An Application for Efficient Error-Free Labeling of Medical Images

Michal Drozdzal, Santi Seguí, Petia Radeva, Carolina Malagelada,
Fernando Azpiroz, and Jordi Vitrià

Abstract. In this chapter we describe an application for efficient error-free labeling of medical images. In this scenario, the compilation of a complete training set for building a realistic model of a given class of samples is not an easy task, making the process tedious and time consuming. For this reason, there is a need for interactive labeling applications that minimize the effort of the user while providing error-free labeling. We propose a new algorithm that is based on data similarity in feature space. This method actively *explores* data in order to find the best label-aligned clustering and *exploits* it to reduce the labeler effort, that is measured by the number of "clicks. Moreover, error-free labeling is guaranteed by the fact that all data and their labels proposals are visually revised by en expert.

Michal Drozdzal · Petia Radeva · Jordi Vitrià
Dept. Matemàtica Aplicada i Anàlisis,
Universitat de Barcelona,
Barcelona, Spain
Computer Vision Center (CVC),
Universitat Autònoma de Barcelona,
Barcelona, Spain
e-mail: {michal.drozdzal,petia.ivanova,jordi.vitria}@ub.edu

Santi Seguí
Computer Vision Center (CVC),
Universitat Autònoma de Barcelona,
Barcelona, Spain
Dept. Matemàtica Aplicada i Anàlisis,
Universitat de Barcelona,
Barcelona, Spain
e-mail: ssegui@cvc.uab.es

Carolina Malagelada · Fernando Azpiroz
Hospital de Vall d'Hebron,
Barcelona, Spain

A.D. Sappa and J. Vitrià: *Multimodal Interact. in Image & Video Appl.*, ISRL 48, pp. 1–16.
DOI: 10.1007/978-3-642-35932-3_1

1 Introduction

1.1 Motivation and Novelty

For many machine learning applications, the compilation of a complete training set for building a realistic model of a given class of samples is not an easy task. The motto ”there is no data like more data” suggests that the best strategy for building this training set is to collect as much data as possible. But in some cases, when dealing with problems of high complexity and variability, the size of this data set can grow very rapidly, making the learning process tedious and time consuming. Time is expended in two different processes: 1) the labeling process, which generally needs human intervention, and 2) the training process, which in some cases exponentially increments computational resources as more data is obtained.

Labeling is a human activity domain[1], in many learning-based tasks an expert knowledge is needed to label the data samples and expert time and effort are expensive. Moreover, when spending long time on data labeling an oracle/expert gets tired and errors can be introduced thus as a result the labeling can be inconsistent. In the literature the problem of effort minimization in data labeling was addressed by using active learning techniques [19]. It is important to note that in active learning the main focus is put on classification of the data samples. In this sense, the learner collects the labels that are best from point of view of the future data generalization problem.

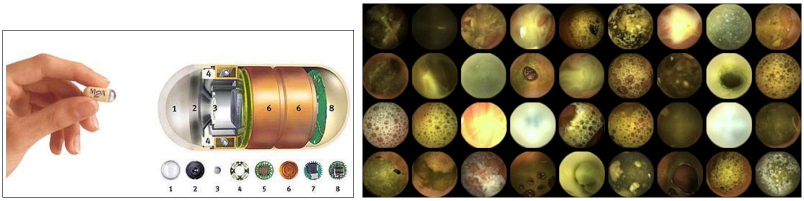
However, there are learning-based applications where all training data must be checked by the oracle to ensure the correctness of the labeling. In this case the main motivation is to minimize the oracle’s effort. This is not a case of direct application of active learning techniques, as they are focused on maximizing classifier performance in test set and not on minimizing the oracle effort when labeling the whole training set. So, active learning techniques are not designed for efficient error-free labeling[2] and thus a need for interactive labeling applications appear that will minimize the effort of the user while providing error-free labeling. An example of application where the correctness of the labeling is of high importance are medical data, where no error can be committed in the training set construction in order to ensure an optimally correct labeling in training process of designed computer-aided classification systems.

1.2 Real-Life Problem

Wireless Capsule Endoscopy (WCE) image analysis (see Fig. 1(a)) is a clear scenario of medical imaging device where it is highly desirable to construct a labeling

[1] Here it is assumed that humans are highly accurate while labeling the data.

[2] The term error-free labeling refers to the fact that all data and its labels proposals are revised by an expert, it is a difference with Active Learning where only some samples are being displayed to the oracle. Clearly it is possible that an expert will miss-label some data according to the criteria of different expert.



(a) (b)

Fig. 1 (a) The wireless video capsule; (b) Non informative frames

error-free training set to learn to classify frames of the endoscopic video. The WCE contains a camera and a full electronic set which allows the radio frequency emission of a video movie in real time. This video, showing the whole trip of the pill along the intestinal tract, is stored into an external device which is carried by the patient. These videos can have duration from 1h to 8h, what means that the capsule captures a total of 7.200 to 60.000 images. WCE videos have been used in the framework of computer-aided systems to differentiate diverse parts of the intestinal tract [11], to measure several intestinal disfunctions [22] and to detect different organic lesions (such as polyps [13], bleeding [12] or general pathologies [4]). In most of these applications, machine learning plays a central role to define robust methods for frame classification and pattern detection.

A common stage in all these research lines is the discrimination of informative frames from non-informative frames. Non-informative frames are defined as frames where the field of view is occluded. Mainly, the occlusion is caused by the presence of intestinal content, such as food in digestion, intestinal juices or bubbles (see Fig. 1(b)). The ability of finding non-informative frames is important since: 1) generally, it helps to reduce time of video analysis, and 2) the majority of non-informative frames are frames with intestinal content that is an indicator for intestinal disfunctions [16, 15].

The main strategy in the search of non-informative frames is the application of machine learning techniques in order to build a two-class classifier. Generally, non-informative frames are characterized by their color information [1]. Robust classifiers can be built when the training set is representative of the data population. The problem that occurs when dealing with WCE videos is the high color variability of non-informative frames in different videos. Therefore, to overcome this problem, we need to construct a wide set of labeled training set of informative and non-informative frames. Here, a naive approach for the labeling process of these video frames could mean the manual annotation of thousands of video frames.

1.3 Problem Set-Up

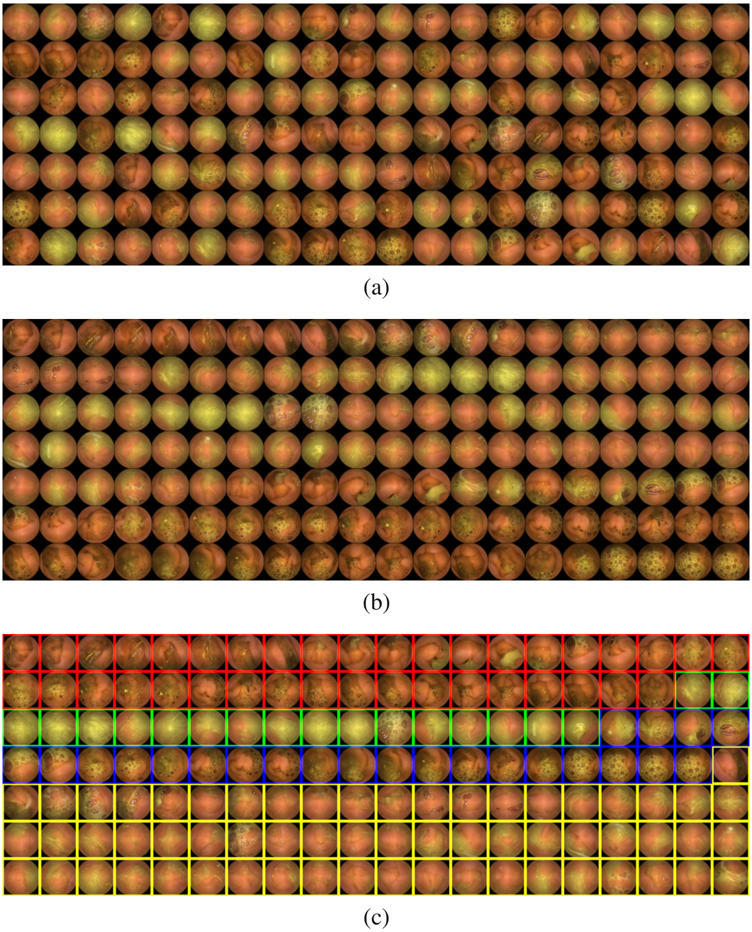
The real life example motivates the need for efficient real error-free labeling. In order to ensure that all data are correctly labeled the oracle has to revise visually all the elements in the data set. A new element will be labeled only if it has been seen by the oracle. The effort of the oracle can be highly minimized by convenient organization of the data before displaying them. First, the system has to come up with a proposal of the label. This proposal is based on the knowledge gained by the system during the labeling process. Then, the human operator faces two possible decisions: to accept the model proposal or to change the label of the sample. In practice, these two options have a non symmetric cost for the human operator: accepting the model proposal can be efficiently implemented with a low cognitive load for the operator assuming it by default, while changing a label has a larger cost consisting in manual intervention while labelling the frame. This cost can be evaluated by the number of its interventions with the system during the labeling process [7]. As a result the oracle has seen all the data samples and has confirmed or has changed the system proposed label so it can be assumed that all labels are correctly assigned. Moreover, in the algorithm discussed in this chapter it is assumed that 1) all data come in a single batch, 2) data distribution is known before starting the labeling process and 3) data can be of more than two classes (possible labels).

1.4 Intuition of Our Approach

The system that minimizes the oracle's effort should address two issues: 1) how the data should be organized while being displayed and 2) which rule the system should follow when giving the proposition of the label.

A natural way to display the data is driven by data similarity and not by randomness (see Fig. 2(a)). It is convenient to display similar samples together. If data was obtained by sequential, in time, process it can be assumed that data acquired in instance i and $i+1$ are similar (see Fig. 2(b)). In case when the data have not been sampled in sequential process or when the samples come from highly dynamic events the similarity can be defined in some feature space. Other possibility is to group similar images using distances between data in some feature space into cluster-structure (see Fig. 2(c) for an example of data grouping in color histogram feature space). The assumption done here is that, in some well defined feature space, it is more probable that the similar samples share same labels than the samples far away in the feature space.

The question is how to find the optimal labeled-aligned cluster structure (see Fig. 3). If we knew all the labels, our problem would become trivial (see Fig. 3(a)): we could present the data to the labeler in an optimal cluster-based organization to minimize the effort. For example: if a cluster is pure (only one class), we can label it (all elements inside the cluster) using only one intervention (e. g. one click). But at the beginning of the labeling process, the labels are unknown (see Fig. 3(b))! This problem, of finding optimal label-aligned cluster structure, can be seen as joint cluster *exploration* and *exploitation* task. *Exploration* is responsible for a discovery of data structure while *exploitation* is in charge of finding an optimal discrimination between classes in the data structure [10].



(a)

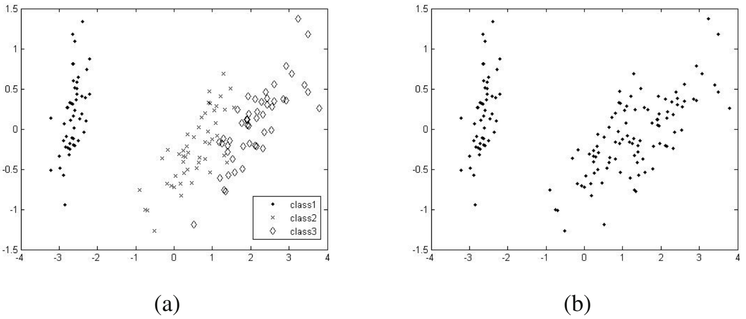
(b)

(c)

Fig. 2 An example of image ordering in case of Wireless Capsule Endoscopy data. Two class classification problem: informative vs. non-informative frames. a) random order, b) sequential in time, c) order according to the frames similarity with respect to the color features (color mark indicates different clusters obtained with k-means algorithm - similar images).

Recently, a novel approach to active label-alligned[3] cluster discovery that does not require any classifier training step has been proposed by Dasgupta and Hsu in [5]. This algorithm, called hierarchical sampling, is a specific case of a more general paradigm called *partition-based sampling* (in statistics also referred as *cluster-based sampling*), characterized by the use of a data structure that represents all possible data clusters or partitions. These methods are aimed to first explore and then to

[3] We will also use the notation of pure cluster(s) referring to well aligned cluster(s) to its label(s).



(a) (b)

Fig. 3 Some example data distribution in 2D (data from IRIS dataset after applying PCA). a) data with all labels uncovered, b) data with unknown labels.

exploit the cluster structure in data. This class of methods, takes advantage of data-structure and aims in active search for an optimal label-aligned partition of the data. Under this assumption, the most critical part is how to find pure clusters as fast as possible by efficiently exploring the data set.

In [5] the chosen data structure is a hierarchical clustering tree based on the Ward's method [23]. This tree, which is computed offline, is a static data structure that is used to guide the active sampling strategy by navigating among all possible prunings of the tree. It has been shown that this active sampling strategy for pure cluster discovery is clearly better than the random sampling strategy in the presence of label-aligned data clusters, and it is not worse (except maybe for some pathological cases) than random sampling when these structures are not present in data.

One of the main advantages of the hierarchical sampling [5] method for active label-aligned data structure discovery is that it is not based on retraining multiple times a classifier (in contrast to many active sampling strategies), but in a sampling process that makes only one assumption on data distribution, that the data are distance-clusterizable in some distance space. The classifier is trained once the exploration of the data structure has produced fairly pure clusters that can be exploited in labeling process resulting in a satisfactory solution.

In the literature there is a vast variety of methods for label proposals where the criteria for choosing the proposal is classifier dependent [19]. Moreover, in general, these strategies are not designed to take advantage of data cluster structure. A simple way to come up with the label proposal in label-aligned cluster structure is the majority label. For each cluster, the system can estimate the majority label using the labels seen so far and treat this label as the most probable label for all the elements inside the cluster. It has been shown in [5] that the cluster-adaptive sampling strategy has theoretical bounds on the empirical estimation of the majority label. These bounds give, at any time, the interval in which the true labeling error lies and the bounds holds with high probability. This property of the sampling algorithm allows for labeling complexity analysis, indicating the number of queries to be seen in order to obtain a given labeling accuracy. The ability for complexity analysis is important since the information about labeling error is a direct estimation of the expected effort needed for labeling of the whole data set.

1.5 Contributions

In this chapter, an application for efficient error-free labeling is presented. The method minimizes the oracle's effort, by using a strategy that minimizes the number of oracle's intervention in the labeling process. This minimization holds when the data can be represented by some label-aligned structure. The algorithm comes with label proposition and the oracle is asked to confirm/change the label. What differs our approach from more classical one, based on active learning, is the necessity of error-free labeling, needed for medical applications, what means that the oracle has to visually revise all the data and their labels. This strategy is build to minimize the number of oracle's interventions.

The chapter is organized as follows, in the next section we reveal some related works. In section 2, we discuss the sampling algorithm and the strategy of showing the images to the oracle. Section 4 presents some experimental results and finally, section 5 concludes.

2 Related Work

The majority of the work that is similar in spirit to our set-up are based around active sampling techniques for active learning. As said before, the main difference is the final objective of each strategy, for active learning it is to get the optimal performance of some classifier with low number of labeled samples. In our case the goal is to label all samples with minimum oracle's effort. Thus, in our case, all samples have to be revised and the goal of the system is two-fold: 1) the data should be displayed in the way that improves oracle performance and 2) the data should come-up with some label proposition. Anyway it is worth revising the literature in the field of active sampling for active learning while it is conceptually similar to our approach. Both are trying to find the data samples that are the most ”favorable” to be displayed and manually labeled.

The active learning strategies vary in the selection strategy in sampling process. For example, *density sampling* methods sample from maximal-density unlabeled regions [17, 18]. On the other hand, *uncertainty sampling* methods sample the regions where the trained classifier is least certain [21]. The combination of density and uncertainty criteria has also been explored and it is called *representative sampling*. This approach explores the clustering structure of uncertain samples for selecting the most representative ones [24]. Using a different heuristic, *instability sampling* approaches are based on sampling from those regions that maximally change the classifier decision boundary [8].

In order to increase the efficiency of these sampling strategies, several research lines have been proposed. One of the most successful lines is to take benefit from the use of ensemble learning methods [14]. For example, *Query-by-committee* [9] selects samples that cause maximum disagreement amongst an ensemble of hypotheses. Another interesting line has been the development of hybrid methods, such as the method presented in [6], where the parameters selection strategy can be adaptively updated after each actively sampled point.

There are works that are not based implicitly on active learning but on concept of interactive labeling with the objective on efficient data labeling. Usage of Self-Organizing Maps (SOMs) to group similar elements in data set and to display them jointly to the user is investigated in [2]. In [25] the authors analyse the use of Bayesian networks to produce refined labeling. There is also a group of works that are based on refining the initial clustering structure with the help of user interaction [20, 3].

3 Methodology

In order to build an application for efficient label-free labeling, the following elements should be considered:

1. An engine responsible for *exploration* and *exploitation* of the data structure (e. g. partition space).
2. A strategy for choosing the elements to be displayed, for both, individual data elements due to structure *exploration* step and a group of elements representing some part of the data structure to enable to the user the possibility to exploit the data structure.
3. An interface to show the data according to displaying strategy and to accept user interactions with the system.

Fig. 4 shows the application flowchart. The core of our approach is based on the Dasgupta's algorithm [5] that is presented in Section 3.1. Section 3.2 presents how to adapt the algorithm output to display to the expert/oracle. In Section 3.3 the interface is presented.

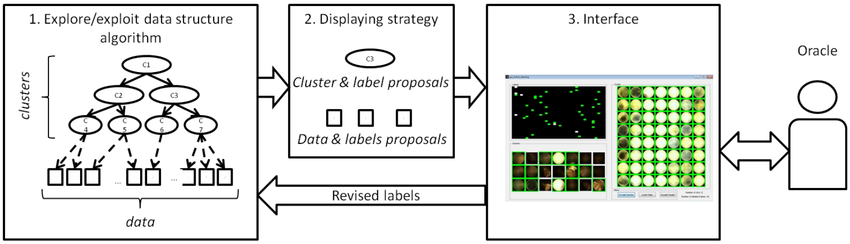


Fig. 4 Application flowchart

3.1 Algorithm for Data Structure Explortaion/Explotation

We propose to base our approach on a modified formulation of *partition based active learning* [5]. The paradigm can be described in its most general terms in the following way: Let $\mathbf{X} = \{\mathbf{x_1}, ..., \mathbf{x_n}\}$ be the set of unlabeled data, $\mathbf{C}$ be a possible partition set of $\mathbf{X}$ and C be an element of this partition such that:

$$\bigcup_{C_k \in \mathbf{C}} C_k = \mathbf{X}$$

The objective under this paradigm is to *efficiently search through the space of partitions* for a partition $\mathbf{C}^{opt}$ of $\mathbf{X}$ such that all data points belonging to an element $C_k \in \mathbf{C}^{opt}$ can be automatically labeled with high confidence by using small number of labeled data. Its basic steps are represented by Algorithm 1.

The algorithm has four associated procedures that need some explanation: `Select`, `Display samples`, `Bound` and `Search`.

The procedure `Bound` estimates the mislabeling error within a given element C of the partition given the samples seen so far. An elegant way of implementing this procedure is to use the bounds derived from the tails of the binomial or multinomial distribution [5].

Suppose there are η possible labels and that their proportions in a given element C of the partition are $p_{C,l}$ for $l = 1, \dots, \eta$. Then, the error induced by assigning all points in C to the majority label is $\varepsilon_C = 1 - max_l(p_{C,l})$. At any given iteration k of the algorithm, we can associate with each element C of the partition $\mathbf{C}^k$ a confidence interval within which we expect the true probability $p_{C,l}$ to lie: $[p^{LB}_{C,l}, p^{UB}_{C,l}] = [max(p_{C,l} - \frac{1}{m} - \sqrt{\frac{p_{C,l}(1-p_{C,l})}{m}}, 0), min(p_{C,l} + \frac{1}{m} + \sqrt{\frac{p_{C,l}(1-p_{C,l})}{m}}, 1)]$, where m is the number of points sampled from C up to that moment. In this way, we have defined a statistical measure about the error introduced if we label automatically the samples of an element C after seeing m labels. The conservative estimate of this error is defined as `Bound`$(C)=1 - p^{LB}_{C,l}$ [5].

The procedure `Select` determines which elements of the partition are sampled. There are several alternatives to implement this procedure, but the most evident choice, taking into account the objective of minimizing the number of queries, is to focus the selection process on those regions of the space that are still under-sampled. A simple implementation of this idea is to choose an element C with probability proportional to ω_C `Bound`(C), where ω_C is the fraction of the unrevised dataset covered by element C (proportional to the number of data points inside an element C). This sampling rule from one side, will reduce querying in elements C of the current partition $\mathbf{C}$ that has a low `Bound`(C), thus reducing labeling efforts in those regions that can be automatically labeled with high confidence. From the other side, the factor ω_C permits to explore the large and fairly pure elements of the data structure where small "missing-clusters" can be hidden.

The most critical part of the algorithm is the definition of a searching strategy for finding $\mathbf{C}^{opt}$. After seeing sufficient number of samples, the algorithm must be able to generate a new, probably better partition $\mathbf{C}^{k+1}$ from $\mathbf{C}^k$. In this framework,

Algorithm 1. *Exploration/exploitation* algorithm

```
Require: A data structure $\mathbf{C}$ to represent any partition of $\mathbf{X}$.
Require: A number $s$ representing the number of samples to be queried at each algorithm step.
$\mathbf{C}^0 \leftarrow \mathbf{X}$
Bound($\mathbf{C}^0$)
$\hat{\mathbf{L}}(\mathbf{X}) \leftarrow 1$ {Arbitrary label for label proposal list.}
$\mathbf{L}(\mathbf{X}) \leftarrow$ empty {Revised labels list.}
$i \leftarrow 0$
$k \leftarrow 0$
while unseen labels do
    $j \leftarrow 0$
    $\mathbf{x} \leftarrow$ empty {List of data points to query.}
    while $j < s$ do
        Select an element $C$ from $\mathbf{C}^k$.
        Sample a random point $p$ from $C$.
        $\mathbf{x} \leftarrow \mathbf{x} \cup p$
        $j \leftarrow j + 1$
    end while
    Find the purest cluster $C^p$ in $\mathbf{C}^k$
    Display samples with label proposals $\hat{\mathbf{L}}(\{\mathbf{x}, C^p\})$ and get true labels in case of
    exploration $\mathbf{L}(\mathbf{x})$ xor exploitation $\mathbf{L}(C^p)$.
    $i \leftarrow i + s$
    if explore then
        for each $C \in \mathbf{C}^k$ do
            Compute Bound($C$), a conservative estimate of the mislabeling error within $C$.
            Update label proposals $\hat{\mathbf{L}}(C)$
        end for
        Search for a new better partition $\mathbf{C}^{k+1}$
    else if exploit then
        $\mathbf{C}^{k+1} \leftarrow \mathbf{C}^k / C^p$
    end if
    $k \leftarrow k + 1$
end while
return the full labeled set $\{\mathbf{X}, \mathbf{L}\}$ to train a classifier.
```

we can define the following order relation between partitions: A partition $\mathbf{C}^{k+1}$ is better than a partition $\mathbf{C}^k$ if $\sum_{C \in \mathbf{C}^{k+1}} \omega_C \texttt{Bound}(C) < \sum_{C \in \mathbf{C}^k} \omega_C \texttt{Bound}(C)$, that is, if the fraction of the dataset that can be automatically labeled with confidence in $\mathbf{C}^{k+1}$ is larger than the one in $\mathbf{C}^k$.

A simple but efficient alternative to limit the size of the search space over possible data partitions when implementing *exploration/exploitation* algorithm, used in [5], is to consider a reduced partition space: instead of considering all possible partitions[4] of $\mathbf{X}$, the authors consider only the subspace formed by the partitions defined by a given pruning of the tree representing a hierarchical clustering of $\mathbf{X}$.

[4] The number of ways a set of n elements can be partitioned into nonempty subsets is called a Bell number and is denoted by B_n. For example, $B_{10} = 115975$.

In this case, the navigation strategy can be also simplified and the `Search` procedure in Algorithm 2 consists of selecting a good pruning of the pre-calculated hierarchical clustering tree at each active learning step (see [5] for more details).

The procedure `Display samples` sends the data samples with labels proposals to the display and waits for the user interaction. After executing this procedure, the algorithm receives a group of true labels that can be used either for structure exploitation either for exploration step. The details on displaying strategies are discussed in section 3.2.

3.2 Displaying Strategy

At each step of the *exploration*/*exploitation* of data structure step, the algorithm produces three outcomes: 1) the samples from impure clusters with the label proposals $\{\mathbf{x},\hat{\mathbf{L}}\}$, 2) the current clustering structure grouping the similar data samples and their labels proposals $\{\mathbf{C},\hat{\mathbf{L}}\}$ and 3) the purity measure of each cluster $\{\mathbf{C},\mathtt{Bound}(\mathtt{C})\}$. The question that arises is how to present all this information to the user (see Fig. 5)?

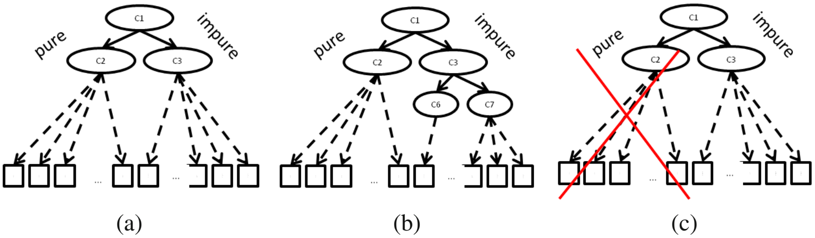


Fig. 5 An illustration of possible actions on data structure. a) hypothetical data structure with one pure and one impure cluster, b) *exploration* - user decides to label samples from impure cluster, as an effect the algorithm descends in hierarchical structure and uncover new clusters, and c) *exploitation* - user decides to revise the labels in pure cluster, and as an effect all data from this cluster are labeled and no further samples will be displayed in sampling process.

It is straight-forward to present to the oracle the samples from impure clusters $\{\mathbf{x},\hat{\mathbf{L}}\}$ while it is a necessity in data structure exploration step (see Fig. 5(b)). If the user decides to label those samples, the algorithm will learn about true labels $\{\mathbf{x},\mathbf{L}\}$. This results in dividing the current cluster structure into more refined one (data structure exploration).

When it comes to the clusters it is infeasible to display all the partitions at once. In this case two different criteria can be applied: one that maximizes the information gain or other that minimizes the oracle's effort. The first one means to display the most impure cluster from the current data structure and to ask the user to correct the labels. But this displaying strategy is contrary to the problem set-up while it is not minimizing the oracles effort. Moreover, the algorithm already provides to the oracle some samples from impure clusters due to data structure exploration step.

The second approach is to present to the user the purest cluster and its labels proposals $\hat{\mathbf{L}}(C^p)$ and asking him/her to correct, hopefully, a few samples and accept the whole cluster. Once the oracle accepts the cluster, all the samples have been assigned a correct label and this part of the space will no longer be sampled by the algorithm. If the data can be represented by the label-aligned data structure with large clusters in this step with a few (or non) oracle's interventions, a whole cluster can be labeled (see Fig. 5(c)). Moreover, if the data can be divided in a few such clusters, the labeling process will be very efficient and cheap in oracle's effort. This is a step of the algorithm that exploits the current data structure.

To be able to jointly *explore/exploit* data structure and minimize the oracle's effort, both data, 1) samples from impure clusters and 2) the purest cluster, should be presented in parallel to the user. In this set-up, the user decides whenever it is convenient to exploit the current data structure or to continue exploring to get finer cluster - label alignment.

3.3 Interface for Interactive Labeling of Endoscopic Frames

The interface is shown on Fig. 6. The interface is composed of 3 fields, one to display the data's labels that have been revised by the oracle. One for displaying the samples from impure clusters, and one to display the purest cluster. The user can choose in which field he/she is willing to interact. If the purest cluster is homogenous, it is favorable to change a few (or none) labels and to accept a large number of labeled samples. Otherwise the oracle should interact in the field of samples from impure clusters and wait for a pure cluster to appear. Once the oracle has revised the labels in the field of samples from impure clusters (or in the purest cluster field) he/she should press the button accept queries (or cluster) and the algorithm will come-up with new data and their proposals.

With respect to the interface two questions remain open:

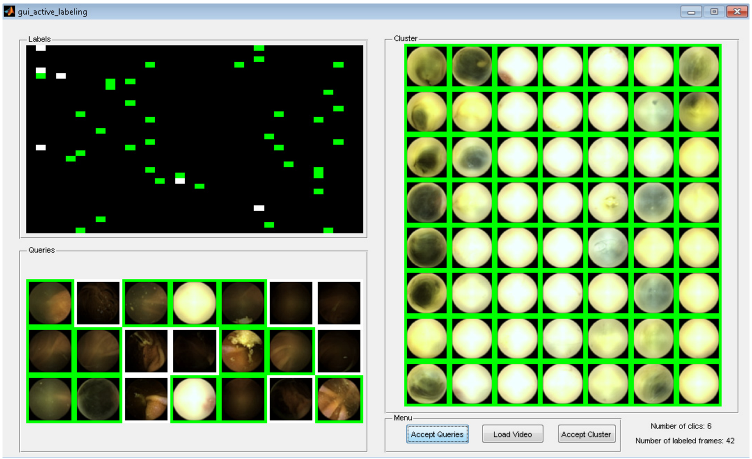
- Number of images to be displayed in each filed.
- Optimal resolution of the image.

These questions are not treated in this chapter while the answers should be adjusted individually to the data set that is being labeled and to the screen's resolution. The general remark is that the parts of the image that are being subject to labeling should be well visible to the user. The user should not spend too much time on visual inspection of a single image and should be able to quickly spot the discriminative (in context of labeling) parts of the image like: color, structure, shape etc. .

4 Experiments

The purpose of the experiments section is to evaluate three possible scenarios:

1. *Random order* - a label has to be provided for each frame individually, the number of oracle's interventions is proportional to the number of samples.



Number of clics: 6

Number of labeled frames: 42

Fig. 6 An example of interface, used to display images during labeling process of Wireless Capsule Endoscopy video. Left-top assigned labels (green intestinal content frames, white clear frames). Left-down file displaying samples from all clusters with the label proposal. Right file showing the purest cluster with label proposal.

2. *Sequential (in time) order* - the label is activated and last until it is changed, the number of oracle's interventions depends on the dynamics of the process that is being observed, if the process is slow in time the number of interventions is proportional to the number of classes in the data, if the process is highly dynamic the number of interventions is proportional to the number of samples.
3. *Proximity in feature space (hierarchical clustering)* - the data are organized into clusters in the feature space, the number of oracle's intervention depends on the cluster structure, if some large fairly-pure clusters are present, the number of interventions is proportional to the number of clusters. If the data can not be organized into fairly-pure label-aligned clusters, the number of interventions is proportional to the number of samples.

To evaluate the scenarios the data from informative vs. non-informative frame problem of Wireless Capsule Endoscopy is used. The scenarios 2 and 3 are further analyzed, in case of scenario 1 it is assumed that the number of oracle's intervention is equal to the number of samples.

In the evaluation one WCE video of 55156 frames was sub-sampled every 50th frame producing a string of images of length 1104 frames. First, the video was presented to the expert according to the Scenario 2 asking her/him to label the informative and non-informative frames. Each sample was displayed in a sequential order with a label proposal, if the proposal was incorrect the user could change the label. In this scenario of reviewing and labeling of all the frames, the oracle needed 57 clicks.

Second, the data was presented to the same user using the application presented in Section 2 asking her/him to label the informative and non-informative frames. In the field of samples from impure clusters, 27 frames were displayed. In this scenario the user needed 45 interventions to revise and label 1104 frames of WCE.

In order to fully appreciate the utility of the application, we tested the proposed labeling scheme in the task of face database creation, where we labeled examples of facial and not-facial images. First, the face hypothesis was generated using the Viola-Jones classifier containing true face detections and some hard false positives cases. The goal was to separate the true from the false detections. In order to do so, each face detection image was represented by using the Histogram of Gradients (HoG) feature. The structure representing the image similarity was calculated in the HoG feature space. In the experiments we used 1064 images (of faces and no-faces). At the beginning, all images were assigned to face class. Using our approach the user was able to review all labels and get perfect labeling with 87 clicks.

5 Conclusions

In this chapter, a new application for error-free labeling with the user in the loop has been presented. The application is based on data similarity in feature space. This method actively *explores* data in order to find the best label-aligned clustering and *exploits* it to reduce the oracle's effort. At each step of the method, the oracle can decide wether it is more convenient to go for data exploitation (the displayed cluster is fairly pure) or for further data structure exploration. The algorithm for each data sample presents a label proposal, based on majority label estimation in current cluster. The error-free labeling is guaranteed by the fact that all data and their labels proposals are visually revised by en expert. Thanks to the clustering structure this revision can be done in an efficient way reducing significantly the time of constructing a wide set of training samples.

This strategy has been compared to the sequential (in time) ordering of the data. This strategy should be used when the data come from steady (or even static) process. On the other hand, the strategy based on proximity in feature space is favorable for the data where some large fairly-pure clusters are expected to be found.

Acknowledgments. This work was supported in part by a research grant from the MICINN Grants TIN2009-14404-C02 and CONSOLIDER-INGENIO 2010 (CSD2007-00018).

References

1. Bashar, M.K., et al.: Automatic detection of informative frames from wireless capsule endoscopy images. Medical Image Analysis 14(3), 449–470 (2010)
2. Bekel, H., Heidemann, G., Ritter, H.: Interactive image data labeling using self-organizing maps in an augmented reality scenario. Neural Networks 18(5-6), 566–574 (2005)
3. Biswas, A., Jacobs, D.: Active image clustering: Seeking constraints from humans to complement algorithms. In: CVPR (2012)
4. Coimbra, M.T., Cunha, J.P.S.: MPEG-7 visual descriptors: Contributions for automated feature extraction in capsule endoscopy. IEEE TCSVT 16(5), 628–637 (2006)
5. Dasgupta, S., Hsu, D.: Hierarchical sampling for active learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 208–215. ACM (2008)
6. Donmez, P., Carbonell, J.G., Bennett, P.N.: Dual Strategy Active Learning. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 116–127. Springer, Heidelberg (2007)
7. Drozdzal, M., Seguí, S., Malagelada, C., Azpiroz, F., Vitrià, J., Radeva, P.: Interactive Labeling of WCE Images. In: Vitrià, J., Sanches, J.M., Hernández, M. (eds.) IbPRIA 2011. LNCS, vol. 6669, pp. 143–150. Springer, Heidelberg (2011)
8. Dwyer, K., Holte, R.: Decision Tree Instability and Active Learning. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 128–139. Springer, Heidelberg (2007)
9. Freund, Y., Sebastian Seung, H., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Mach. Learn. 28, 133–168 (1997)
10. Hospedales, T.M., Gong, S., Xiang, T.: Finding rare classes: Active learning with generative and discriminative models. IEEE Transactions on Knowledge and Data Engineering 99(PrePrints) (2011)
11. Igual, L., et al.: Automatic discrimination of duodenum in wireless capsule video endoscopy. IFMBE Proceedings 22, 1536–1539 (2008)
12. Jung, Y.S., et al.: Active blood detection in a high resolution capsule endoscopy using color spectrum transformation. In: ICBEI, pp. 859–862 (2008)
13. Kang, J., Doraiswami, R.: Real-time image processing system for endoscopic applications. In: IEEE CCECE 2003, vol. 3, pp. 1469–1472 (2003)
14. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley-Interscience (2004)
15. Malagelada, C., De lorio, F., Seguí, S., Mendez, S., Drozdzal, M., Vitria, J., Radeva, P., Santos, J., Accarino, A., Malagelada, J.R., Azpiroz, F.: Functional gut disorders or disordered gut function? small bowel dysmotility evidenced by an original technique. Neurogastroenterology and Motility 24(3), 223-e105 (2012)
16. Malagelada, C., et al.: New insight into intestinal motor function via noninvasive endoluminal image analysis. Gastroenterology 135(4), 1155–1162 (2008)
17. McCallum, A., Nigam, K.: Employing em and pool-based active learning for text classification. In: Proceedings of the Fifteenth International Conference on Machine Learning, ICML 1998, pp. 350–358. Morgan Kaufmann Publishers Inc., San Francisco (1998)
18. Nguyen, H.T., Smeulders, A.: Active learning using pre-clustering. In: Proceedings of the Twenty-First International Conference on Machine Learning, ICML 2004, p. 79. ACM, New York (2004)
19. Settles, B.: Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison (2009)

20. Tian, Y.: A face annotation framework with partial clustering and interactive labeling. In: International Conf. on Computer Vision and Pattern Recognition, p. 7 (2007)
21. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. J. Mach. Learn. Res. 2, 45–66 (2002)
22. Vilarino, F., et al.: Intestinal motility assessment with video capsule endoscopy: Automatic annotation of phasic intestinal contractions. IEEE TMI 29(2), 246–259 (2010)
23. Ward Jr., J.H.: Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58(301), 236–244 (1963)
24. Xu, Z., Yu, G., Tresp, V., Xu, X., Wang, J.: Representative Sampling for Text Classification Using Support Vector Machines. In: Sebastiani, F. (ed.) ECIR 2003. LNCS, vol. 2633, pp. 393–407. Springer, Heidelberg (2003)
25. Zhang, L., Tong, Y., Ji, Q.: Active Image Labeling and Its Application to Facial Action Labeling. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 706–719. Springer, Heidelberg (2008)