CAAI Transactions on Intelligence Technology 2 (2017) 1–11

Contents lists available at ScienceDirect

CAAI Transactions on Intelligence Technology

journal homepage: http://www.journals.elsevier.com/caai-transactions-on-intelligence-technology/

Original article

Relative attribute based incremental learning for image recognition

Emrah Ergul

Electronics & Communication Eng. Dept. of Kocaeli University, Umuttepe Campus, 41380 Kocaeli, Turkey

ARTICLE INFO

Article history:
Received 6 November 2016
Received in revised form
15 January 2017
Accepted 19 January 2017
Available online 30 January 2017

Keywords:
Image classification
Incremental learning
Relative attribute
Visual recognition

ABSTRACT

In this study, we propose an incremental learning approach based on a machine-machine interaction via relative attribute feedbacks that exploit comparative relationships among top level image categories. One machine acts as 'Student (S)' with initially limited information and it endeavors to capture the task domain gradually by questioning its mentor on a pool of unlabeled data. The other machine is 'Teacher (T)' with the implicit knowledge for helping S on learning the class models. T initiates relative attributes as a communication channel by randomly sorting the classes on attribute space in an unsupervised manner. S starts modeling the categories in this intermediate level by using only a limited number of labeled data. Thereafter, it first selects an entropy-based sample from the pool of unlabeled data and triggers the conversation by propagating the selected image with its belief class in a query. Since T already knows the ground truth labels, it not only decides whether the belief is true or false, but it also provides an attribute-based feedback to S in each case without revealing the true label of the query sample if the belief is false. So the number of training data is increased virtually by dropping the falsely predicted sample back into the unlabeled pool. Next, S updates the attribute space which, in fact, has an impact on T's future responses, and then the category models are updated concurrently for the next run. We experience the weakly supervised algorithm on the real world datasets of faces and natural scenes in comparison with direct attribute prediction and semi-supervised learning approaches, and a noteworthy performance increase is achieved.

1. Introduction

Instead of mapping low level features (color/edge histograms, bag of words, Fourier, Wavelet transform features etc.) directly to the class labels, attribute-based approaches are nowadays popular instruments in the computer vision applications [1–8]. Attributes are mainly human interpretable and visually detectable entities on the visual data. They are used for intermediate level data representations between machine-alone understandable low level features and the human specific top level class labels. Attributes are introduced mostly because of sparse category labeled training data and large variations among low level features caused by both viewing conditions and high intra-class variance [1,9].

Since many top level categories share attributes as the common platforms, they can be learned with much more data samples than class labels. In addition to traditional problems like classification and retrieval [2,6], this leads to transferring the previously learned visual attributes to learn/detect a novel class which has not been seen during the training stage but we still know its attribute values, called zero-shot learning [10,11]. Besides, we can even infer an unfamiliar class which we have not known its attribute ground-truths [12,13] at all. We may also produce more semantic queries by using the visual attributes to perceive scenes/objects in the image search engines [14].

In this work, we propose a new approach which tries to recover from aforementioned insufficiencies of the supervision caused by the human factor, and the lack of connections in the attribute-based class modeling. It is based on the relative attributes for incremental visual recognition. We generate the relative attributes in an unsupervised manner by randomly ordering the target classes. On the other hand, we model the visual classes in a weakly supervised environment where a very limited number of training data is used. Meanwhile, we update both the attribute space and class models concurrently in an iterative algorithm. To do so, we establish a machine-machine interaction protocol that projects a Student (S)-Teacher (T) relationship that is available in the real world.

E-mail addresses: 106103002@kocaeli.edu.tr, emergul13@yahoo.com.
Peer review under responsibility of Chongqing University of Technology.

http://dx.doi.org/10.1016/j.trit.2017.01.001

1.1. Contributions

We present three contributions in this paper. First, the attribute space and the class models are learned simultaneously in an iterative manner. This is very intuitive since the class models are constructed on the attribute space, and there must be a connection in a combined learning procedure. The second contribution is the bilateral learning paradigm where S updates the attribute space which, in fact, influences T's future attribute-based explanations. We can say that T is also learning or it should generate its responses with respect to S' inferences on the attributes. Also note that attributes are initialized randomly and T's attribute-based explanations are prepared online in an unsupervised manner. The last contribution is about the virtual growth of the training set. Unlike semi-supervised methods where each sample is used only once for the learning procedure, T does not reveal the true label of the selected image if the belief is false. So the number of training data is increased by dropping the falsely predicted sample back into the unlabeled pool. In the experiments, we see that the number of iterations is almost doubled, and this leads to a better performance in visual recognition.

The general structure of the rest of the paper is as follows: Section 2 summarizes the related work about attribute-based applications with their important aspects. Then, we explain the proposed work in details in section 3, and the experimental evaluations are presented with comparative results on two popular datasets in section 4. We conclude the paper with our clear inferences in section 5.

2. Related work

The literature of attribute-based computer vision problems can be generally summarized in the types and extraction methods of attributes, the applications and datasets on which they are implemented, and evaluation criteria in the experiments, as shown in Fig. 1.

2.1. Attribute types

The importance of an attribute for each class varies for recognition purposes. Thus, we may produce an image/category-attribute co-occurrence matrix such that attribute membership for each class/image and the corresponding correlations with respect to the attributes may be captured. These relations can be produced in a supervised manner beforehand; or in a semi-supervised/unsupervised fashion. Some researchers suggest that binary correlations would be sufficient while others claim real valued ranking scores are essential to measure the attribute strength among categories/samples.

Relative attributes are first introduced by Parikh and Grauman in Ref. [6] with the assumption that semantically more enriched data descriptions and discrimination will be achieved if we use relative class memberships on attributes, instead of binary relations. Biswas and Parikh propose an interactive learning scheme for both object and face recognition with relative attributes in Ref. [15]. They improve this semi-supervised work in Ref. [10] by weighting negative candidate samples, selecting the query images and updating the attributes.

Kovashka and Grauman [16] open a new perspective to learn binary attributes by modeling them with many visual definitions instead of training only one discriminative model for each which is called shades of an attribute. Jayaraman and Grauman [17] study zero-shot classification paradigm with the assumption that traditional attribute scores are not suitable enough to predict them perfectly in unseen images. Chao-Yeh and Grauman [18] convey the zero-shot discussion on a new idea that class-sensitive attribute classifiers can transfer the learning to infer the new ones unobserved during training. Liang and Grauman [19] explore active learning strategies to learn relative attributes better without exploiting all pairwise comparisons.

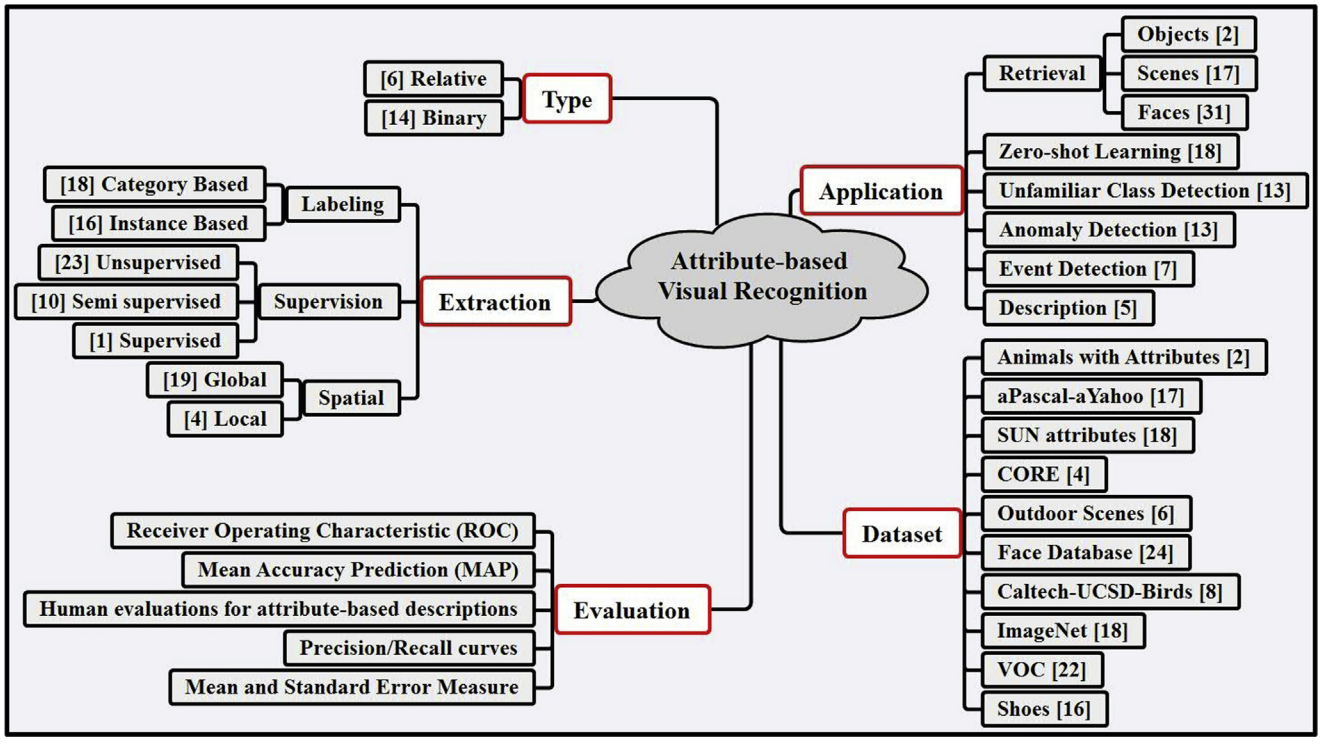

Fig. 1. Summary of the literature work on attribute-based computer vision problems.

2.2. Attribute extraction

In supervised settings, the attribute labeling can be constructed on whether the instances individually or the visual categories generally. Akata et al. [8] try to solve the attribute-based classification problem by constructing a composite framework which embeds mapping images from both sides of low-level features and class labels into a common attribute space at once. Jing et al. [20] generate a deep Convolutional Neural Network (CNN) architecture to learn appearance and motion based binary attributes. Kiapour et al. [21] experiment a challenging task to extract deep learning features that are based on retrieval and similarity learning methods for this task. Xinlei and Abhinav [22] propose an approach to utilize web data for learning visual attributes through CNN.

Unlike supervised learning where the attributes are named specifically, one may not name the attributes but only use them to discriminate the data classes. Farhadi et al. [5] embed the object recognition problem into describing objects and mainly focus on attribute learning. Ma et al. [11] implement an algorithm to learn class-level relative attributes in an unsupervised manner where they produce relative attributes as many as the number of pairwise class combinations in a greedy fashion. Karayel and Arica [23] follow the similar way of [11,24] but they produce the attributes completely randomly and attribute selection is implemented afterwards. Yu et al. [25] formalize a category-attribute co-occurrence matrix for cross-category generalization.

One may relate the attributes with each other such that hierarchical structure is established based on spatial layout, heritage and encapsulation aspects. Ran et al. [26] combine category-specific attributes and category-level information for generic instance search in the web. Victor et al. [27] extract relative attributes in different layers of the CNNs architecture by encoding the location, sparsity and the relationship between visual attributes. Local attributes, besides global ones, are also considered when the spatial ordering is important for recognition. But in this case, extra segmentation methods are required such as bounding boxes [4,7] or intensive labeling approaches [28] lead to additional cost to learn local attributes.

3. Materials and methods

3.1. Overview of the proposed method

In this work, we introduce a new approach for incremental visual recognition to model visual categories simultaneously with the relative attributes in an incremental learning procedure. To do so, we establish a machine-machine interaction protocol that projects a Student (S)-Teacher (T) relationship available in the real world. Relative attributes are first initialized by T in a way that sorts the image classes randomly in an unsupervised manner. Because we do not aim to locate the attributes specifically on spatial regions of the visual data, they are learned from global low-level features (i.e. GIST). The classes are then modeled by S with a limited number of category labeled data in the mid-level attribute space. Although it is T's initiative to generate random class orderings for the attribute learning, S then takes the responsibility to update both the attribute space and class models concurrently in an iterative algorithm. While S selects a sample, $\boldsymbol{x}^{(\boldsymbol{q})}$, from a pool of unlabeled data each time and propagates it with the belief class, T paves the way for S' corrective actions on class modeling with its attribute-based, comparative and affirmative/rejective responses.

Unlike [10,15] where the attribute-based explanations are prepared by the Amazon Mechanical Turk (AMT) workers before the learning procedure, the T machine decides online what explanation should be given to S for mentoring. While doing updates on class models, S also appends some candidate samples to its labeled dataset. The candidates are selected out of the unlabeled data, Uset, temporarily with respect to the conditions on the current attributes by constructing a weighting cube. This alternate routine which describes the attribute-class model updates continues until S runs out of the unlabeled data that leads to incremental learning for recognition. The flow chart of the algorithm is given in Fig. 2.

3.2. Preliminaries of initiative setup

Relative attributes are used as a communication channel in bidirectional interactions between Student (S) and Teacher (T) machines. To do so, we first divide the training set randomly, apart from test set, $X = \{\boldsymbol{x}^{(\boldsymbol{i})}, \mathrm{y}^{(\mathrm{i})}\}$; $\boldsymbol{x}^{(\boldsymbol{i})} \in \Re^{\mathrm{d}}$, $\mathrm{y}^{(\mathrm{i})} = \{1, 2, \ldots, \mathrm{K}\}$ *and* $\mathrm{i} = \{1, 2, \ldots, \mathrm{N}\}$, randomly into three discrete subsets: Student (Sset), Teacher (Uset) and Validation (Vset) sets. Sset consists of a very limited number of class-labeled images per category, and S uses this subset to achieve initial class models. On the other hand, S does not know the class labels of Uset and it basically represents T's teaching materials about the task domain. S utilizes Uset for both picking a sample in its belief propagation through questioning and for selecting some candidates in a weighting scheme to update its current category models. On the contrary, T handles this subset for preparing the best attribute-based response to indicate why the belief is correct or not. Vset is used for optimizing the free parameters of the algorithm and the initial setup is summarized in Fig. 3.

Another important detail of the proposed work is to construct an attribute space which constitutes the communication channel between S and T machines. Given a class based ordering, $a_m = \{\mathrm{c}^{(1)} > \mathrm{c}^{(2)} \approx \mathrm{c}^{(3)} > \ldots \approx \mathrm{c}^{(\mathrm{j})}\}$; $\mathrm{c}^{(\mathrm{j})} \in C$ and $C = \{1, 2, \ldots, K\}$, which relates every category to each other with a less/more and similarity conditions, we use the Newton's method in Ref. [6] for a relative attribute as:

$$\mathrm{r_m(x)} = w_m^T x^{(i)} \tag{1}$$

$$\forall\,(\mathrm{i,j}) \in O_m : w_m^T x_i^a > w_m^T x_j^b;\quad i \in c_a, j \in c_b, c_a > c_b \tag{2}$$

$$\forall\,(\mathrm{i,j}) \in S_m : w_m^T x_i^a \approx w_m^T x_j^b; i \in c_a, j \in c_b, c_a \approx c_b \tag{3}$$

$$\mathrm{argmin}_{\mathrm{W_m}} \left(\frac{1}{2} \left\| W_m^T \right\|_{L_2}^2 + Z\left(\Sigma \varepsilon_{ij}^2 + \Sigma \gamma_{ij}^2 \right) \right) \tag{4}$$

$$\text{st.} \quad \begin{array}{l} w_m^T (x_i - x_j) \geq 1 - \varepsilon_{ij};\ \forall\,(i,j) \in O_m \\ \left| w_m^T (x_i - x_j) \right| \leq \gamma_{ij};\ \forall\,(i,j) \in S_m \\ \varepsilon_{ij} \geqslant 0; \gamma_{ij} \geq 0 \end{array} \tag{5}$$

where $\mathrm{r_m}$ is the real valued ranking score for the training instance, $\boldsymbol{x_i}$, on the attribute basis, a_m, $\boldsymbol{w_m} \in \Re^d$ is the parameter vector of the relative attribute model. O_m is the set which includes pairs of data instances holding for the more/less conditions while S_m refers to the set of similar samples. So the optimum solution would then order the classes on the weight vector, $\boldsymbol{w_m}$, by minimizing the cost function in (4); where Z is the constant that regulates the balance between weight decreasing and the non-negative slack variables, $\varepsilon_{\mathrm{ij}}$ and γ_{ij}. This results in maximizing the margin between classes in the order definition of a_m.

T initializes the relative attribute space on Sset by randomly generated class ordering [23] for each attribute in an unsupervised manner, and shares this information with S. After preparation of the subsets for training and random initialization of the attribute space by T, it is now time for S to initialize category models by using Sset.

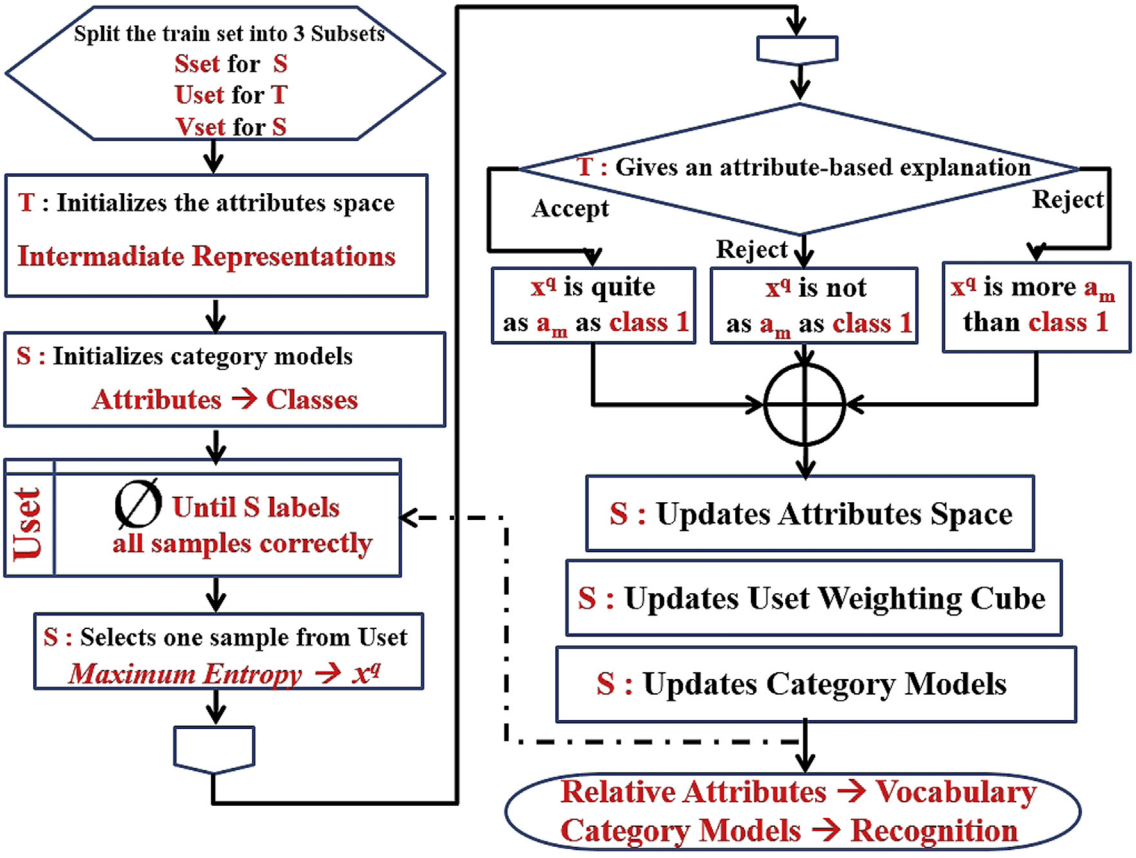

Fig. 2. Flow chart of the proposed algorithm.

S uses RBF kernel-based one-vs-all SVM [29] for class discriminants and the free parameters are optimized in a grid search method with 10-fold cross validation on Vset. To summarize, low level features, $\boldsymbol{x}^{(i)}$, are first projected into attribute space, $A = \{a_m\} \in \Re^M$, and classification is implemented in this mid-level with Sset at the beginning.

3.3. Student–teacher interactions for visual recognition

Incremental learning corresponds to gradually updating the attribute space and category models alternately in S-T interactions until S labels all of the supplementary data set, Uset, correctly with the help of T's attribute-based explanations. In detail, S picks a sample, $\boldsymbol{x}^{(q)}$, out of Uset which will be as much informative as possible for its class learning and this is achieved by maximum entropy, H, [10]:

$$H = -\sum_{y} p\left(\boldsymbol{x}^{(i)}\right)_y \log_2\left(p\left(\boldsymbol{x}^{(i)}\right)_y\right) \tag{6}$$

$$\boldsymbol{x}^{(q)} = argmax_{\boldsymbol{x}^{(i)}} H \tag{7}$$

where $p(\boldsymbol{x}^{(i)})_y$ is the probability output of SVM for the sample, $\boldsymbol{x}^{(i)}$, for each class, y. Hence, the class label $c^{(j)}$ with the highest probability, y^*, is attached to the query sample, $\boldsymbol{x}^{(q)}$, and it is questioned for an explanation like 'Does $\boldsymbol{x}^{(q)}$ belong to class $c^{(j)}$?' As aforementioned, the falsely predicted sample is dropped back into Uset which helps increasing the training set virtually.

When S machine often picks the same set of query samples, $\boldsymbol{x}^{(q)}$, out of Uset, T machine would generate similar attribute-based explanations for the S' beliefs. Because S uses these explanations for the updates of attribute space, overfitting may occur respectively. To avoid overfitting at this point, we propose three enhancements for S. First, S machine also propagates its best prediction once in a while (i.e. one in ten iterations) on the contrary to the selection method which is based on the maximum entropy. Secondly, S restricts one sample image to be selected often by deploying a counter for each. For instance, if S sends the $\boldsymbol{x}^{(q)}$ back to the Uset its counter will be initialized (for example set to three). The counters are then decreased by one through iterations, and the previously selected images are not allowed to be chosen again until their counters reach zero, i.e. not for the three next runs in this case. Finally, S keeps track of the selection history of samples in the current Uset. Hence, the image is not propagated again with a class label which has been predicted falsely in the previous iterations.

After S propagates its belief in a query it is now T's turn to reply, and this process is displayed in Fig. 4. The attributes are computed relatively with respect to the class labels and T already knows the class labels of both Sset and Uset. So one may append an attribute-based explanation to the decision of 'accepted' or 'rejected' by selecting an attribute and its comparative term. Unlike semi-supervised learning [30,31], S is influenced in the complementary attribute space for category modeling in a way that the query sample, $\boldsymbol{x}^{(q)}$, is sent back to Uset if the prediction is false, or it is added to Sset, otherwise. T is then responsible to select the best attribute-condition pair for the explanation and this is achieved by first projecting $\boldsymbol{x}^{(q)}$ and the datasets, Sset and Uset, on each attribute axes with $r_a(\boldsymbol{x}^{(i)}) = (\boldsymbol{x}^{(i)})^T \boldsymbol{w_a}$, where $r_a(\boldsymbol{x})$ is the ordering function computed as in Ref. [6] and $\boldsymbol{w_a}$ is the weight vector of attribute a_m. Thereafter, the samples are sorted increasingly with

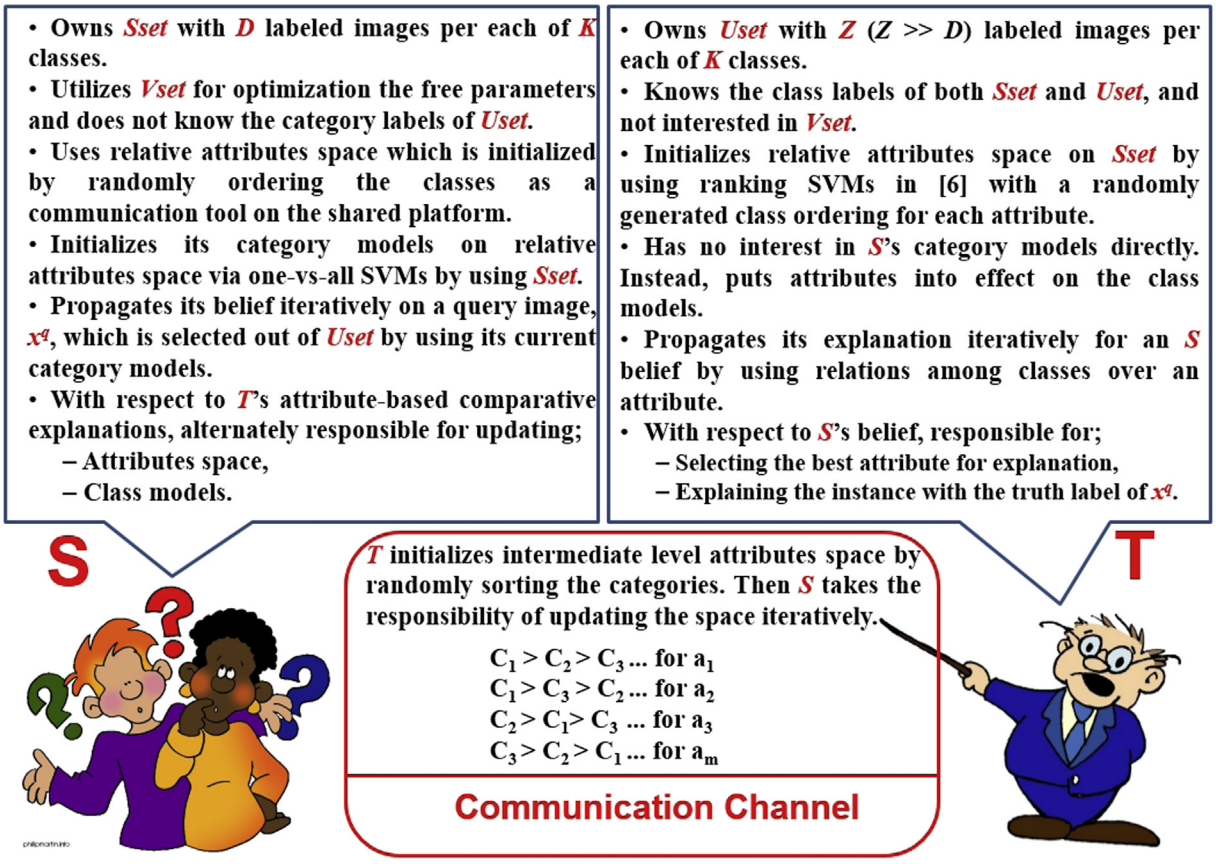

Fig. 3. Preliminaries of the proposed incremental learning algorithm.

their ranks, r_a, as in Ref. [15] and the voting method is implemented as:

$$\text{vote}\left(\boldsymbol{x}^{(q)}; a_m, \boldsymbol{x}^{(p)}\right) = \sum_j indnum\left(\boldsymbol{x}^{(q)}, a_m\right) - indnum\left(\boldsymbol{x}^{(p)}, a_m\right); \boldsymbol{x}^{(p)} \in C_{predicted} \tag{8}$$

where indnum $(\cdot)$ refers to the index number of samples on the attribute axes in the ordered list. The intuition is that if $\boldsymbol{x}^{(q)}$ does not belong to the falsely predicted class, $C_{predicted}$, but it belongs to another class, C_{true}, where $C_{true} > C_{predicted}$ on a_m, then we expect its vote to be positively high for that attribute. On the contrary, the vote should be as low as possible in its absolute form when the prediction is true. So T has 2 M options (i.e. more or less conditions for M attributes) to choose for false and M options for true beliefs since the condition is already selected as 'similar'.

Finally, T may mislead S by just using the probabilities computed in (8). For example, giving an explanation like 'No, $\boldsymbol{x}^{(q)}$ is not as natural (a_m) as $c^{(1)}$' contradicts the reality when it is the case '$C_{true} > c^{(1)}$ on naturalness'. Because T knows the category orderings on attributes in addition to C_{true} of $\boldsymbol{x}^{(q)}$, the response may be re-established with the next highest probability. Also note that there is no need of such considerations when the belief is true, and the attribute with the lowest vote (i.e. unsigned) is selected with the condition of similarity.

3.4. Updating parameters of the learning algorithm

3.4.1. Attribute space fine-tuning

Basically, the student-teacher interaction produces a pair of query-response through the iterations which triggers updating the parameters of attribute space and category models alternately. Assuming that T accepts S belief, $\boldsymbol{x}^{(q)}$ is appended to Sset and subtracted from Uset since the answer reveals the ground truth category label of $\boldsymbol{x}^{(q)}$. Thus, all attributes can be updated by creating new pairs with $\boldsymbol{x}^{(q)}$ in its class orderings. The situation becomes interesting when T rejects the prediction because the true class label of $\boldsymbol{x}^{(q)}$ is not given in the answer and it is not added to Sset. But S still finds additional pairs to update only the selected attribute by perceptual induction. For instance in 'No, $\boldsymbol{x}^{(q)}$ is more natural than class $c^{(1)}$', it is known that $\boldsymbol{x}^{(q)}$ is not in the predicted class with 'more' condition on attribute 'naturalness'. So one may simply produce new ordering pairs $O^m_{new} = \{(\boldsymbol{x}^{(q)}, \boldsymbol{x}^{(j)})\}$, where $\boldsymbol{x}^{(q)} > \boldsymbol{x}^{(j)}$, such that query image visualizes attribute a_m stronger than $\forall \boldsymbol{x}^{(j)} \in c^{(1)}$ on Sset.

Additionally, S may infer even more pairs by using its prior knowledge. When the class-based ordering on a_m is $c^{(3)} > c^{(2)} > c^{(1)} > c^{(4)}$ and the previous answer is given for false prediction, then $\boldsymbol{x}^{(q)}$ is actually more natural than $\forall \boldsymbol{x}^{(j)} \in c^{(1)} \cup c^{(4)}$. The same idea holds for the 'less' condition except reverse ordering rule. Hence, this would increase the number of pairing samples for attribute learning which leads to better discrimination. Also note that newly added pairs are deleted for the next iteration since S has not discovered the class label of $\boldsymbol{x}^{(q)}$ yet, unlike in case of true prediction. Besides, previously updated attribute weight vectors, $\boldsymbol{w}_a$, are carried through iterations for incremental learning instead

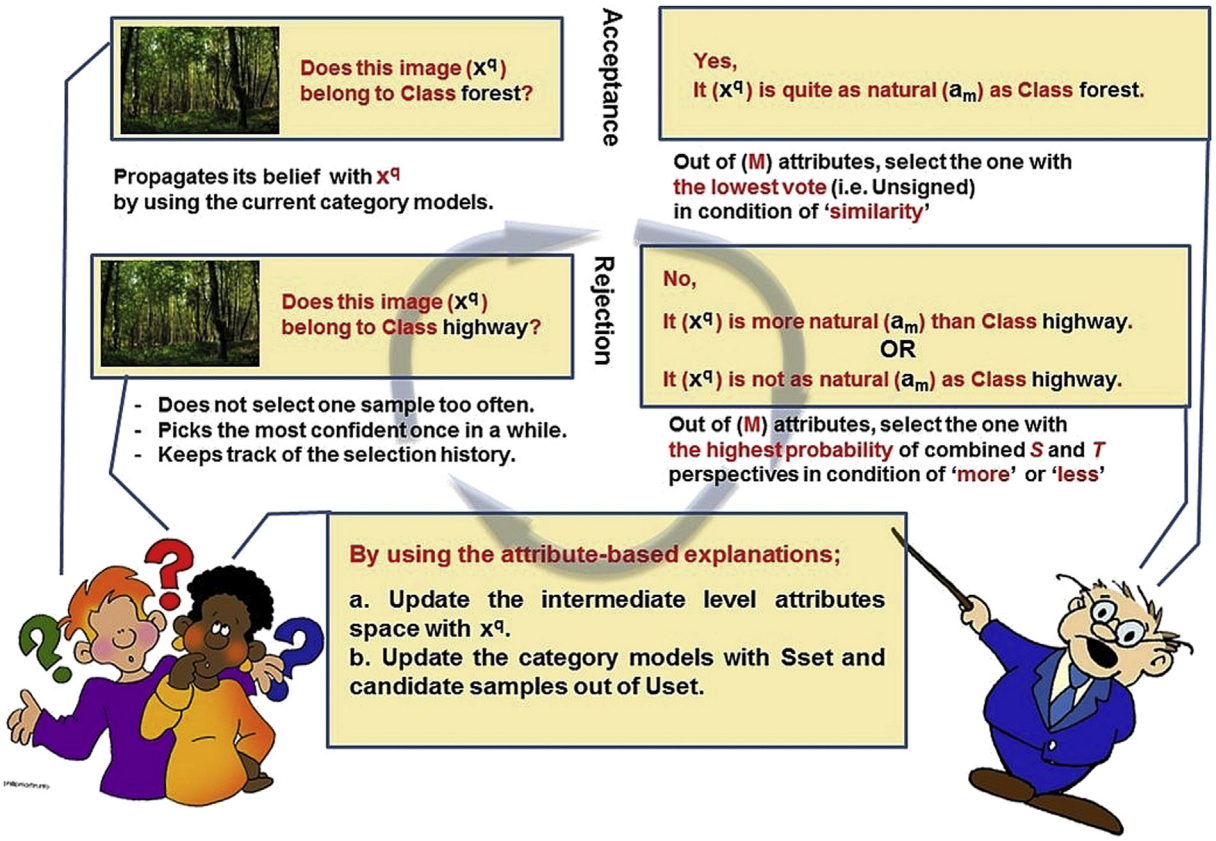

Fig. 4. Student − Teacher interaction scenario.

of random initialization, and only the paring samples which violate the constraints of ranking margin are used in the attribute update procedure. It accounts for avoidance of the overfitting when using the same ordering pairs through iterations.

3.4.2. Selecting candidates for updating the class models

Like in attribute updating, S makes inferences out of T's answers but now it is for picking the candidate samples from Uset temporarily to update class models. To do so, S produces a Uset 'Weighting Cube' on the Uset in which the weights are summed cumulatively in iterations for each candidate with respect to positive and negative sides, in addition to target categories. The weighting scheme is visualized in Fig. 5. In detail, the candidate samples, $X_{candidates} = \{\boldsymbol{x}^{(\boldsymbol{p})}\}_{p=1,2,\ldots,U}$; $\boldsymbol{x}^{(\boldsymbol{p})} \in Uset$, are chosen on the attribute base, a_m, which has been given in T's explanation by first computing the ranks of all samples, $r_a(\boldsymbol{x}^{(\boldsymbol{i})}) = (\boldsymbol{x}^{(\boldsymbol{i})})^T \boldsymbol{w_a}$; $\boldsymbol{x}^{(\boldsymbol{i})} \in (Uset + Sset)$. Afterwards, the samples are sorted and positions of the Uset samples in the sorted list are checked with $\boldsymbol{x}^{(\boldsymbol{q})}$ as it is the benchmark on the condition embedded in T's response. If S predicts $\boldsymbol{x}^{(\boldsymbol{q})}$ correctly to be $c^{(1)}$, for instance, the explanatory condition will be 'similarity', and the candidates should be even more similar to $c^{(1)}$ than $\boldsymbol{x}^{(\boldsymbol{q})}$ on the a_m. This is achieved by first finding the median image of $c^{(1)}$ (i.e. $\mu(\boldsymbol{x}^{(\boldsymbol{j})})$, see Fig. 5a.) and then marking the symmetric area which is centered on the median point and bounded by $\boldsymbol{x}^{(\boldsymbol{q})}$. The Uset samples projected on this region are selected as the 'positive' candidates for $c^{(1)}$ in this iteration only if they are not pinned as 'negative' for $c^{(1)}$ previously.

On the other hand, the 'negative' candidates for the predicted class can be achieved in false predictions easily in a similar manner except relative conditioning which determines the side moving away from $\boldsymbol{x}^{(\boldsymbol{q})}$ (i.e. to the left for 'less' and to the right for 'more'). The intuition for 'less' condition is that if $\boldsymbol{x}^{(\boldsymbol{q})}$ is not as a_m as $c^{(3)}$, the samples which are even less a_m than $\boldsymbol{x}^{(\boldsymbol{q})}$ would not belong to $c^{(3)}$, either. The same idea holds for the 'more' condition in the opposite direction (see Fig. 5b and c). After selecting the candidate positive/negative samples, $\boldsymbol{x}^{(\boldsymbol{p})}$, for the predicted class, $C_{predicted}$, S computes their cumulative weights as:

$$w_{UB}(\boldsymbol{x}^{(\boldsymbol{p})}; a_m, \boldsymbol{x}^{(\boldsymbol{q})}) = \sum_{iter=1}^{UB} \Big| indnum\,(\boldsymbol{x}^{(\boldsymbol{q})}, a_m) - indnum\,(\boldsymbol{x}^{(\boldsymbol{p})}, a_m) \Big| (1 - \exp^{\frac{-iter}{\sigma}}) \quad (9)$$

where UB is the upper bound of the iterations and σ is the scale factor. Summation yields to cumulative votes over iterations which is independent on the selected attribute, a_m. So the weights happen to be the distance that refers to the number of images between the candidate and $\boldsymbol{x}^{(\boldsymbol{q})}$. Also note that the exponential factor in (9) ensures that weights are added with an increasing ratio. Eventually, S selects some candidate samples out of Uset with their weights in each iteration, and these candidates will be appended to Sset for updating the category models.

3.4.3. Updating the class models

So far, S fine-tunes the attribute space and picks some candidate samples with their weights from Uset which will be appended to Sset as a temporary extension. Now it is time to update the category models, and this is achieved by one-vs-all L_2 RBF kernel based SVM

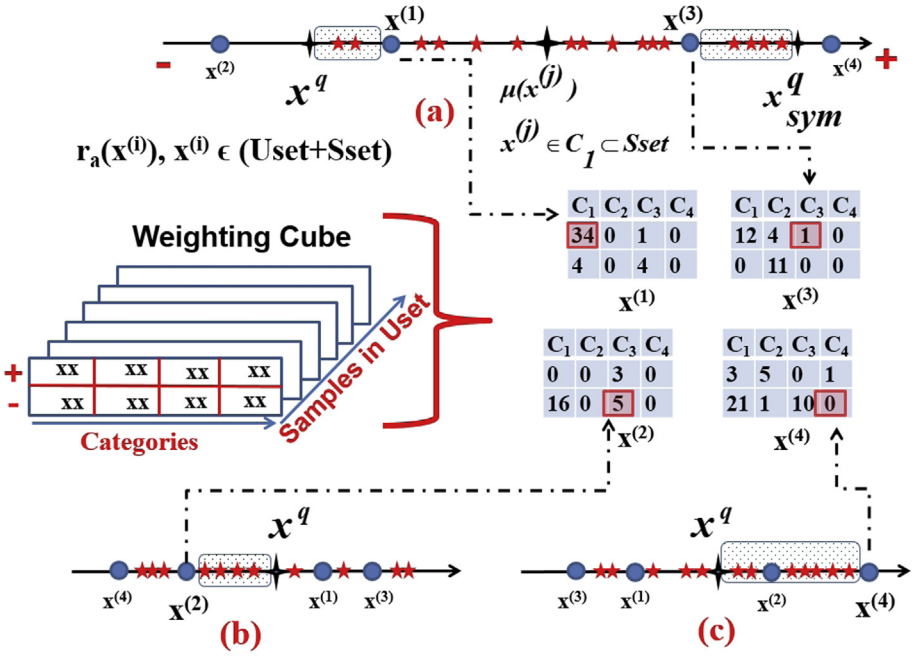

Fig. 5. Candidate selection with a weighting scheme. $\{x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}\} \in$ Uset are the selected candidates for an explanation of: a) $\boldsymbol{x}^{(q)}$ is quite as a_m as $c^{(1)}$, b) $\boldsymbol{x}^{(q)}$ is not as a_m as $c^{(3)}$, c) $\boldsymbol{x}^{(q)}$ is more a_m than $c^{(4)}$.

[29]. In detail, the samples in the Sset are known with their labels by S, so their weights are naturally 1 like in Ref. [15]. Hence, the candidate weights are first normalized to [0,1] such that they are summed to 1 for true predictions while the maximum value is set to 1 for false predictions because absolutely negative samples exist in the candidate list.

Once S predicts $\boldsymbol{x}^{(q)}$ truly the candidates are then labeled as $C_{predicted}$ with their normalized weights. The situation becomes confusing when T rejects the belief because S simply assumes that the candidates are not be in $C_{predicted}$, either. For example, if T answers '$\boldsymbol{x}^{(q)}$ is not as a_m as $c^{(3)}$' and the class-ordering is $c^{(2)} > c^{(3)} > c^{(4)} > c^{(1)}$ on a_m, S may expect that the candidates $\boldsymbol{x}^{(p)}$, in addition to $\boldsymbol{x}^{(q)}$, may come from $c^{(4)}$ or $c^{(1)}$. The decision is trivial when there is only one candidate category (i.e. $c^{(1)}$ if $c^{(3)}$ is interchanged with $c^{(4)}$) and the negative side of the $C_{predicted}$ vote is taken directly as the candidate weight. But S should make its best guess for $\boldsymbol{x}^{(p)}$ if there are more possible classes. In detail, S picks the best one among candidate classes by using its current category models if there is no weighting for these classes yet. Otherwise, the one with the highest vote, computed by the difference between positive and negative weights of $\boldsymbol{x}^{(p)}$ on the candidate class, is chosen directly.

So long as S determines the weights and class labels for the candidate samples, $\boldsymbol{x}^{(p)}$, which are appended to Sset temporarily, SVM algorithm is implemented on the new training set by initializing the model parameters with the previous ones in each iteration. The free parameters are optimized in a grid search method with 10-fold cross validation on Vset as mentioned in the 'preliminaries (see 3.2)' subsection. Also note that class models and attribute space are not updated each time when S achieves worse accuracy performance than the previous setups on the Vset. The intuition is that T's explanations sometimes may not help S improve its visual recognition because of overfitting, and it can downgrade the performance in iterations. The alternate update procedure continues iteratively until S empties Uset with its correct predictions.

4. Performance evaluation

4.1. Experimental setup

We use two real world datasets to evaluate the proposed algorithm for comparative experiments with the other attribute based methods: Outdoor Scene Recognition (OSR) [6,10,11] that contains 2688 images in 8 scene categories, and Public Figure Face (PUBFIG) [6,15,23] which has 800 images from 8 face classes. The global low level features (i.e. 512D GIST and additional 30D LAB color histograms for PUBFIG) are also provided with the datasets. We repeat the experiments 30 times with different training/test splits, and the average results are noted for generalization. In each run, there exist 15 samples in the Sset, 25 images in the Uset through random selection, and the rest is reserved for the test. Vset is generated from Sset + Uset by injecting Gaussian noise. Furthermore, the supervised attributes which are given with the datasets are also used (6 for OSR and 11 for PUBFIG [6]) for the performance evaluation. In the implementation, LibSVM [29] algorithm is utilized for category modeling while Newton's method in Ref. [6] is used for the extraction of relative attributes. Also note that RBF kernel outputs are whitened as in Ref. [32] for normalization before SVM.

Basically, the proposed work is compared with two framework methods of attribute-based recognition: Direct Attribute Prediction (DAP) and Semi-supervised Attribute Classification (SSAC). DAP offers a two-phase approach where the attribute space and class models are learned independently. Hence, Sset and Uset are now combined in a joined set. We first learn the relative attributes; thereafter categories are modeled on this mid-level via the joined set. SSAC, on the other side, represents 'in-between' methodology between DAP and the proposed work. The attribute space is generated with the joined set, Sset + Uset, and the attribute weight vectors, **w**, are fixed during the class modeling like in DAP. S initializes the classifiers on the attribute space by using Sset at the beginning, and it picks a sample out of Uset with its current best guess in an iterative manner. So, one sample is transferred from

Uset to Sset with the belief class label and S updates the class models with the new Sset in each run.

4.2. Multi-class classification

Our main concerns in multi-class classification are whether: *a*. Attributes constitute a fruitful intermediate space for data representation, *b*. Unsupervised attributes are more discriminative than the crowd sourced attribute labeling in supervision, *c*. Learning attributes and category models incrementally in a machine-machine interaction increases the recognition accuracy when compared with DAP and SSAC frameworks.

The plot of mean accuracy vs. varying number of attributes for OSR and PUBFIG datasets are displayed in Fig. 6. With regards to the supervised attributes alone, all three algorithms seem to achieve the similar performance despite the fact that the proposed and SSAC algorithms maintain a better start. We believe that it is due to the limited number of attributes and the class orderings on attributes are akin to each other. We learn only 28 unsupervised attributes in order to compare the proposed work with the other studies in literature. For these, all methods achieve much lover accuracy at the beginning but they make up the difference quickly when compared with the supervised attributes. As expected, the performance increases little about 1–2% for DAP and 3–4% for SSAC overall, whereas an obvious rise is experienced with the proposed algorithm, that is up to 6–8% for the 28 unsupervised attributes. The same evaluation is still true if we combine both. When the unsupervised attributes are appended, the accuracy rises up again about 3% with respect to the single usage of unsupervised attributes.

As SSAC and the proposed method are common in learning the class models iteratively, we display their plots of accuracy vs. iteration step in Fig. 7. Remembering that the relative attributes are learned with Sset + Uset initially, SSAC keeps them fixed during the class model updates whereas S machine updates the attribute space and classifiers concurrently with respect to T's attribute-based explanations in the proposed work. Also note that the query image, $\boldsymbol{x}^{(q)}$, is transferred with its belief class label directly into the Sset each time in the SSAC algorithm. So the iteration bound is restricted simply to the number of samples in the Uset, i.e. 200 for 25 images per category. On the contrary, the bound varies in the proposed method since the query image is dropped back into the Uset when it is predicted falsely.

In Fig. 7, it is observed that the SSAC makes a better start of about 2–4% than the proposed method. Because SSAC uses the joined set, Sset + Uset, for the attribute space initialization, we assume that SSAC has started to learn the attributes more consistently with a larger dataset, as expected. But the accuracy trend of SSAC is not as steep as the one of the proposed work during further iterations. For example in the PUBFIG experiments, the best accuracy level of the SSAC is already captured at about iteration # 230 and 4% final increase is achieved in the proposed algorithm. We conclude that the

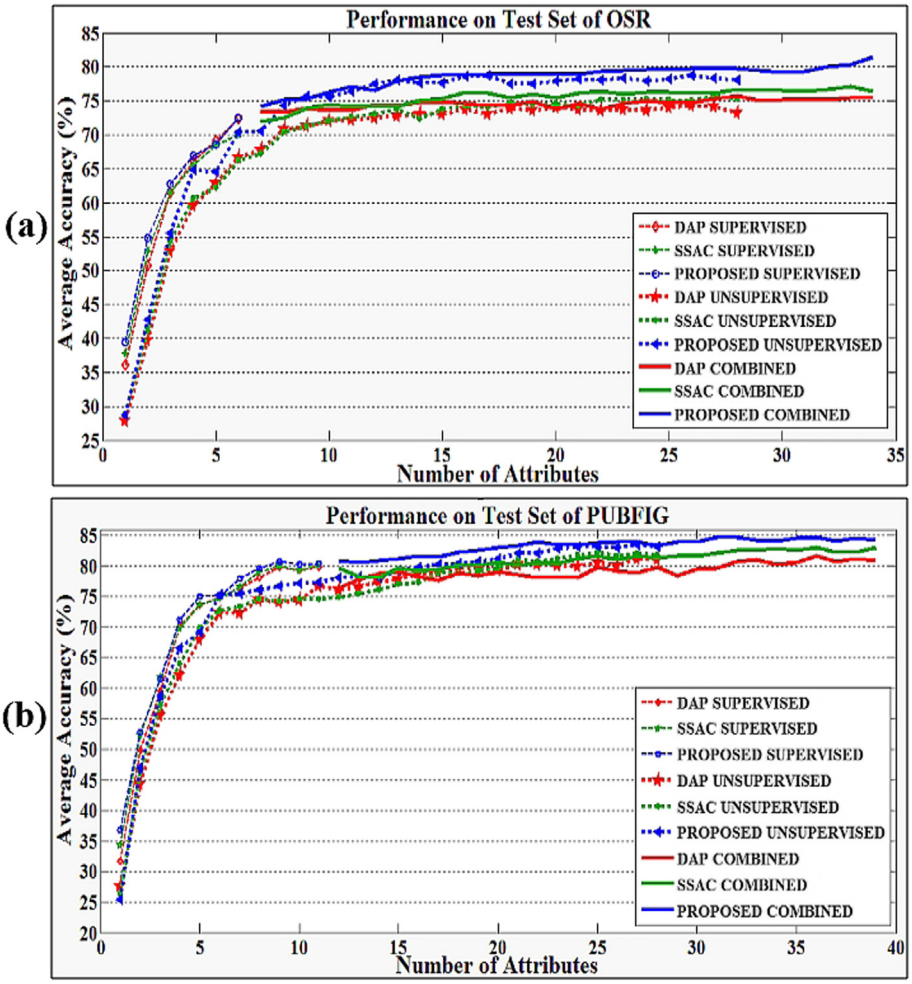

Fig. 6. Accuracy performance analysis with varying number of attributes on: a) OSR, b) PUBFIG.

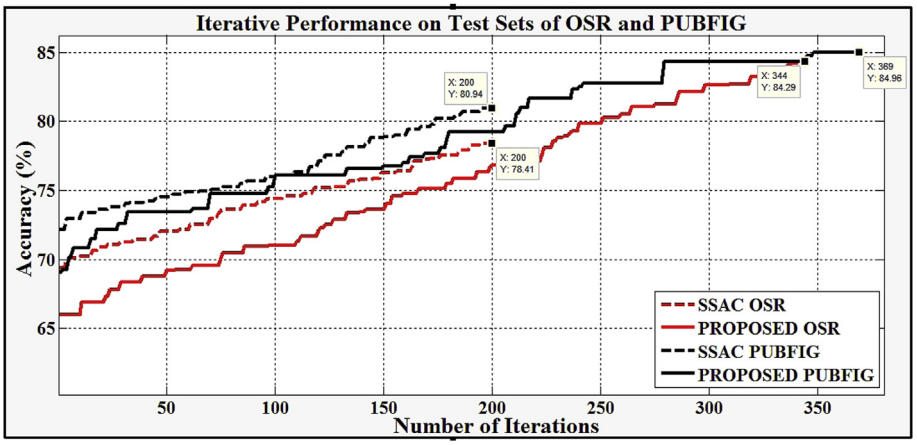

Fig. 7. Iterative performance analysis of SSAC and the proposed work on OSR and PUBFIG datasets.

performance is increased by expanding the training set virtually while some Uset samples are used multiple times in iterations.

Receiver Operating Characteristic (ROC) curve is frequently used in literature to evaluate the performance of classifiers. Basically, the ratio of false and true positive samples is plotted by changing thresholds in a step-wise manner. The classifier is regarded as more successful when its plot rises up earlier and sharper than the others. We compare the performances of the DAP, SSAC and proposed work with supervised and combined (i.e. supervised + unsupervised) attributes on ROC curves for both datasets in Fig. 8. As seen, the accuracy is increased obviously when all attributes are used together, and this confirms that the

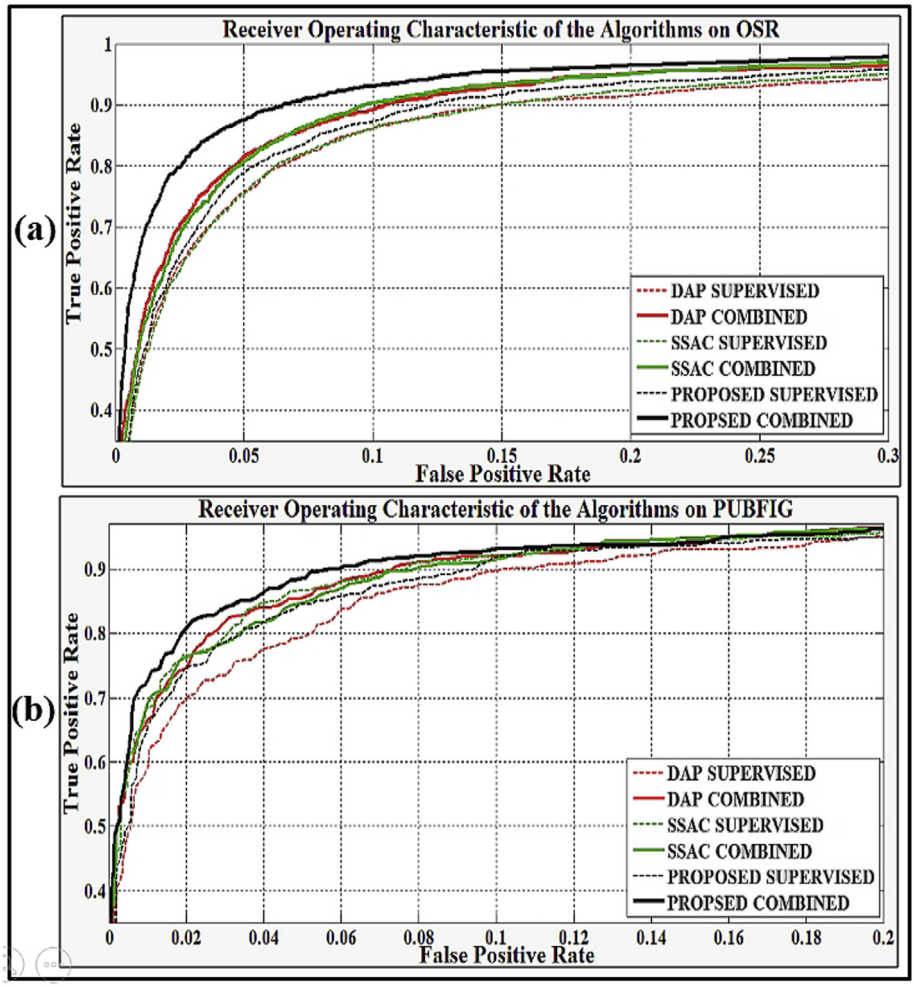

Fig. 8. ROC analysis of the DAP, SSAC and the proposed algorithms on: a) OSR, b) PUBFIG.

Table 1
Performance (%) comparison of the algorithms.

Algorithm	# of attributes	Mean accuracy OSR	Mean accuracy PUBFIG
SAT [6]	6–11	72.82±0.58	74.40±1.04
UAT [11]	28–28	76.57±1.93	78.60±2.26
SAT + UAT [11]	34–39	77.88±2.78	78.83±1.64
RAS [23]	34–39	78.64±2.14	81.25±2.96
UFA [33]	28–28	78.35±1.70	80.60±1.14
SRAA [34]	34–39	78.86±2.52	82.07±1.38
Our SAT	6–11	72.56±0.61	77.21±0.82
Our UAT	28–28	78.08±2.42	81.90±1.25
Our SAT + UAT	34–39	**81.51**±1.41	**84.05**±0.56

The bold numbers in the last two columns (i.e. mean accuracy OSR and PUBFIG, respectively) indicate the BEST mean accuracy percentages for the OSR and PUBFIG datasets.

unsupervised attributes add discriminative power in dimensionality. Additionally, the proposed algorithm outperforms the others clearly while SSAC is slightly better than the DAP method.

Finally, the proposed method is compared with the similar approaches in literature on the same experimental setup, and the mean ± std accuracy results of 30 repeats are listed in Table 1. BINs, PCA and FLD algorithms are actually used for dimension reduction. Nevertheless, the basis vectors which are extracted during their implementations help representing the data in a new feature space, so they are included as baselines for this reason. The other methods generate supervised/unsupervised attributes in the mid-level for visual recognition, like the proposed work. The results in Table 1 show that the proposed method outperforms the other approaches for about minimum of 2% although the attributes are produced by random class orderings unlike in Ref. [11], and there is no specific attribute selection, differently from Ref. [23]. The intuition behind such a notable result is that we shape the attribute space simultaneously with the class models and the number of training samples is increased during attribute-based feedbacks. We also assume that the expanded number of unsupervised attributes with distinct class orderings establish a better representation without human laboring, leading to more effective classifiers.

5. Conclusion

In this study, we use attributes as the mid-level representation, instead of low-level features, in an incremental learning procedure for visual recognition. Attributes constitute a communication channel through S-T machine interactions. The conversation mainly indicates a question-answer protocol where S selects an informative sample from the unlabeled pool of data, Uset, each time and T gives attribute-based explanations for mentoring. Also note that the training set is expanded virtually since the query image is dropped back into the Uset without exposing its ground-truth category label when S predicts it falsely. So we introduce a student centered learning method where S is actually responsible for updating the attribute space and class models simultaneously in an iterative manner. Although T initiates the relative attributes by sorting the image categories randomly in an unsupervised fashion, S updates the attributes with the selected query images from then on. This means that T also learns in time how the student interprets the representative space on which the class discriminants are built. Additionally, S does not use only its labeled dataset, Sset, for updating the class models, but it also implements a candidate selection method to append some samples from Uset to Sset in a weighting scheme.

We experiment the proposed work comparatively with the frameworks DAP and SSAC, and other similar approaches in literature on two popular mid-scale datasets, OSR and PUBFIG. Because classification is the focal point, the mean accuracy vs. iteration step/number of attributes and the ROC curves are the main themes for analysis during multiple experiments. To summarize, the proposed method outperforms other works significantly, and it is concluded that the unsupervised attributes exploit the recognition performance when learned incrementally and simultaneously with the class models even though they are achieved by random class orderings without human laboring.

Acknowledgements

This work was supported in part by the Scientific and Technological Research Council of Turkey (TUBITAK) 2214B.14.2 TBT.0.06.01.214.83.

Appendix A. Supplementary data

Supplementary data related to this article can be found at http://dx.doi.org/10.1016/j.trit.2017.01.001.

References

[1] V. Ferrari and A. Zisserman, Learning visual attributes, Neural Information Processing Systems Conference, 3–8 December 2007, Vancouver, Canada, pp. 433–440.
[2] C.H. Lampert, H. Nickisch, S. Harmeling, Attribute-based classification for zero-shot visual object categorization, J. IEEE Trans. Pat. Anal. Mach. Intel. 36 (2014) 453–465.
[3] C.H. Lampert, H. Nickisch and Harmeling S, Learning to detect unseen object classes by between-class attribute transfer, IEEE CVPR Conference, 20–25 June 2009, Miami, FL, USA, pp. 951 – 958.
[4] A. Farhadi, I. Endres and D. Hoiem, Attribute-centric recognition for cross-category generalization, IEEE CVPR Conference, 13–18 June 2010, San Francisco, CA, USA, pp. 2352–2359.
[5] A. Farhadi, I. Endres, D. Hoiem and D. Forsyth, Describing objects by their attributes, IEEE CVPR Conference; 20–25 June 2009, Miami, FL, USA, pp. 1778–1785.
[6] D. Parikh and K. Grauman, Relative attributes, IEEE ICCV Conference, 6–13 November 2011, Barcelona, Spain, pp. 503–510.
[7] G. Sharma, F. Jurie and C. Schmid, Expanded parts model for human attribute and action recognition in still images, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 652–659.
[8] Z. Akata, F. Perronnin, Z. Harchaoui and C. Schmid, Label embedding for attribute based classification, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 819–826.
[9] A. Bosch, M. Xavier, R. Marti, A review: which is the best way to organize/classify images by content, J. Image Vis. Comput. 25 (2007) 778–791.
[10] A. Parkash and D. Parikh, Attributes for classifier feedback, Springer ECCV Conference, 7–13 October 2012, Florence, Italy, pp. 354–368.
[11] S. Ma, S. Sclaroff and N.I. Cinbis, Unsupervised learning of discriminative relative visual attributes, Springer ECCV Conference, 7–13 October 2012, Florence, Italy, pp. 61–70.
[12] Y. Wang and G. Mori, A discriminative latent model of object classes and attributes, Springer ECCV Conference, 5–11 September 2010, Crete, Greece, pp. 155–168.
[13] C. Wah and S. Belongie, Attribute-based detection of unfamiliar classes with humans in the loop, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 779–786.
[14] M. Rastegari, D. Parikh and A. Farhadi, Multi-attribute queries: to merge or not to merge, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 3310–3317.
[15] A. Biswas and D. Parikh, Simultaneous active learning of classifiers and attributes via relative feedback, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 644–651.
[16] A. Kovashka, K. Grauman, Discovering attribute shades of meaning with the crowd, Int. J. Comput. Vis. 114 (2015) 56–73.
[17] D. Jayaraman and K. Grauman, Zero-shot recognition with unreliable attributes, Neural Information Processing Systems Conference, 8–13 December 2014, Montreal, Canada, pp. 3464–3472.
[18] C. Chao-Yeh and K. Grauman, Inferring analogous attributes, IEEE CVPR Conference, 23–28 June 2014, Columbus, OH, USA, pp. 200–207.
[19] L. Liang and K. Grauman, Beyond comparing image pairs: setwise active learning for relative attributes, IEEE CVPR Conference, 23–28 June 2014, Columbus, OH, USA, pp. 208–215.
[20] S. Jing, K. Kai, C.L. Chen, and W. Xiaogang, Deeply learned attributes for crowded scene understanding, IEEE CVPR Conference, 7–12 June 2015, Boston, MA, USA, pp. 4657–4666.
[21] M.H. Kiapour, H. Xufeng, L. Svetlana, C.B. Alexander and L.B. Tamara, Where to

buy it: matching street clothing photos in online Shops, IEEE ICCV Conference, 7–13 December 2015, Santiago, CA, USA, pp. 3343–3351.
[22] C. Xinlei and G. Abhinav, Webly supervised learning of convolutional networks, IEEE ICCV Conference, 7–13 December 2015, Santiago, CA, USA, pp. 1431–1439.
[23] M. Karayel and N. Arica, Random attributes for image classification, IEEE SIU Conference, 24–26 April 2013, Girne, TRNC, pp. 1–4.
[24] N. Kumar, A.C. Berg, P.N. Belhumeur and S.K. Nayar, Attribute and simile classifiers for face verification, IEEE ICCV Conference, 29 September – 2 October 2009, Kyoto, Japan, pp. 365–372.
[25] F.X. Yu, L. Cao, R.S. Feris, J.R. Smith and S. Chang, Designing category-level attributes for discriminative visual recognition, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 771–778..
[26] T. Ran, W.M.S. Arnold and C. Shih-Fu, Attributes and categories for generic instance search from one example, IEEE CVPR Conference, 7–12 June 2015, Boston, MA, USA, pp. 177–186.
[27] E. Victor, C.N. Juan and G. Bernard, On the relationship between visual attributes and convolutional networks, IEEE CVPR Conference, 7–12 June 2015, Boston, MA, USA, pp. 1256–1264.
[28] S. Zhiyuan, Y. Yongxin, M.H. Timothy and X. Tao, Weakly supervised learning of objects, attributes and their associations, Springer ECCV Conference, 6–12 September 2014, Zurich, Switzerland, pp. 472–487.
[29] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, ACM T Intel. Syst. Tech. 27 (2011) 1–27.
[30] J. Choi, M. Rastegari, A. Farhadi and L.S. Davis, Adding unlabeled samples to categories by learned attributes, IEEE CVPR Conference, 23–28 June 2013, Portland, OR, USA, pp. 875–882.
[31] A. Shrivastava, S. Singh and A. Gupta, Constrained semi-supervised learning using attributes and comparative attributes, Springer ECCV Conference, 7–13 October 2012, Florence, Italy, pp. 369–383..
[32] A. Coates, Y.N.G. Andrew, H. Lee, An analysis of single-layer networks in unsupervised feature learning, J. Mach. Learn Res. 15 (2011) 215–223.
[33] E. Ergul, S. Erturk and N. Arica, Unsupervised relative attribute extraction, IEEE SIU Conference, 24–26 April 2013, Girne, TRNC, pp. 1–4.
[34] E. Ergul, M. Karayel, O. Timus, E. Kiyak, Unsupervised feature learning for mid-level data representation, J. Nav. Sci. Eng. 12 (2016) 51–79.

Emrah Ergul received the B.S. degree in Electrical and Electronics Engineering from Turkish Naval Academy, Istanbul, Turkey, in 2001. From 2001 to 2007, he served in the Navy Fleet Command, Kocaeli, Turkey as a weapon systems and electronics officer. He received the M.S. degree in Computer Engineering from Naval Science and Engineering Institute, Istanbul, Turkey, in 2009. He also studied at the Computer Science Dept. of University of Sheffield, UK for one semester with respect to Erasmus Student Exchange Program while pursuing the M.S. degree. He studied in the Computer Vision and Robotics Laboratory at the Beckman Institute of the University of Illinois, Urbana-Champaign IL, USA, between 2013 and 2014 with a scholarship granted by The Scientific and Technological Research Council of Turkey. He received the Ph.D. degree in Electronics and Communication Engineering from Kocaeli University, Kocaeli, Turkey in 2016. He has been working for Navy Inventory Control Center, Kocaeli, Turkey since 2009 for stock estimation and process control. His research interest includes computer vision for image retrieval and classification, logistics decision support systems and data exchange standards for inventory planning, pattern recognition analysis, data mining and machine learning applications for the recommender systems.