Selectively Guiding Visual Concept Discovery

Maggie Wigness, Bruce A. Draper, and J. Ross Beveridge
Colorado State University
Fort Collins, Colorado
mwigness,draper,ross@cs.colostate.edu

Abstract

Labeling data to train visual concept classifiers requires significant human effort. Active learning addresses labeling overhead by selecting a meaningful subset of data, but often these approaches assume that the set of visual concepts is known in advance. Clustering approaches perform bottom-up discovery of concepts, and reduce labeling effort by moving from instance-based to group-based labeling. Unfortunately, clustering techniques assume a one-to-one mapping between clusters and visual concepts even though learned groups are often not coherent and fail to represent all concepts. We introduce Selective Guidance, a technique that hierarchically clusters data and selectively queries labels of coherent clusters representing different visual concepts. Unlike most active learning and clustering techniques, Selective Guidance does not require any a priori knowledge. Using benchmark data sets we show that Selective Guidance achieves classification accuracy better than active learning and clustering approaches with fewer labeling queries.

1. Introduction

Supervised multi-class visual concept classifiers require large amounts of labeled training data to yield high classification accuracy. Although the collection of visual data has become trivial, the task of labeling large sets of data requires significant human effort. Thus, unsupervised and semi-supervised approaches are emerging with the goal of training accurate classifiers while significantly reducing the labeling workload relative to supervised approaches.

Active learning has been used to reduce the labeling workload without significantly compromising the performance of visual concept classification [5, 6, 7, 8, 12]. Active learning frameworks iteratively search for subsets of unlabeled data to query and label. Selection of the data subset typically involves identifying a diverse set of images that a supervised classifier is uncertain how to label.

While active learning frameworks have reduced the labeling workload, they often require heavy assumptions about the training data. Labeled data is needed to train the initial supervised classifier, so the number and types of visual concepts must be known in advance. These assumptions limit active learning frameworks to data that have been at least broadly analyzed.

Clustering algorithms look for reoccurring patterns in data that indicate similarity, and thereby discover visual concepts bottom-up in unlabeled data [2, 10, 16]. Clustering therefore does not require that visual concepts are known in advance, but many clustering techniques do assume that the number of visual concepts is known. Clustering reduces labeling effort since clusters are labeled instead of individual data instances.

Providing a single label to a cluster of images is most meaningful when the cluster is *pure*, i.e., contains images from the same visual concept. Unfortunately, achieving a perfect partition of large visual data sets is difficult. Variations in intra-class and inter-class similarity make some concepts easy to group while others are incredibly challenging. Impure clusters produce weak training data since not all instances will match the label assigned to the cluster, and there is no guarantee every visual concept will dominate a cluster. Cluster impurity has lead to the emergence of active clustering which iteratively collects pair-wise constraints [1, 17] or examples of true/false positives [4] to improve the clustered output. Over time the clustered output becomes purer, but human effort is introduced to achieve this.

This paper presents a novel technique, Selective Guidance, that discovers visual concepts bottom-up and efficiently labels these concepts to create training data that achieves high classification accuracy. We do this by hierarchically clustering data to create a dendrogram in which to search for visual concepts. Clusters are iteratively selected for labeling by evaluating the likelihood of information gain in terms of exploitation, i.e., collecting labels for a large number of samples, and exploration, i.e., the discovery of new visual concepts. Using benchmark data sets we show that Selective Guidance is able to collect labeled training data that achieves classification accuracy better than existing active learning and clustering techniques with fewer labeling queries. This is achieved because Selective Guidance provides labels to more samples than active learning and discovers more pure visual concepts than clustering techniques. Further, unlike existing techniques, Selective Guidance requires no advance knowledge of the data, making it ideal for completely unlabeled data.

2. Related Work

Two broad techniques are commonly used to solve the problem of collecting labels for unlabeled data: active learning and clustering. Both are motivated by the need to reduce the labeling workload relative to supervised approaches, but make different assumptions about the a priori knowledge that is available.

2.1. Active Learning Frameworks

The primary goal of active learning is to reduce the high cost of annotation while maintaining classification accuracy. Pool-based frameworks assume that an initial set of labeled instances is available, called the *seed set*, in addition to a set of unlabeled data known as the *active pool*. These frameworks iteratively train classifiers by querying for labels on subsets of data from the active pool that are expected to improve classification accuracy.

A variety of selection criteria have proven successful in the visual domain. Holub et al. [5] select samples that minimize the expected entropy of the active pool, whereas Jain and Kapoor [6] select uncertain samples using entropy from a variant of k-nearest neighbor classification. Other uncertainty sampling techniques use the posterior mean and variance from Gaussian processes classification [8], and a multi-class SVM for margin sampling, i.e., the difference between the best (most likely) and second best classes (BvSB) [7]. Li and Guo [12] combine uncertainty sampling with information density to select uncertain samples in dense areas of feature space which are more likely to fit the expected distribution of testing data.

These frameworks yield high classification accuracy with fewer training samples, but many require the number and types of classes to be known a priori to initially train a supervised classifier. These assumptions limit the frameworks to data that are not completely unlabeled. The exception is the BvSB framework [7]. The seed set used in BvSB is chosen randomly from the unlabeled data without regard to class which does not guarantee the labeled subset of training data will include all visual concepts until BvSB discovers them. Thus, BvSB is a rare active learning framework that has been demonstrated under the assumption that data are completely unlabeled.

2.2. Clustering

The primary goal of clustering is to partition unlabeled data into visual concepts. A perfect partition would result in every group containing data from the same concept, and no two groups representing the same concept. This perfect one-to-one mapping produces groups that can be labeled, allowing multiple images to be labeled simultaneously.

The low intra-class similarity and high inter-class similarity found in visual data makes perfect partitions of data difficult. Many techniques focus on feature representation as this plays a crucial role in learning good partitions. Tuytelaars et al. [16] show that different normalization, interest point detectors and dimensionality reduction affect the output. Dai et al. [2] adapt supervised ensemble based learning for unlabeled data to learn improved proximity matrices. Lee and Grauman [10] iteratively learn groups of concepts in order of difficulty and emphasize the use of context descriptors [11] in addition to other features.

Perfect data partitions are difficult to achieve indicated by the evaluation of average cluster purity for these clustering techniques. This measure becomes particularly important when discussing the labeling process, since only a single label is given to a group of images. Active clustering techniques iteratively refine the clustered output after collecting feedback from an annotator about data samples. Feedback has included binary must-link or cannot-link constraints [1, 17], and identification of samples that are true or false positives relative to the majority concept of their cluster [4]. As more feedback is collected, the clustered output gets closer to an ideal one-to-one mapping.

On the whole, clustering techniques make fewer assumptions than active learning because an initial set of labeled data is not required. However, most clustering techniques still assume the number of concepts is known to learn a one-to-one mapping. While this reduces the labeling effort, cluster impurities increase the cognitive load of the labeling process and produce weak training data. Even when assuming the number of visual concepts is known, some concepts may go undiscovered if they do not dominate a cluster. Active clustering has made advances to improve cluster purity, but as a result the human effort increases.

3. Selective Guidance Approach

Selective Guidance (SG) is designed to generalize the task of collecting labeled training data to completely unlabeled data sets. The algorithm makes no assumptions regarding the total number of instances, the number and types of visual concepts or the underlying distribution of instances per class in the data set. Thus, SG does not require any a priori knowledge of the data, but is capable of discovering coherent groups of visual concepts that can be labeled with minimal human effort.

SG uses hierarchical clustering to create a space of potential visual concepts. Discovery and labeling are done iteratively. On each iteration, clusters are evaluated based on their expected information gain. Two estimates of information gain are used: 1) *exploitation* of the unlabeled data and 2) *exploration* of the visual concepts in the unlabeled data. The cluster expected to provide the most information gain is selected for labeling. Details of this algorithm are discussed in the remainder of this section.

3.1. Unsupervised Learning of Visual Concepts

Clustering lends itself well to unsupervised visual concept discovery because it identifies reoccurring patterns within data which can be indicators of interesting concepts. Spectral clustering or k-means require information that SG assumes is unavailable, such as the number of classes in the data. Also, the poor cluster purity achieved using these clustering algorithms (as in [2, 11, 16]) lead us to believe that partitional clustering is not well suited for SG. Instead, SG uses hierarchical clustering to create a hierarchy of m groups, $\mathcal{H} = \{c_1, c_2, \ldots, c_m\}$, from the set of n training samples, $\mathcal{T} = \{x_1, x_2, \ldots, x_n\}$. Each c_i contains at least two training samples, meaning $m \approx n - 1$. The two sample constraint is introduced since SG is designed to avoid instance-based labeling.

Hierarchical clustering gives SG three important properties. First, the number of clusters does not have to be defined in advance. Second, for multi-class data sets the hierarchy will contain both pure and impure clusters. Third, the one-to-one mapping constraint is relaxed which allows visual concepts to group at different locations and levels of the hierarchy.

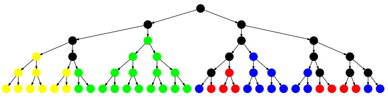
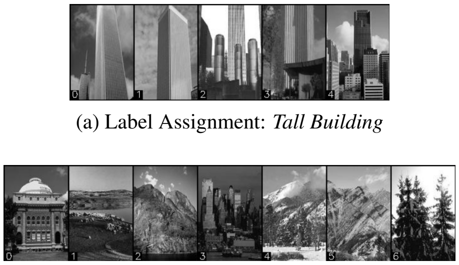


Figure 1: Cluster hierarchy where yellow, green, red and blue indicate pure clusters of different concepts and black represents impure clusters.

Figure 1 is a toy example of a hierarchical clustering for four classes that illustrates these properties. Black circles in the hierarchy indicate impure clusters while the other colors indicate pure clusters for a particular class. Trivially, the root node of the hierarchy must be impure since it contains all images which come from different classes. *Green* appears to be relatively self-similar and easy to group indicated by the formation of a pure cluster high in the hierarchy, while *red* appears to be less self-similar and/or highly similar to *blue* which requires a finer granularity of grouping to form pure clusters.

3.2. Cluster Selection

An SG query asks for a single label for a cluster. Labeling the entirety of $\mathcal{H}$ is redundant, since clusters are subsets of one another. Also, providing a single label to an impure cluster is not well-defined and creates noisy training data. Therefore, if a pure cluster is queried, as in Figure 2a, its visual concept label is provided, and if an impure cluster is queried, as in Figure 2b, the label "mixed" is provided. Thus, SG queries are easy in terms of cognitive load. However, a "mixed" label does not add any information to the set of collected labeled data, so the task of SG is to identify and label a small set of pure clusters in $\mathcal{H}$ to quickly collect meaningful information about $\mathcal{T}$.



(a) Label Assignment: *Tall Building*

(b) Label Assignment: *Mixed*

Figure 2: Examples of pure and impure clusters from the 13-Scenes data set.

Cluster selection is iterative. On each iteration, clusters in $\mathcal{H}$ are in either the purely labeled set $\mathcal{L}$, the mixed labeled set $\mathcal{M}$ or the unlabeled set $\mathcal{U}$. SG is similar to active learning in the sense that $\mathcal{U}$ is the equivalent of an active pool, and samples belonging to clusters in $\mathcal{L}$ can be used as training data for a classifier. On each iteration, a selectivity score based on the expected information gain is given to all clusters in $\mathcal{U}$. The cluster with the maximum score is queried and labeled, and $\mathcal{L}$, $\mathcal{M}$ and $\mathcal{U}$ are updated accordingly. Note that labels of pure clusters are inherited by descendants, and labels of impure clusters are inherited by ancestors. Two selectivity scores, one focusing on exploitation and the other on exploration, are discussed later in this section. First, however, we discuss our approach to estimating cluster purity.

3.2.1 Evaluating Cluster Purity

Our estimation of cluster purity is predicated on the belief that data near each other in feature space are more likely to represent the same visual concept than data that are further away, where near and far depend on the relative density of the data. Thus, clusters in $\mathcal{H}$ that contain samples that are approximate nearest neighbors (ANN) in feature space are likely to be pure. We derive a purity measure from the Proximity Forest data structure [14] which was designed for fast

ANN lookups in general metric spaces. Although any ANN algorithm would suffice for image vector data, the Proximity Forest was shown to return more accurate querying results than other common ANN algorithms [14], and allows SG to later generalize to a broader range of data representations (e.g., videos represented as points on Grassmann manifolds [13]).

A Proximity Forest is a set of T randomized metric trees. Each tree partitions data into τ sized leaf nodes to encode approximate nearest neighborhoods. Thus, data that coexist in leaf nodes of multiple metric trees likely represent the same visual concept. Treating all x_i as nodes in a graph, a weighted edge between x_i and x_j defines the connectivity of the samples, i.e., the number of times the samples coexist in forest leaf nodes. Connectivity is extended to clusters and called the Proximity Forest Connectivity (PFC) measure.

Formally, PFC is based on the calculation of weighted edges between points x_i and x_j, defined as

$$w(x_i, x_j) = \sum_{k=1}^{T} common\_leaf_k(x_i, x_j), \qquad (1)$$

where the function $common\_leaf_k$ finds the leaf nodes in tree k that x_i and x_j belong to, and returns 1 if the leaf nodes are the same, and 0 otherwise. The average edge weight from $x_i \in c_i$ to all $x_j \in c_i$,

$$c(x_i) = \frac{1}{|c_i|} \sum_{\forall x_j \in c_i} w(x_i, x_j), \qquad (2)$$

defines how connected a sample is to its cluster. Finally, since PFC estimates cluster purity, the connectivity of c_i is the average connectivity of all samples in c_i,

$$pfc(c_i) = \frac{1}{|c_i|} \sum_{\forall x_i \in c_i} c(x_i). \qquad (3)$$

Higher PFC scores suggest a greater likelihood of being pure. PFC is computed without supervision since the only information used is the relative distances between samples. Unfortunately, PFC is not size invariant. The connectivity sum in Equation 2 is dependent on the value of τ, meaning x_i can maximally be connected to $\tau - 1$ other data points. However, the connectivity for point x_i is normalized by the size of its cluster c_i. Thus, smaller clusters with the same connectivity as larger clusters will receive higher PFC scores, causing clusters near the leaves of the hierarchy to be favored when evaluating purity. This favoring is accurate, but provides little benefit in terms of minimizing the labeling effort. Further, PFC is not probabilistic, but instead a relative score. The next section discusses details of how SG uses PFC to create a probability of purity for the exploitation score.

3.2.2 Exploitation

Exploitation seeks to label large numbers of samples quickly, making cluster purity and cluster size important factors when calculating information gain. The exploitation score for cluster c is defined as

$$exploit(c) = p_c * l_c, \qquad (4)$$

where p_c is the probability that c is pure and l_c is the number of unlabeled samples in c. Note that l_c is not the size of c since some samples may already be labeled if a descendant of c was selected for labeling in a previous iteration.

p_c is calculated by iteratively modeling PFC scores and cluster sizes from clusters that exist in $\mathcal{L}$ and $\mathcal{M}$. Adding cluster size to the model helps ensure that cluster selectivity is not limited to the bottom of the hierarchy since PFC is not size invariant. As more clusters are labeled, more information is available to predict the ranges of PFC scores and sizes most likely to be pure. The information is modeled using a 10x10 uniform grid of Gaussian radial basis functions (RBF). One axis of the grid represents the range of PFC scores and the other represents the range of cluster sizes. Each axis is normalized to $[0.0, 1.0]$ with an even distribution of grid point centers along these axes.

The RBF grid is modeled online as labeling queries are processed. Each grid point is modeled as the average weighted purity of the current labeled clusters. That is, after t labeling queries, grid point g_i has a modeled purity value of

$$p(g_i) = \frac{1}{t} \sum_{i=0}^{t} \phi(r_i) * v_i, \qquad (5)$$

where r_i is the distance between the grid point center and the cluster queried at iteration i. v_i is 0 if the cluster was labeled "mixed" or 1 if it was given a non-mixed label, and $\phi(r_i)$ is the Gaussian RBF formally defined as $\exp^{-(r_i/\sigma)^2}$, which weights clusters closer to the center of g_i more heavily than clusters further from the center of g_i. For all experiments in this paper, $\sigma = 0.1$.

The value of p_c is calculated from the RBF grid as

$$p_c = \phi(r_c) * p(g_i), \qquad (6)$$

where g_i is selected as the grid point that minimizes r_c. The exploitation score describes the expected number of samples that will receive labels if a cluster is given a non-mixed label, and emphasizes labeling as many samples in as few queries as possible. Focusing solely on exploitation, however, favors the discovery of visual concepts that are easy to group and that dominate the data set, possibly disregarding less common concepts.

3.2.3 Exploration

Exploration focuses on how to discover different visual concepts quickly. Exploration is modeled with the assumption

Data Set	# Concepts	Total Instances	Training Size	Testing Size	Classifier
UCI-Pendigits	10	10,992	5,100	2,000	SVM-Linear
UCI-Letters	26	20,000	7,100	5,000	SVM-RBF
13-Scenes	13	3,859	2,500	500	SVM-Linear
Leaf-100 and Face-100	10	100	70	30	NN
Leaf-250 and Face-250	25	250	175	75	NN

Table 1: Experiment details for benchmark data sets.

that often, different visual concepts will be located in different areas of feature space. Thus, when selecting a cluster from $\mathcal{U}$ to be labeled, it should be far away from the clusters that already exist in $\mathcal{L}$ to try and identify a new visual concept.

The exploration selectivity score is based on a distance value and defined as

$$explore(c) = \min_{\forall c_i \in \mathcal{L}} d(c_i, c), \qquad (7)$$

where d is the Euclidean distance between two cluster centroids. After two non-mixed labeling queries, unlabeled clusters will have multiple distances between the clusters in $\mathcal{L}$. The minimum distance from c to any cluster in $\mathcal{L}$ is used since it represents the difference between c and its most similar labeled neighbor. The cluster in $\mathcal{U}$ with the maximum exploration score represents the cluster that is most dissimilar to its nearest neighbor and expected to be most likely to represent a visual concept that has not been discovered yet.

3.2.4 Combination of Selection Criteria

Both exploitation and exploration provide benefits to the overall goal of SG. For this reason, both selection criteria are combined to provide an overall selection criteria score. The two terms are combined using a weight α that has a range $[0, 1]$. Formally, the combination is defined as

$$SG(c) = \alpha * exploit(c) + (1 - \alpha) * explore(c), \qquad (8)$$

and the cluster with the highest selection score is selected to be labeled. For all experiments in this paper, the exploitation and exploration terms are weighted evenly by setting $\alpha = 0.5$.

Note that neither selectivity score requires retraining a supervised classifier after each labeling query. The only supervised modeling comes at the level of cluster purity within the RBF grid for the exploitation term.

4. Experiments

SG, active learning and active clustering all iteratively query for information to collect labeled training data. Each query is slightly different. SG asks for labels for clusters, active learning asks for labels of individual images and active clustering asks for constraints about whether images should be grouped together. Each querying task, however, is designed to collect information about the training data so it can be used to classify new unseen testing data. The goal is to maximize accuracy while keeping the number of queries to a minimum. Thus, the experiments in this paper compare the classification accuracy achieved by each method as a function of the number of labeling queries.

The focus of comparison is placed on the process of collecting labeled data, so the same classifier is trained for each method being compared. For each experiment, a classifier is also trained using the full set of training data available to indicate the performance of a completely supervised approach. The details of each experiment are given in Table 1. Each experiment is averaged over 20 trials of random training and testing partitions.

Comparisons are made against the following techniques:

SG : Our proposed Selective Guidance approach that iteratively labels clusters.

Wards$_k$: Baseline hierarchical clustering with Wards linkage that cuts the dendrogram to form k clusters (one-to-one mapping) without any annotator feedback.

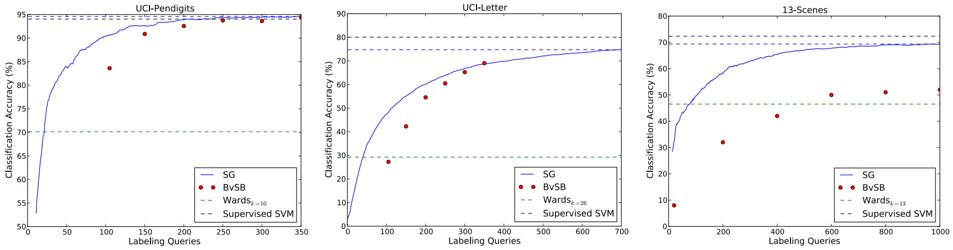
BvSB : Active learning framework that iteratively queries for labels of uncertain samples [7].

FAST-Active-HACC-H1 : Active clustering approach that iteratively queries for must-link or cannot-link constraints to improve clustered output [1].

Wards$_k$ and FAST-Active-HACC-H1 output a set of clusters. Each cluster is assigned the label that represents the dominating visual concept of its images. This label is transferred to all images in the cluster and then used to train a classifier. Classification accuracy depends in part on the number of concepts discovered. Since SG is only capable of collecting labeled data from all k concepts after k labeling iterations, classification accuracy for SG is only shown starting at labeling query k.

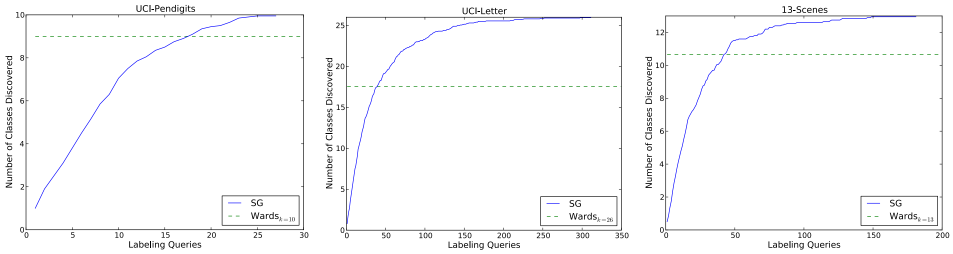
4.1. Selective Guidance vs Active Learning

Three experiments are replicated to make direct comparisons to the BvSB active learning framework. BvSB does not make assumptions about the number or types of visual concepts in the training set, making it a good candidate for



(a) (b) (c)

Figure 3: Classification accuracy per labeling query for the (a) Pendigits, (b) Letter and (c) 13-Scenes data sets.



(a) (b) (c)

Figure 4: Concepts discovered per labeling query for the (a) Pendigits, (b) Letter and (c) 13-Scenes data sets.

comparison to SG. The total number of labeling queries answered in the BvSB framework is the sum of the size of the seed set and the total number of images labeled during active learning iterations. The authors [7] do not report the first classification accuracy result for BvSB until after the seed set has been labeled.

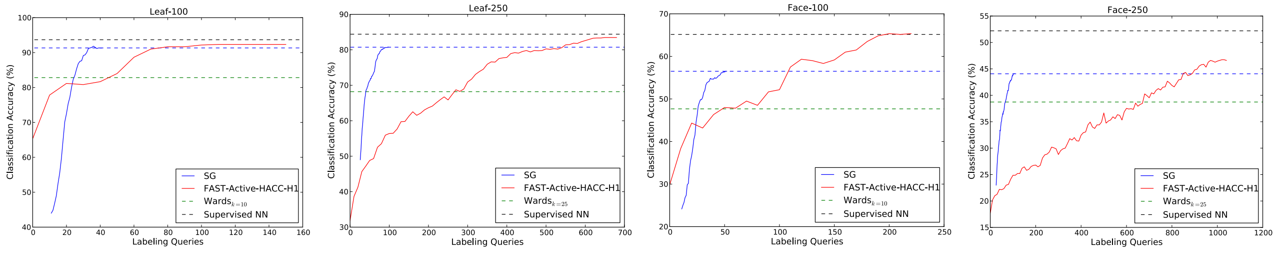
The first two experiments use UCI data sets, Pendigits and Letter. The third experiment uses the 13-Scenes [3] data set which contains images from 13 categories of natural scenes. GIST [15] features are used to represent each scene image just as in the BvSB framework. The exact training and testing partition is not reported for the 13-Scenes data set. We report SG results using a 2,500 training and 500 testing partition, but found similar trends across several other partitions.

Figure 3 shows the classification accuracy per labeling query for the three experiments. For all three experiments, SG outperforms BvSB early on in the labeling process. This suggests that given a time constraint where an annotator is only able to provide a limited number of labeling queries, SG would likely collect a more diverse set of informative data faster than BvSB. On the whole, even after many labeling queries are answered, SG never performs any worse than BvSB. Further, SG always approaches the classification accuracy of a completely supervised approach, but does so with significantly fewer labeling queries.

The performance gap for the 13-Scenes experiment is significantly larger than that seen on the UCI experiments. BvSB never converges to the performance of SG even after 1,000 labeling queries. This is likely due to the fact that the BvSB framework is set up to select 20 samples at each iteration. The authors claim that even though potentially redundant data are being labeled at each active learning round, BvSB still achieves improved results over random selection [7]. This may be true, but because SG first learns to group some of these redundant data, they can be labeled simultaneously with a single labeling query.

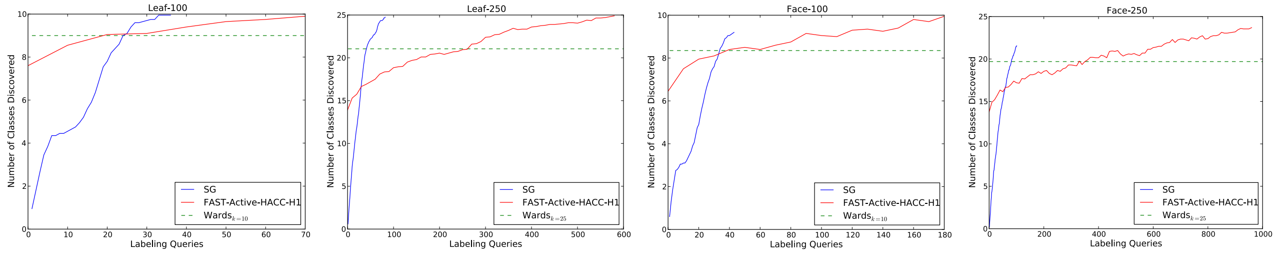
As expected, both SG and BvSB outperform the baseline clustering approach. The baseline method assigns the dominating class label to each cluster regardless of purity which causes weak training data to be collected. For the three experiments, the average cluster purity achieved by Wards$_k$ is 0.81 ± 0.20 for Pendigits, 0.41 ± 0.27 for Letter and 0.50 ± 0.18 for 13-Scenes. This means that on average only about half of the training data received accurate labels for the Letter and 13-Scenes experiments. Looking beyond average cluster purity, Figure 4 shows the number of visual concepts that actually dominate the learned clusters. For all experiments, SG eventually discovers clusters that represent all visual concepts. Wards$_k$ on the other hand, leaves at least one visual concept from each data set undiscovered. The noisy labels and undiscovered concepts affect the ability of Wards$_k$ to accurately train the SVM classifier.

The better classification performance achieved by SG likely comes from the fact that SG provides more images with labels than BvSB after the same number of labeling



(a) (b) (c) (d)

Figure 5: Classification accuracy per labeling query for the (a) Leaf-100, (b) Leaf-250, (c) Face-100 and (d) Face-250 data sets.



(a) (b) (c) (d)

Figure 6: Concepts discovered per labeling query for the (a) Leaf-100, (b) Leaf-250, (c) Face-100 and (d) Face-250 data sets.

queries. Since BvSB is an instance based labeling technique, t images are given labels after t labeling queries. Using SG, 4,906 images from the Pendigits data set were labeled and 2,682 images from the Letter data set were labeled after only 350 labeling queries. Although less extreme, SG provided labels to 1,254 images for the 13-Scenes data set after 1,000 queries. In all cases, SG collects more labeled image samples than BvSB using the same number of labeling queries.

4.2. Selective Guidance vs Active Clustering

Four experiments are performed to compare SG to the active clustering FAST-Active-HACC-H1 [1] approach. FAST-Active-HACC-H1 assumes the number of visual concepts in the data set is known and iteratively re-clusters the data to search for an ideal one-to-one mapping, and labeling queries are defined as binary constraints between two images. At each iteration two images are selected and an annotator determines if they are from the same or different classes.

Two experiments are run using subsets of leaf species from the Leafsnap corpora[1]. The remaining two experiments use subsets from the PubFig [9] data set which includes images of real-world human faces collected from the Internet. 70% of the images from each concept are used as the training set and the remaining 30% are used as the testing set. Publicly available proximity matrices[2] are used for all methods in the experiments.

Figure 5 shows the classification accuracy results for all four experiments. FAST-Active-HACC-H1 performs better than SG when very few queries are answered for the 10 class data sets, Figures 5a and 5c, but within 20 labeling queries the performances of the methods cross and SG performs better. This performance cross can be explained by looking at how quickly each method discovers the different visual concepts, seen in Figures 6a and 6c. SG can only discover visual concepts after a pure cluster is queried and labeled. FAST-Active-HACC-H1 produces a set of clusters after each labeling query, and the number of discovered concepts is based on the set of concepts that dominate each cluster regardless of cluster purity. It appears that many classes in these 10 class data sets are relatively self-similar since FAST-Active-HACC-H1 discovers ≈ 80% and 60% (figures 6a and 6c respectively) of the visual concepts in the initial clustering without any labeling queries. SG however, either selected several clusters that represented the same concept or impure clusters early in the labeling process. After 20 queries, SG has discovered more concepts than FAST-Active-HACC-H1 which is the same point the classification performances cross.

In the 25 class data sets, Figures 5b and 5d, SG hits its

[1] http://leafsnap.com

[2] www.umiacs.umd.edu/~arijit/projects/Active_clustering/active_clustering.html

peak performance with significantly fewer labeling queries than it takes FAST-Active-HACC-H1 to reach this same classification performance. While the discovery for these 25 class data sets, Figures 6b and 6d, have a similar trend to the 10 class experiments, it is likely that the average cluster purity achieved by FAST-Active-HACC-H1 is much lower to begin with. So although FAST-Active-HACC-H1 discovers more concepts than SG with a small number of labeling queries, the collected data is likely very weak whereas SG collects only pure labeled data resulting in better classification performance.

Once again both approaches outperform the baseline method. Although the baseline method only requires k queries after clustering is complete, the lower classification performance caused by not discovering all concepts (seen in Figure 6) and producing weak training data is a major trade-off. FAST-Active-HACC-H1 eventually always outperforms SG and approaches the performance of the supervised classifier, but this is the nature of iterative feedback. The same is true for active learning approaches because eventually all unlabeled samples can be queried and labeled. This is not the case for SG because some training samples may never be labeled if they do not exist in a pure cluster that has at least two images. Notice however, that for the Leaf data sets, SG also approaches the performance of the supervised classifier but with far fewer labeling queries than FAST-Active-HACC-H1.

On the whole, SG outperforms FAST-Active-HACC-H1 on many levels. SG is able to discover a large number of visual concepts at a faster rate than FAST-Active-HACC-H1. Although SG never discovers all visual concepts in the Face subsets, FAST-Active-HACC-H1 requires many labeling queries to approach discovery of all concepts as well, indicating that the data set is very challenging. In all experiments FAST-Active-HACC-H1 does not reach the peak performance achieved by SG until a significant number of labeling queries are answered. In fact, the number of binary constraints answered is typically much larger than the total number of samples in the training set, which is the number of queries a completely supervised classifier requires.

5. Conclusions

Reducing the labeling overhead to collect training data has been addressed in a variety of ways. Active learning selects subsets of instances to label from an unlabeled pool of images. Clustering techniques group unlabeled data by similarities to label multiple images simultaneously. However, many techniques do not generalize to completely unlabeled data sets because they assume knowledge of the number and/or types of visual concepts in the unlabeled data set. We introduce Selective Guidance (SG) as an assumption-free visual concept discovery approach that minimizes human labeling effort. Using benchmark data sets, we showed that SG labels more individual samples with the same number of queries as the BvSB active learning framework, discovers more visual concepts than clustering, and classifies more accurately than active learning or clustering techniques.

Acknowledgments

This research has been supported by funding (W911NF-10-2-0066) from the DARPA Mind's Eye program.

References

[1] A. Biswas and D. Jacobs. Active image clustering: Seeking constraints from humans to complement algorithms. In *CVPR*, pages 2152–2159, 2012. 1, 2, 5, 7

[2] D. Dai, M. Prasad, C. Leistner, and L. Van Gool. Ensemble partitioning for unsupervised image categorization. In *ECCV*, pages 483–496, 2012. 1, 2, 3

[3] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, pages 524–531, 2005. 6

[4] A. Gilbert and R. Bowden. igroup: Weakly supervised image and video grouping. In *ICCV*, pages 2166–2173, 2011. 1, 2

[5] A. Holub, P. Perona, and M. Burl. Entropy-based active learning for object recognition. In *CVPR Workshops*, pages 1–8, 2008. 1, 2

[6] P. Jain and A. Kapoor. Active learning for large multi-class problems. In *CVPR*, pages 762–769, 2009. 1, 2

[7] A. Joshi, F. Porikli, and N. Papanikolopoulos. Multi-class active learning for image classification. In *CVPR*, pages 2372–2379, 2009. 1, 2, 5, 6

[8] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with gaussian processes for object categorization. In *ICCV*, pages 1–8, 2007. 1, 2

[9] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372, 2009. 7

[10] Y. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *CVPR*, pages 1721–1728, 2011. 1, 2

[11] Y. J. Lee and K. Grauman. Object-graphs for context-aware category discovery. In *CVPR*, pages 1–8, 2010. 2, 3

[12] X. Li and Y. Guo. Adaptive active learning for image classification. In *CVPR*, 2013. 1, 2

[13] Y. Lui, R. Beveridge, and M. Kirby. Action classification on product manifolds. In *CVPR*, 2010. 4

[14] S. O'Hara and B. Draper. Are you using the right nearest neighbor algorithm? In *WACV*, 2013. 3, 4

[15] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001. 6

[16] T. Tuytelaars, C. Lampert, M. Blaschko, and W. Buntine. Unsupervised object discovery: A comparison. *IJCV*, 88(2):284–302, 2010. 1, 2, 3

[17] C. Xiong, D. Johnson, and J. J. Corso. Spectral active clustering via purification of the k-nearest neighbor graph. In *ECDM*, 2012. 1, 2