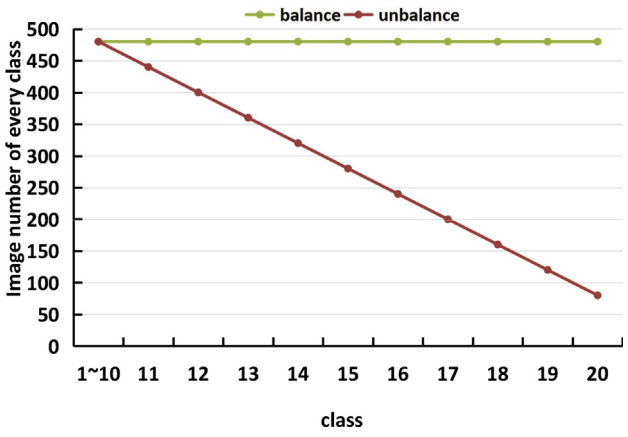

Fig. 9. Data partition comparison of experiments about balance data and unbalance data in HOD-20.

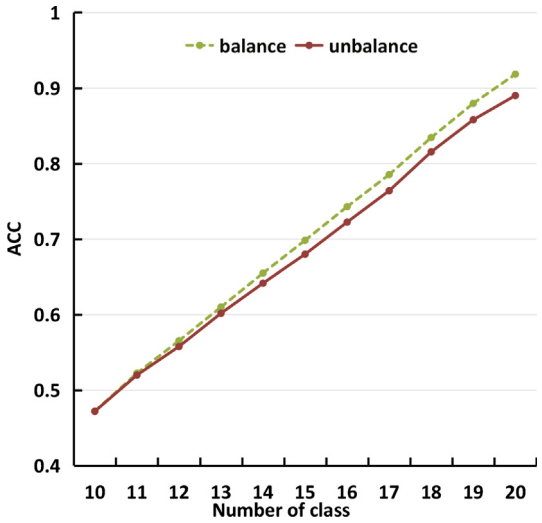

Fig. 10. Performance comparison of experiments about balance data and unbalance data in HOD-20.

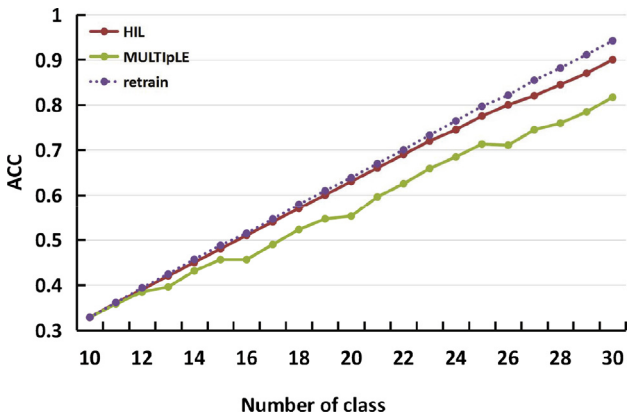

Fig. 11. Performance comparison of class-incremental learning methods in ImageNet-30.

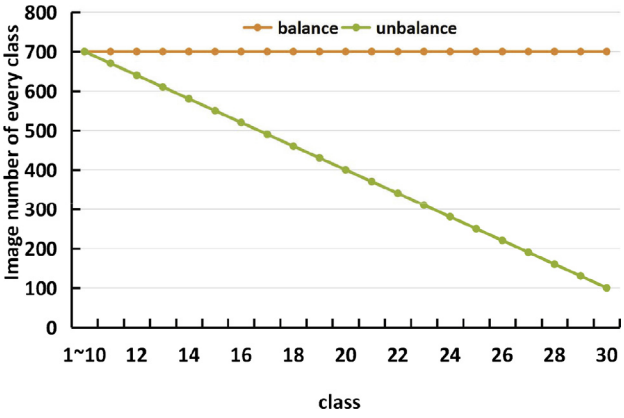

Fig. 12. Data partition comparison of experiments about balance data and unbalance data in ImageNet-30.

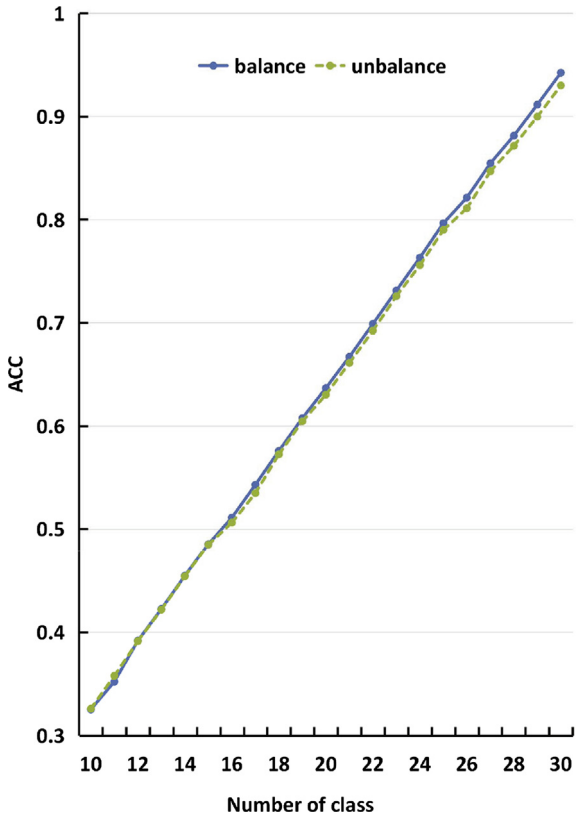

Fig. 13. Performance comparison of experiments about balance data and unbalance data in Image-30.

but also better adjusts the classification-planes to guarantee the classification performance on both old data and new data. (2) The accuracy of our HIL system is comparable with HIL-Offline. The possible reason is that incremental learning method learns new information based on previous model without retraining all previous images. However, our system adjusts what has been learned according to new examples instead of learning a new model, and thus the training time of incremental learning method is much shorter than HIL-Offline. Whether the dataset is balanced or not is one key factor for the performance of incremental learning. In order to verify the robustness of our proposed method, we further analyze the difference in performance of balance data and unbalance data. The data partition is shown in Fig. 9 and the results are shown in Fig. 10. We train a 10-class model as our source model and learn both new class and new samples in a hybrid incremental learning way. For the balance data, each step of hybrid incremental learning will learn a new class $((m-10)*40+80$ images, where m represents the order of this learned class and new samples of all learned class (40 images per class). For the unbalance data, each step of hybrid incremental learning will learn a new class (80 images) and new samples of all learned class (40 images per class).. As shown in (Fig. 10), the results of balance data is slightly better than unbalance data. We can see that data skew influences the performance, but the influence is small.

5.5.2. ImagNet-30

In order to further verify the robustness of our method, we conducted the experiment on another dataset, namely ImageNet-30. Similarly, we firstly compared our method with other baselines in class-incremental fashion, and then data-incremental fashion in a random sequence order. Fig. 11 showed the experimental results of different methods in class-incremental way. Our source model has learned 10 source conceptsand then learned 20 new concepts in class-incremental fashion. From these results, we can observe that: (1) The performance on the test dataset shows steadily increasing performance, indicating the algorithm is able to learn the new classes. (2) The accuracy of our on-line system HIL

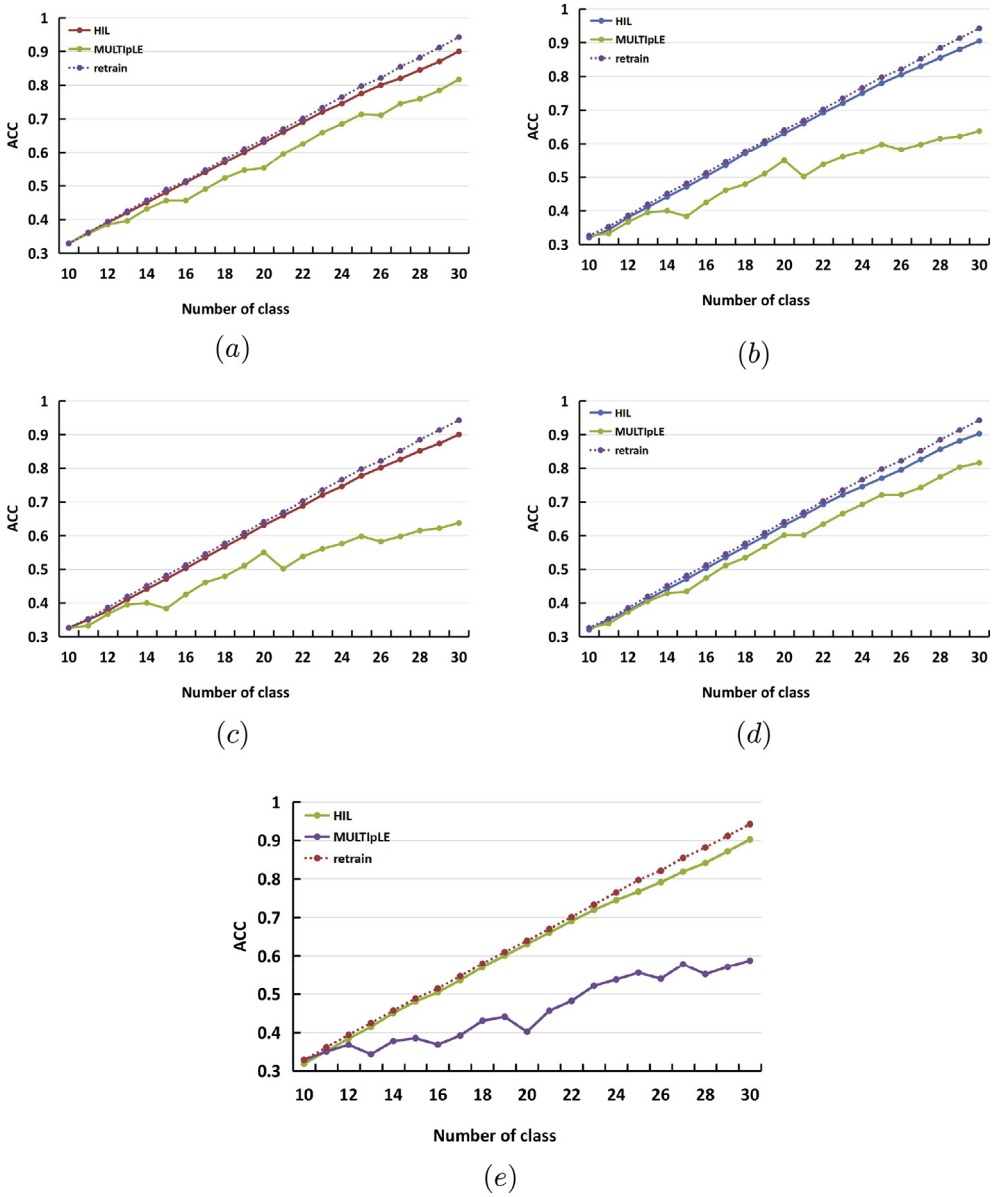

Fig. 14. Five different order of class-incremental learning in the ImageNet-30 dataset.

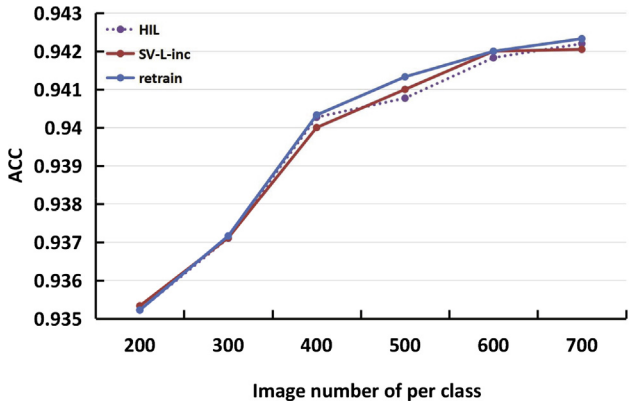

Fig. 15. Performance comparison of data-incremental learning methods in ImageNet-30.

is higher than MULTIpLE. (3) The accuracy of our HIL system is slight lower than HIL-Offline. The performance in this dataset is consistent with HOD-20.

Similarly, we also analyze the difference in performance of balance data and unbalance data The data partition is shown in Fig. 12 and the results are shown in Fig. 13 We train a 10-class model as our source model and learn both new class and new samples in a hybrid incremental learning way. For the balance data, each step of hybrid incremental learning will learn a new class $((m-10)*30+100$ images, where m represents the order of this learned class and new samples of all learned class (30 images per class). For the unbalance data, each step of hybrid incremental learning will learn a new class (100 images) and new samples of all learned class (30 images per class).. As shown in (Fig. 13), the results of balance data is slightly better than unbalance data. We can see that data skew influences the performance, but the influence is small.

To analyze the influence of different learning sequences, we conducted the class incremental learning based on multiple different training order in the ImageNet. We implemented the incremental learning process in 5 different random sequences. For random each sequence, we used the first 10 classes to train a source model and the other 20 classes as the data for incremental learning. Fig. 14 shows the results of 5 randomly selected training orders in ImageNet-30. The experimental results have showed that the difference in the performance is small in our HIL method.

Fig. 15 shows the results of different methods in data-incremental learning way. To avoid the influence of data imbalance, we added a fixed amount of new data to every source concept. Particularly, for each step, the model learned 100 test images of every source concepts. After five steps, the amount of data for each source class increases from 200 to 700 images.From Fig. 15, we can see that (1) HIL-Offline (retrain) achieves better performance compared with other two methods, since they utilized all the information to retrain the model when added new classes. The time cost is very expensive. (2) The accuracy of our HIL and SV-L-Inc. methods are comparable with HIL-Offline (retrain), when the number of images per class is above 500. This verified that our proposed hybrid increasing learning method is capable of utilizing limiting information for object recognition. (3) Compared with the SV-L-Inc., which can not learn new class, our method can deal with new class and new samples simultaneously in a hybrid way.

6. Conclusions

In this work, we proposed a hybrid incremental learning method for hand-held object recognition, which combined the class-incremental learning and data-incremental learning in a unified fashion. Our method is capable of finding a balance between stability and plasticity to guarantee that our framework could learn new information without forgetting previously acquired knowledge. The framework added new classification-planes to learn unknown concepts, and adjusted all source classification-planes to improve recognition ability of known concepts. The experimental results demonstrate the effectiveness of our model in hybrid learning.

This work is an effort to human-computer interaction. We have a long way to go because of the organization of learned information. Human could learn much more than isolated concepts according to these additional relational and attribute information. Human computer interaction systems should also take these information into consideration for a more comprehensive study. In addition, Gobet et al. [48] proposed that the configuration of smaller units of information into large coordinated units might be important in many processes of perception, learning and cognition in humans. Therefore, for intelligent human-machine interaction systems, a reasonable learning mechanism is to automatically learn new information incrementally and make associations between concepts automatically. Incremental learning on multimodal also can be a task that worth to study, for the reason that multimodal information is a good way for the intelligence technology to obtain a more complete and accurate perception of environment. For the recent progress on deep learning, how to realize a real incremental learning method on neural network is also a valuable task. Incremental learning also provide a way to carefully control the growth of the network capacity, which is also meaningful to the study of deep learning.

This work can be extended in the following two directions: (1) In order to make the performance of HIL comparable or even better than HIL-Offline, a possible way is to develop an end-to-end framework for simultaneous CNN based feature learning and classification. Such framework will make the learned classifier more discriminative, since it can benefit from the learned discriminative visual features. (2) We will extend our hybrid framework to more recognition tasks with more classes. (3) There are various implicit or explicit relations among concepts in the real world and various attribute information. (4) Because of the second term in our objective function, which can only handle one new class at one time, our method can not learn multiple class at once, we will further study our objective function to solve this problem.We will incorporate these concept relations and their attribute information into this framework to make human–machine interaction systems more intelligent.

Conflict of interest

We wish to confirm that there are no known conflicts of interest associated with this publication.

Acknowledgements

This work was supported in part by the Beijing Natural Science Foundation (4174106), in part by the National Natural Science Foundation of China under Grant (61532018 and 61602437), in part by the Lenovo Outstanding Young Scientists Program, in part by National Program for Special Support of Eminent Professionals and National Program for Support of Top-notch Young Professionals, in part by the China Postdoctoral Science Foundation (2017T100110) and in part by the State Key Laboratory of Robotics.

References

[1] X. Lv, S. Jiang, L. Herranz, S. Wang, Rgb-d hand-held object recognition based on heterogeneous feature fusion, J. Comput. Sci. Technol. 30 (2) (2015) 340–352.

[2] X. Lv, S. Jiang, L. Herranz, S. Wang, Hand-object sense: a hand-held object recognition system based on rgb-d information, in: Proceedings of ACM International Conference on Multimedia, 2015, pp. 765–766.

[3] X. Ren, C. Gu, Figure-ground segmentation improves handled object recognition in egocentric video, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3137–3144.

[4] T. Mensink, J.J. Verbeek, F. Perronnin, G. Csurka, Distance-based image classification: generalizing to new classes at near-zero cost, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2624–2637.

[5] A. Bordes, S. Ertekin, J. Weston, L. Bottou, Fast kernel classifiers with online and active learning, J. Mach. Learn. Res. 6 (2005) 1579–1619.

[6] G. Cauwenberghs, T.A. Poggio, Incremental and decremental support vector machine learning, in: Conference on Neural Information Processing Systems, 2000, pp. 409–415.

[7] N. Littlestone, P.M. Long, M.K. Warmuth, On-line learning of linear functions, in: Proceedings of the Twenty-third Annual ACM Symposium on Theory of Computing, 1991, pp. 465–475.

[8] N. Murata, M. Kawanabe, A. Ziehe, K.-R. Müller, S. ichi Amari, On-line learning in changing environments with applications in supervised and unsupervised learning, vol. 15, 2002, pp. 743–760.

[9] S. Ruping, Incremental learning with support vector machines, in: Proceedings IEEE International Conference on Data Mining, 2001, pp. 641–642.

[10] I. Kuzborskij, F. Orabona, B. Caputo, From n to n+1: multiclass transfer incremental learning, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3358–3365.

[11] C. Lampert, H. Nickisch, S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 951–958.

[12] M. Ristin, M. Guillaumin, J. Gall, L. Van Gool, Incremental learning of random forests for large-scale image classification, IEEE Trans. Pattern Anal. Mach. Intell. (99) (2015) 1.

[13] S. Grossberg, Nonlinear neural networks: principles, mechanisms, and architectures, Neural Netw. 1 (1) (1988) 17–61.

[14] R. Polikar, L. Upda, S.S. Upda, V. Honavar, Learn++: an incremental learning algorithm for supervised neural networks, IEEE Trans. Syst. Man Cybernet. Part C 31 (4) (2001) 497–508.

[15] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[16] B. Morisset, R.B. Rusu, A. Sundaresan, K.K. Hauser, M. Agrawal, J.-C. Latombe, M. Beetz, Leaving flatland: toward real-time 3D navigation, in: Proceedings IEEE Conference on Robotics and Automation, 2009, pp. 3786–3793.

[17] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, V. Lepetit, Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes, in: IEEE International Conference on Computer Vision, 2011, pp. 858–865.

[18] A. Wang, J. Lu, J. Cai, T.-J. Cham, G. Wang, Large-margin multi-modal deep learning for RGB-D object recognition, IEEE Trans. Multimedia 17 (11) (2015) 1887–1898.

[19] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R. Howard, W. Hubbard, L. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Comput. 1 (1989) 541–551.

[20] P. Jing, Y. Su, L. Nie, X. Bai, J. Liu, M. Wang, Low-rank multi-view embedding learning for micro-video popularity prediction, IEEE Trans. Knowl. Data Eng. 30 (8) (2018) 1519–1532.

[21] W. Li, W. Nie, Y. Su, Human action recognition based on selected spatio-temporal features via bidirectional, IEEE Access 6 (2018) 44211–44220.

[22] P. Jing, Y. Su, L. Nie, H. Gu, Predicting image memorability through adaptive transfer learning from external sources, IEEE Trans. Multimedia 19 (5) (2017) 1050–1062.

[23] Z. Cheng, Y. Ding, X. He, L. Zhu, X. Song, M.S. Kankanhalli, A 3ncf: an adaptive aspect attention model for rating prediction, in: IJCAI, 2018, pp. 3748–3754.

[24] Z. Cheng, J. Shen, On effective location-aware music recommendation, ACM Trans. Inform. Syst. 34 (2) (2016) 13.

[25] M.A. Friedl, C.E. Brodley, Decision tree classification of land-cover from remotely-sensed data, Rem. Sens. Environ. 61 (3) (1997) 399–409.

[26] P. Rajendran, M. Madheswaran, Hybrid medical image classification using association rule mining with decision tree algorithm, Computing Research Repository abs/1001.3503.

[27] T.-N. Doan, F. Poulet, Large scale image classification. fast feature extraction, multi-codebook approach and multi-core SVM training, in: Advances in Knowledge Discovery and Management, 2014, pp. 155–172.

[28] J. Tang, L. Jin, Z. Li, S. Gao, RGB-D object recognition via incorporating latent data structure and prior knowledge, IEEE Trans. Multimedia 17 (11) (2015) 1899–1908.

[29] J. Wen, X. Li, X. Gao, D. Tao, Incremental learning of weighted tensor subspace for visual tracking, in: Proceedings IEEE International Conference on Systems, Man and Cybernetics, 2009, pp. 3688–3693.

[30] J.G. Paiva, W.R. Schwartz, H. Pedrini, R. Minghim, An approach to supporting incremental visual data classification, IEEE Trans. Visual. Comput. Graph 21 (1) (2015) 4–17.

[31] G.-J. Qi, X.-S. Hua, Y.R.J. Tang, H.-J. Zhang, Two-dimensional multilabel active learning with an efficient online adaptation model for image classification, IEEE Trans. Pattern Anal. Mach. Intell. 31 (10) (2009) 1880–1897.

[32] G.-J. Qi, C.C. Aggarwal, Q. Tian, H. Ji, T.S. Huang, Exploring context and content links in social media: a latent space method, IEEE Trans. Pattern Anal. Mach. Intell. 34 (5) (2012) 850–862.

[33] G.-J. Qi, W. Liu, C.C. Aggarwal, T.S. Huang, Joint intermodal and intramodal label transfers for extremely rare or unseen classes, Comput. Res. Repository abs/1703.07519.

[34] G.-J. Qi, C.C. Aggarwal, Y. Rui, Q. Tian, S. Chang, T.S. Huang, Towards cross-category knowledge propagation for learning visual concepts, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2011, pp. 897–904.

[35] T. Xiao, J. Zhang, K. Yang, Y. Peng, Z. Zhang, Error-driven incremental learning in deep convolutional neural network for large-scale image classification, in: ACM International Conference on Multimedia, 2014, pp. 177–186.

[36] M. Pratama, J. Lu, E. Lughofer, G. Zhang, M.J. Er, Incremental learning of concept drift using evolving type-2 recurrent fuzzy neural network, IEEE Trans. Fuzzy Syst. 17 (5) (2016) 1.

[37] V. Lomonaco, D. Maltoni, Comparing incremental learning strategies for convolutional neural networks, Artificial Neural Networks in Pattern Recognition, vol. 9896, 2016, pp. 175–184.

[38] E.H.-C. Wang, A. Kuh, A smart algorithm for incremental learning, in: Proceedings IEEE International Joint Conference on Neural Networks, vol. III, 1992, pp. III-121–III-126.

[39] A. Engelbrecht, R. Brits, A clustering approach to incremental learning for feedforward neural networks, IEEE International Joint Conference on Neural Networks, vol. 3, 2001, pp. 2019–2024.

[40] Grippo, Convergent on-line algorithms for supervised learning in neural networks, IEEE Transactions on Neural Networks 11.

[41] M.T. Vo, Incremental learning using the time delay neural network, in: Proceedings International Conference on Acoustics, Speech and Signal Processing, 1994, pp. II-629–II-632.

[42] B.-T. Zhang, An incremental learning algorithm that optimizes network size and sample sum in one trial, IEEE International Conference on Neural Networks, vol. I, 1994, pp. 215–220.

[43] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, Inc., New York, 1995.

[44] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Process. Lett. 9 (3) (1999) 293–300.

[45] R.T. Rockafellar, Lagrange multipliers and optimality, Soc. Ind. Appl. Math. Rev. 35 (1993) 183–238.

[46] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[47] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Conference on Neural Information Processing Systems, 2012, pp. 1106–1114.

[48] F. Gobet, P. Lane, S. Croker, P. Cheng, G. Jones, I. Oliver, J. Pine, Chunking mechanisms in human learning, Trends Cogn. Sci. 5 (6) (2001) 236–243.