Multi-Instance Active Learning with Online Labeling

by

Kimia Salmani Niasar, B.S

Master's Thesis

In

Computer Science

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
Degree of

Master of Science

Approved

Dr. Mohan Sridharan

Committee Chair

Dr. Richard Watson

Dominik Casadonte

Dean of the Graduate School

August, 2013

ACKNOWLEDGEMENTS

I wish to acknowledge the help and support of my advisor Dr. Mohan Sridharan for his support of my Master's study and research, patience and guidance.

Beside my advisor, I would like to thank Dr. Richard Watson, my committee member for his encouragement, insightful words, and questions during my defense session and Dr. Akbar Siami Namin who has guided me through this difficult period of my life with his support and help.

I would like to thank my fellow labmates in Texas Tech Robotic lab who helped me with editing my thesis and defense practice talk especially Daniel Holman who reviewed my thesis comprehensively and enlightened me with his useful comments and Shiqi Zhang, Li Xiang and Sarah Rainge for their practical opinions upon my defense slides.

Last but not the least, I would like to thank my parents who always supported me spiritually and financially throughout my life and also my sisters especially Nafise Salmani who has been an incredible mentor to me during the recent two years.

TABLE OF CONTENTS

Acknowledgements ii
Abstract v
List of Tables vii
List of Figures viii
List of Abbreviations ix
1. Introduction 1
1.1 Motivation 1
1.2 Statement of the Problem 1
1.3 Objective of the Thesis 1
1.4 Thesis Outline 2
2. Literature Review 3
2.1 Overview 3
2.2 Background 3
2.2.1 What is Machine Learning? 3
2.2.1.1 Unsupervised Learning 3
2.2.1.2 Reinforcement Learning 4
2.2.1.3 Supervised Learning 4
2.3 Related Work 6
2.3.1 Multiple-Instance Learning 6
2.3.2 Active Learning 8
2.3.3 Verbal Understanding 9
2.4 Summary 10
3. Multiple-Instance Learning 11
3.1 Overview 11
3.2 Learning a Model 12
3.3 Evaluation 14
3.4 Summary 14
4. Active Learning 16
4.1 Overview 16
4.2 Active Learning for Human 16

4.3 Approaches . . . 16
4.3.1 Membership Query Synthesis . . . 17
4.3.2 Stream-Based Selective Sampling . . . 17
4.3.3 Pool-Based Sampling . . . 18
4.4 Uncertainty Sampling . . . 19
4.5 Summary . . . 19
5. Proposed Framework . . . 20
5.1 Overview . . . 20
5.2 Bag Uncertainty . . . 20
5.3 Applied Active Learning Approaches . . . 21
5.4 Verbal Understanding . . . 22
5.5 Summary . . . 25
6. Experimental Setup and Results . . . 26
6.1 Overview . . . 26
6.2 Database . . . 26
6.3 Experimental Setup . . . 27
6.4 Experimental Results . . . 28
6.5 Discussion of The Results . . . 38
7. Conclusion and Future Work . . . 40
7.1 Conclusion . . . 40
7.2 Future Work . . . 40
Bibliography . . . 41

ABSTRACT

Robots are increasingly being used to automate different processes in domains such as navigation, health care and reconnaissance. Real-world domains characterized by non-determinism and unforeseen changes frequently make it difficult for robots to operate without considerable domain knowledge or human feedback. At the same time, humans may not have the time and expertise to provide accurate and elaborate domain knowledge, and it may be difficult to obtain many labeled training samples of relevant aspects of the domain. For widespread deployment, robots thus need the ability to incrementally and automatically extract relevant domain knowledge from multimodal sensor inputs, acquiring and using human feedback when such feedback is necessary and available. This thesis describes an algorithm for such incremental learning in the context of building models of relevant domain objects, using visual cues extracted from images of indoor and outdoor scenes and verbal cues extracted from limited high-level human feedback.

Our algorithm builds on an existing framework called *Multiple Instance Active Learning*. *Multiple-instance learning* (MIL) supports learning from sets (also known as *bags*) of instances assigned positive and negative labels. In the context of learning a model for a specific object from images, at least one region in an image (i.e., the bag) assigned a positive label contains the target object, while none of the regions in an image assigned a negative label contain the object under consideration. In prior work, an object model learned from the labeled training data was used to identify training instances that are known to contain the corresponding object but are associated with a high uncertainty, i.e., a low probability of containing the object. These instances were converted to labeled positive bags to simulate active learning and improve the learned object model. However, the ground truth label for specific image regions is not likely to be available in new data obtained by a robot. Existing algorithms thus do not truly support incremental learning, which is essential for robot application domains.

To address the limitations of prior work, we introduce the concept of *Bag Uncertainty*. Our algorithm applies the object models learned through MIL on *previously unseen and unlabeled* images that are processed incrementally. For each learned object, these images are assigned a probability that represents the likelihood that it contains the corresponding object in different regions. Images with a high level of uncertainty (with regard to the pres-

ence or absence of the learned object) are assigned a proportionately higher priority for human feedback. Simplistic verbal queries are posed to humans to solicit the bag labels or specific properties of the image regions. If humans do not (or cannot) provide feedback, the algorithm continues using the learned models on subsequent images. If a human participant does provide feedback, lexical tools are used to process the verbal feedback and associate the new information with a specific image or regions in an image, thus generating a labeled bag for revising the learned object model. As a result, the robot is able to operate without human feedback as far as possible, soliciting help when it is needed and available. We experimentally compared our algorithm with a state of the art MIL algorithm on a set of natural scenes and object categories drawn from the benchmark IAPR TC-12 dataset. Experimental results show that our algorithm learns better models and thus provides significantly higher object recognition accuracy than the existing algorithm.

LIST OF TABLES

5.1 An Example to Calculate Similarities Lexically 24
5.2 An Example to Calculate Similarities With Respect to the Content 25
6.1 The Average AUROC Improvement on Object Tree 29
6.2 The Average AUROC Improvement on Object Street 30
6.3 The Average AUROC Improvement on Object Window 31
6.4 The Average AUROC Improvement on Object Door 32
6.5 The Average AUROC Improvement on Object Cloud 34
6.6 The Average AUROC Improvement on Object River 35
6.7 The Average AUROC Improvement on Object Rock 36
6.8 The Average AUROC Improvement on Object Bottle 37
6.9 The Average AUROC Improvement on Object Car 38
6.10 Table of Results 39

LIST OF FIGURES

2.1 Reinforcement Learning 5
2.2 Supervised Learning 6
2.3 Classification 7
2.4 Diverse Density 8
2.5 Multimodal Learning Framework 9
3.1 Positive Bag Concept Depiction 11
3.2 Negative Bag Concept Depiction 12
3.3 Segmented Image 13
4.1 Three Main Active Learning Approaches 17
4.2 Pool-Based Active Learning Illustration 18
5.1 Proposed Framework 21
5.2 POS and BIO Tag Example 24
5.3 Hypernym Relation in Wordnet 24
6.1 Segmented Images From SAIAPR TC-12 26
6.2 Comparison on Object Tree 29
6.3 Comparison on Object Street 30
6.4 Comparison on Object Window 31
6.5 Comparison on Object Door 32
6.6 Comparison on Object Cloud 33
6.7 Comparison on Object River 34
6.8 Comparison on Object Rock 35
6.9 Comparison on Object Bottle 36
6.10 Comparison on Object Car 37

LIST OF ABBREVIATIONS

MIL – Multiple Instance Learning
HRI – Human Robot Interaction
ML – Machine Learning
FBI – Federal Bureau of Investigation
MILR – Multiple-Instance Logistic Regression
LBFGS – Limited-memory Broyden-Fletcher-Goldfarb-Shanno
ROI – Region Of Interest
POS – Part Of Speech
MIU – Multiple-Instance Uncertainty
BU – Bag Uncertainty
AUROC – Area Under the Receiver Operating Characteristic
DD – Diverse Density

CHAPTER 1
INTRODUCTION

1.1 Motivation

The idea of interaction with robots as facile as humans requires a higher level of sophistication in robots. The more robots can learn and behave like humans, the more they pace toward independency.

Consider a spacious room with variety of known and unknown objects. When a man walks into this room looking for a particular object he may or may not have a mental image of the object. If he knows what the object looks like, he may or may not be able to find it. If someone is close by, he could ask if currently observable objects with some degree of uncertainty are the goal object. He could also simply store the knowledge in his mind to ask later. On the other hand if he does not know what the object looks like, he needs to build a mental image of the object initially by asking questions regarding the objects he observe.

In order for the robot to act like humans (above), it needs to be capable of learning from known and unknown object categories, calculate the resemblance between them and interact with human (oracle) to make queries on uncertain objects.

1.2 Statement Of The Problem

The existing algorithms address the learning from known object categories and the resemblance calculation. In these methods, at least a number of objects should be discovered from the category. What if no object from the category has been learned before? How to build the premier mental image (in learning problems for machines it has been referred to as a model) for the unknown object category? We address these questions in this research.

1.3 Objective Of The Thesis

In this thesis, we introduce a new concept of MI active learning, *Bag Uncertainty*, which raises the learning rate for some object categories while decreasing the labeling effort. In our approach, we increase the training set's size with unknown objects. Therefore, the

model is built upon observed and new objects of the category. Furthermore, we address human robot interaction for labeling recently observed objects by using verbal cues.

1.4 Thesis Outline

This thesis is consisting of 6 chapters. Chapter 1 provides the motivation behind this research. It also describes the problem and objective of this research.

Chapter 2 presents a summary of machine learning and its subcategories. It also provides a comprehensive review of previous studies and drawbacks on Multi-Instance Learning, active learning and verbal understanding.

Chapter 3 explains an instance of supervised learning which has been used distinctively in our research known as MI learning.

Chapter 4 defines active learning, its objective and approaches, membership query synthesis, stream-based selective sampling and pool-based sampling. It also introduces the most common active learning framework used in our research distinguished by uncertainty sampling.

Chapter 5 presents the applied active learning approaches utilized accompanied with the description on MIL and verbal understanding which led to a description of our proposed framework.

Chapter 6 presents and discusses the experimental results. The experimental assumption and setups are given along with the dataset features. A complete comparison between the best method introduced in previous research and BU was conducted under a similar condition to evaluate BU.

CHAPTER 2
LITERATURE REVIEW

2.1 Overview

Machine learning and its subcategories such as supervised learning, reinforcement learning and unsupervised learning are reviewed. MI learning, active learning, verbal understanding and their combination functionality have been investigated through a comprehensive review.

2.2 Background

2.2.1 What is Machine Learning?

As the world moves forward, most applications in many fields moves toward autonomy. *Introduction to Machine Learning* book [2] referred to autonomy as:

"the act of operating independent of external control"

Although human presence and feedback is usually necessary for many applications to accomplish their tasks, machine learning helps to omit this necessity as times goes by.

Machine Learning (ML) methods promote autonomy in a more intelligent direction, such as making predictions [2] or decisions. ML applications are utilized in many fields like marketing, stock, weather forecasting, object recognitions. Facial recognition is a rather complicated field in object recognition which has been utilized to accomplish many goals including criminal identification, finding face locations on camera devices, and Facebook.

In the non-stop world, everything is dynamic, including the environment. In order to be intelligent, a system needs to adapt itself automatically to the environment by learning, so that the system designer need not foresee and provide solution to all possible situations. Learning is obtained by three existing methods: Supervised Learning, Unsupervised Learning and Reinforcement Learning. We will provide more information regarding these three subcategories in following sections.

2.2.1.1 Unsupervised Learning

In supervised learning, the aim is to learn a mapping from the input to an output whose correct values are provided by a supervisor. In unsupervised learning, there is no such

supervisor and we only have input data. The aim is to find regularities in the input. There is a structure to the input space such that certain patterns occur more than others, and we want to see what generally happens and what does not. In statistics, this is called *density estimation* [2]. However unsupervised learning also encompasses many other techniques that seek to summarize and explain key features of the data. Fergus et. al. [17] describes a method to recognize and learn object categories from unlabeled and unsegmented images using probabilistic representation of all aspects of the object such as appearance, shape and relative scale. Fergus [17] utilizes expectation maximization (EM) algorithm for learning. Approaches to unsupervised learning include:

- clustering (e.g., k-means, mixture models, hierarchical clustering),
- blind signal separation using feature extraction techniques for dimensionality reduction (e.g., Principal component analysis, Independent component analysis, Non-negative matrix factorization, Singular value decomposition) [1].

2.2.1.2 Reinforcement Learning

RL is an enormous area with variety of approach algorithms. Sutton and Barto [20] provide a comprehensive description on this area. Basically, in RL the agent take an action and the environment -could be human or surrounded area- gives feedback or reward in order to encourage or prohibit the agent from repeating that action in that particular state. Figure 2.1 shows the concept of reinforcement learning.

2.2.1.3 Supervised Learning

Supervised learning is the machine learning task of inferring a function from labeled training data [12]. The training data consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations. Figure 2.2 shows an example of supervised learning. One predicament of using supervised learning is its natural necessity for huge number of labeled samples. Providing labels for each instance is troublesome, expensive and sometimes

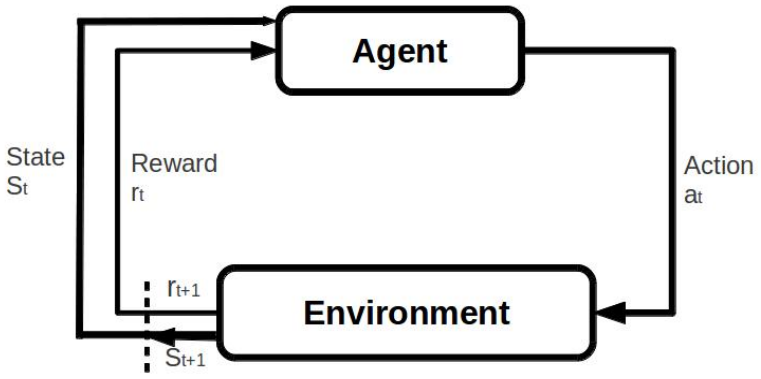

Figure 2.1: Reinforcement Learning

inaccurate. MI learning is categorized as a supervised learning method which is discussed primarily in Chapter 3. *Classification* and *Regression* are two basic supervised learning problems [2]:

Classification . In classification problems there are new entries (in our case, instance images) that need to be classified into two or more suitable categories based on a previously learned model. Figure 2.3 shows classification of different objects with respect to their geometric shapes. Objects shown in the image could also be classified regarding other aspects such as color or size. Classifying objects in an image requires a visual classifier to divide the images into region of interests (ROI). Gupta [8] proposes an approach for learning visual classifiers by addressing image regions to corresponding objects and relationships between pairs of those objects.

Regression . In supervised learning classification problems, we need to find out the resemblance of a new instant with the trained model for a special object class category. Inputs are the instant's features like colors, area, convexity and other information that could be extracted from the instant image. The output is the probability of the instant to be classified as the object class category. Such problems that the output is a number rather than a

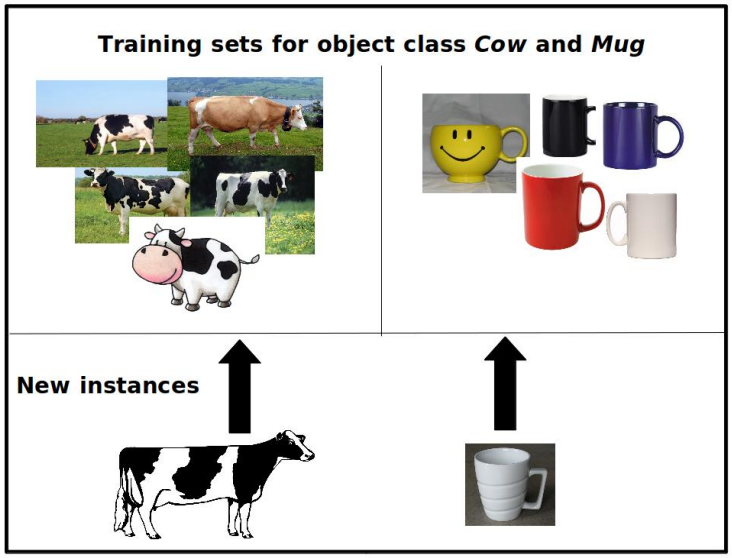

Figure 2.2: Training images for two object categories are shown, the new instances are evaluated using the inferred function. The optimal scenario will categorize them correctly

category are *regression* problems [2].

Let X denote the instant features and Y be the probability of the instant. Using training data, a machine learning program fits a function to this data to learn Y as a function of X.

$$f(x) = y = ax + b$$

2.3 Related Work

2.3.1 Multiple-Instance Learning

A multiple-instance learning algorithm was introduced by Maron et. al [10]. MIL is a supervised learning method which consists of two special concepts: *instances* and *bags*. Each bag encompasses several instances and labeling takes place for bags instead of instances. This method decreases the labeling effort effectively. The framework introduced by Maron et. al [10] for MIL, called *Diverse Density* (DD), is a method for finding commonalities between the positive bags. DD is utilized in Maron's later work [11] in order to classify natural scenes. Another contribution of this research is the development of a

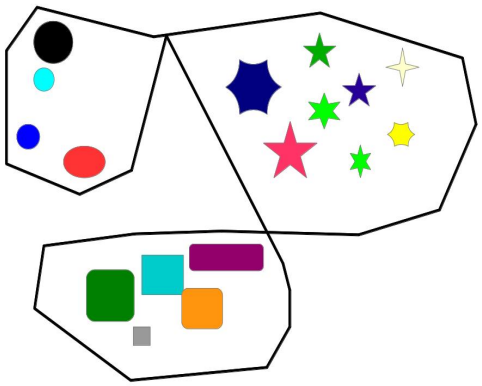

Figure 2.3: Classification of objects with respect to their geometric shape.

bag generator: a mechanism which transforms an image into a set of instances. In this approach, each training image is a bag which is divided into various sub-images; each sub-image is described in several ways such as *single blob with neighbors*, *single blob with no neighbors*, *disjunctive blob with neighbors*, etc. Diverse Density is used to find areas in feature space that is the closest to at least one instance from every positive bag and far from every negative bag. The algorithm searches the feature space for points with high Diverse Density. Figure 2.4 depicts the DD concept. The intersection of the positive bags is the positive instance in every positive bag. However, this approach only responds to the classification of the natural scenes and no object category. Yang et. al [22] have applied MIL to classify a broader range of images including object images. Bags are generated by selecting some regions from the image with respect to their variances, extracting two sub-pictures from each region and smooth and sample them in an $h \times h$ matrix which is then treated as an h^2-dimensional feature vector. Each feature vector represents an instance and compose a bag for the image. Nevertheless, all experiments shown in this paper were based on gray-scale images; therefore there was no proposed solution to the color representation schemes. Zhang and Golman [23] proposed a new algorithm for MIL, called EM-DD, which outperforms the previous methods with respect to both accuracy and computation time. In this method, the first step (E-step) is to pick one instance from each bag which is most likely to be the most effective in the bag's label, called hypothesis h. In the second

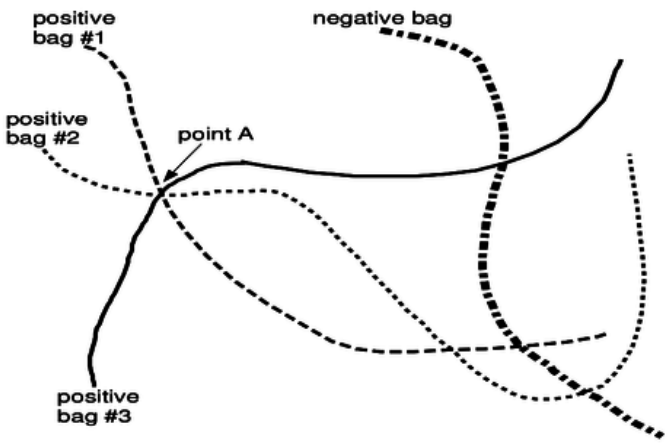

Figure 2.4: A multiple-instance learning algorithm: Diverse Density[1]

step (M-step), gradient search of the DD algorithm is applied in order to find a new h' that maximize $DD(h)$. These two steps repeats until the algorithm converges.

Labeling expense could be yet lowered by applying MI *active* learning [3]. Settles et. al [3] introduced another framework for MI, called MI Logistic Regression. Our research is based on Settle's work. We are using MILR with a new active learning method, called *BU*. All the active learning methods introduced in previous work [3] depend on the initial training set, while in our approach, queries occur on the new observed images. The new image is added to the training set if it helps the learning.

2.3.2 **Active Learning**

In Supervised learning, the goal of *active learning* is to minimize the overall labeling for the learner to reach a higher accuracy. There are several active learning approaches [18] such as *pool-based*, *membership query* and *stream-based selective*. In MI active learning introduced by Settles [3], *pool-based* active learning is applied. In pool-based active learning, the learner is trained with a small set of labeled data and selectively asks queries from a larger pool of unlabeled data which is non-changeable or static. Druck et. al [7] utilizes active learning to solicit labels for features instead of instances. For example *water*, *garbage*, *included* are considered features and their label is *utilities*. Feature Active learning follows

[1]Figure is adopted from Maron et. al [10]

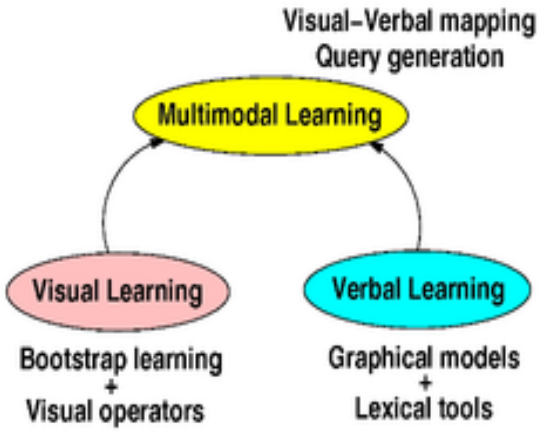

Figure 2.5: Multimodal Learning Framework[2]

a pool-based scenario. Their algorithm may skip a query at a different cost than labeling, since a user may not know how to label a feature.

Active learning could be applied in different algorithms in order to improve the algorithms accuracy. For example, Seddiquie and Gupta [19] presents a method for active learning to learn appearance and contextual models concurrently for multi-class classification. In this approach, instead of selecting the question which clarifies the labels of uncertain region, a contextual interaction model between image regions is built which solicit labels for those regions.

2.3.3 **Verbal Understanding**

Swaminathan et. al [21] proposes a framework that combines visual and verbal learning, as shown in the figure 2.5. In this framework an association between visual and verbal vocabularies are learned to supply the labels for the subsequent sensory cue. Furthermore, feature vectors consisting of the probability distribution over visual and verbal vocabularies lead to learn a mapping model to object category (i.e. *normal*, *typical*, *suspicious*). However the experiments of this research involved simple objects. The verbal interaction of our research follows Swaminathan's work with slight changes in vocabularies, object properties and categories. Verbal understanding as *Natural language Understanding* (NLU) is an

[2]Figure is adopted from Swaminathan et. al [21]

obstacle for humans to verbally interact with the robot effortlessly. Cantrell [16] describes a natural language architecture which presents novel capabilities in NLU which ease the verbal interaction and raise the level of the robots understanding with ambiguous verbal cues.

2.4 Summary

A comprehensive overview of related work and background work is provided. MIL learning lowers the labeling effort as a supervised learning method and is conducted with different frameworks. MIL and the framework presented by Settles [3] is explained in detail in subsequent chapter. Active learning methods and approaches are introduced and elaborated in chapter 4.

CHAPTER 3
MULTIPLE-INSTANCE LEARNING

3.1 Overview

This chapter provides more detail on MIL. We describe how a model is learned and how a new image is evaluated in accordance with the model. This section is completely adopted from Settles et. al [3] framework that we selected for our approach.

As mentioned earlier, in supervised learning, each labeling occurs for only one instance which makes the labeling process tedious and troublesome. Multiple-instance learning (MIL) gives a solution to this issue.

MIL proposes a solution to the troublesome labeling. Instead of labeling only one *instance*, labeling *bags*, a set of instances, has been utilized [10]. In the MIL problems domain, instances are categorized into bags with labels for training. The presumption is that, all the instances in the bags labeled negative are negative and at least one of the instances in the bags labeled positive is positive. *Negative* and *positive* concepts in MIL for instances and bags are shown in figure 3.1 and 3.2. In every problem domain, instances and bags could address different concepts. For example bags could address images and instances, objects in the image; or bags could address a context and instances, each paragraph in the context [3]. Figure 3.3 shows an image and its segmented instances.

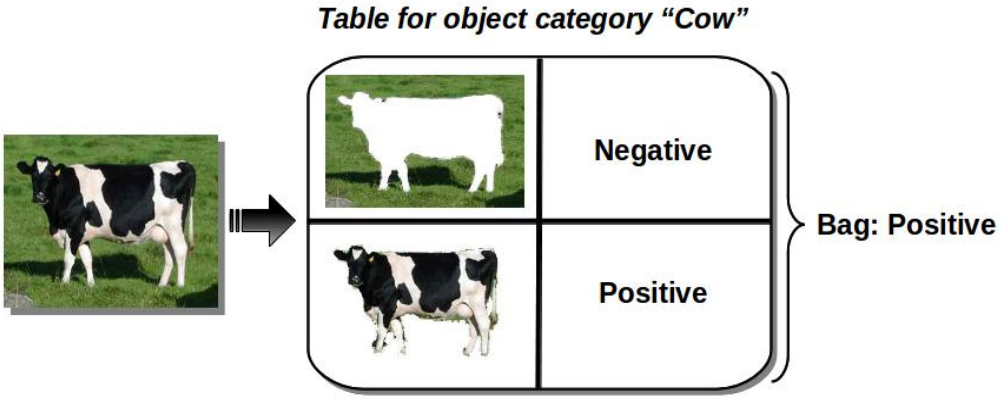

Figure 3.1: Positive bag concept depiction

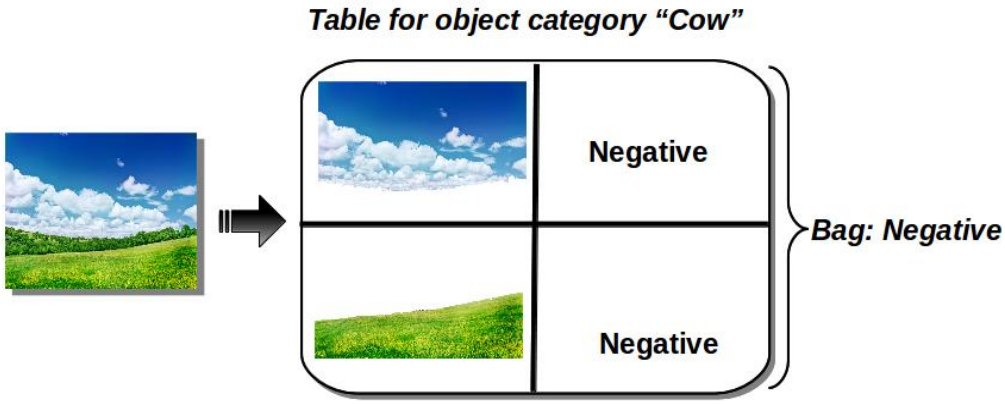

Figure 3.2: Negative bag concept depiction

In supervised learning, every problem consists of one or multiple model(s) which is/are learned from a set of objects (an object refers to anything which could be learned). Models are addressed with a *vector of parameters*, also called weights, which are set randomly initially and improve after training. These parameters correspond to *feature vectors*.

Every instance in a bag is defined with a feature vector. In our research, each feature vector consists of:

$$< r_i, c_i, < hsb_{i1}, hsb_{i2}, hsb_{i3}, ..., hsb_{i64} >$$

where r_i boundary/area, c_i convexity and $hsb_{i1}...hsb_{i64}$ HSB color histogram divided into 64 bins.

3.2 Learning a Model

Assume label y_i is positive for bag B_i consisting of n instances with respect to a special object, the conditional probability is $P(y_i = 1|B_i = B_{i1}, B_{i2}, ..., B_{in})$. Therefore, a substituted method is to compute the conditional probability of each instance, B_{ij}, of Bag B_i: $P(y_{ij} = 1|B_{ij})$, and utilize a *combining function* (such as softmax) to combine posterior probabilities of the instances of the bag and conjecture the posterior probability $p(y_i = 1|B_i)$ for the bag. Classifiers are trained using *multiple_Instance logistic regression* [3]. MILR uses logistic regression with parameters $\theta = (w, b)$ to conjecture conditional probabilities for

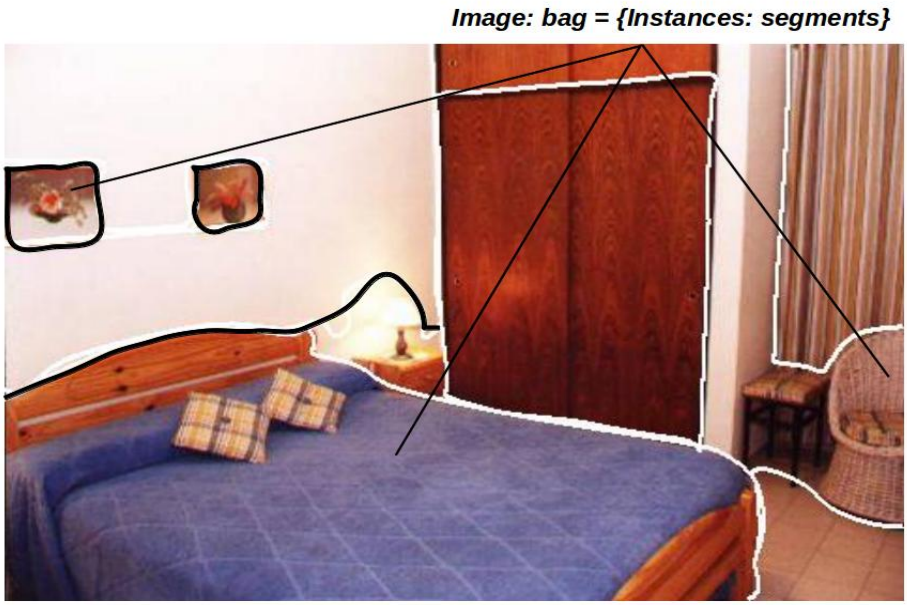

Figure 3.3: In content-based image retrieval, images are represented as bags and instances correspond to segmented image regions.

each instances:

$$p_{ij} = P(y_{ij} = 1|B_{ij}) = \frac{1}{1 + e^{-(w.B_{ij}+b)}}$$

where B_{ij} represents the feature vector for the jth instance in the ith bag and w is the vector of weights affiliated with the feature vectors. Weights (model parameters) are initially generated randomly using Gaussian distribution. To combine the instance probabilities into bag probabilities, MILR uses the softmax function:

$$p_i = P(y_i = 1|B_i) = softmax_\alpha(p_{i1}, ..., p_{in}) = \frac{\sum_{j=1}^{n} p_{ij} e^{\alpha p_{ij}}}{\sum_{j=1}^{n} e^{\alpha p_{ij}}}$$

where α is a constant greater than zero.

Since the equations above represent smooth functions of the model parameters θ, the instance labels in the positive bags are not known, however the parameter values are learnable using Limited_memory Broyden_Fletcher_Goldfarb_Shanno [14] (LBFGS) gradient-based

optimization method and an objective function:

$$E(\theta) = \frac{1}{2}\sum_i (y_i - p_i)^2$$

where $y_i \in \{0,1\}$ is the known label of the bag B_i and $E(\theta)$ is the error over bags which is tried to be minimized during the learning. The best model parameter is the one with the lowest error [3].

3.3 Evaluation

During training, model parameters are adjusted to best match bags to the specific object class features. Therefore, every new bag is evaluated with the adjusted model parameters to be recognized as a positive or negative bag with respect to specific object class. Conjecture conditional probabilities for each instances is:

$$p_{ij} = P(y_{ij} = 1|B_{ij}, M) = \frac{1}{1 + e^{-(w_j.B_{ij}+b)}}$$

where B_{ij} represents the feature vector for the jth instance in the ith bag, M refers to trained model and w is the vector of the trained model affiliated with the feature vector [3]. The conditional probabilities of the instances are combined using Softmax to estimate the bags posterior probability:

$$p_i = P(y_i = 1|B_i) = Softmax_\alpha(p_{i1}, ..., p_{in}) = \frac{\sum_{j=1}^{n} p_{ij} e^{\alpha p_{ij}}}{\sum_{j=1}^{n} e^{\alpha p_{ij}}},$$

The question of "how to label the *test bag* positive or negative" is not answered yet. We will address this question in Chapter 5 after explaining *Active learning*.

3.4 Summary

As discussed in this chapter, MIL lowers the labeling effort effectively in comparison with other supervised learning methods. We discussed parameter tuning (referred to as weights) and bag evaluation. Another approach, called *active learning*, is discussed in the next chapter which can improve the MIL toward the lower labeling and higher accuracy. Therefore the combination of MIL and active learning is a detour towards the higher accu-

racy.

CHAPTER 4
ACTIVE LEARNING

4.1 Overview

Chapters 3 and 4 provide pre-knowledge for Chapter 5. This chapter describes the active learning concept embodied in an example. A comprehensive review of the active learning and its approaches such as, membership query, stream-based selective and pool-based sampling. Uncertainty sampling is provided as an active learning measurement.

4.2 Active Learning for Human

How do Humans Learn by Asking? Consider a curious child who ask questions about every event around him. This child asks due to lack of experience and knowledge about that particular subject. When the child is with his parents, he asks what he does not know or has doubts about, and the parents try to answer him so that their child can boost his knowledge in that particular subject. Now, consider substituting the child with a system or robot and the parents with an *oracle*. An oracle refers to anything that contains all the knowledge that we are or are not aware about. This is an informal description of *Active Learning*.

Active learning, also called "query learning" or "optimal experimental design" is a subfield of machine learning. In active learning, the learning algorithm has been given a chance to be curious and query about the data which is being learned [18].

4.3 Approaches

Why Active Learning Is Desirable to Have in Learning Algorithms? In supervised learning methods, an algorithm is trained on hundreds (even thousands) of labeled instances. As mentioned earlier, the process of labeling instances is not always trivial. In more sophisticated supervised learning tasks, such as *speech recognition*, *information extraction*, *classification and filtering*, labeling is expensive and time-consuming [18]. AL systems attempt to promote supervised learning by asking queries in the form of unlabeled instances to be labeled by an oracle, so that the active learner can reach higher accuracy using as few labeled instances as possible; in this case, the labeling expense is minimized. There are several different approaches for AL systems to ask queries: *Membership Query Synthesis*, *Stream-Based selective sampling*, and *Pool-Based Sampling*. We will provide more details

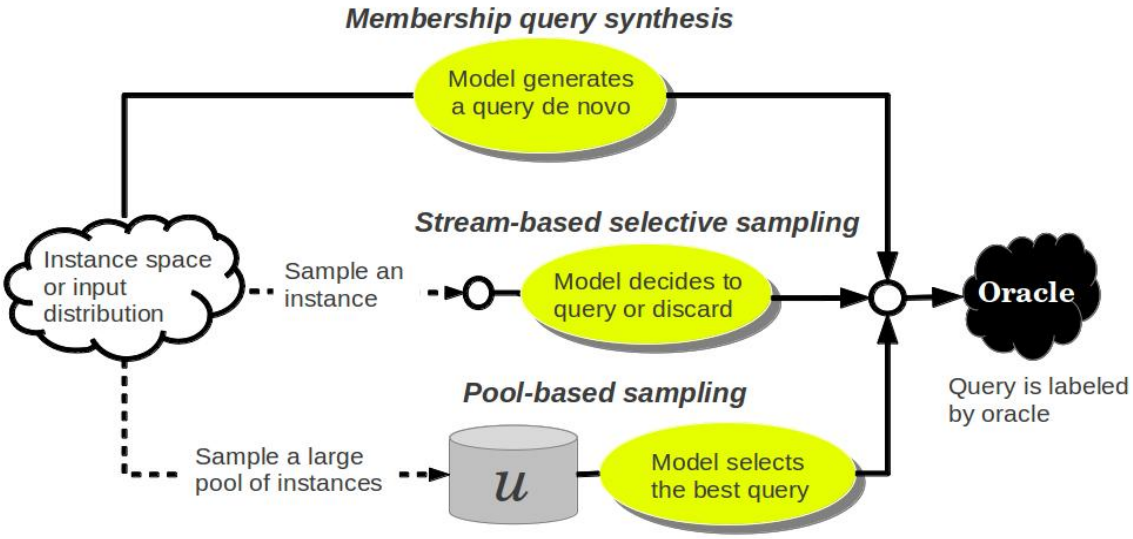

Figure 4.1: Three main active learning approaches

on mentioned approaches in this chapter. Figure 4.1 distinguishes the three approaches.

4.3.1 Membership Query Synthesis

One of the active learning approaches is learning with *membership queries*. As shown in figure 4.1, in this scenario, the learner may ask queries to label any unlabeled instances in the input space, including queries that the learner constructs. Query Synthesis is reasonable for many problems, but labeling instances can be awkward using human annotator as the oracle [18].

4.3.2 Stream-Based Selective Sampling

An alternative to query synthesis is *selective sampling*. In this approach an unlabeled instance is first sampled from the actual distribution, and then the learner can decide whether to query its label or disregard it, thus requiring unlabeled instance acquisition to be free or inexpensive. This approach is called *stream-based* or *sequential* active learning. Depending on the input distribution, this approach can behave like membership query learning, if the input distribution is uniform.

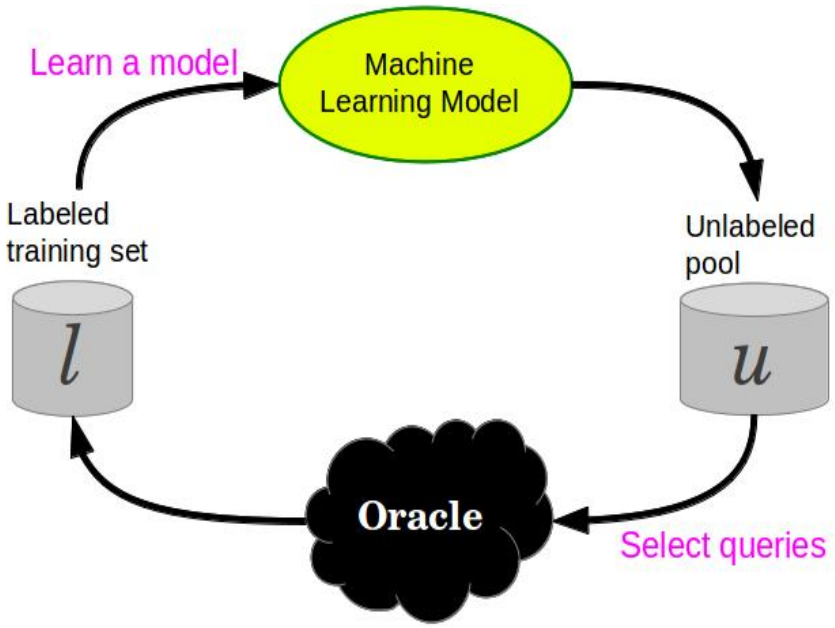

Figure 4.2: Pool-based active learning illustration

The decision whether or not to query an instance can be framed in several ways [18]. One approach is to define a *region of uncertainty* and query every instance that fall in that area. The simplest version of this approach is defining a threshold; every instance which has a higher evaluation than the specified threshold is selected to be queried. Another approach is to use *informativeness measure* or *query strategy* and make a biased decision, such that more informative instances have higher probability to be queried.

4.3.3 **Pool-Based Sampling**

The main difference between pool-base and stream-based active learning is that the former assesses the data sequentially and makes query decisions individually, whereas the latter evaluates the entire collection before selecting the best query.

In *pool-Based sampling*, a presumption is that there is a small set of labeled data $\boldsymbol{l}$ and a larger pool of unlabeled data, also called $\boldsymbol{u}$. Queries are selectively chosen from the unlabeled pool which is assumed to be non-changing [18]. Frequently, the queries are selected in a greedy manner, according to an informativeness measure used to assess all the instances in the pool [3]. Figure 4.2 depicts the pool-based active learning concept.

All active learning approaches utilize assessing informativeness of unlabeled instances. There has been several proposed methods of formulating the query strategies, such as *Uncertainty Sampling*, *Query-By-Committee*, *Expected Model Change*, *Expected Error Reduction*, *Variance Reduction*, *Density-Weighted Methods* [18]. Since we are only using the *Uncertainty Sampling* in this research, we will discuss it in the next section.

4.4 Uncertainty Sampling

The most commonly uncertainty measurement is *uncertainty sampling* [18]. In this measurement, the active learner asks queries about instances with the least amount of certainty for labeling. This approach utilizes probabilistic methods to assess the uncertainty for each instance. When evaluating uncertainty for binary classification, instances with probability closer to 0.5 are classified as the most uncertain ones. *Gini measure* calculates the uncertainty for binary classification as follows:

$$U = 1 - (p^2 + (1-p)^2)$$

where p is the probability. For classification problems with three or more class labels, a more general uncertainty sampling can query the instance with *least confident* prediction:

$$x^*_{LC} = \arg\max_x 1 - P_\theta(\hat{y}|x)$$

where $\hat{y} = \arg\max_y P_\theta(y|x)$, or the class label with the highest posterior probability under the model θ [18].

4.5 Summary

Active learning increases accuracy by allowing the system to send queries. The query may be asked in different manners as described in three approaches and each query may address special need according to different measurements. This chapter discussed one of the uncertainty measurements, *uncertainty sampling*. In the next chapter we will discuss a new uncertainty measurement for MI active learning.

CHAPTER 5
PROPOSED FRAMEWORK

5.1 Overview

The concepts elaborated in Chapter 3 and 4 are utilized in this section for the proposed framework. The concept of Bag Uncertainty is introduced as a new uncertainty measurement for our proposed MI active learning framework in section 5.2. The proposed framework applies two active learning approaches, pool-based and stream-based sampling. An overview of the proposed framework is shown in figure 5.1.

5.2 Bag Uncertainty

Every new image that is observed by the robot noted as "*test image*" or "*test bag*", is segmented into ROIs. Each ROI forms an instance with extracted feature vector. The *test bag*'s probability of being positive is calculated using the methods introduced in chapter 3. Subsequently, a certain threshold (will be elaborated next) is applied to decided whether to add the bag to the training set or ignore it. In this framework, we consider the conditional probability of every newly observed bag, regardless of their positive or negative labeling, to simulate the actual condition in the real-world. In order to improve accuracy with active learning (Chapter 4), we chose uncertainty sampling using Gini measure to calculate the uncertainty for bags:

$$U_i = 1 - (p_i^2 + (1 - p_i)^2)$$

where p_i is the ith bag conditional probability. The Gini score is between 0 and 0.5 and the most uncertain bag's score is 0.5. We call the approach ***bag uncertainty*** in MI active learning.

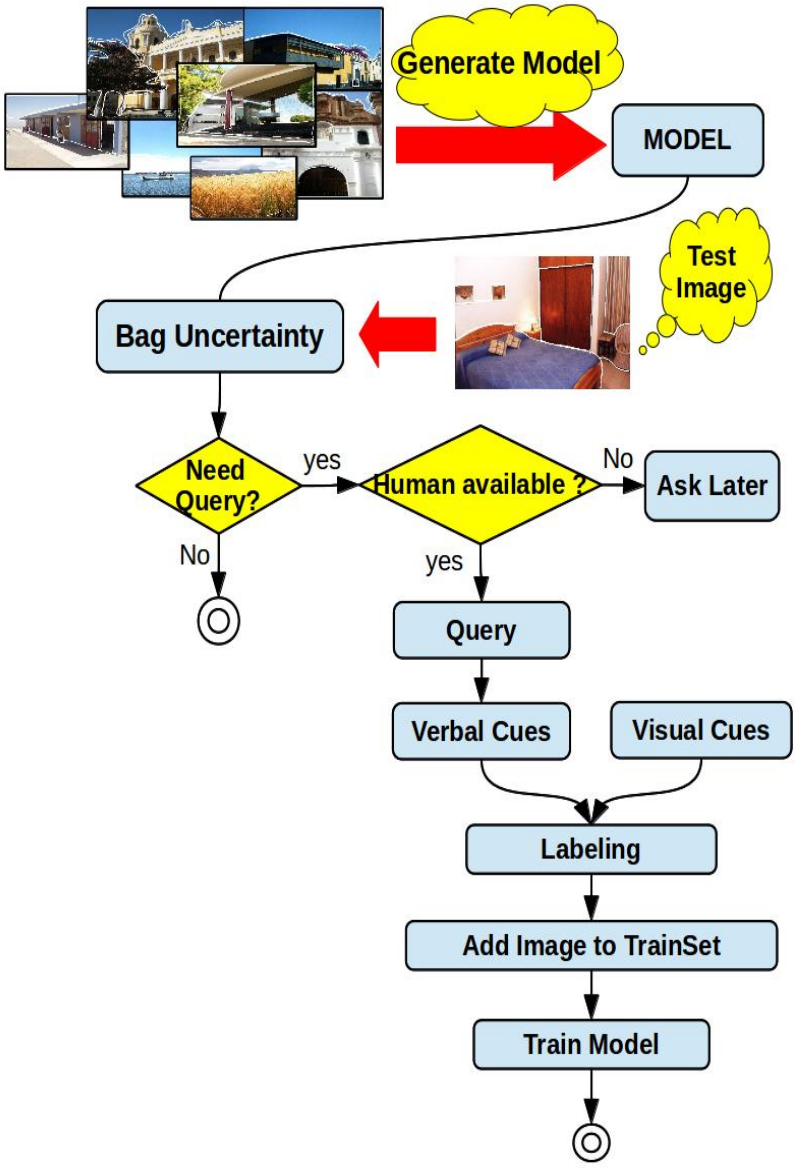

Figure 5.1: Proposed framework

5.3 Applied Active Learning Approaches

As mentioned earlier in Chapter 4, there are several approaches that pursue active learning. We apply a combination of stream-based and pool-based active learning in order to

reach the best simulation.

Stream-Based. After calculating the probability and subsequent uncertainty for the entered bag, a threshold is applied as an approach in stream-based active learning to either query the bag or ignore it. The query is only possible if the oracle is available; otherwise the bag identity and the uncertainty are stored until the oracle became available.

Pool-Based. During the absence of the oracle, queries are propagated into an unlabeled pool of bags, thus propelling us toward pool-based active learning.

5.4 Verbal Understanding

Querying and labeling instances in a bag take place through verbal interaction. Human feedback may involve one or more sentences with respect to a specific object. Negative sentences label the bag and each of its instances as negative. However, in positive expressions, the distinguished features of the object (such as color) are inspected to distinct the instance as the object. The assumption is that every positive sentence contains a color and object label. Each color label corresponds to specific RGB numbers. In order to label the instances, the RGB histogram of each instance in the bag is extracted and the instance with the least Euclidean distance is labeled positive for the object class.

$$i_p^* = \arg\min_i \sqrt{(R_i - R_c)^2 + (G_i - G_c)^2 + (B_i - B_c)^2}$$

where i is the instance, R_i, G_i, B_i are the RGB numbers of the instance i and R_c, G_c, B_c are RGB numbers of the mentioned color in the sentence. The queried test image is added to training set and the model is trained again. As illustrated in the figure 5.1, labeling requires both verbal and visual cues. Visual cues are extracted from the image while verbal cues require verbal understanding.

Given a sentence (a set of words), individual words can own several roles, such as *noun* and *verb*. Therefore, words requisite identity tags to illuminate its role in the sentence. Each word in a sentence is accompanied by a BIO and a Part Of Speech (POS) tag. BIO tags are according to IOB2 convention [15]. *B*, *I* and *O* represents beginning, inside and outside of the property label, 2 types of property labels are considered in our algorithm, category (CAT) and color (COL). POS tags are extracted for each word using Stanford Log-Linear POS Tagger [9] with Penn Treebank tag set [13], i.e., tags of the form: *NN, JJ, DT, VB*. Consider:

"There is a blue door in the image"

The sentence above is a prototype of a human feedback to label a specific instance in the bag. Every sentence provides knowledge of a special object, which in this case, the provided knowledge is that: *"the door in this image is the blue object"*. Provided knowledge may also deliver a negative content by using *not* or *no*. Figure 5.2 illustrates the POS and BIO tags for the above sentence.

Given the verbal entries, the semantic meaning of the vocabularies is extracted using a lexical database, called WordNet [5, 6]. WordNet is a large lexical database for nouns, verbs, adjectives and adverbs which are grouped into sets of cognitive synonyms (synsets). Synsets consist of all possible conceptual-semantics. WordNet's synsets are linked to other synsets with lexical relations i.e., *synonyms, antonyms, hypernyms* and *hyponyms*.

Color values such as *red, green* and *blue* have the same hypernym *chromatic_color* and thus have an *is_a* relationship since red is a chromatic color. Color values such as *black* and *white* have the same hypernym *achromatic_color* and second level hypernym of some colors such as *aqua, lime* and *olive* is *chromatic_color*, since olive is a type of green and green is a chromatic color. Figure 5.3 represents the *is_a* relation in Wordnet. The similarities are calculated between the given vocabularies and the vocabularies in the dictionary. Measuring similarity between two words is not only based on their *lexical* similarity but also their *content*.

As illustrated in Table 5.1, similarities are between 0 and 1. The dictionary, mentioned earlier, consists of all color and object vocabularies which are most likely to be used. Each word is represented in the following format:

word#pos#sense

where "*pos*" elucidate the part of speech as *n* for noun, *a* for adjective, *v* for verb and "*sense*" refers to the word's special semantic id.

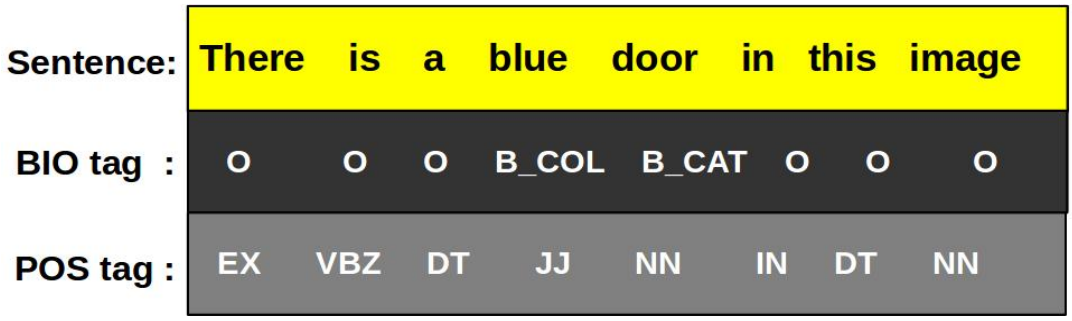

Figure 5.2: Part of speech and BIO tag for words in a sentence.

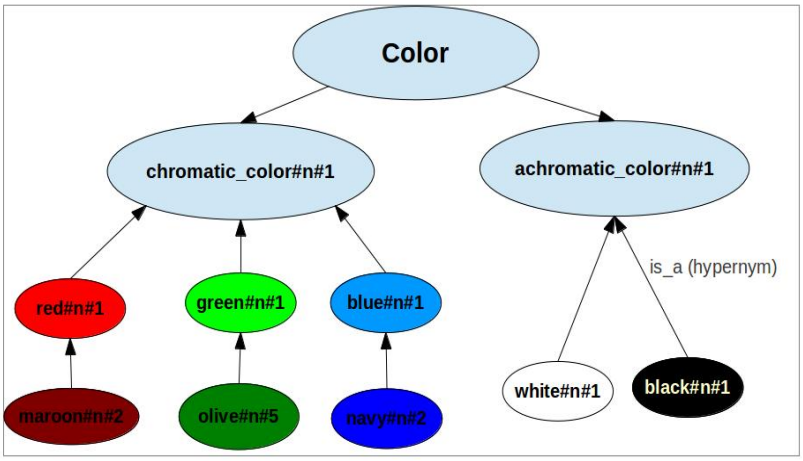

Figure 5.3: Hypernym relation in Wordnet

Table 5.1: An example to calculate similarities lexically.

Word	*red#n#1*	*Green#n#1*	*Blue#n#1*	*Aqua#n#1*
Blue	0.3	0.3	1	0.5
Green	0.3	0.3	1	0.25
Aqua	0.25	0.5	0.25	1

For example, when considering the word *blue* there may be many different semantics; however, it is necessary that only the semantic id (which refers to the blue as a color) be mentioned in the color's dictionary. As shown in table 5.2, the color *blue* and the word blue with semantic id 2 are not recognized as similar by Wordnet's similarity function.

Table 5.2: An example to calculate similarities with respect to the content.

Word	*red#n#2*	*Green#n#2*	*Blue#n#2*	*Aqua#n#2*
Blue	0.07	0.07	0.07	0
Green	0.07	0.07	0.07	0
Aqua	0.07	0.06	0.06	0

5.5 **Summary**

A new MI active learning framework is introduced. This framework utilizes verbal understanding to label newly entered images, calculates their probability of being positive and decides to add them to the training set; therefore, it is a combination of MI, active learning and verbal understanding. The overview of the framework is provided in the figure 5.1. The experimental results show that this method performs better than previous methods.

CHAPTER 6
EXPERIMENTAL SETUP AND RESULTS

6.1 Overview

This chapter provides experimental results of MI uncertainty, bag uncertainty and a comparison between them. We present information on the employed database and its features in section 6.2. Section 6.3 provides the default conditions under which the experiments were conducted. Finally, the experimental results are provided in section 6.4 and the discussion of the results are illucidated in section 6.5.

6.2 Database

We applied *IAPR TC-12 Benchmark* [4], a database with approximately 20000 images in 40 folders. Figure 6.1 shows examples of the images in the IAPR TC-12 Benchmark. Each folder contains:

Figure 6.1: Examples of images from the SAIAPR TC-12 collection.[1]

- **Segmentation masks**. One per region: 99,535 files; one per image: 20,000 files. Each object of reasonable size is segmented by using ISATOOL. On average, 5 objects per image have been segmented.

[1]Images are obtained from `http://imageclef.org/SIAPRdata`

- **Annotations**. One per region: 99,535 regions were manually annotated. Each segmented region is assigned a label from a carefully defined vocabulary, see [4]; the annotation vocabulary has been organized according to a conceptual hierarchy. For annotation the annotator went through the hierarchy from top to bottom looking for the best label for each object.

- **Spatial relationships**. One per image: 20,000 files. The following relationships have been calculated for each pair of regions in every image: adjacent, disjoint, beside, X-aligned, above, below and Y-aligned.

- **Visual features**. A vector of features per region: 99,535 vectors of attributes. The following features have been extracted from each region: area, boundary/area, width and height of the region, average and standard deviation in x and y, convexity, average, standard deviation and skewness in both color spaces RGB and CIE-Lab.

6.3 Experimental Setup

The results are obtained using 15 folders for training and evaluating the model and approximately 4 folders for the learning set (new images are randomly selected from these folders). We modified the *visual features* to help us improve the learning process. Our visual features consist of: boundary/area, convexity, HSB color histogram divided into 64 bins and finally, the instance label. Each image has a corresponding segmented image and each region is represented by a 67-dimensions vector.

We compare *Bag Uncertainty* (BU) against the best strategy introduced in previous research [3] *MI Uncertainty* (MIU). To highlight the difference between these two methods, incremental learning in BU occurs with a new learning set consisting of unseen images; while in MIU, it takes place on the instances of the positive bags of the train set. The MILR model uses $\alpha = 2.5$ for the softmax function and is trained by minimizing squared loss via L-BFGS [14].

Following the previous work [3], we evaluate our method by constructing learning curves that plot the area under the ROC curve (AUROC) as a function of instances queried for each object class and selection strategy. The primary point in all experiments is the AUROC for a model trained from labeled bags in the training set and with no queries. We average results over 10 independent repetitions for each class object category; the learner

starts with 200 randomly drawn positive bags and 200 random negative bags. The model is then evaluated on the rest of the unlabeled bags. The unlabeled query bags (bags in learning set) are selected randomly from 2515 new images. The threshold 0.3 is applied for the stream-based active learning on the queried bag. In other words, images with bag uncertainty greater than or equal to 0.3 are queried and added to the training set.

6.4 Experimental Results

This section provides experimental proof of this research. There are 9 case studies consisting of object categories and natural scenes. Both methods, BU and MIU, are compared on each of the case studies. Plots compare the AUROC of BU and MIU after 10,20,...,50 queries and tables present the improvement from the initial point. As mentioned earlier, BU enlarges the train set by newly seen images; therefore we expect this method to outperform MIU when a case study appears in different shapes or colors, e.g. door and car. However, this expectation is not satisfied unless there is a reasonable amount of positive bags in the learning set so that they could be seen and learned. We will discuss each of the case studies' results with respect to their plots and tables:

Tree: This object could be considered as a natural scene. Since almost all the trees are green (and we are learning a model mostly by colors); consequently, as shown in figure 6.2, BU and MIU correlate closely.

Street: Despite discussed object category *tree*, *streets* may have slightly different features; since learning set may contain a new aspect of *street* which lead to a better model, therefore, BU performs slightly better than MIU (figure 6.3).

Window and door: These object categories act similar to *street* only with the higher variation in the features.

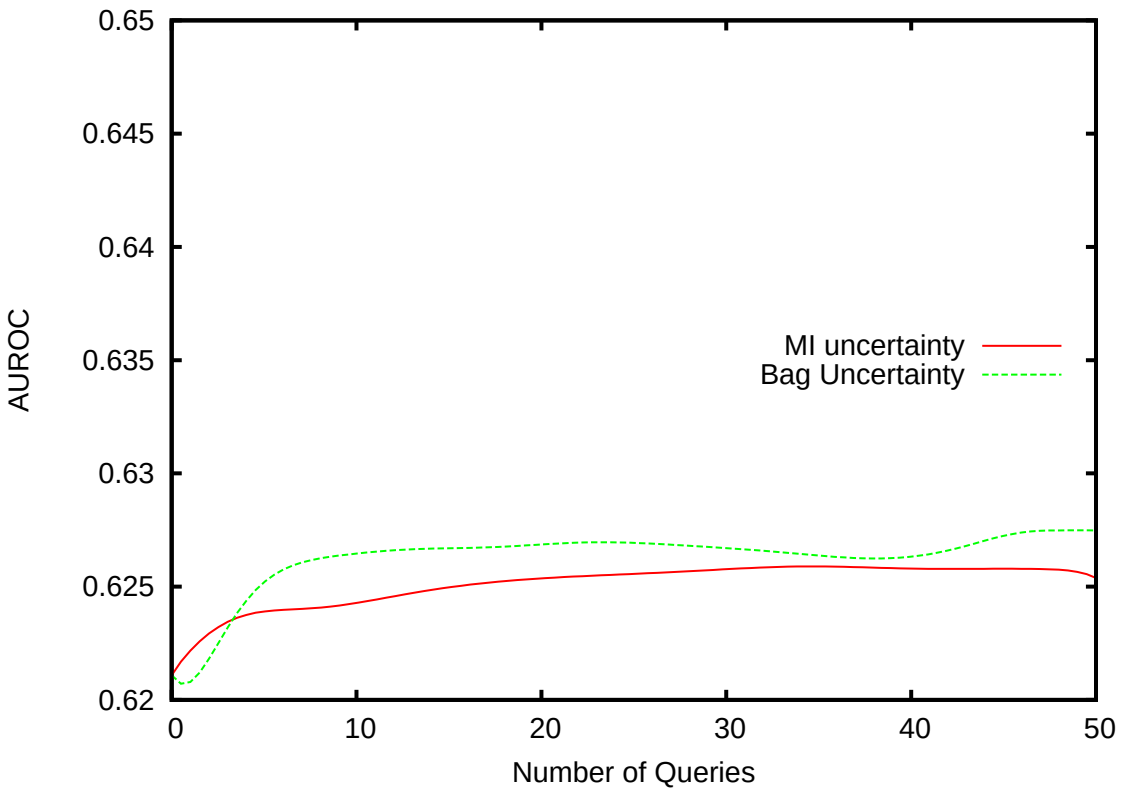

Figure 6.2: Comparison between MIU and BU query strategies for object class *tree*

Table 6.1: The average AUROC improvement over 10 repetitions for object class *tree*.

Instance Queries	*MIU*	*BU*
10	0.00289	0.00534
20	0.00427	0.00604
30	0.00466	0.00558
40	0.00462	0.00513
50	0.00429	0.00639

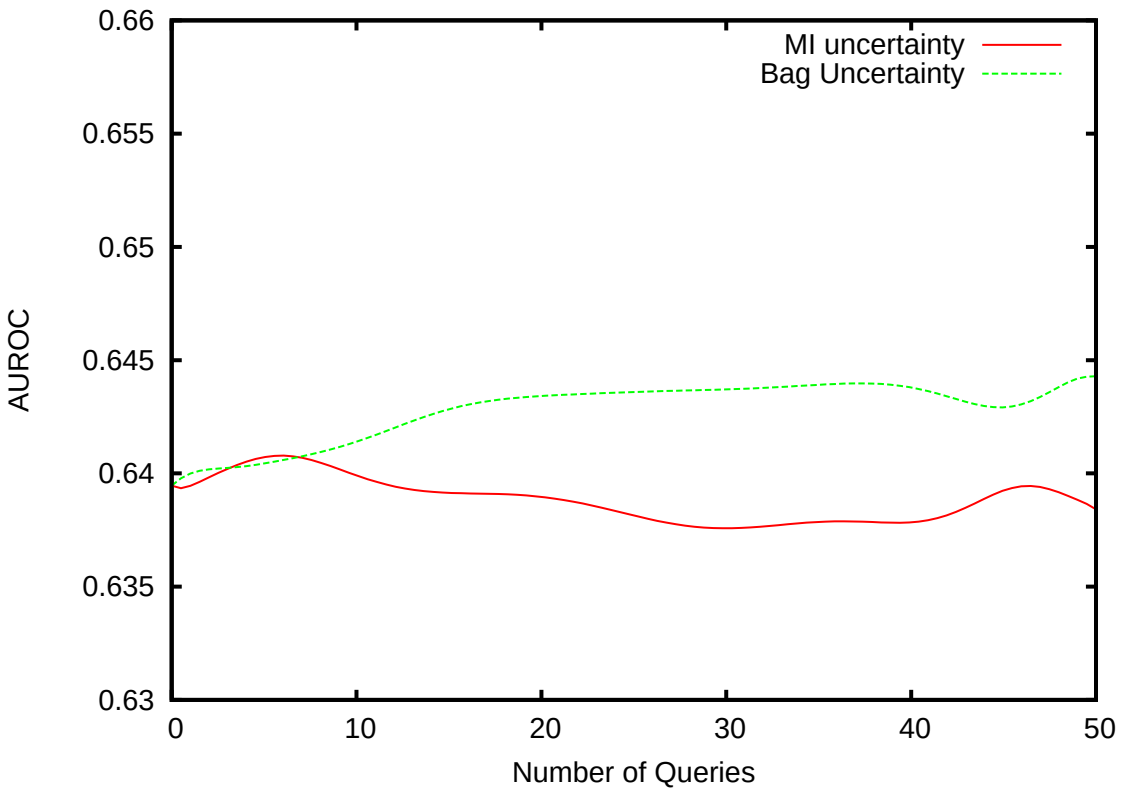

Figure 6.3: Comparison between MIU and BU query strategies for object class *street*

Table 6.2: The average AUROC improvement over 10 repetitions for object class *street*.

Instance Queries	*MIU*	*BU*
10	-0.0003	0.00156
20	-0.00042	0.00418
30	-0.00236	0.00417
40	-0.00243	0.00472
50	-0.00105	0.00483

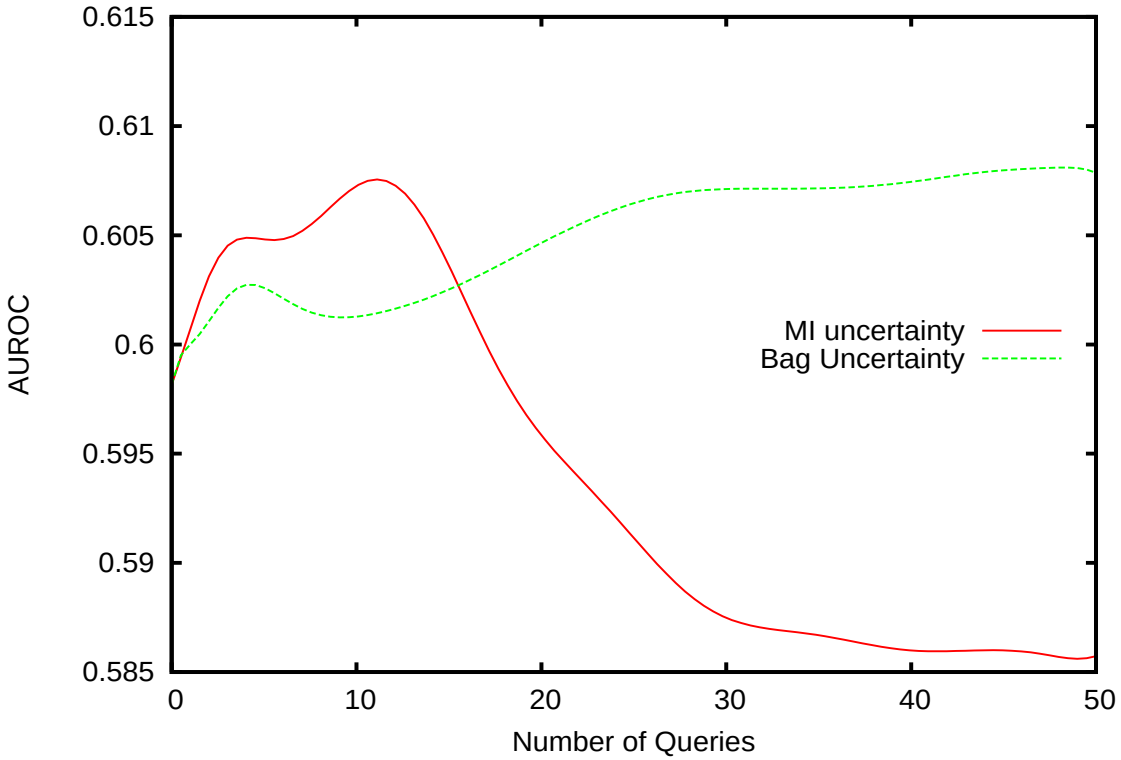

Figure 6.4: Comparison between MIU and BU query strategies for object class *window*

Table 6.3: The average AUROC improvement over 10 repetitions for object class *window*.

Instance Queries	*MIU*	*BU*
10	0.01191	0.00302
20	-0.00339	0.00729
30	-0.0117	0.00901
40	-0.01244	0.00918
50	-0.01243	0.0097

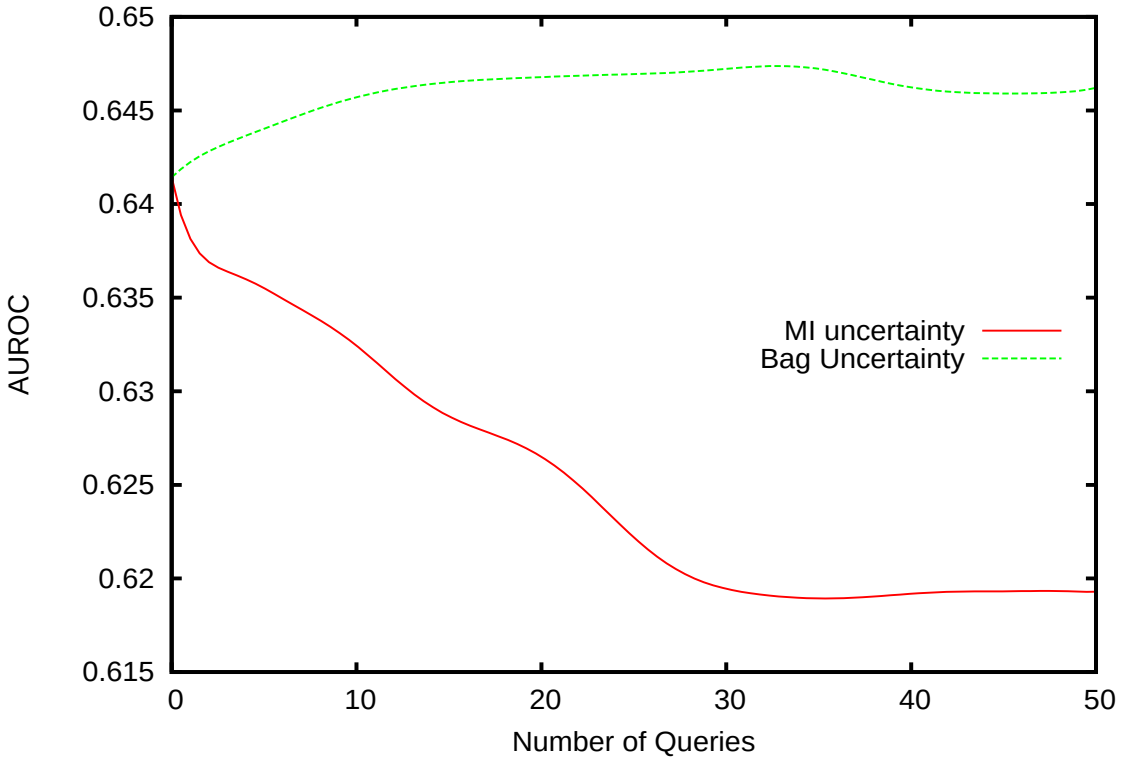

Figure 6.5: Comparison between MIU and BU query strategies for object class *door*

Table 6.4: The average AUROC improvement over 10 repetitions for object class *door*.

Instance Queries	*MIU*	*BU*
10	-0.00617	0.00466
20	-0.01374	0.0056
30	-0.02172	0.0056
40	-0.02207	0.00466
50	-0.02214	0.00478

Cloud, river and rock: These are categorized as natural scenes (tree, an instance of natural scene, is discussed earlier). Cloud can have different features depending on the weather which can not be captured thoroughly in the training set. BU improves the train set by unseen features of cloud and performs better than MIU (figure 6.6). Likewise, this condition applies to river and rock (figures 6.7 and 6.8).

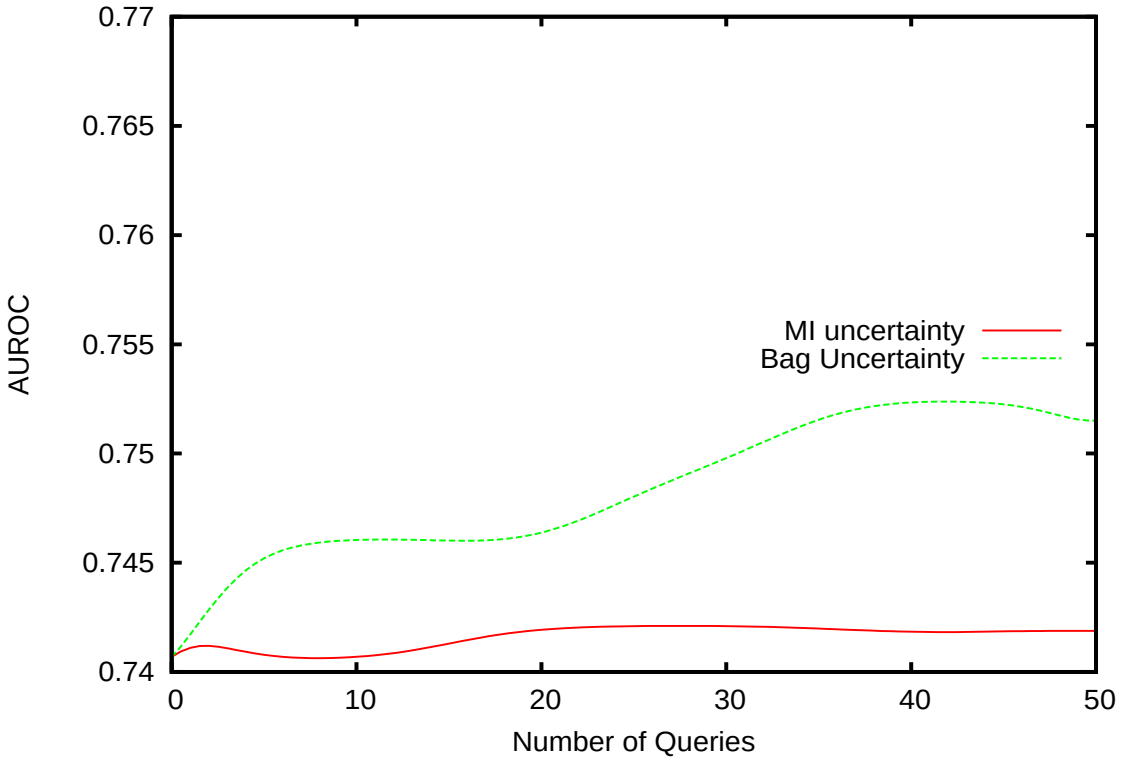

Figure 6.6:]
Comparison between MIU and BU query strategies for object class *cloud*

Table 6.5: The average AUROC improvement over 10 repetitions for object class *cloud*.

Instance Queries	*MIU*	*BU*
10	0.0001	0.00541
20	0.00131	0.00513
30	0.00143	0.00838
40	0.00106	0.01176
50	0.00119	0.01081

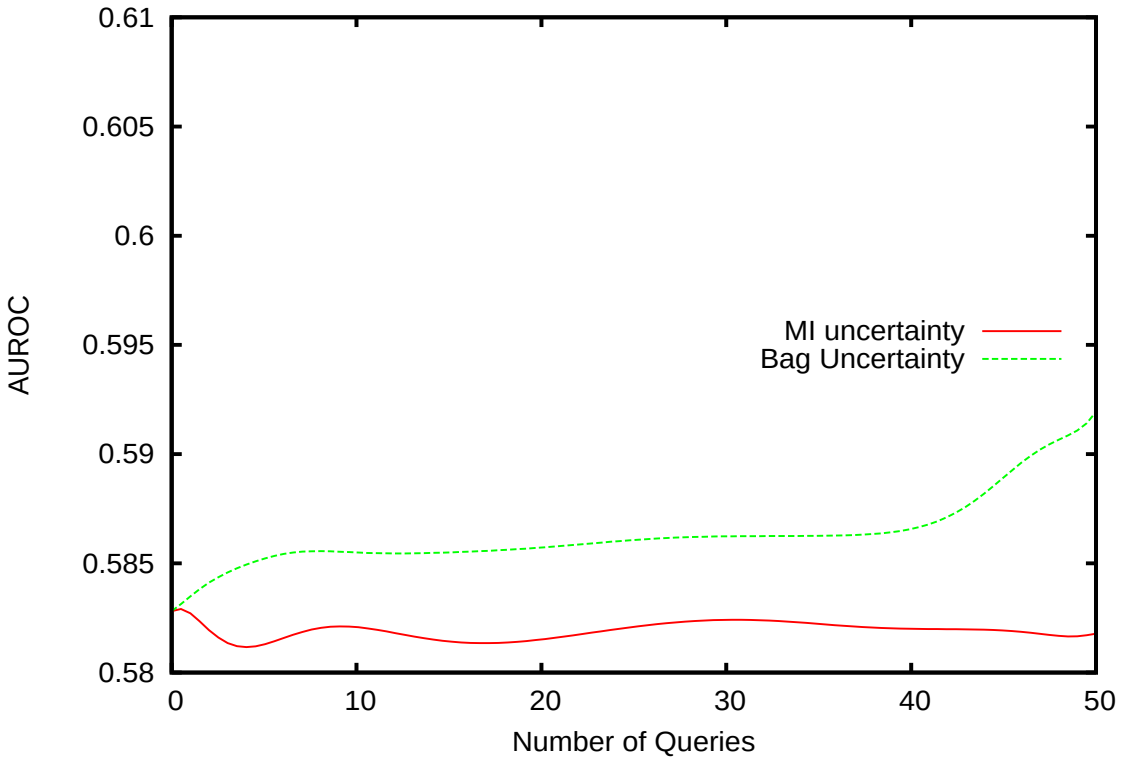

Figure 6.7: Comparison between MIU and BU query strategies for object class *river*

Table 6.6: The average AUROC improvement over 10 repetitions for object class *river*.

Instance Queries	*MIU*	*BU*
10	-0.00005	0.00226
20	-0.00137	0.00289
30	-0.00027	0.00344
40	-0.00083	0.00344
50	-0.00103	0.00913

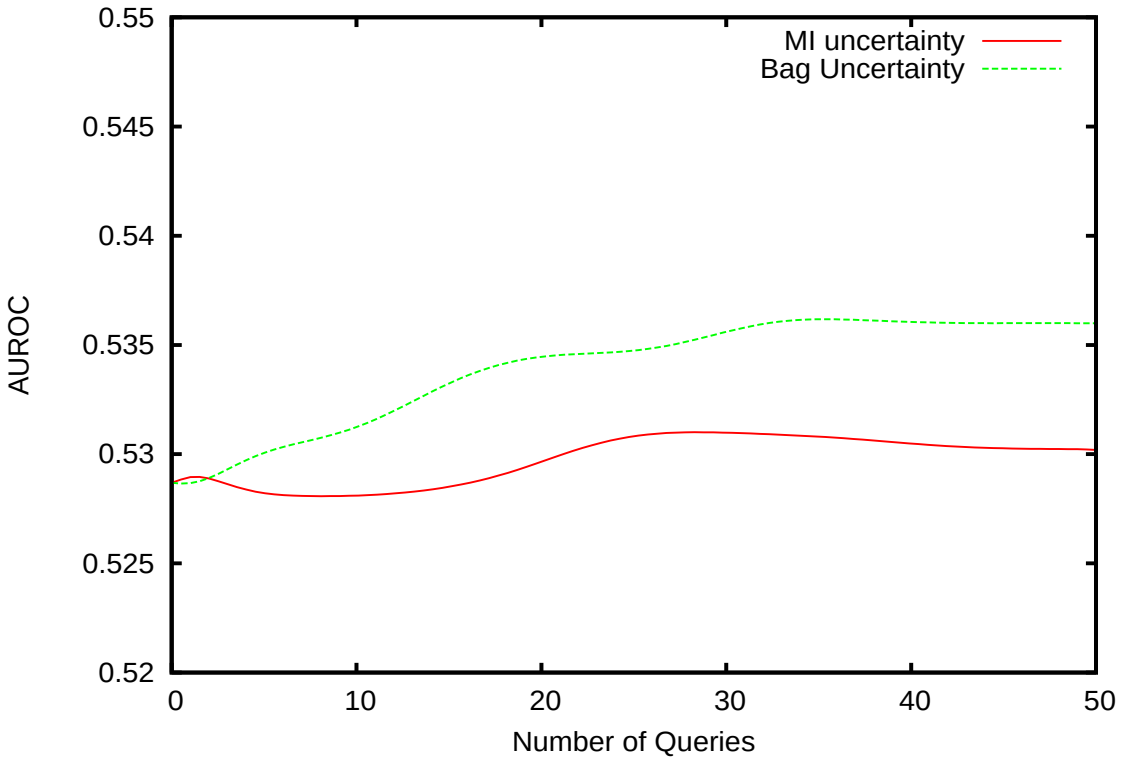

Figure 6.8: Comparison between MIU and BU query strategies for object class *rock*

Table 6.7: The average AUROC improvement over 10 repetitions for object class *rock*.

Instance Queries	*MIU*	*BU*
10	-0.00062	0.00223
20	0.00029	0.00603
30	0.00227	0.00751
40	0.00169	0.00743
50	0.0015	0.0073

Bottle and car: The initial dent in the BU curve may be caused from several negative bags, since the learning set bags are assigned randomly. However, after 19 queries BU outperforms MIU due to the BU's characteristic in expanding the knowledge on objects (figures 6.7 and 6.8).

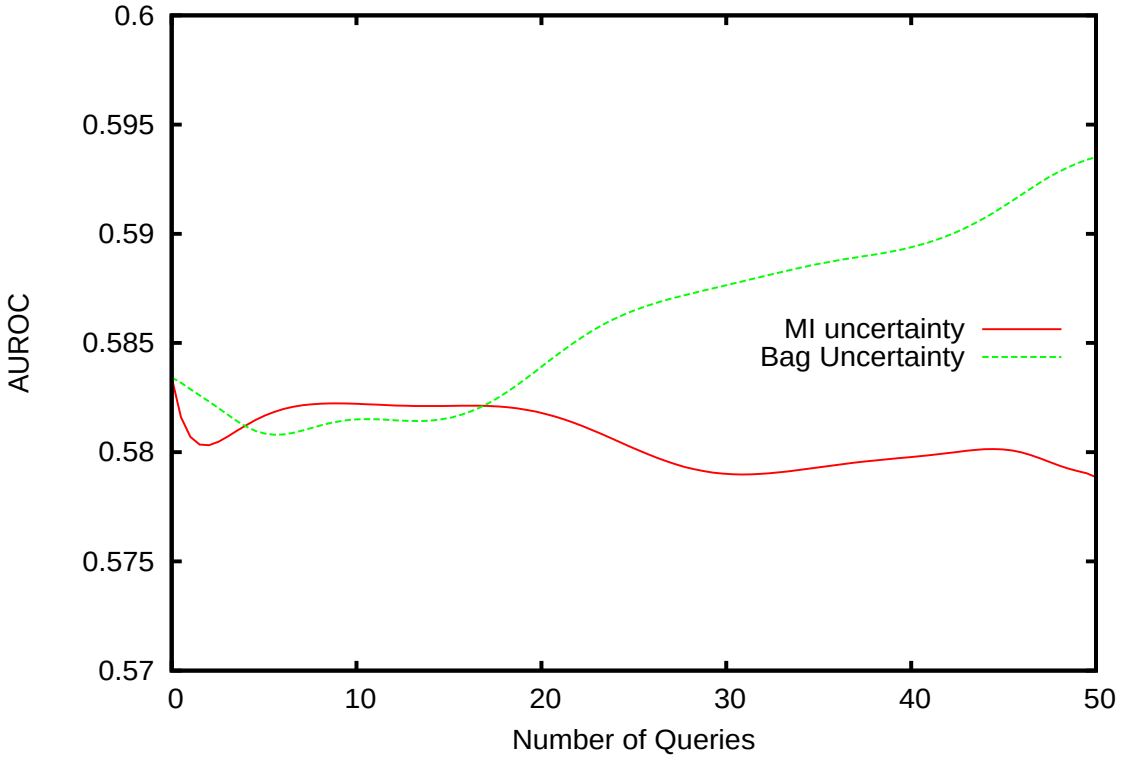

Figure 6.9: Comparison between MIU and BU query strategies for object class *bottle*

Table 6.8: The average AUROC improvement over 10 repetitions for object class *bottle*.

Instance Queries	*MIU*	*BU*
10	-0.00113	-0.00097
20	-0.00134	0.00008
30	-0.00471	0.0043
40	-0.00388	0.00657
50	-0.00456	0.01009

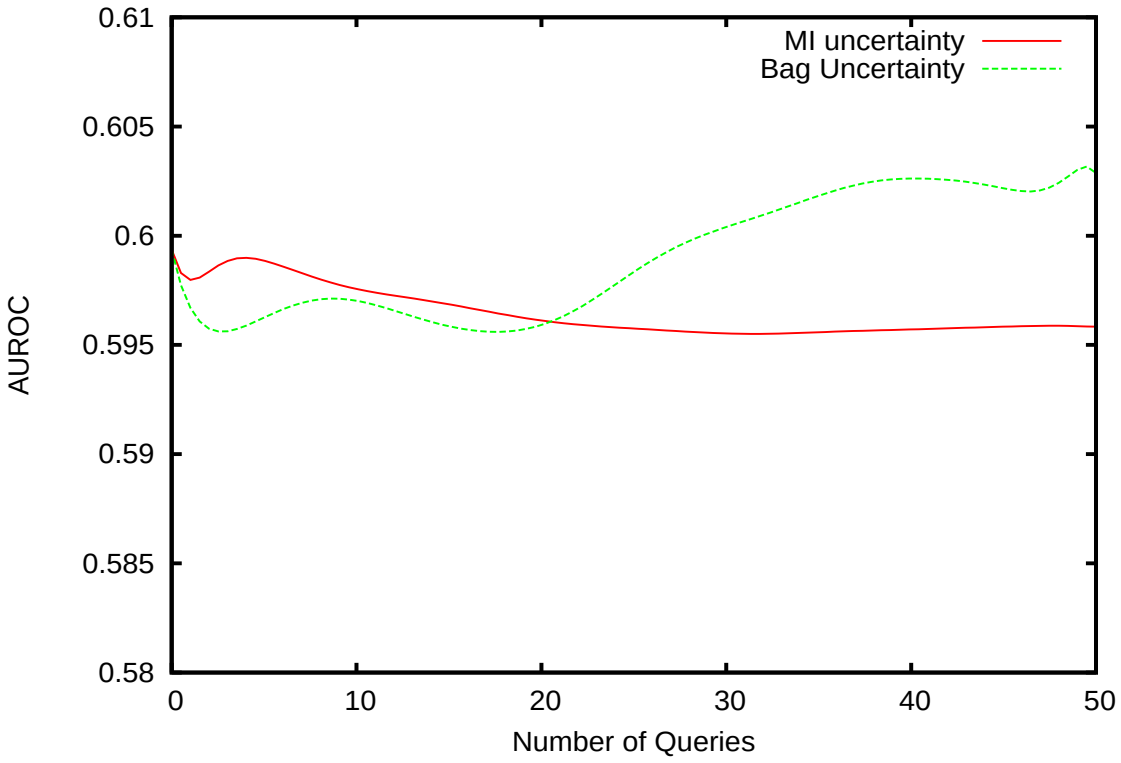

Figure 6.10: Comparison between MIU and BU query strategies for object class *car*

Table 6.9: The average AUROC improvement over 10 repetitions for object class *car*.

Instance Queries	*MIU*	*BU*
10	-0.00269	-0.00187
20	-0.00365	-0.00409
30	-0.00418	0.00082
40	-0.0037	0.00342
50	-0.00352	0.00351

6.5 **Discussion of The Results**

Addition of a newly observed scene as a bag, either positive or negative, promotes the trained model from multiple aspects. Initially, increasing the train set's size by actual bags which consist of several instances raises the chance of decreasing unknown scene to the robot. Both positive and negative bags affect the learning to increase.

Although online labeling with addressed features such as color and the existence of multiple objects with the same color in the image could draw instance labeling error, it would not affect the learning due to the MIL characteristic.

According to the tables and plots, BU outperforms the existing method known as MIU. However BU is reliant on the number of observed negative bags. In other words, If a robot observe negative bags and no positive bag while learning, this could drop the learning results since there was nothing to be learned. Total results after 50 queries are depicted in table 6.10.

Table 6.10: The average AUROC improvement after 50 queries over the initial MI learner. Numbers are averaged across all tasks for each object. The winning algorithm at each point is indicated by bold font.

Instance Queries	*MIU*	*BU*
Cloud	0.00119	**0.01081**
Street	-0.00105	**0.00483**
Rock	0.0015	**0.0073**
Tree	0.00429	**0.00639**
Window	-0.01243	**0.0097**
Bottle	-0.00456	**0.01009**
Door	-0.02214	**0.00478**
House	**0.01283**	0.001984
Car	-0.00352	**0.00351**
River	-0.00103	**0.00913**

CHAPTER 7
CONCLUSION AND FUTURE WORK

7.1 Conclusion

The previous approaches involved the background knowledge for actively learning of objects. Furthermore, the knowledge was restricted to what was observed in the training set in the beginning. In existing methods, the most uncertain instance is chosen to be queried and added to the training set as an individual bag; therefore, no new aspect of an object can be captured to improve the accuracy since there are no new image from the object involved. We tried to address these problems in this research by following steps:

- Proposing a new uncertainty measurement called *bag uncertainty* which estimates the uncertainty for unseen bags instead of uncertain instances chosen from training set.
- Proposing a new framework applying bag uncertainty with a combination of stream-based active learning and pool-based active learning. Stream-based active learning selects queries from the learning set randomly and pool-based active learning selects a query with highest uncertainty.
- Labeling occurs on new uncertain bags by verbal interaction with an oracle.

7.2 Future Work

The research described in this thesis is a step towards more natural interaction between robots and humans who may not have the time and expertise to provide elaborate and accurate feedback. Future work will investigate the use of knowledge representation architectures to encode prior knowledge of the application domain and specific scenes, thus enabling robots to make best use of human feedback when it is available and necessary. Another direction of research is to consider natural language representations that support efficient and incremental learning. The long-term objective is to enable human-robot interaction and collaboration in complex real-world application domains.

BIBLIOGRAPHY

[1] Ranjan Acharyya. *A New Approach for Blind Source Separation of Convolutive Sources - Wavelet Based Separation Using Shrinkage Function*. VDM Verlag Dr. Mueller e.K., 2008.

[2] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2004.

[3] Mark Craven Burr Settles and Soumya Ray. Multiple-instance active learning. In *Advances in neural information processing systems*, pages 1289–1296, 2007.

[4] Hugo Jair Escalante, Carlos A. Hernndez, Jesus A. Gonzalez, A. Lpez-Lpez, Manuel Montes, Eduardo F. Morales, L. Enrique Sucar, Luis Villaseor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer Vision and Image Understanding*, In press, 2009. doi: http://dx.doi.org/10.1016/j.cviu.2009.03.008.

[5] Christiane Fellbaum. Wordnet: a lexical database for english. *Communications of the ACM*, 38:39–41, 1995.

[6] Christiane Fellbaum. Wordnet: an electronic lexical database. In press, 1998.

[7] Andrew McCallum Gregory Druck, Burr Settles. Active Learning by Labeling Features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 81–90, 2009.

[8] Abhinav Gupta and Larry S. Davis. Beyond Nouns: Exploiting Prepositions and Comparative Adjectives for Learning Visual Classifiers. In *European Conference on Computer Vision (ECCV)*, 2008.

[9] Dan Klein Kristina Toutanova and Yoram Singer Christopher D Manning. Feature-Rich Part-of-Speech Tagging With a Cyclic Dependency Network. In *In Proceedings of HLT-NAACL*, pages 252–259, 2003.

[10] Oded Maron and Tomas Lozano-Perez. A Framework for Multiple-Instance Learning. In *Advances in Neural Information Processing Systems*, 1998.

[11] Oded Maron and Aparna Lakshmi Ratan. Multiple-instance learning for natural scene classification. In *In The Fifteenth International Conference on Machine Learning*, pages 341–349. Morgan Kaufmann, 1998.

[12] Afshin Rostamizadeh Mehryar Mohri and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT press, Cambridge, Massachusetts, 2012.

[13] Mary Ann Marcinkiewicz Mitchell P Marcus and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. volume 19, pages 313–330. MIT Press, 1993.

[14] Jorge. Nocedal and Stephen J Wright. *Numerical optimization*, volume 2. Springer New York, 1999.

[15] Adwait Ratnaparkhi. *Maximum Entropy Models for Natural Language Ambiguity Resolution*. PhD thesis, 1998.

[16] Paul Schermerhorn Rehj Cantrell, Matthias Scheutz and Xuan Wu. Robust spoken instruction understanding for hri. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, pages 275–282. IEEE, 2010.

[17] Pietro Perona Robert Fergus and Andrew Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–264. IEEE, 2003.

[18] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[19] Behjat Siddiquie and Abhinav Gupta. Beyond active noun tagging: Modeling contextual interactions for multi-class active learning. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2979–2986. IEEE, 2010.

[20] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: an Introduction*. The MIT press, Cambridge, Massachusetts, 1998.

[21] Ranjini Swaminathan and Mohan Sridharan. Towards Robust Human-Robot Interaction Using Multimodal Cues. In *Human Agent Robot Teamwork (HART) Workshop at the International Conference on Human-Robot Interaction (HRI)*, 2012.

[22] Cheng Yang. Image database retrieval with multiple-instance learning techniques. In *Proc. International Conference on Data Engineering, 2000*, pages 233–243, 2000.

[23] Qi Zhang and Sally A. Goldman. Em-dd: An improved multiple-instance learning technique. In *In Advances in Neural Information Processing Systems*, pages 1073–1080. MIT Press, 2001.