# Unsupervised Active Learning Based on Hierarchical Graph-Theoretic Clustering

Weiming Hu, *Senior Member, IEEE*, Wei Hu, Nianhua Xie, and Steve Maybank, *Senior Member, IEEE*

***Abstract*—Most existing active learning approaches are supervised. Supervised active learning has the following problems: inefficiency in dealing with the semantic gap between the distribution of samples in the feature space and their labels, lack of ability in selecting new samples that belong to new categories that have not yet appeared in the training samples, and lack of adaptability to changes in the semantic interpretation of sample categories. To tackle these problems, we propose an unsupervised active learning framework based on hierarchical graph-theoretic clustering. In the framework, two promising graph-theoretic clustering algorithms, namely, dominant-set clustering and spectral clustering, are combined in a hierarchical fashion. Our framework has some advantages, such as ease of implementation, flexibility in architecture, and adaptability to changes in the labeling. Evaluations on data sets for network intrusion detection, image classification, and video classification have demonstrated that our active learning framework can effectively reduce the workload of manual classification while maintaining a high accuracy of automatic classification. It is shown that, overall, our framework outperforms the support-vector-machine-based supervised active learning, particularly in terms of dealing much more efficiently with new samples whose categories have not yet appeared in the training samples.**

***Index Terms*—Active learning, dominant-set clustering, image and video classification, network intrusion detection, spectral clustering.**

## I. Introduction

ACTIVE learning [1], [15], [23] is, with a number of available labeled samples, the automatic selection of highly informative unseen samples for manual classification and the automatic classification of other samples. The number of samples for manual classification should be as small as possible, and the automatic classification of samples should be as accurate as possible. Active learning can be applied to many fields such as image retrieval, video annotation, gene sequence analysis, audio processing, and network intrusion detection.

Manuscript received July 21, 2008; revised December 7, 2008. First published March 24, 2009; current version published September 16, 2009. This work was supported in part by National Science Foundation of China (NSFC) under Grants 60825204 and 60672040 and in part by the National High-Tech R&D Program of China (or 863 Program) under Grant 2006AA01Z453. This paper was recommended by Associate Editor P. S. Sastry.

W. Hu, W. Hu, and N. Xie are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China (e-mail: wmhu@nlpr.ia.ac.cn; whu@nlpr.ia.ac.cn; nhxie@nlpr.ia.ac.cn).

S. Maybank is with the School of Computer Science and Information Systems, Birkbeck College, University of London, WC1E 7HX London, U.K. (e-mail: sjmaybank@dcs.bbk.ac.uk).

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

Digital Object Identifier 10.1109/TSMCB.2009.2013197

Most existing active learning approaches are supervised [6], [10], [18], [47]–[49]. They are described in the following as either uncertainty based or committee based, depending on the strategy for selecting samples for manual classification [35]:

1) Uncertainty-based approaches [6], [10], [18], [36], [40], [46] generally use the available labeled samples to construct a classifier. If the uncertainty in classification of a new sample using this classifier is high, e.g., the sample falls close to a classification boundary of a support vector machine (SVM), then the sample is selected for manual classification. Zhang and Chen [23] use biased kernel regression to estimate the probability of each unlabeled sample that has an attribute. Knowledge gain is then defined to determine the sample that has the highest uncertainty for the classifier. Lewis and Gale [44] select an unlabeled sample as the most uncertain one when the posterior probability that the sample with a specific pattern belongs to a specific category is closest to 0.5. Roy and McCallum [45] present an active learning algorithm that selects samples for manual classification such that the expected future error is minimized. Schohn and Cohn [42] describe an active learning heuristic that greatly enhances the generalization behavior of SVMs on several practical document classification tasks. Campbell *et al.* [41] propose an algorithm for the training of SVMs using the active selection of samples for manual classification. Tong and Koller [43] introduce an algorithm for performing active learning with SVMs, providing a theoretical motivation for the algorithm using the notion of a version space.
2) Committee-based approaches [17], [37] repeatedly and jointly use labeled samples to construct a committee of classifiers. If a new sample is assigned to quite different categories by the members of this committee, the sample is considered to be highly informative and is selected for manual classification. Tur *et al.* [17] use the SVM classifier and an implementation of the AdaBoost algorithm as the members of the committee of classifiers. The lowest scored sample is selected for manual classification. Fine *et al.* [38] show that recent algorithms for approximating the volume of convex bodies and uniformly sampling from convex bodies using random walks can yield an efficient implementation for committee-based active learning. Freund *et al.* [39] show that if the two-member committee algorithm achieves information gain with positive lower bound, then the automatic classification error exponentially decreases as the number of manual classifications increases.

Supervised active learning approaches have the following problems.

1) These approaches cannot deal effectively with the semantic gap between the distribution of samples in the feature space and their labels—the relative positions of samples in the feature space may only be weakly related to their label information. We use the classification of arthropods to illustrate this semantic gap. It is known that locusts and ladybirds are insects and that spiders are arachnids. Samples of either locust, ladybird, or spider can form a cluster in the feature space. As the shape of a locust is very different from the shape of a ladybird and the shape of a ladybird is similar to the shape of a spider, the clusters for locusts and ladybirds, which are both insects, are far apart in the feature space, but the clusters for ladybirds and spiders, which belong to different categories, are nearby in the feature space. This is an example of a semantic gap in that the semantic meanings of categories are not reflected in their positions in the feature space. In such cases, supervised active learning that depends on sample labels in training may fail to identify the highly informative samples.
2) Supervised active learning cannot select new samples that belong to new categories that have not yet appeared in the training samples. Supervised active learning generally selects, for manual classification, these new samples that fall close to decision boundaries. It fails to identify new samples that belong to new categories that have not yet appeared in the training samples, as such samples are usually far from the training samples but not near to the decision boundaries.
3) Supervised active learning cannot adapt to changes in the labeling, arising from a new semantic interpretation of sample categories—for example, ladybirds and spiders could be relabeled as arthropods, belonging to the same category. Current supervised active learning algorithms must reconstruct the classifier to take account of such changes in the labeling.

The first aforementioned problem influences the accuracy of supervised active learning, and the second and third problems make supervised active learning difficult to adapt to changing environments.

To deal with these problems, we propose a novel active learning framework based on unsupervised learning [50]. Our basic idea is to ignore the relations between the relative positions of samples in the feature space and the labeling of samples. In our unsupervised active learning framework, the available labeled samples are clustered, and each cluster is given the label that most of the samples in it share. If a new sample does not fall into any of these clusters, the sample is considered as highly informative and thus selected for manual classification; otherwise, the sample is given the label of the cluster that it falls into. For the example of classifying arthropods, if the label of the ladybird cluster is insect, and if a new sample falls into this cluster, then it is classified as an insect. Different clusters may have the same label, and the labeling information is reflected by the distributed labeled clusters. The semantic gap between the distribution of samples in the feature space and their labels is then tackled. When the labeling of samples is changed, corresponding to a new semantic interpretation of sample categories, it is only necessary to change the labels of the clusters. Reclustering the samples is not needed. Furthermore, new samples that belong to new categories that have not yet appeared in the training samples are likely to be selected for manual classification, because new samples are compared with the clusters rather than the decision boundaries, and the features of the samples that belong to new categories are usually quite different from the features of the clusters.

Selection of a proper unsupervised learning algorithm is critical for our active learning framework. In unsupervised learning, graph-theoretic clustering [2], [5], [22] has recently attracted broad attention due to its clear intuitiveness, strong theoretical foundations, and successful applications in many fields such as image segmentation and video analysis [16], [24]. There exist many graph-theoretic clustering algorithms such as complete link [8], minimum cut [19], information theoretic [9], normalized cut [16], spectral clustering [21], and dominant-set clustering [13]. Spectral clustering is one of the most popular clustering algorithms. It has high accuracy, but its computational complexity is high, and thus, it is not suitable for large-scale clustering. Dominant-set clustering is a newly proposed algorithm that is based on a novel definition of clusters corresponding to dominant sets. It has low computational complexity, and it is flexible enough to allow online clustering. Spectral clustering and dominant-set clustering complement each other, so we combine them in a hierarchical fashion to construct a hierarchical graph-theoretic clustering-based active learning framework.

The remainder of this paper is organized as follows. Section II briefly introduces the two graph-theoretical clustering algorithms: dominant-set clustering and spectral clustering. Section III describes the proposed unsupervised active learning framework. Section IV shows the experimental results. Section V concludes this paper.

## II. Graph-Theoretic Clustering

A set $(V)$ of $n$ samples can form the vertices of an undirected edge-weighted graph $G = (V, E)$, where $E$ is the set of weighted edges that link different vertices. The edge weights reflect the similarities between samples. Let $w_{ij}$ be the edge weight between samples $i$ and $j (w_{ij} \geq 0)$. A symmetric affinity matrix $A = (a_{ij})^{n \times n}$ is used to represent the graph $G$, where $a_{ij} = w_{ij}$ if $(i, j) \in E$ and $a_{ii} = 0, \forall i \in V$.

The aim of graph-theoretic clustering is to cluster the vertices in $V$ according to the affinity matrix $A$. Concepts and algorithms for dominant-set clustering and spectral clustering are briefly introduced hereinafter for the convenience of the reader.

### A. Dominant-Set Clustering

The dominant set is a novel combinatorial concept proposed by Pavan and Pelillo [13]. The characteristics of dominant-set clustering rest with the definition of dominant sets.

*1) Definitions:* Let $\mathbb{S}$ be a nonempty vertex set, where $\mathbb{S} \subseteq V$. For any vertex $i \in \mathbb{S}$, the average weighted degree of $i$ relative to $\mathbb{S}$ is defined as

$$D_{\mathbb{S}}(i) = \frac{1}{|\mathbb{S}|} \sum_{k \in \mathbb{S}} a_{ik} \tag{1}$$

where $|\mathbb{S}|$ is the number of vertices in $\mathbb{S}$. For a vertex $j \notin \mathbb{S}$, the similarity $\phi_{\mathbb{S}}(i,j)$ between vertices $i$ and $j$ relative to $\mathbb{S}$ is defined as $\phi_{\mathbb{S}}(i,j) = a_{ij} - D_{\mathbb{S}}(i)$. Then, the weight $w_{\mathbb{S}}(i)$ of $i \in \mathbb{S}$ relative to $\mathbb{S}$ is defined as

$$w_{\mathbb{S}}(i) = \begin{cases} 1, & \text{if } |\mathbb{S}| = 1 \\ \sum\limits_{k \in \mathbb{S}-\{i\}} \phi_{\mathbb{S}-\{i\}}(k,i) w_{\mathbb{S}-\{i\}}(k), & \text{otherwise.} \end{cases} \tag{2}$$

Equation (2), which is a recursive definition, indicates that, to examine the weight of $i$ relative to $\mathbb{S}$, the influence of set $\mathbb{S} - \{i\}$ on $i$ is examined. The more the influence, the more the importance of $i$ in $\mathbb{S}$. According to $w_{\mathbb{S}}(i)$, the total weight of $\mathbb{S}$ is defined as

$$W(\mathbb{S}) = \sum_{i \in \mathbb{S}} w_{\mathbb{S}}(i). \tag{3}$$

The set $\mathbb{S}$ is defined as a dominant set if it satisfies the following conditions: 1) $\forall \mathrm{T} \subseteq \mathbb{S}$, $W(\mathrm{T}) > 0$; 2) $\forall i \in \mathbb{S}$, $w_{\mathbb{S}}(i) > 0$; and 3) $\forall i \notin \mathbb{S}$, $w_{\mathbb{S} \cup \{i\}}(i) < 0$. Condition 1) indicates that vertices in each subset of $\mathbb{S}$ are closely and firmly united. Condition 2) indicates that $\mathbb{S}$ has large attraction to each vertex in $\mathbb{S}$. Condition 3) indicates that $\mathbb{S}$ has no large attraction to any vertex outside $\mathbb{S}$. Conditions 1) and 2) describe the internal homogeneity of $\mathbb{S}$. Condition 3) describes the external heterogeneity of $\mathbb{S}$. Therefore, it is proper that a dominant set is treated as a cluster of vertices.

*2) Algorithm:* Depending on the definition of the dominant set, a dominant set is found from a graph by quadratic programming [13].

Let $u$ be an $n$-dimensional vector, where $n$ is the number of vertices in a graph, and let $A$ be the affinity matrix of the graph. The following quadratic program is considered:

$$\begin{aligned} &\max f(u) = u^T A u \\ &\text{s.t. } u \geq 0 \quad \sum_{i=1}^{n} u_i = 1 \end{aligned} \tag{4}$$

where $f(u)$ is the object function of the program. Let $u^*$ denote a local maximum of $f$, and let $\Omega_{u^*}$ be the vertex support set of $u^* : \Omega_{u^*} = \{V_i : u_i^* > 0\}$. It is proved that the vertex support set $\Omega_{u^*}$ corresponds to a dominant set in the graph. Then, a dominant set is found by solving (4). The local maximum value $f(u^*)$ of the objective function indicates the cohesiveness of the dominant-set cluster corresponding to $u^*$.

The following iterative equation is used to solve (4):

$$u_i(t+1) = u_i(t) \frac{(Au(t))_i}{u(t)^T A u(t)} \tag{5}$$

where $t$ indexes the number of iterations. The solution for (4) is succinct, and its computational demand is low compared with that of other graph-theoretic clustering algorithms that rely on an eigen analysis of $A$.

Dominant-set clustering is a bipartition procedure. A dominant set is found and then removed from the graph; and then, a second dominant set is found from the remaining part of the graph, and so on. The procedure continues until each vertex in the graph is assigned to a dominant-set cluster. The number of clusters is automatically determined.

### *B. Spectral Clustering*

Spectral clustering is described in [3], [11], [12], [16], and [21]. It seeks an optimal partition of the graph $G = (V, E)$, where the number $K$ of clusters is specified in advance. The algorithm for spectral clustering is outlined as follows:

**Step 1:** Calculate the diagonal degree matrix $D$ for the affinity matrix $A$ by

$$D_{ii} = \sum_{j=1}^{n} A_{ij}, \quad 1 \leq i \leq n; \quad D_{ij} = 0, \quad \text{if} \quad i \neq j \tag{6}$$

and construct the matrix $L$ by $L = D^{-1/2} A D^{-1/2}$.

**Step 2:** Apply the eigenvalue decomposition to $L$, and select the $K$ largest eigenvalues and the corresponding eigenvectors $z_1, \ldots, z_K$. The eigenvectors are represented as column vectors. Then, form the matrix $Z = [z_1, \ldots, z_K] \in \mathbb{R}^{n \times K}$, in which the columns are the $K$ eigenvectors.

**Step 3:** Normalize each row of $Z$ to unit length:

$$\tilde{Z}_{ij} = Z_{ij} \Big/ \left( \sum_{k=1}^{K} Z_{ik}^2 \right)^{1/2}, \qquad 1 \leq i, j \leq n. \tag{7}$$

Each row of $\tilde{Z}$ is considered as an embedding of the corresponding original sample.

**Step 4:** Cluster the $n$ row vectors of $\tilde{Z}$ into $K$ clusters, using any suitable clustering algorithm. Correspondingly, $K$ clusters of original samples are acquired.

Ng *et al.* [34] use $K$-means clustering to cluster the $n$ row vectors of $\tilde{Z}$ in Step 4 of the aforementioned algorithm. The result of $K$-means clustering depends heavily on the initialization. To handle this problem, Yu and Shi [21] propose an effective multiclass spectral clustering algorithm. The algorithm inputs the embedding matrix $\tilde{Z}$ and outputs a partition indication matrix $P \in \mathbb{R}^{n \times K}$, where $P_{ik}$ is a measure of the tendency of sample $i$ to be in cluster $k$. The algorithm follows.

**Step 1:** Initialize an orthogonal matrix $R = [r_1, r_2, \ldots, r_K]$ $(r_i \in \mathbb{R}^{K \times 1})$ by:

Randomly selecting $i \in \{1, 2, \ldots, n\}$, set $r_1 = [\tilde{Z}_{i1}, \ldots, \tilde{Z}_{iK}]^T$, and $c = 0^{n \times 1}$;

For $k = 2, \ldots, K$ do

a) $c = c + abs(\tilde{Z} r_{k-1})$ (8)

b) $i = \arg\min_j (c_{j,1})$ (9)

c) $r_k = [\tilde{Z}_{i,1}, \ldots, \tilde{Z}_{i,K}]^{\mathrm{T}}$ (10)

**Step 2:** Find the optimal discrete solution $P$ that is an $n \times K$ matrix by

$P = \tilde{Z}R$, $X = 0^{n \times K}$.

For $i = 1, \ldots, n$, do

a) $l = \arg\max_k P(i,k),\ k \in \{1, \ldots, K\}$ (11)

b) $X_{i,l} = 1$

**Step 3:** Find the optimal orthogonal matrix $R$ by:

SVD decomposition for $X^T P : X^T P = U \Omega V^T$, producing three $K \times K$ matrices $U$, $\Omega$, and $V$, where $\Omega$ is a diagonal matrix.

$R = VU^T$.

**Step 4:** If the sum of the diagonal elements of $\Omega$ converges, then stop and output $P$; otherwise, go to Step 2.

In practice, the aforesaid algorithm converges only after several iterations. Each sample $i$ is assigned to the cluster $k$ for which $P_{ik}$ is a maximum.

## III. Unsupervised Active Learning Framework

### A. Overview of the Framework

As discussed in Section I, to solve the problems in supervised active learning, i.e., inefficiency in dealing with the semantic gap between the distribution of samples in the feature space and their labels, failure to identify new samples whose categories have not yet appeared in the training samples, and lack of adaptability to changes in the labeling, we propose a novel active learning framework based on unsupervised learning. Clustering algorithms are used to cluster the available labeled samples, and then, the "high-quality" clusters are selected. In each of these high-quality clusters, the samples are similar, and most of the samples have the same label. Each high-quality cluster is given the label that most of the samples in it share. If a new sample falls into one of the existing high-quality clusters, it is given the label of that cluster. If the new sample does not fall into an existing high-quality cluster, it is considered as highly informative and selected for manual classification. In our framework, label information is not used for clustering. Thus, the relation between the distribution of samples in the feature space and their labels is decoupled, thus avoiding the semantic gap problem.

Our active learning framework is hierarchical. Two-layer graph-theoretic clustering is used. Dominant-set clustering and spectral clustering are chosen for the two layers, because they complement each other. Dominant-set clustering with a strong definition of dominant sets can obtain high-quality clusters at the earlier steps in the bipartition clustering procedure. However, at the later steps, the quality of the produced clusters goes down remarkably. If spectral clustering is applied to the samples that are not clustered at the earlier steps of dominant-set clustering, better clusters are obtained. Thus, we use dominant-set clustering for the first layer of clustering and then spectral clustering for the second layer of clustering.

In our framework, the first layer consists of a set of dominant-set clustering machines (DSCMs), and the second layer consists of one spectral clustering machine (SCM). For a DSCM, a set of high-quality clusters is obtained by dominant-set clustering.

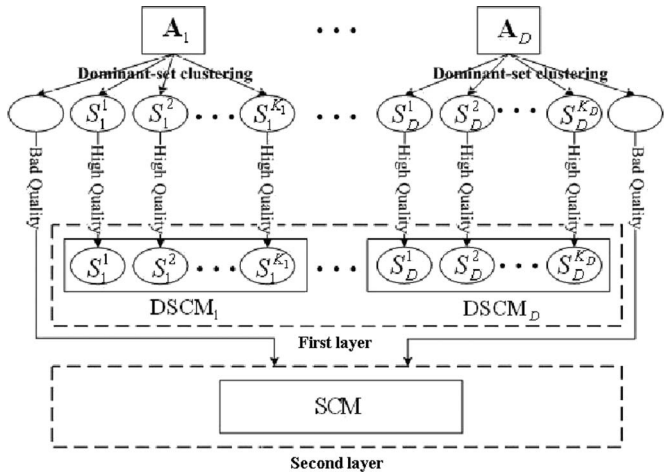



Fig. 1. Initialization phase of our framework.

For an SCM, a set of clusters is obtained by, using spectral clustering, clustering the samples that are not properly dealt with in the first layer. When a new sample is input, DSCMs are used to examine whether it falls into a cluster in the first layer. If the sample does not fall into any cluster in the first layer, it is delivered to the second layer and dealt with by the SCM. If the sample cannot be assigned to any cluster in the SCM, it is selected for manual classification.

The active learning framework has two phases: the initialization phase, in which the DSCMs and the SCM are constructed, and the functioning phase, in which new samples are tested and the framework is updated.

### B. Initialization Phase

Fig. 1 shows the process of initialization. The available labeled samples are split into $D$ smaller subsets if the set of labeled samples is large. This is because the clustering efficiency is low if the number of samples is large. The number of samples in a subset is empirically chosen to ensure that the samples in the subset can effectively be clustered. If the set of labeled samples is small, it is not needed to split it into subsets. For each subset $d$, the corresponding affinity matrix $A_d$ is obtained, and dominant-set clustering is applied to $A_d$. As only high-quality clusters are needed, dominant-set clustering is terminated when the quality of the most recent dominant set is low (it is noted that this approach for terminating the dominant-set clustering is an improvement on the standard termination for which each vertex in the graph has to be assigned to a dominant set).

To evaluate the quality of clusters, we define the cluster purity based on the definition of the importance factors of labels in clusters. Suppose that cluster $i$ contains $n_i^l$ samples with the label $l$. Let $N^l$ be the number of samples with label $l$ in all the samples for the current clustering and $N_i$ be the number of samples in cluster $i$. We define the importance factor $f_i^l$ of label $l$ with respect to cluster $i$ as

$$f_i^l = \frac{n_i^l}{N_i} \times \frac{n_i^l}{N^l}. \tag{12}$$

This means that the more the proportion of the samples with label $l$ in cluster $i$ to the total samples in cluster $i$ and the more proportion of the samples with label $l$ in cluster $i$ to the total samples with label $l$ in the set of samples for the current clustering, the more important the label $l$ with respect to cluster $i$. Let $l_1$ and $l_2$ be the two labels whose importance factors $f_i^{l_1}$ and $f_i^{l_2}$ are the largest and the second to largest with respect to cluster $i$, respectively. We define the purity $\rho_i$ of cluster $i$ as $\rho_i = f_i^{l_1}/f_i^{l_2}$. This means that if labels $l_1$ and $l_2$ are similarly important (i.e., $\rho_i$ is small), cluster $i$ is considered to be impure, and if $\rho_i$ is large (i.e., label $l_1$ predominates over cluster $i$), cluster $i$ is considered to be pure, and it is labeled as $l_1$.

During the period that dominant-set clustering is carried out for each subset of labeled samples, the process of bipartition terminates when the purity of the most recent cluster is less than a predefined threshold $\theta_P$. Then, for the subset $d$ of the labeled samples, $K_d$ dominant-set clusters $\{S_d^1, S_d^2, \ldots, S_d^{K_d}\}$ with high quality and their corresponding labels $\{l_d^1, l_d^2, \ldots, l_d^{K_d}\}$ are obtained (in our approach, high quality means pure). These dominant-set clusters are put together to form the DSCM $d$. This way, we obtain $D$ DSCMs $\mathrm{DSCM}_1, \mathrm{DSCM}_2, \ldots, \mathrm{DSCM}_D$, which are kept in the first layer.

In each subset of labeled samples, there are a number of samples that do not fall into the high-quality clusters constructed by dominant-set clustering. All such unclustered samples in all the subsets are collected and clustered in the second layer. The purities of the resulting clusters and the embedding matrix of the samples in the SCM are calculated and stored for use in the functioning phase.

### C. Functioning Phase

The functioning phase is the core of our active learning framework. Its task is to decide whether a new sample is informative. If so, then it is selected for manual classification; otherwise, it is automatically classified. The DSCMs in the first layer are used to find the cluster that the new sample best matches. If the new sample is an outlier to the best-matching cluster, it is fed to the second layer and dealt with by the SCM. If the SCM assigns it to a cluster with low purity, the sample is selected for manual classification and added to its best-matching cluster in the SCM to improve the SCM's capability. If the number of samples in all the clusters in the SCM is large enough (exceeds a predefined threshold $\theta_S$), then the samples in the SCM are promoted to the first layer to construct a new DSCM. The threshold $\theta_S$ is empirically selected to ensure that reclustering the samples in the SCM clusters using the dominant-set clustering can give a good result. Fig. 2 shows an overview of the functioning phase. In the following, functions in the first and second layers and framework upgrade in the functioning phase are detailed.

*1) Functions in the First Layer:* When a new sample $h$ is input to the first layer, it is decided whether the sample $h$ falls into one of the dominant-set clusters in a DSCM $d$ in the first layer. Referring to [14], the decision algorithm is described as follows. Let $a$ be the affinity vector describing the similarities

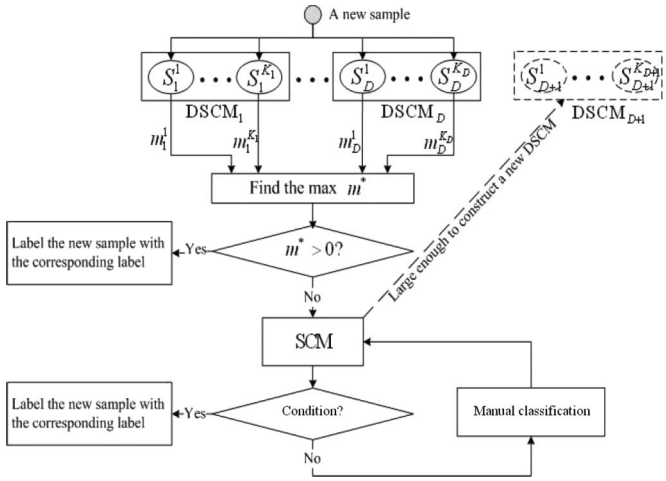



Fig. 2. Functioning phase of our framework.

between $h$ and the existing samples in the $k$th cluster $S_d^k$ in the DSCM $d$. Then, the membership $m_d^k$ of $h$ related to cluster $k$ is defined as

$$m_d^k = \frac{|S_d^k| - 1}{|S_d^k| + 1}\left(\frac{a^T u^k}{f(u^k)} - 1\right) \tag{13}$$

where $|S_d^k|$ denotes the number of samples in $S_d^k$, $f(*)$ is the object function of the program (4), and $u^k$ is the vector whose vertex support set is $S_d^k$. We find, from all the clusters in all the DSCMs, the cluster $S^*$ to which the sample $h$ has the largest membership $m^*$

$$S^* = \underset{d,k}{\operatorname{argmax}}\left(m_d^k\right). \tag{14}$$

If the membership $m^*$ corresponding to $S^*$ is less than or equal to zero, then $h$ is an outlier for all the DSCMs, and it is fed to the second layer. Otherwise, $h$ falls into the cluster $S^*$, and $h$ is given the same label as the cluster $S^*$. The aforementioned sample decision algorithm demands little computational resource, and it is thus suitable for large-scale online tasks.

*2) Functions in the Second Layer:* In the second layer, an embedding for each new sample $h$ fed from the first layer is constructed referring to [4]. Based on the constructed embedding, it is determined whether $h$ falls into one of the clusters in the SCM. Let $A$ be the affinity matrix corresponding to the $n$ samples existing in the SCM. Let $a$ be the affinity vector describing the similarities between the new sample $h$ and the samples in the SCM. A vector $w \in \mathbb{R}^n$ is calculated by

$$w(i) = a(i)\left(\sum_{j=1}^{n} A_{ij} + a(i)\right)^{-1/2}. \tag{15}$$

A variable $r$ is calculated by

$$r = \frac{n+1}{n}\left(\sum_{j=1}^{n} \mathrm{a}(j)\right)^{-1/2}. \tag{16}$$

Then, the embedding vector $y \in \mathbb{R}^K$ of the new sample $h$ is calculated by $y = rw^T \tilde{Z} \Lambda^{-1}$, where $\Lambda \in \mathbb{R}^{K \times K}$ is the diagonal matrix consisting of the $K$ largest eigenvalues of the matrix $A$, and $\tilde{Z}$ is the existing embedding matrix corresponding to $A$. The embedding vector $y$ is added to the existing embedding matrix $\tilde{Z}$ as the last row of the extended embedding matrix $\hat{Z}$.

From the extended embedding matrix $\hat{Z}$, an extended partition indication matrix is obtained using the algorithm introduced in Section II-B. The last row of the extended partition indication matrix indicates the tendencies that the new sample is clustered into all the clusters in the SCM. Let the last row of the extended partition indication matrix be vector $v$. We find the maximum and the second to maximum components in the vector $v$. Let these two components $v(I_1)$ and $v(I_2)$ correspond to clusters $I_1$ and $I_2$ in the SCM, respectively. If $v(I_1)/v(I_2)$ is small, it is regarded that the new sample $h$ has no sufficient tendency to fall into cluster $I_1$, and sample $h$ cannot automatically be classified according to the clusters in the second layer. Instead, it is regarded as an informative sample that is suitable for manual classification. If $v(I_1)/v(I_2)$ is large, and if the purity of cluster $I_1$ is large, then sample $h$ is given the label of cluster $I_1$; otherwise, sample $h$ is manually labeled.

*3) Framework Upgrade:* After manual classification of a new sample is complete, it is checked whether the new sample belongs to a new category that has not yet appeared in the samples. If so, then a new cluster in the SCM is created, and the sample is added into the cluster; otherwise, the new sample with its label is added to the cluster $I_1$ into which the sample has the maximum tendency to fall, and the purity of cluster $I_1$ is recalculated. This way, the discriminative capability of the SCM is improved, and new samples whose categories have not yet appeared in the samples can be dealt with. When the number of samples in all the clusters in the SCM exceeds the threshold $\theta_S$, all of the samples are taken out to construct a new graph to which dominant-set clustering is applied to obtain a number of high-quality clusters. These high-quality clusters are promoted to the first layer and form a new DSCM. The samples that cannot form high-quality clusters are kept in the second layer to form a smaller SCM. This way, the discriminative capability of the first layer of the framework is increased. Thus, with the development of the active learning process, the framework is continuously upgraded, and the discriminative capability of the first layer and that of the second layer both increase gradually.

### D. Discussion

We make the following comments on our framework.

1) **The hierarchical architecture, in which dominant-set clustering and spectral clustering are used in the first and second layers, respectively, is reasonable.** First, dominant-set clustering can generate clusters of very high quality at the earlier steps of the clustering procedure due to the strong definition of dominant sets. The DSCM clusters are of a much higher quality than the SCM clusters. The discriminative capability of DSCMs is higher than that of the SCM. Thus, DSCMs should be put in the first layer to make the framework more efficient in dealing with new samples. Second, for spectral clustering, the number of clusters must be fixed beforehand, and it cannot be large if the computational cost has to be kept low. As the number of the samples that are not clustered into high-quality dominant-set clusters is relatively small, the number of clusters required for representing the distribution of these samples is small, and spectral clustering can generate better clusters for these samples. Therefore, spectral clustering can be used as a supplement to dominant-set clustering, and thus it is put in the second layer.
2) **The architecture is very flexible. It can be easily upgraded online.** During the active learning process in our framework, manually provided valuable labeling information is frequently added to the framework, and its discriminative capability is gradually enhanced. In most previous supervised active learning approaches, the classifiers have to be entirely retrained from time to time to make use of the manual labeling information. In our framework, the DSCMs that were previously constructed need not be altered at all. The upgrade of the first layer only adds new DSCMs. Our framework upgrading online is more flexible and more adaptable to changing environments than the supervised active learning approaches that are upgraded offline.
3) **Our active learning framework is easily extended.** Any clustering algorithm that produces high-quality clusters can be introduced into the framework. The dominant-set clustering algorithm is just one particular choice. The hierarchical strategy can easily combine different clustering algorithms. As long as the different unsupervised clustering algorithms complement each other to some extent, they can be arranged on different layers to deal with new samples in turn. In our framework, just two layers are used, but there is no limit to the number of layers.
4) **Our framework can deal with the semantic gap in active learning.** In our unsupervised active learning framework, the effect of the clustering is to divide the feature space into many parts among which the labels are distributed. Samples that are far from each other can have the same label, while samples that are near to each other can have different labels. Thus, the semantic gap between the distribution of samples in the feature space and the labeling is partly solved.
5) **Our framework is adaptable to changes in the labeling of samples, arising from a new semantic interpretation.** This is demonstrated by the example of classifying arthropods. If ladybird and spider samples are given the same label under a new semantic interpretation, a change in the architecture of the framework is not needed; the only requirement is to memorize the new labels of the clusters. However, supervised active learning needs to reconstruct the classifier according to the new labeling.
6) **New samples whose category has not yet appeared in the training samples can be dealt with.** In our

TABLE I
NUMBERS OF SAMPLES IN VARIOUS CATEGORIES IN THE KDDCUP99 SET

| | Normal behaviors | Attack | | | | Total |
|---|---|---|---|---|---|---|
| | | DOS | U2R | R2L | PROBE | |
| Training set | 97278 | 391458 | 52 | 1126 | 4107 | 494021 |
| Test set | 60593 | 223298 | 39 | 5993 | 2733 | 292300 |

framework, new samples for manual classification are selected by contrasting new samples with the current clusters. The features of the samples belonging to a new category are usually quite different from the features of the current clusters. Thus, these samples can be identified. Once a new category has manually been established, later samples are automatically classified by the upgraded framework. The supervised active learning approaches that select, for manual classification, those new samples that fall close to decision boundaries may fail to identify samples from new categories, as they are usually far from the training samples but not near to decision boundaries.

## IV. EVALUATIONS

We use the data from network intrusion detection, image classification, and video classification to evaluate the performance of our unsupervised active learning framework.

### A. Intrusion Detection

Network intrusion detection [7] is particularly suitable for active learning, because network data are of huge size, and the network environment is always changing. Batch classification of a large amount of network data is unable to respond to these changes.

We use the KDDCUP99 set, which is a standard network intrusion detection data set, to evaluate our active learning framework for intrusion detection. In this data set, each TCP/IP connection was labeled, and 41 features were extracted. There are attack samples and samples of normal behaviors. The attack samples are divided into four categories: DOS, U2R, R2L, and PROBE. Table I shows the numbers of samples in various categories in the training set and in the test set. For any two samples with the same category, their similarity is set to $\exp(-x)$, where $x$ is their Euclidean distance. For any two samples with different categories, their similarity is set to zero. The threshold $\theta_P$ for cluster purity is set to 100. The number of clusters for the spectral clustering in the second layer is set to the number of sample categories. This is because, after the first layer of clustering, only a small number of samples are left in the second layer, and only a small number of clusters are required to represent the distribution of the samples in the second layer. As there are five categories of network behaviors in the data set, the number of clusters in the second layer is set to five. The threshold $\theta_S$ is set to 3000.

*1) Initialization:* In the initialization phase, we randomly select five subsets of samples from the training set to construct five DSCMs. Table II shows the numbers of samples in various categories in each of the subsets, the total numbers of samples in various categories used in the initialization phase, and the ratio of the number of selected samples in each category to the number of all the samples in the category in the training set (the latter number is shown in Table I). The samples of U2R and R2L have to be repeatedly and jointly used in the sample subsets, as there are so few samples in these two categories in the training set. Samples in the other categories in the subsets are disjoint. In total, 16 178 prelabeled samples, i.e., only 3.3% of the samples in the training set, are used to initialize the active learning framework. The initialization algorithm introduced in Section III-B is used to construct the DSCMs and the SCM.

TABLE II
NUMBERS OF SAMPLES IN VARIOUS CATEGORIES USED IN THE INITIALIZATION PHASE

| | Normal | Attack | | | | Total |
|---|---|---|---|---|---|---|
| | | DOS | U2R | R2L | PROBE | |
| Each subset | 1000 | 1500 | 52 | 500 | 500 | 3552 |
| Total samples | 5000 | 7500 | 52 | 1126 | 2500 | 16178 |
| Selected ratio | 5.14% | 1.92% | 100% | 100% | 60.87% | 3.27% |

TABLE III
NUMBERS OF SAMPLES IN DIFFERENT CATEGORIES SELECTED FOR MANUAL CLASSIFICATION

| | Normal | Attack | | | | Total |
|---|---|---|---|---|---|---|
| | | DOS | U2R | R2L | PROBE | |
| Numbers | 634 | 2178 | 5 | 573 | 78 | 3468 |
| Ratios | 1.05% | 0.98% | 12.82% | 9.56% | 3.28% | 1.19% |

*2) Manual Classification:* Samples in the test set are used to evaluate the active learning capability of our framework. In the functioning phase, all the 292 300 samples in the test set are input as new samples into our framework in a random order. When the manual classification of a sample is requested, the chosen label is the label of that sample in the test set. Table III shows the numbers of samples in the different categories selected for manual classification in the whole functioning phase, and the ratio of the number of selected samples in each category to the total number of samples in the category in the test set (the latter number is shown in Table I). It can be seen that 3468 samples are selected for manual classification in all. This means that our framework reduces human efforts more than 84 $(292\,300/3468)$ times for the test set.

From Table III, it can be seen that the categories U2R and R2L have much higher manual classification ratios than the other categories. For the category U2R, samples in both the training set and the test set are too few. This affects sample clustering in both the layers in our active learning framework. As for the category R2L, the semantic gap problem is severe. Table IV shows the number of the samples in each subcategory of the category R2L in the training and test sets. Although these samples have a common label R2L, they are widely distributed in the feature space. For example, the *Wareclient* samples are far from the *Guess_passwd* samples. From the manual classification rate for the category R2L, it can be seen that when the semantic gap is large, our active learning framework

TABLE IV
NUMBERS OF R2L SAMPLES IN SUBCATEGORIES IN TRAINING AND TEST SETS

| R2L | Training set | Test set |
|---|---|---|
| ftp_write | 8 | 3 |
| Guess_passwd | 53 | 4367 |
| Imap | 12 | 1 |
| Multihop | 7 | 18 |
| Phf | 4 | 2 |
| Spy | 2 | 0 |
| Wareclient | 1020 | 0 |
| Warezmaster | 20 | 1602 |

TABLE V
CLASSIFICATION RESULTS OF THE INTRUSION DETECTION SAMPLES THAT OUR FRAMEWORK AUTOMATICALLY CLASSIFIES

| | Normal | DOS | U2R | R2L | PROBE | *Recall* |
|---|---|---|---|---|---|---|
| Normal | 57879 | 1453 | 104 | 147 | 376 | 96.53% |
| DOS | 6071 | 214034 | 249 | 359 | 407 | 96.80% |
| U2R | 4 | 1 | 23 | 4 | 2 | 67.65% |
| R2L | 173 | 99 | 2 | 5141 | 5 | 94.85% |
| PROBE | 57 | 3 | 0 | 0 | 2239 | 97.39% |
| *Precision* | 90.18% | 99.28% | 6.08% | 90.98% | 73.92% | 96.71% |

may pick out a relatively large number of samples for manual classification. This is reasonable, because the semantic gap causes multiple clusters for the samples with the same label, and the labeled samples used to represent these clusters must be sufficient.

*3) Accuracy:* Table V shows the classification results of the samples that our active learning framework automatically classifies, i.e., without manual classification, tested on the test set. In the table, each row (excluding the last row) shows the number of samples in each category into which our framework automatically classifies the samples in one category. For example, Row 4 shows that 34 U2R samples are automatically classified into four normal samples, one DOS sample, 23 U2R samples, four R2L samples, and two PROBE samples. The last column in Table V indicates each category's recall that is the ratio of the number of correctly classified samples in this category to the total number of samples in this category in the test set. The last row indicates each category's precision that is the ratio of the number of correctly classified samples in this category to the total number of samples assigned to this category, consisting of the samples correctly classified as this category and incorrectly classified as this category from the other categories. It can be seen that the recall and precision for all categories except the category U2R are high. The reason why the recall and precision for the category U2R are low is that there are few training samples in this category. It is noted that, although there are large semantic gaps in the category R2L, the recall and precision for this category are still high. Thus, our framework deals effectively with the semantic gap problem.

Two measures are commonly used to judge the performance of network intrusion detection algorithms. One is the detection rate that is the ratio of the number of attacks correctly detected to the number of all attacks. The other is the false-alarm rate that is the ratio of the number of normal network behaviors incorrectly classified as attacks to the number of all the normal behaviors. Table VI shows the detection rate and the false-alarm rate of our framework and other published detection rates and false-alarm rates tested on the KDDCUP data set. The detection rate and the false-alarm rate of our framework are 97.25% and 3.47%, respectively. This result is among the best ones for intrusion detection. It is significantly better than the result of the SVM algorithm [27] that is commonly used for active learning. It is noted that, for our framework, the sum of the number of samples used for initialization and the number of samples manually classified is much less than the total number of samples in the training set. However, the other algorithms listed in Table VI use all the training samples. Therefore, our framework has very encouraging performance for intrusion detection.

TABLE VI
COMPARISON BETWEEN OUR FRAMEWORK AND OTHER ALGORITHMS

| Methods | Detection rates | False alarm rates |
|---|---|---|
| Genetic Clustering [25] | 79% | 0.3% |
| Hierarchical SOM [26] | 90.04%-93.46% | 2.19%-3.99% |
| SVM [27] | 91%-98% | 6%-10% |
| Bagged C5 [28,29] | 91.81% | 0.55% |
| RSS-DSS [30] | 89.2%-94.4% | 0.27%-3.5% |
| Our framework | 97.25% | 3.47% |

TABLE VII
PERFORMANCE OF SPECTRAL CLUSTERING AND DOMINANT-SET CLUSTERING

| Approaches | Detection rates | False alarm rates |
|---|---|---|
| Spectral clustering | 90.31% | 7.86% |
| Dominant-set clustering | 92.32% | 6.59% |
| Our framework | 97.25% | 3.47% |

Table VII shows the detection and false-alarm rates when spectral clustering or dominant-set clustering is separately applied to the KDDCUP data set. It can be seen that the detection and false-alarm rates of our framework are much better than those that are separately obtained by spectral clustering or dominant-set clustering. This is because our hierarchical graph-theoretic clustering combines the merits of spectral clustering and dominant-set clustering.

## B. Image Classification

Image classification is an important prerequisite for applications such as image retrieval. In this section, we evaluate the performance of our active learning framework for classifying images.

We use color correlograms [31] of images as features for classification. A color correlogram includes the color histogram of an image and the distances between pixels, which contain spatial information about the image. For simplicity, the self-color correlogram, in which only the distances between pixels falling into the same bin of the color histogram are included, is used in our experiments.

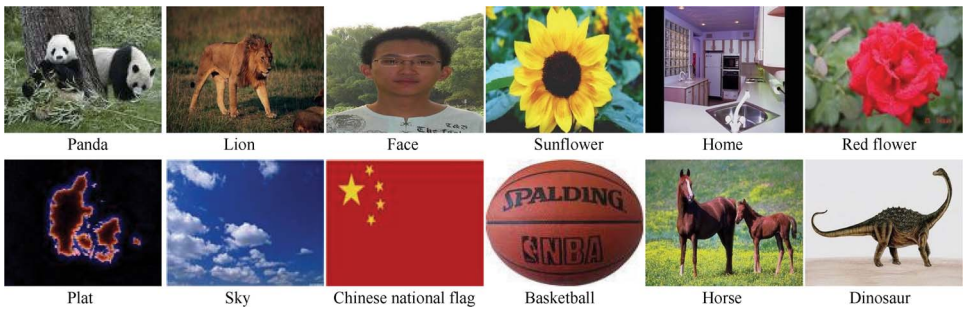



Fig. 3. Categories of images.

TABLE VIII
NUMBERS OF IMAGES IN VARIOUS CATEGORIES

| Categories | Total number of images | Number of images for initialization | Number of images for functioning |
|---|---|---|---|
| Panda | 500 | 100 | 400 |
| Lion | 400 | 100 | 300 |
| Face | 1700 | 500 | 1200 |
| Sunflower | 300 | 100 | 200 |
| Home | 200 | 60 | 140 |
| Red flower | 1000 | 200 | 800 |
| Plat | 300 | 100 | 200 |
| Sky | 600 | 100 | 500 |
| National flag | 400 | 80 | 320 |
| Basketball | 500 | 100 | 400 |
| Horse | 300 | 0 | 300 |
| Dinosaur | 300 | 0 | 300 |
| *Total-1* | 5900 | 1440 | 4460 |
| *Total-2* | 6500 | 1440 | 5060 |

TABLE IX
NUMBERS OF IMAGES SELECTED FOR MANUAL CLASSIFICATION USING OUR FRAMEWORK

| Categories | Manual classification | Manual classification ratio |
|---|---|---|
| Panda | 23 | 5.75% |
| Lion | 42 | 14.00% |
| Face | 56 | 4.67% |
| Sunflower | 50 | 25.00% |
| Home | 8 | 5.71% |
| Red flower | 0 | 0.00% |
| Plat | 0 | 0.00% |
| Sky | 0 | 0.00% |
| National flag | 6 | 1.87% |
| Basketball | 26 | 6.50% |
| Horse | 98 | 32.67% |
| Dinosaur | 104 | 34.67% |
| *Total-1* | 211 | 4.73% |
| *Total-2* | 413 | 8.16% |

We downloaded, from the Internet, 6500 images that can be classified into 12 categories: pandas, lions, faces, sunflowers, homes, red flowers, plat, sky, flags, basketballs, horses, and dinosaurs. Fig. 3 shows one example image from each category. Table VIII shows the numbers of various categories' images separately used in the initialization phase and in the functioning phase. No images of horses and dinosaurs are used in the initialization phase. This is intentional to test our framework's ability to identify new images that belong to new categories that have not yet appeared in the training samples. The second to the last row, namely, "Total-1," shows the total numbers of images for each column, but omitting the images of horses and dinosaurs, while the last row, namely, "Total-2," shows the totals for all the categories. As the number of images used in the initialization phase is small, the images are not split into subsets, i.e., only one DSCM is used. The threshold for cluster purity $\theta_P$ is set to 64. The number of clusters in the second layer is set to 12, which is equal to the number of image categories.

In the following, we first evaluate the performance of our framework for classifying images, calculating the manual classification rate and the accuracy rate for automatic classification, and then make a comparison between our framework and the SVM-based active learning.

*1) Performance Evaluation:* Table IX shows the numbers of images that are selected for manual classification in different categories and, for each category, the ratio of the number of images selected for manual classification to the total number of images used in the functioning phase. In the table, the second to the last row, namely, "Total-1," shows the total number of manually classified images for the second column and the average manual classification rate for the third column, but omitting the images of horses and dinosaurs, while the last row, namely, "Total-2," shows the total and the average for all the categories.

Table X shows the classification results that are automatically obtained by our framework, i.e., with manual classification. In our framework, some samples are classified by dominant-set clustering, and the other samples are classified by spectral clustering. The second column in the table shows the numbers of images that dominant-set clustering correctly and incorrectly classifies, respectively. The third column shows the numbers of images that spectral clustering correctly and incorrectly classifies, respectively. The fourth column gives the classification results for all the images in the category indicated by the leftmost entry of each row. For example, Row 2 shows that our framework automatically classifies 377 panda images into 375 panda images and two face images. The fifth and sixth columns

TABLE X
CLASSIFICATION RESULTS OF IMAGES AUTOMATICALLY CLASSIFIED BY OUR FRAMEWORK

| Categories | Dominant-set (True/False) | Spectral (True/False) | Classifications | Recall | Precision |
|---|---|---|---|---|---|
| Panda | 250/0 | 125/2 | Panda: 375; Face: 2 | 99.47% | 89.71% |
| Lion | 4/1 | 226/27 | Lion: 230; Panda: 6; Face: 6; Sunflower: 2; Home: 10; Red flower: 4 | 89.15% | 93.50% |
| Face | 10/1 | 1106/27 | Face: 1116; Panda: 9; Home: 19 | 97.55% | 98.59% |
| Sunflower | 0/0 | 138/12 | Sunflower: 138; Panda: 2; lion: 4; Face: 4; Home: 2 | 92.00% | 98.57% |
| Home | 34/2 | 92/4 | Home: 126; Face: 4; Red flower: 2 | 95.45% | 80.25% |
| Red flower | 26/2 | 772/0 | Red flower: 798; Lion: 2 | 99.75% | 99.01% |
| Plat | 200/0 | 0/0 | Plat: 200 | 100.00% | 100.00% |
| Sky | 215/0 | 285/0 | Sky: 500 | 100.00% | 100.00% |
| National flag | 210/0 | 104/0 | National flag: 314 | 100.00% | 100.00% |
| Basketball | 296/0 | 76/2 | Basketball: 372; Red flower: 2 | 99.47% | 100.00% |
| Horse | 0/0 | 163/39 | Panda: 23; Lion: 16; Horse: 163 | 80.69% | 100.00% |
| Dinosaur | 0/0 | 157/39 | Panda: 39; Dinosaur: 157 | 80.19% | 100.00% |
| *Total-1* | 1245/6 | 2924/74 | / | 98.12% | 98.12% |
| *Total-2* | 1245/6 | 3244/152 | / | 96.60% | 96.60% |

show the recall and precision rates for automatically classifying images in each category. In the table, the second to the last row, namely, "Total-1," shows the total numbers of images for the second and third columns and the average recall and precision for the fifth and sixth columns, but omitting the images of horses and dinosaurs, while the last row, namely, "Total-2," shows the totals and the averages for all the categories.

From Tables IX and X, the following points are deduced.

1) There are few errors for dominant-set clustering. This supports the use of dominant-set clustering in the first layer of the framework.
2) Many images are automatically classified by spectral clustering, and the corresponding classification accuracy is high. Therefore, there is good complementarity between dominant-set clustering and spectral clustering.
3) If the images of horses and dinosaurs are omitted, then the average accuracy rate for automatically classifying images is 98.12%; otherwise, it is 96.60%. The performance of our framework for automatically classifying images is good.
4) If the images of horses and dinosaurs are omitted, then the average manual classification rate is 4.73%; otherwise, it is 8.16%. Our framework efficiently reduces human burden for manually classifying images.

Furthermore, in the experiments, it is shown that, for the samples of horses and dinosaurs, almost all such samples that are input first into our framework are selected for manual classification, and afterward, most of such new samples are automatically labeled while a good classification accuracy rate is maintained. This indicates that our framework effectively deals with new samples that belong to new categories that have not yet appeared.

*2) Comparison:* For classifying images, we made a comparison between our framework and the SVM-based active learning that is the most typical of the existing supervised active learning approaches. The SVM-based active learning for classifying images uses the same image features and the same images in the initialization phase and the functioning phase as those in our active learning framework. To implement the SVM-based active learning, we use LibSVM [32], which is a simple and efficient software for SVM classification and regression. In LibSVM, the SVM type is set to "C-svc," and the kernel type is set to "radial basis function." The optimal parameters of LibSVM can be automatically estimated, and the estimated optimal values for the parameters are "$c = 4$ and $\gamma = 0.125$." For each test image, LibSVM outputs the probabilities that the test sample is in the various categories. The largest and the second to the largest probabilities are selected. If the ratio of these two probabilities is small, the test image is selected for manual classification.

TABLE XI
NUMBERS OF IMAGES SELECTED FOR MANUAL CLASSIFICATION USING SVM

| Categories | Manual classification | Manual classification rate |
|---|---|---|
| Panda | 20 | 5.00% |
| Lion | 26 | 8.67* |
| Face | 9 | 0.75% |
| Sunflower | 34 | 17.00% |
| Home | 26 | 18.57% |
| Red flower | 11 | 1.38% |
| Plat | 0 | 0.00% |
| Sky | 4 | 0.80% |
| National flag | 8 | 2.50% |
| Basketball | 18 | 4.50% |
| Horse | 124 | 41.33% |
| Dinosaur | 117 | 39.00% |
| *Total-1* | 156 | 3.50% |
| *Total-2* | 397 | 7.85% |

Table XI shows the numbers of images manually classified in various categories and the manual classification rates for the SVM-based active learning. Table XII shows the classification results of the images that the SVM-based active learning automatically classifies. From the comparisons between Tables IX

TABLE XII
CLASSIFICATION RESULTS OF IMAGES AUTOMATICALLY CLASSIFIED BY SVM

| Categories | SVM True/False | Classifications | Recall | Precision |
|---|---|---|---|---|
| Panda | 374/6 | Panda: 374; Lion: 2; Face: 4 | 98.42% | 63.82% |
| Lion | 251/23 | Lion: 251; Panda: 2; Face: 12; Sunflower: 5; Home: 4 | 91.61% | 65.54% |
| Faces | 1184/7 | Face: 1184; Lion: 2; Home: 5 | 99.41% | 92.79% |
| Sunflower | 148/18 | Sunflower: 148; Face: 16; Lion: 2 | 89.16% | 94.87% |
| Home | 84/30 | Home: 84; Face: 28; Red flower: 2 | 73.68% | 82.35% |
| Red flower | 773/16 | Red flower: 773; Lion: 2; Face: 14 | 97.97% | 98.60% |
| Plat | 200/0 | Plat: 200 | 100.00% | 100.00% |
| Sky | 490/6 | Sky: 490; Face: 6 | 98.79% | 100.00% |
| National flag | 312/0 | National flag: 312 | 100.00% | 100.00% |
| Basketball | 374/8 | Basketball: 374; Lion: 2; Red flower: 6 | 97.91% | 100.00% |
| Horse | 2/174 | Panda: 45; Lion: 104; Face: 12; Sunflower: 3; Home: 9; Red flower: 3; Horse: 174 | 1.14% | 100.00% |
| Dinosaur | 1/182 | Panda: 165; Lion: 18; Dinosaur: 1 | 0.55% | 100.00% |
| *Total-1* | 4190/114 | / | 97.35% | 97.35% |
| *Total-2* | 4193/470 | / | 89.86% | 89.86% |

and XI, and between Tables X and XII, it can be seen that when only the images whose categories appeared in the initialization phase are counted, i.e., the images of horses and dinosaurs are omitted, the recall and precision of our framework are higher than those of the SVM-based approach, while the manual classification rate of our framework is also higher than that of the SVM-based approach. The two approaches have the same performance in this case. This is because there is almost no semantic gap between these image samples. The ability of our framework to deal with the semantic gap does not function for these image samples. When only the images of horses and dinosaurs are counted, most of these images are mistakenly classified by the SVM-based approach, resulting in a much lower classification accuracy for these images than that of our framework, while its manual classification rate is higher than our framework. Therefore, our framework has much more ability to deal with new images whose categories have not yet appeared in the training images. Overall, our framework outperforms the SVM-based active learning for classifying images.

### C. Video Classification

Video classification [20] is an important prerequisite for vision applications such as video annotation. In this section, we evaluate our active learning framework for classifying videos.

We use $T$-bin histograms [33] of videos as features. A number of images are selected from each video at a uniform frame interval. For each of these images, a color correlogram is extracted. The color correlograms are used to cluster these images into $T$ clusters. For each cluster, we construct a model that is a weighted sum of the color correlograms of the images in the cluster. For a given video, the correlogram of each image frame is compared with each model, and for each model, the number of images whose correlograms best match the model is counted. These numbers, i.e., one for each model, form a $T$-dimensional vector whose normalized version is used as the feature for the video.

We collected 740 sports videos from 17 categories: body mechanics, football, boxing, volleyball, billiards, shooting, skating, shadowboxing, kickboxing, badminton, weight lifting, swimming, extreme games, judo, aerobics, tennis, and table tennis, as shown in Fig. 4, where each category of videos is represented by one image. Table XIII shows the numbers of various categories' videos separately used in the initialization phase and in the functioning phase. No videos of tennis and table tennis are used in the initialization phase. This is intentional to test our framework's ability to identify videos that belong to new categories that have not yet appeared in the training videos. The second to the last row, namely, "Total-1," shows the total number of videos for each column, but omitting the videos of tennis and table tennis, while the last row, namely, "Total-2," shows the totals for all the categories. As the number of videos used in the initialization phase is small, the videos are not split into subsets. The threshold $\theta_P$ is set to 16. The number of clusters in the second layer is set to 17, which is equal to the number of video categories.

*1) Performance Evaluation:* Table XIV shows the numbers of videos selected for manual classification in different categories and, for each category, the ratio of the number of videos selected for manual classification to the total number of videos used in the functioning phase. The second to the last row, namely, "Total-1," shows the total number of manually classified videos for the second column and the average manual classification rate for the third column, but omitting the videos of tennis and table tennis, while the last row, namely, "Total-2," shows the total and the average for all the categories. Table XV shows the classification results of the videos that our active learning framework automatically classifies, i.e., without manual classification. The definitions of each column and the last two rows in the table are the same as those in Table X. From Tables XIV and XV, the same characteristics of our active learning framework are deduced, as shown in classifying images, i.e., there are few errors for dominant-set clustering; spectral clustering is a good complementarity to


Fig. 4. Categories of videos.

TABLE XIII
NUMBERS OF VIDEOS IN VARIOUS CATEGORIES

| Categories | Total number of videos | Number of videos for initialization | Number of videos for functioning |
|---|---|---|---|
| Body mechanics | 64 | 20 | 44 |
| Football | 56 | 20 | 36 |
| Boxing | 32 | 12 | 20 |
| Volleyball | 32 | 12 | 20 |
| Billiards | 68 | 20 | 48 |
| Shooting | 36 | 12 | 24 |
| Skating | 64 | 20 | 44 |
| Shadowboxing | 32 | 12 | 20 |
| Kickboxing | 32 | 12 | 20 |
| Badminton | 32 | 12 | 20 |
| Weight lifting | 28 | 12 | 16 |
| Swimming | 24 | 12 | 12 |
| Extreme games | 116 | 48 | 68 |
| Judo | 36 | 12 | 24 |
| Aerobics | 24 | 12 | 12 |
| Tennis | 32 | 0 | 32 |
| Table tennis | 32 | 0 | 32 |
| *Total-1* | 676 | 248 | 428 |
| *Total-2* | 740 | 248 | 492 |

TABLE XIV
NUMBERS OF VIDEOS SELECTED FOR MANUAL CLASSIFICATION USING OUR FRAMEWORK

| Categories | Manual classification | Manual classification rate |
|---|---|---|
| Body mechanics | 3 | 6.82% |
| Football | 3 | 8.33% |
| Boxing | 6 | 30.00% |
| Volleyball | 1 | 5.00% |
| Billiards | 4 | 8.33% |
| Shooting | 4 | 16.67% |
| Skating | 4 | 9.09% |
| Shadowboxing | 0 | 0.00% |
| Kickboxing | 4 | 20.00% |
| Badminton | 2 | 10.00% |
| Weight lifting | 1 | 6.25% |
| Swimming | 0 | 0.00% |
| Extreme games | 5 | 7.35% |
| Judo | 1 | 4.17% |
| Aerobics | 1 | 8.33% |
| Tennis | 22 | 68.75% |
| Table tennis | 21 | 65.63% |
| *Total-1* | 39 | 9.11% |
| *Total-2* | 82 | 16.67% |

dominant-set clustering; our framework can effectively deal with new samples whose categories have not yet appeared. In the experiments, it is shown that the misclassifications in Table XV are due to similar features extracted from videos in different categories.

*2) Comparison:* Our active learning framework for classifying videos is compared with the SVM-based active learning approach. In the SVM-based active learning for video classification, the values for the optimal parameters of LibSVM are automatically estimated as "$c = 512$ and $\gamma = 0.00048828125$." As in Section IV-B2, we select the largest probability and the second to the largest probability from the probabilities that a test video is classified into various categories, and we check whether the ratio of these two probabilities is small enough to determine whether the test video is to be selected for manual classification.

Table XVI shows the numbers of videos selected for manual classification and the manual classification rates for the SVM-based approach. Table XVII shows the classification results of the videos that the SVM-based approach automatically classifies. From the comparisons between Tables XIV and XVI, and between Tables XV and XVII, it can be seen that when the videos of tennis and table tennis are omitted, the average manual classification rates of the SVM-based approach and our framework are 10.98% and 9.11%, and the average classification accuracy rates of the SVM-based approach and our framework are 89.76% and 89.20%, respectively. Therefore, when only the videos whose categories have appeared in the

TABLE XV
CLASSIFICATION RESULTS OF VIDEOS AUTOMATICALLY CLASSIFIED USING OUR FRAMEWORK

| Categories | Dominant-set True/False | Spectral True/False | Classifications | Recall | Precision |
|---|---|---|---|---|---|
| Body mechanics | 35/3 | 1/2 | Body mechanics: 36; Swimming: 2; Extreme games: 2; Aerobics: 1 | 87.80% | 97.30% |
| Football | 19/8 | 6/0 | Football: 25; Body mechanics: 1; Badminton: 2; Extreme games: 5 | 75.76% | 83.33% |
| Boxing | 6/1 | 4/3 | Boxing: 10; Kickboxing: 4 | 71.43% | 100.00% |
| Volleyball | 19/0 | 0/0 | Volleyball: 19 | 100.00% | 100.00% |
| Billiards | 18/10 | 11/5 | Billiards: 29; Football: 5; Shooting: 10 | 65.91% | 96.67% |
| Shooting | 12/0 | 7/1 | Shooting: 19; Billiards: 1 | 95.00% | 65.52% |
| Skating | 30/3 | 6/1 | Skating: 36; Extreme games: 4 | 90.00% | 97.30% |
| Shadowboxing | 20/0 | 0/0 | Shadowboxing: 20 | 100.00% | 95.24% |
| Kickboxing | 11/0 | 5/0 | Kickboxing: 16 | 100.00% | 80.00% |
| Badminton | 18/0 | 0/0 | Badminton: 18 | 100.00% | 90.00% |
| Weight lifting | 13/0 | 2/0 | Weight lifting: 15 | 100.00% | 100.00% |
| Swimming | 7/1 | 4/0 | Swimming: 11; Shadowboxing: 1 | 91.67% | 73.33% |
| Extreme games | 46/1 | 15/1 | Extreme games: 61; Skating: 1; Swimming: 1 | 96.83% | 83.56% |
| Judo | 23/0 | 0/0 | Judo: 23 | 100.00% | 100.00% |
| Aerobics | 9/1 | 0/1 | Aerobics: 9; Swimming: 1; Extreme games: 1 | 81.82% | 90.00% |
| Tennis | 0/0 | 6/4 | Tennis: 6; Skating: 2; Extreme games: 2 | 60.00% | 100.00% |
| Table tennis | 0/0 | 11/0 | Table tennis: 11 | 100.00% | 100.00% |
| *Total-1* | 286/28 | 61/14 | / | 89.20% | 89.20% |
| *Total-2* | 286/28 | 78/18 | / | 88.78% | 88.78% |

TABLE XVI
NUMBERS OF VIDEOS SELECTED FOR MANUAL CLASSIFICATION USING SVM

| Categories | Manual classification | Manual classification rate |
|---|---|---|
| Body mechanics | 5 | 11.36% |
| Football | 3 | 8.33% |
| Boxing | 5 | 25.00% |
| Volleyball | 1 | 5.00% |
| Billiards | 10 | 20.83% |
| Shooting | 9 | 37.50% |
| Skating | 6 | 13.64% |
| Shadowboxing | 0 | 0.00% |
| Kickboxing | 2 | 10.00% |
| Badminton | 0 | 0.00% |
| Weight lifting | 0 | 0.00% |
| Swimming | 3 | 25.00% |
| Extreme games | 1 | 1.47% |
| Judo | 1 | 4.17% |
| Aerobics | 1 | 8.33% |
| Tennis | 23 | 71.88% |
| Table tennis | 5 | 15.63% |
| *Total-1* | 47 | 10.98% |
| *Total-2* | 75 | 15.12% |

initialization phase are counted, the manual classification rate of our framework is much lower than that of the SVM-based approach, and the recall and precision of our framework are slightly lower than those of the SVM-based approach. The reason for this is that our framework deals better with the semantic gaps existing in the video samples. When only the videos of tennis and table tennis are counted, most of these videos are mistakenly classified by the SVM-based approach, resulting in a much lower classification accuracy of the SVM-based approach than that of our framework. Thus, our framework outperforms the SVM-based active learning for classifying videos, particularly in terms of dealing much more efficiently with new videos whose categories have not yet appeared in the training samples.

## V. CONCLUSION

In this paper, a new active learning framework based on unsupervised learning has been proposed. In the framework, two promising graph-theoretic clustering approaches, namely, dominant-set clustering and spectral clustering, are applied in a two-layer structure and complement each other to boost the active learning capability. Our active learning framework is flexible to upgrade, adaptable to changes in the labeling, and easy to implement. Experiments on network intrusion detection, image classification, and video classification have shown that our framework can greatly reduce the workload of manual classification while maintaining favorable accuracy for the automatic classification of samples. It has been shown that, overall, our framework outperforms the SVM-based supervised active learning, particularly in terms of dealing much more efficiently with new samples whose categories have not yet appeared in the training samples.

Our future work will focus on the following.

1) We will investigate the merging of clusters with similar characteristics in the functioning phase—discarding redundant samples in these clusters.
2) We will combine supervised learning and unsupervised learning to make active learning more efficient.

TABLE XVII
CLASSIFICATION RESULTS OF VIDEOS AUTOMATICALLY CLASSIFIED BY SVM

| Categories | SVM True/False | Classifications | Recall | Precision |
|---|---|---|---|---|
| Body mechanics | 35/4 | Body mechanics: 35; Badminton: 1; Extreme games: 2; Aerobics: 1 | 89.74% | 94.59% |
| Football | 21/12 | Football: 21; Body mechanics: 1; Badminton: 2; Extreme games: 9 | 63.64% | 84.00% |
| Boxing | 13/2 | Boxing: 13; Kickboxing: 2 | 86.67% | 100.00% |
| Volleyball | 19/0 | Volleyball: 19 | 100.00% | 100.00% |
| Billiards | 27/11 | Billiards: 27; Football: 3; Shooting: 8 | 71.05% | 100.00% |
| Shooting | 13/2 | Shooting: 13; Body mechanics: 1; Football: 1 | 86.67% | 61.90% |
| Skating | 36/2 | Skating: 36; Extreme games: 2 | 94.74% | 97.30% |
| Shadowboxing | 20/0 | Shadowboxing: 20 | 100.00% | 90.91% |
| Kickboxing | 17/1 | Kickboxing: 17; Extreme games: 1 | 94.44% | 89.47% |
| Badminton | 20/0 | Badminton: 20 | 100.00% | 86.96% |
| Weight lifting | 16/0 | Weight lifting: 16 | 100.00% | 100.00% |
| Swimming | 7/2 | Swimming: 7; Shadowboxing: 2 | 77.78% | 87.50% |
| Extreme games | 66/1 | Extreme games: 66; Skating: 1 | 98.51% | 83.54% |
| Judo | 23/0 | Judo: 23 | 100.00% | 100.00% |
| Aerobics | 9/2 | Aerobics: 9; Swimming: 1; Extreme games: 1 | 81.82% | 90.00% |
| Tennis | 1/8 | Skating: 2; Extreme games: 6; Tennis: 1 | 11.11% | 100.00% |
| Table tennis | 1/26 | Extreme games: 26; Table tennis: 1 | 3.70% | 100.00% |
| *Total-1* | 342/39 | / | 89.76% | 89.76% |
| *Total-2* | 344/73 | / | 82.49% | 82.49% |

## REFERENCES


[1] M. Almgren and E. Jonsson, "Using active learning in intrusion detection," in *Proc. 17th IEEE Comput. Security Found. Workshop*, 2004, pp. 88–98.

[2] J. G. Auguston and J. Minker, "An analysis of some graph theoretical clustering techniques," *J. Assoc. Comput. Mach.*, vol. 17, no. 4, pp. 571–588, Oct. 1970.

[3] F. R. Bach and M. I. Jordan, "Learning spectral clustering," in *Proc. Neural Inf. Process. Syst.*, Cambridge, MA: MIT Press, 2004, vol. 16, pp. 305–312.

[4] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. L. Roux and M. Ouimet, "Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering," in *Proc. Neural Inf. Process. Syst.*, Cambridge, MA, MIT Press, 2004, vol. 16, pp. 177–184.

[5] R. C. Dubes and A. K. Jain, "Clustering methodologies in exploratory data analysis," *J. Advances Comput.*, vol. 19, pp. 113–228, 1980.

[6] S. C. H. Hoi and M. R. Lyu, "A semi-supervised active learning framework for image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005, vol. 2, pp. 20–25.

[7] W. M. Hu, W. Hu, and S. Maybank, "AdaBoost-based algorithm for network intrusion detection," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 2, pp. 577–583, Apr. 2008.

[8] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice–Hall, 1988.

[9] R. Jenssen, T. Eltoft, and J. C. Principe, "Information theoretic spectral clustering," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2004, vol. 1, pp. 25–29.

[10] F. Jing, M. Li, H. J. Zhang, and B. Zhang, "A unified framework for image retrieval using keyword and visual features," *IEEE Trans. Image Process.*, vol. 14, no. 7, pp. 979–989, Jul. 2005.

[11] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. Neural Inf. Process. Syst.*, Cambridge, MA: MIT Press, 2005, vol. 17, pp. 1601–1608.

[12] U. Luxburg, O. Bousquet and M. Belkin, "Limits of spectral clustering," in *Proc. Neural Inf. Process. Syst.*, Cambridge, MA: MIT Press, 2005, vol. 17, pp. 857–864.

[13] M. Pavan and M. Pelillo, "A new graph-theoretic approach to clustering and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2003, vol. 1, pp. 18–20.

[14] M. Pavan and M. Pelillo, "Efficient out-of-sample extension of dominant-set clusters," in *Proc. Neural Inf. Process. Syst.*, Cambridge, MA: MIT Press, 2005, vol. 17, pp. 1057–1064.

[15] Y. Rong and A. Hauptmann, "Multi-class active learning for video semantic feature extraction," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2004, vol. 1, pp. 27–30.

[16] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[17] G. Tur, R. E. Schapire, and D. Hakkani-Tur, "Active learning for spoken language understanding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, vol. 1, pp. 6–10.

[18] L. Wang, K. L. Chan, and Y. P. Tan, "Image retrieval with SVM active learning embedding Euclidean search," in *Proc. Int. Conf. Image Process.*, 2003, vol. 1, pp. 14–17.

[19] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1101–1113, Nov. 1993.

[20] R. Yan, J. Yang, and A. Hauptmann, "Automatically labeling video data using multi-class active learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, vol. 1, pp. 516–523.

[21] S. X. Yu and J. Shi, "Multiclass spectral clustering," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, vol. 1, pp. 313–319.

[22] C. T. Zahn, "Graph-theoretic methods for detecting and describing gestalt clusters," *IEEE Trans. Comput.*, vol. C-20, no. 1, pp. 68–86, Jan. 1971.

[23] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," *IEEE Trans. Multimedia*, vol. 4, no. 2, pp. 260–268, Jun. 2002.

[24] D.-Q. Zhang, C.-Y. Lin, S.-F. Chang, and J. R. Smith, "Semantic video clustering across sources using bipartite spectral clustering," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2004, vol. 1, pp. 117–120.

[25] Y. G. Liu, K. F. Chen, X. F. Liao, and W. Zhang, "A genetic clustering method for intrusion detection," *Pattern Recognit.*, vol. 37, no. 5, pp. 927–942, May 2004.

[26] S. T. Sarasamma, Q. A. Zhu, and J. Huff, "Hierarchical Kohonen net for anomaly detection in network security," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 2, pp. 302–312, Apr. 2005.

[27] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data," in *Applications of Data Mining in Computer Security*, D. Barbara and S. Jajodia, Eds. Norwell, MA: Kluwer, 2002, ch. 4.

[28] C. Elkan, "Results of the kdd99 classifier learning contest," *SIGKDD Explorations*, vol. 1, no. 2, pp. 63–64, 2000.

[29] B. Pfahringer, "Winning the kdd99 classification cup: Bagged boosting," *SIGKDD Explorations*, vol. 1, no. 2, pp. 65–66, 2000.

[30] D. Song, M. I. Heywood, and A. N. Zincir-Heywood, "Training genetic programming on half a million patterns: An example from anomaly detection," *IEEE Trans. Evol. Comput.*, vol. 9, no. 3, pp. 225–239, Jun. 2005.
[31] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlogram," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 1997, pp. 762–768.
[32] T.-K. Huang, R. C. Weng, and C.-J. Lin, "Generalized Bradley–Terry models and multi-class probability estimates," *J. Mach. Learn. Res.*, vol. 7, pp. 85–115, Jan. 2006.
[33] P. Muneesawang and L. Guan, "iARM—An interactive video retrieval system," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jun. 2004, vol. 1, pp. 285–288.
[34] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing System*, vol. 14, T. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2001, pp. 849–856.
[35] D. Angluin, "Queries and concept learning," *Mach. Learn.*, vol. 2, no. 4, pp. 319–342, Apr. 1988.
[36] D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 1994, pp. 148–156.
[37] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. ACM Workshop Comput. Learn. Theory*, 1992, pp. 287–294.
[38] S. Fine, R. Gilad-Bachrach, and E. Shamir, "Query by committee, linear separation and random walks," *Theor. Comput. Sci.*, vol. 284, no. 1, pp. 25–51, Jul. 2002.
[39] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Mach. Learn.*, vol. 28, no. 2/3, pp. 133–168, 1997.
[40] T. Graepel and R. Herbrich, "The kernel Gibbs sampler," in *Proc. Advances Neural Inf. Process. Syst.*, 2000, vol. 13, pp. 514–520.
[41] C. Campbell, N. Cristianini, and A. J. Smola, "Query learning with large margin classifiers," in *Proc. Int. Conf. Mach. Learn.*, 2000, pp. 111–118.
[42] G. Schohn and D. Cohn, "Less is more: Active learning with support vector machines," in *Proc. Int. Conf. Mach. Learn.*, Stanford, CA, 2000, pp. 839–846.
[43] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," in *Proc. Int. Conf. Mach. Learn.*, Stanford, CA, 2000, pp. 999–1006.
[44] D. Lewis and W. Gale, "A sequential algorithm for training text classifiers," in *Proc. ACM-SIGIR Int. Conf. Res. Develop. Inf. Retrieval*, 1994, pp. 3–12.
[45] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. Int. Conf. Mach. Learn.*, 2001, pp. 441–448.
[46] D. Lewis and W. Gale, "Training text classifiers by uncertainty sampling," in *Proc. ACM Int. Conf. Res. Develop. Inf. Retrieval*, 1994, pp. 3–12.
[47] S. C. H. Hoi, R. Jin, J. K. Zhu, and M. R. Lyu, "Batch mode active learning and its application to medical image classification," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 417–424.
[48] K.-S. Goh, E. Y. Chang, and W.-C. Lai, "Multimodal concept-dependent active learning for image retrieval," in *Proc. ACM Int. Conf. Multimedia*, New York, Oct. 2004, pp. 564–571.
[49] Y. Guo and D. Schuurmans, "Discriminative batch mode active learning," in *Proc. Advances Neural Inf. Process. Syst.*, 2007, pp. 1–8.
[50] W. Hu and W. Hu, "HIGCALS: A hierarchical graph-theoretic clustering active learning system," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2006, vol. 5, pp. 3895–3900.

**Weiming Hu** (SM'07) received the Ph.D. degree from the Zhejiang University, Hangzhou, China.

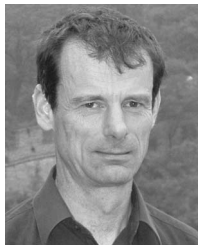
From April 1998 to March 2000, he was a Postdoctoral Research Fellow with the Institute of Computer Science and Technology, Founder Research and Design Center, Peking University, Beijing, China. Since April 1998, he has been with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, where he is currently a Professor and a Ph.D. Student Supervisor. His research interests include visual surveillance, filtering of objectionable Internet information, retrieval of multimedia, and understanding of Internet behaviors. He has published more than 100 papers in national and international journals and international conferences.

**Wei Hu** received the B.E. degree in automation engineering from Tsinghua University, Beijing, China, in 2003 and the M.S. degrees in electronic engineering and computer science from the Institute of Automation, Chinese Academy of Sciences, Beijing, and the State University of New York, Stony Brook, in 2006 and 2008, respectively.

He is currently with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests include computer vision and machine learning.

**Nianhua Xie** received the B.E. degree in automation engineering from the Beijing Institute of Technology, Beijing, China, in 2005. He is currently working toward the Ph.D. degree with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing.

His research interests include image processing, computer vision, and machine learning.



**Steve Maybank** (SM'06) received the B.A. degree in mathematics from King's College, Cambridge, U.K., in 1976 and the Ph.D. degree in computer science from Birkbeck College, University of London, London, U.K., in 1988.

He was with the Pattern Recognition Group, Marconi Command and Control Systems, Frimley, U.K., in 1980 and with the GEC Hirst Research Centre, Wembley, U.K., in 1989. During 1993–1995, he was a Royal Society/Engineering and Physical Sciences Research Council Industrial Fellow with the Department of Engineering Science, University of Oxford, Oxford, U.K. In 1995, he was with the University of Reading, Reading, U.K., as a Lecturer with the Department of Computer Science. In 2004, he was a Professor with the School of Computer Science and Information Systems, Birkbeck College. His research interests include the geometry of multiple images, camera calibration, visual surveillance, information geometry, and applications of statistics to computer vision.