Active Learning for Image Ranking Over Relative Visual Attributes

by

Lucy Liang

Supervisor: Kristen Grauman
Department of Computer Science
University of Texas at Austin

Abstract

Visual attributes are human-nameable, cross-categorical image concepts used by people everyday to describe objects. For instance, cats are "furry" while turtles are not. Some shoes are "shiny" while others are not. These human-intuitive attributes have significant uses in the computer vision field. Yet it is even more intuitive, and informative, to be able to ascribe the *degree* of some attribute in an image *relative* to that of others. If asked to describe a donkey, we are much less inclined to name all the attributes associated with the animal and are more likely to describe it in terms of a similar animal, using relative descriptions such as "like a horse but *shorter* and with *longer ears*." In this way, we can use relative attributes to build a richer vocabulary for textually describing an assortment of images.

Currently, there exist rank learning methods that are useful for modeling relative visual attributes, but these methods generally require training samples in the form of relative partial orderings, labeled by humans. However, annotating training samples can be very time-consuming.

In this work, we investigate three active learning methods for image ranking over relative visual attributes designed to minimize the annotation effort. The active learners seek to select training samples that provide the most useful information by choosing the most ambiguous image samples for annotation at each step in the learning loop. We accomplish this by using a low margin-based approach to sample selection. Two of our three active learners operate exclusively under this condition. For the final proposed active learner, we introduce a novel form of active sample selection that selects training samples that are both visually diverse and satisfy the low-margin property.

Experimental results indicate that on average, the active learner employing a combination of the low-margin and visual-diversity property performs best, as it outperforms the other active learners that operate solely under the low-margin condition. In addition, all three active learners consistently outperform the baseline approach of random sample selection, supporting the effectiveness of our proposed active learning methods as applied to image ranking over relative visual attributes.

Acknowledgement

I am extremely grateful to my advisor, Kristen Grauman, for her continuous guidance. I would also like to thank Adriana Kovashka for assisting me through several technical hurdles along the way.

Finally, the state of my thesis would not have been what it is today without the encouragement and support of Nick Collins. Thank you for always being there for me.

Contents

List of Figures

Chapter 1

Introduction

Visual attributes are human-nameable mid-level image concepts (e.g., furry, shiny, round) that can be shared across multiple image categories. They can be machine-learnable and therefore have several applications for visual recognition systems in describing and categorizing images [11, 20, 21]. Since visual attributes are observable properties of images that are nameable but not themselves categorical, combinations of a handful of visual attributes can be used to describe a massive variety of images across multiple categories [11, 27]. This offers much more flexibility for object recognition than that gained from simple identity assignment.

For instance, we can train an identifier for ducks by showing the learning system multiple pictures of ducks. If we would like to train an identifier for chickens next, we could show the system multiple pictures of chickens, and so on. Clearly, this strategy will become expensive if we intend to train identifiers for a wide variety of other kinds of birds later on. Instead, it would save huge time and effort if we simply train *attribute* identifiers capable of describing various visual characteristics of birds. For example, we know that ducks have webbed feet and chickens do not, chickens have pointy beaks and ducks do not, and penguins have both webbed feet and pointy beaks. Then instead of training three separate identifiers for ducks, chickens, and penguins, we can instead train just two identifiers for the attributes *has webbed feet* and *has pointy beak* to uniquely identify the three birds.

Yet it is often even more intuitive, and informative, to describe images based on the *relative strengths* (as opposed to *absolute presence*) of visual attributes compared to those of others (e.g., X is rounder than Y but less round than Z). Visual attributes that vary across a spectrum are known as relative visual attributes [27]. They provide a more human-intuitive, semantically meaningful way to represent and work with visual data, making it easier to describe objects.

For instance, suppose someone asks you to describe a donkey. You are probably less inclined to describe all the features of the donkey (e.g., “It walks on four legs,

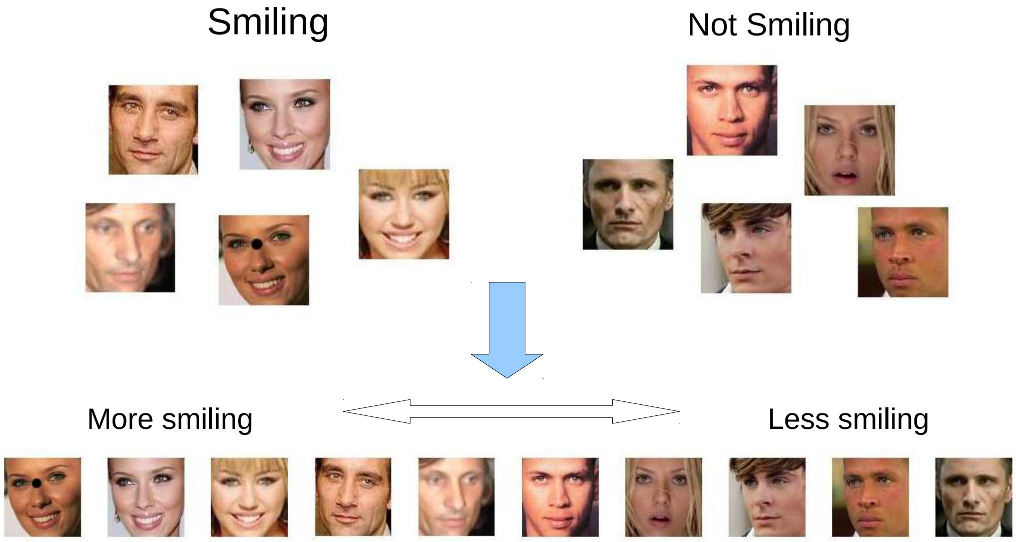


Figure 1.1: **Relative vs Binary Attributes** The top set of images are divided into two groups: *is smiling* and *is not smiling*. When we use this attribute as a binary classifier, we can overgeneralize and miss lots of other information. Compare this to the bottom set of images, which are ranked across a spectrum according to the relative attribute, *smiling*. Now we see just how much some images are more or less smiling than others, giving us a much clearer idea of how the images relate to each other over this relative attribute.

has a short mane, large snout, two eyes, two long ears, hooves, etc.") and more likely to compare it to a similar-looking animal such as a horse using relative attributes (e.g., "It looks like a horse but has a less thick mane, longer ears, etc").

Relative attributes also provide much more information than binary attributes since they vary across a spectrum (see Figure 1.1), whereas binary attributes are limited to providing only two pieces of information (e.g., "Does this object have X or not?"). This is especially helpful in cases where the absolute prescence or absence of an attribute is ambiguous. This greatly expands our ability to learn identifiers for unseen classes of images across a spectrum of categories [27].

Consider Figure 1.2. Suppose we want to learn an identifier for boots and already have knowledge of pumps, clogs, and their association with the attribute, *pointy*. The left image of the pump is clearly pointy while the right image of the clog is clearly not, but what would we assign the middle image of the boot? It remains ambiguous whether boots are pointy or not pointy. It is much easier and intuitive to just say "Boots are *less pointy* than pumps but *more pointy* than clogs."

In addition, we can extend the usefulness of relative attributes to practical applications such as image search [19]. Suppose we want to look for images of shoes similar to ones we already own, but more long on the leg and less shiny; using

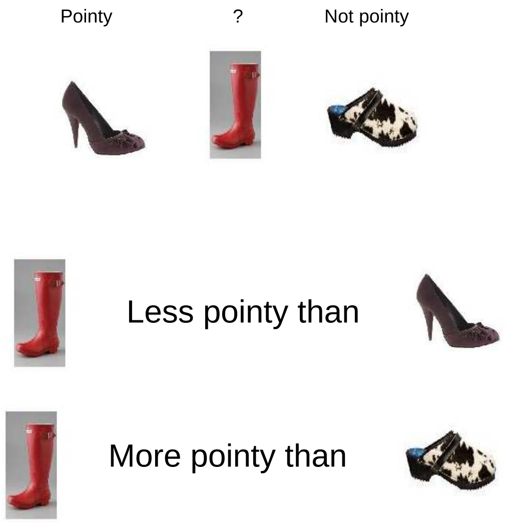


Figure 1.2: **Learning an Identifier with Relative Attributes** It is difficult to assign a value for to boot in the middle over the attribute, *is pointy*, since it is ambiguous whether the boot is pointy or not. It is much intuitive to use relative attributes and define the boot as: *less pointy* than the pump but *more pointy* than the clog.

relative attributes makes the process much more natural.

There exist a variety of methods for learning to rank [22, 17, 13, 16] that could also be applied to learn relative visual attributes. These ranking functions are generally trained by collecting annotations from humans in the form of click data, such that user preferences over a query (text, image, etc.) are used to learn a function that ranks items in terms of relevancy. Clustering and distance-based approaches are then used to induce a ranking over these images.

However, this form of data collection is less intuitive than judging over relative strengths of attributes. We prefer to ask questions in the form of "Is the face in Image 1 chubbier than the face in Image 2?" or "Of these images, which ones are most [some attribute]? Which ones are second most [some attribute]? ..." such that ranking by relative attributes over image samples come more naturally. To this end, we use a large margin based approach to rank learning that can be trained using relative comparisons among training examples [18], which offer a robust method for supervised learning over high-dimensional data. Training labels are simply in the form of sets of partial orderings (see Figure 1.3), and the goal is to learn a ranking function that imposes those desired orderings as well as generalizes to new, unseen images.

However, the labelling process necessary to learn an adequate ranking function is costly. Manually assigning partial orderings to sets of images can be expensive, and for many real world applications, the size of a candidate training pool of images

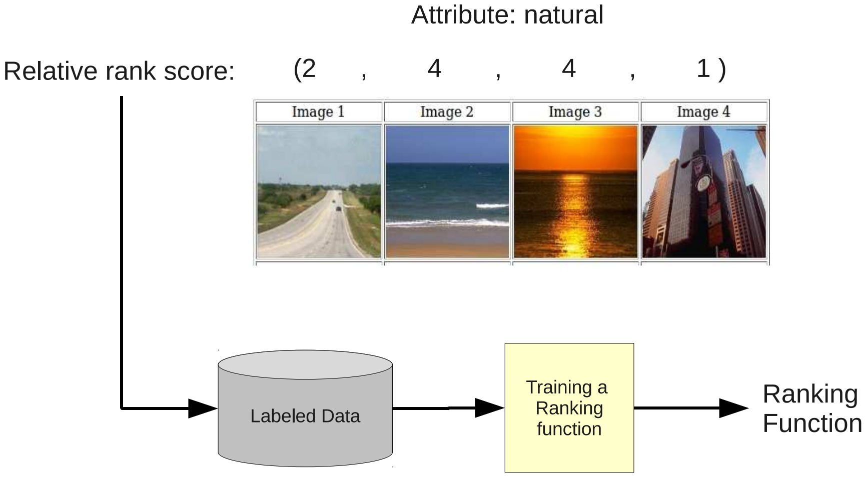


Figure 1.3: **Example of a Labeled Sample** To train a ranking function, we need annotations in the form of partial orderings over a set of images based on their relative strengths of the attribute. Notice here that the assigned rank values (2, 4, 4, 1) are not "absolute" scores of any sort, but are relative only among the set of images in this training sample.

may extend to the tens and thousands, making annotating all of them unfeasible. Multiply that by the number of attributes necessary to relate a number of different image instances and classes, and we quickly see how overwhelming the process can get.

Our goal is to introduce an active learning algorithm that selects the best images to use as training samples, thereby greatly minimizing the total number of training images and human labeling effort necessary to train an accurate ranking function.

To give an intuition of the role of active learning, consider the following example. Suppose we have an image database of peoples' faces containing 10% male faces and 90% female faces, and we conveniently want to learn a ranking function that ranks faces based on masculinity. If we simply selected image samples at random, we would select images with female faces 90% of the time, which might train a ranking function that is really good at judging relative levels of masculinity among female faces, but most likely we want one that is just as good at judging relative levels of masculinity among male faces. A good portion of the 90% of the time spent annotating female faces could have gone into annotating both male and female faces, which would yield a ranking function with much more generalizability using approximately the same amount of effort.

This is why we seek to implement sample selection techniques that make the most out of annotator effort at each step in the learning process so as not to waste any work. Factors to consider include whether or not we have seen a similar image before, how uncertain we are over these images' current rankings, how much information they will provide once ranked, and how easy or difficult it will be for annotators to rank the samples.

So far, most work on active learning has been over classification tasks(e.g., [31, 26, 34]) with some on ranking (e.g., [5, 38, 23]), although few have been put into practice for image-ranking in general(e.g., [33, 28]), and none have been applied to image ranking over visual attributes in particular.

Here, we investigate two active rank learning approaches and propose a third approach toward smart selection of image samples. All three approaches utilize a margin-based criterion, which seeks to pick out the image samples that were most difficult to rank at the previous iteration (i.e., images having the lowest rank margin) and would therefore provide the most useful information for training an updated ranking function at the next iteration once annotated.

The first approach, the *myopic active learner*, actively selects images samples on a pair-by-pair basis (considering only rank margins between pairs of data). The second approach, the *far-sighted active learner*, actively selects image samples that minimize the cumulative rank margin within a batch (based on [38]). The third approach, the *diversity-based far-sighted active learner* is an extension of the far-sighted learner that actively selects for low-margin image samples that are also visually diverse.

We demonstrate the effectiveness of these active learning methods in three unique domains: the Outdoor Scenes [25] data set, the Public Figure Face [20] data set, and the Shoes [2] data set. In addition, we apply our active learning models to up to 27 distinct relative visual attributes over the three data sets to show the performance of our learning method across various visual contexts. The three active learners will be tested against each other as well as against three baseline approaches discussed further in Section 4.2.

Results indicate that all three active learners consistently outperform the baseline approach of random sample selection, suggesting their effectiveness in image ranking over relative attributes. Among the active learners, on average, our proposed diversity-based far-sighted learner outperforms the far-sighted learner, and the far-sighted learner outperforms the myopic learner, suggesting that a combination of both the low-margin condition and visual-diversity condition is most effective for learning to rank images.

In this work, I will investigate low margin-based active learning methods and apply them to the novel problem of image ranking over relative visual attributes as well as introduce a new active learning method for ranking over relative attributes. Chapter 2 discusses related work and explains the context for my contributions. Chapter 3 describes the process of learning a ranking function and introduces the three active learners in detail. Chapter 4 presents the experiments and results. We conclude the paper with Chapter 5, which summarizes the findings and suggests potential candidates for future work in this area.

Chapter 2

Related Work

The importance of visual attributes has been well-explored in computer vision. Several works have been dedicated to discovering these types of attributes from public sources [2, 36]. These attributes can benefit object recognition [11, 3] and facial recognition [20]. They also allow us to describe classes of images using human-intuitive semantics that enable zero-shot learning of, and knowledge transfer between, classes and concepts [21, 29].

In the case of relative information, several works use relative data as applied to images in terms of relative degree of image similarity [20, 35], although these do not involve relative visual attributes since they require training a system to a specific model or image category, as opposed to category-independent relative attributes. In more recent work, relative visual attributes have been explored with applications to zero-shot learning [27] and image search [19].

With regards to learning to rank, there have been several works dedicated to rank learning algorithms including those making use of margin-based criterion [18], bipartite or k-partite ranking with boosting [28, 12], distance-based approaches [13], and works investigating supervised learning using both the pairwise and listwise approaches [7, 38]. In practical applications, rank learning has largely been applied to document and text retrieval [18, 22, 6] and to some extent image retrieval [17, 16]. For the latter, annotations generally take the form of click data, evaluating a human user's preferences towards an image query. However, the ranking functions in these works are intended for ranking images in terms of their relevance to a specific query and not over relative visual attributes. While some recent works *have* investigated rank learning over relative visual attributes [27, 19], they do not investigate active learning as we will do here.

Much work has been done on both margin-based [1, 31, 30] and diversity-based [37, 4, 24, 15] active learning techniques for classification over documents and images [39, 26, 34, 14, 9]. The goal of these works is to actively learn classifiers that distinguish between relevant and irrelevant documents or images with regards to some query by selecting data items that would provide the most information for learning an updated classifier once annotated. Margin-based active learning for *ranking* has also received attention [5, 38]. The goal there is to actively learn ranking functions that rank items according to some criteria by selecting data items that would provide the most information for learning an updated ranking function once annotated.

Most applications of active learning to ranking, however, have largely fallen in the context of text and document retrieval [10, 23]. Works applied to active learning for image ranking in particular are more rare. While a few exist [33, 28], they use "bipartite" ranking methods that are still largely based on traditional classification strategies. Questions are of the form "is this image relevant or irrelevant?" and then a ranking is extracted from what are essentially binary labels (each image only receives one of two labels, "relevant" or "irrelevant"). On the other hand, our approaches allow multiple images within a batch of samples to be uniquely ranked relative to each other, which is not only more informative, but also a more natural way of ranking as opposed to assigning absolute scores to individual images. Also, like most other works involving image-ranking, these only focus on active learning for image ranking over query relevance and not over relative visual attributes. This is a very important distinction for the following reasons:

- In relevancy judgment, the highest ranked items are the most important, and therefore the ordering goes one way. On the other hand, ranking over relative attributes is a bidirectional process (e.g., "X is more masculine-looking than Y" is just as important as "Z is less masculine or more feminine-looking than Y").

- Ranking over relative attributes is more complicated, because it is more subject to individual differences in opinion (e.g., "Are lips that are thinner but wider considered bigger or smaller than lips that are thicker but narrower?"). In many cases, there may not be a perfect, indisputable, ground-truth ranking at all.

- Relevance judgment in information retrieval is still a membership problem (e.g., "How relevant is this data item to this topic or class?"), where everything is based on some ideal model. This is not the same with relative attributes, where concepts are cross-categorical and have no real "anchor." The same set of features may or may not indicate greater presence of the attribute under different conditions. It may be the the case that what constitutes greater attribute presence is dependent on the context (e.g., "There is greater presence of maleness in a face when it possesses more facial hair, but only if it also has more facial musculature").

My main contribution is to apply three active learning strategies to image ranking over relative visual attributes, which to the best of our knowledge, has not yet been investigated.

The first two active learning strategies, which we call the myopic and far-sighted active learners, are adapted from Yu's [38] approximated selective sampling (ASEL) and selective sampling (SEL) methods, respectively. The former method approaches low margin-based sample selection on a pair-by-pair basis, considering only margin distances between pairs of samples, while the latter method considers the cumulative margin distance among all members within a batch of samples. Yu's active learning strategies appear promising, but to our knowledge, they have only so far been applied to artificial rank data or labels that did not arise from relative judgements. Instead of asking annotators to rank sets of images up front, these experiments asked annotators to impose absolute (not relative to anything) rank scores on individual query items. Here, we seek to apply the active learning methods to actual images, querying annotators with sets of images that must be ranked based on the relative strengths of some visual attribute as opposed to assignment of absolute rank scores.

The third proposed active learning strategy is called the diversity-based far-sighted learner. It adheres to the low-margin condition while simultaneously selecting for visually diverse image samples.

Chapter 3

Approach

We first explain how we train a ranking function for relative visual attributes (Section 3.1). Next, we discuss the general active learning setup (Section 3.2). Finally, we introduce three types of active learners: the myopic active learner (Section 3.3.1), the far-sighted active learner (Section 3.3.2), and the diversity-based far-sighted active learner (Section 3.3.3).

3.1 Training a Ranking Function for Images

Suppose we are given a set of images $I = \{i\}$ represented in $\mathbb{R}^n$ by feature vectors $\{x_i\}$ describing their texture, color, or other low-level cues. Let "$x_i >_a x_j$" denote "x_i has more of attribute a than x_j" and "$x_i =_a x_j$" denote "x_i has the same amount of attribute a as x_j." Then given a set of pairs $O = \{(x_i, x_j)\}$, if $(x_i, x_j) \in O$, then $x_i >_a x_j$, and given a set of pairs $S = \{(x_i, x_j)\}$, if $(x_i, x_j) \in S$, then $x_i =_a x_j$ (see Figure 3.1).

The goal is to learn a ranking function F:

$$F(x_i) = w^T x_i \tag{3.1}$$

that best satisfies the following constraints:

$$\forall (x_i, x_j) \in O : w^T x_i > w^T x_j \tag{3.2}$$

$$\forall (x_i, x_j) \in S : w^T x_i = w^T x_j, \tag{3.3}$$

where $w \in \mathbb{R}^n$ is the weight vector to be learned. In other words, we aim to learn a weight vector, w, that ranks the images in the desired order dictated by O and S by projecting their corresponding feature vectors onto w.

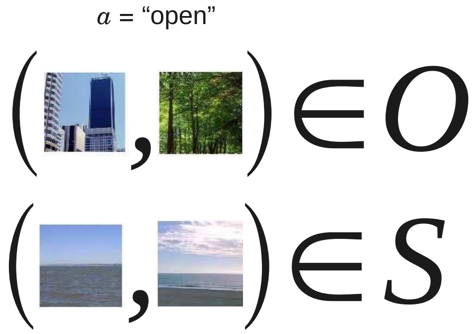


Figure 3.1: **Example of Pairs from Sets O and S for Attribute: Open.** The top pair of images belong in the set of pairs, O, because the left image of buildings is more open than the right image of trees. The bottom pair of images belong in the set of pairs, S, because both images of coasts are equally open.

This NP-hard problem can be approximated by using the method described in [18] and introducing non-negative slack variables, $\gamma_{i,j}$ and $\epsilon_{i,j}$. The reformulation also imposes a parameter, C, as a trade-off constant between maximizing the distance between the closest data pairs (x_i, x_j) when projected onto w (we refer to this distance as the *rank margin*, or $\frac{1}{\|w\|}$), and minimizing the training error by satisfying the constraints in equations 3.2 and 3.3 (see Figure 3.2). This becomes the following optimization problem.

Minimize:

$$\frac{1}{2}\|w^T\| + C\left(\sum \epsilon_{i,j}^2 + \sum \gamma_{i,j}^2\right) \tag{3.4}$$

Subject to:

$$\forall (x_i, x_j) \in O : w^T x_i \geq w^T x_j + 1 - \epsilon_{i,j} \tag{3.5}$$

$$\forall (x_i, x_j) \in S : |w^T x_i - w^T x_j| \leq \gamma_{i,j} \tag{3.6}$$

$$\forall (i, j) : \epsilon_{i,j} \geq 0, \gamma_{i,j} \geq 0. \tag{3.7}$$

By rearranging constraints of Equations 3.5 and 3.6 to:

$$\forall (x_i, x_j) \in O : w^T (x_i - x_j) \geq 1 - \epsilon_{i,j} \tag{3.8}$$

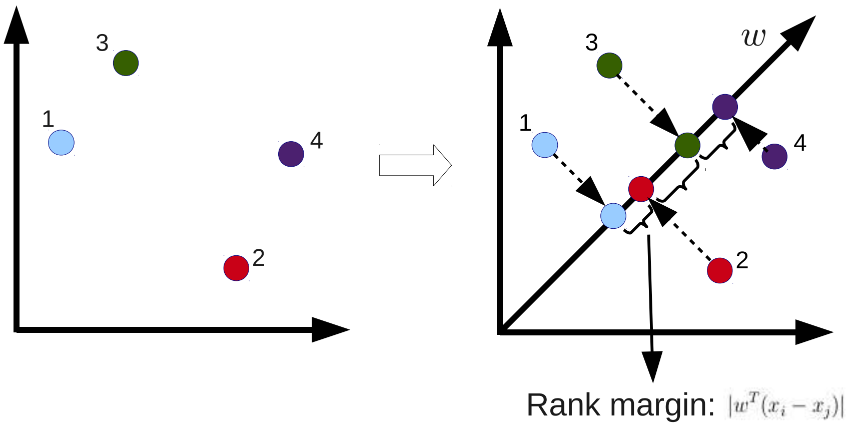


Figure 3.2: **Learning a Ranking Function** The rank margin is the distance between the two closest projections onto w. When learning a ranking function, we seek to derive a w that imposes the desired ordering while maximizing the margin to increase the ranking function's generalizability.

$$\forall (x_i, x_j) \in S : |w^T(x_i - x_j)| \leq \gamma_{i,j}, \tag{3.9}$$

these now take the form of support vector machine (SVM) classification constraints on pairwise difference vectors $(x_i - x_j)$. The rank margin mentioned above $(\frac{1}{\|w\|})$, which in classification problems represents the distance from support vectors to the boundary, here in *ranking* represents the distance between the closest data pairs when projected onto the learned weight vector, w. This problem is now solvable using Newton's method [8]. Although we use linear ranking functions in our experiments here, the formulas above are kernelizable and thus extend to nonlinear ranking functions.

By minimizing the slack variables while adhering to the constraints in Equations 3.8 and 3.9, we learn a ranking function that satisfies those constraints with minimal error. By maximizing the rank margin $\frac{1}{\|w\|}$ (or minimizing $\|w\|$), we learn a ranking function that has maximum generalizability (see Figure 3.3).

3.2 Active Learning

Recall that O is the set of pairs (x_i, x_j) of feature vectors such that x_i has more of some attribute than x_j, and S is the set of pairs (x_i, x_j) of feature vectors such that x_i has the same amount of the attribute as x_j. Once we establish how to learn a ranking function, the next step is to determine how to populate these sets such that we can best learn an accurate ranking function.

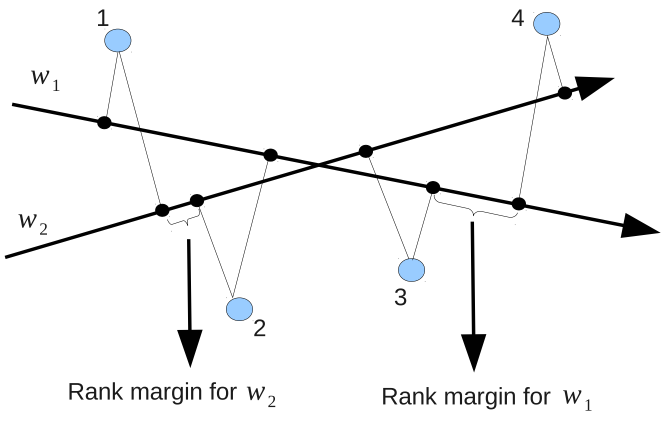


Figure 3.3: **Maximizing the Margin** Here, two candidate weight vectors, w_1 and w_2 rank four points. w_1 is the better candidate, because it yields the largest rank margin.

Like most supervised learning settings, we can simply query a human annotator for labels. For pairwise rankings, questions may be in the form of “Is Image 1 more or less [some attribute] than Image 2?” For batch selections, questions may be in the form of “Of these set of images, assign them rank labels from 1-4 with 4 being most [some attribute] and 1 being least [some attribute]” or “Of these set of images, select the ones with the most [some attribute], then do so again with the remaining images until all have been processed.” Annotating pairs of images at a time is easier, but annotating batches of images provides more information at a time [7].

Theoretically, one can annotate hundreds and thousands of images with rank labels in order to train an accurate ranking function. Realistically, this is inconvenient, highly time consuming, and likely frustrating for the annotator. It would save much human effort if we aim to select only image samples for labeling that will provide the most information once annotated, thus populating O and S with only the most useful samples.

Consider this scenario. Suppose we have a set of images of scenes, including pictures of mountains, forests, oceans, highways, and tall buildings. At first, we select two images, one of a mountain and one of a highway, for an annotator to rank with regards to the attribute, *natural*. The annotator ranks the image of the mountain as more natural than the image of the highway, and we train a ranking function using this new information. The rudimentary ranking function is now likely to rank pictures of similar-looking mountains as more natural than images of similar-looking highways.

Now suppose for the second round, we choose two more images of a mountain and highway that look pretty similar to those from the first set. The annotator once again ranks the image of the mountain as more natural than the image of the highway. We train a new ranking function using this updated information, and produce one that does not act much differently from the first one. This is because we selected, for our second sample set, images whose relative rankings the predictive model was already pretty sure of since it had seen something similar before. Had we selected images whose rankings were more uncertain, such as pictures of oceans or tall buildings, the refined ranking function would be much more useful the second time around. Essentially, at each step in the learning process, we use newly labeled image samples to fine tune the ranking function, but this is only effective if the image samples actually provide new and meaningful information.

This is the sample selection problem that active learning seeks to solve. The goal is to choose only image samples whose annotation will provide not only new, but maximum information towards training a ranking function. To determine which new training samples to select for the current iteration, we must utilize information gained from the previous iterations, which in this case is the most recently learned ranking function (see Figures 3.4 and 3.5).

In classification problems, the learning machine might select samples that lie closest to the classification boundary, because they were the most difficult to classify and therefore are likely to provide the most information once annotated. We apply a related rationale to image ranking.

In the image ranking problem, the distance between adjacent feature vector pairs when projected onto the weight vector, w, represents the confidence level in those assigned rankings. The smaller the distance, or margin, between a data pair, the less certain the machine is about how that data pair was ranked. Therefore, the closest feature vector projections, or those with the smallest margin, make up the support vectors of ranking — samples that were most difficult to rank and therefore likely to provide the most information once annotated. This is the low margin-based sample selection method that all three of our active learners will employ in the upcoming experiments. Descriptions of these active learners follow in the next section.

3.3 Actively Learning to Rank

We now introduce three different types of active learners that utilize the low margin-based sample selection approach discussed in the previous section. Section 3.3.1 presents a conventional pairwise low-margin approach, or the *myopic active learner*. Section 3.3.2 presents a listwise low-margin approach that considers cumulative margin spaces beyond that between a data pair, or the *far-sighted active learner*. Section 3.3.3 introduces a listwise low-margin approach that is also sensitive to visual similarity among images, or the *diversity-based far-sighted active learner*, also called the *far-sighted-D active learner*.

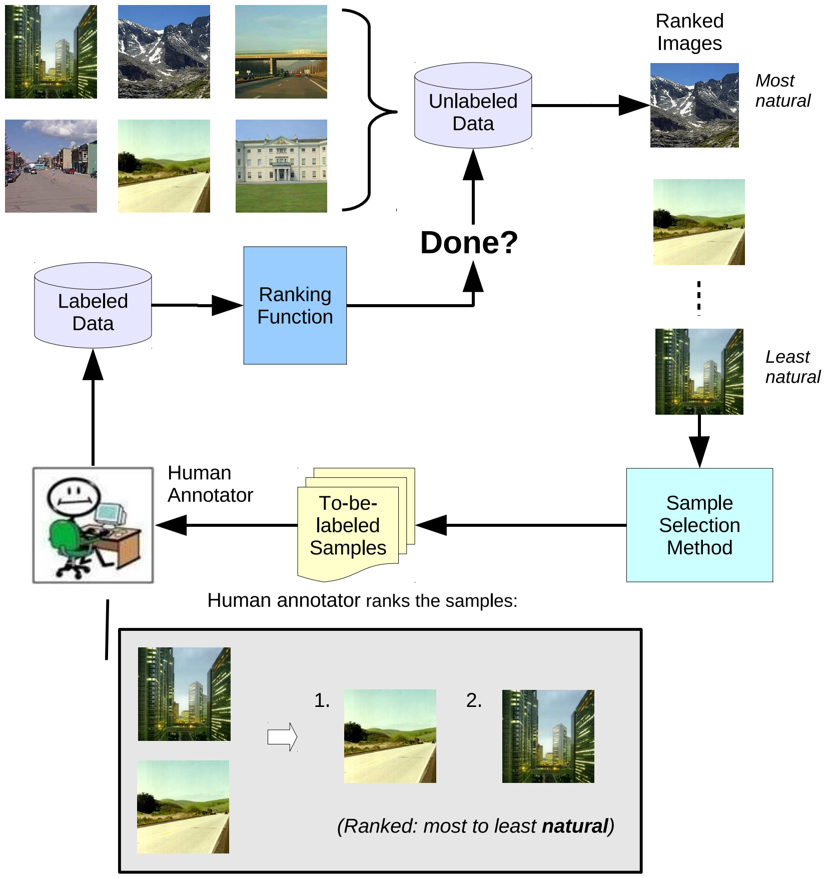


Figure 3.4: **Active Learning for Image Ranking.** The active learning loop involves selecting a sample of images from an unlabeled pool of data for the annotator to rank, adding the annotation to an accumulating labeled set, using the updated labeled set to learn a new ranking function, applying what we just learned (i.e., the ranking function) to the unlabeled set, and using the result to actively select for a new set of samples for annotation.

3.3.1 Myopic Active Learner - A Pairwise Low-Margin Approach

Annotation for ranking requires at least a sample size of $s = 2$, or one data pair. The myopic active learner employs the low margin-based sample selection approach presented in Section 3.2 by selecting the data pair whose members lie closest to each other when projected onto the weight vector, w (see Figure 3.6).

Although annotating a pair of images is quick and easy, annotating a batch of $s > 2$ images provides much more information per iteration. The myopic learner can extend to selecting for samples of size $s > 2$ by simply selecting the $\frac{s}{2}$ data pairs with the lowest margins (see Figure 3.7). Note that s must be even. Figure 3.8

1. Randomly select a number of samples from an unlabeled set U to order.
2. User orders all s samples.
3. Add the samples with the orders to the labeled set L and learn a ranking function F (see Section 3.1) from it.
4. If F satisfies some condition, return it. Otherwise, select s number of samples from U using an active sample selection method.
5. Repeat from step 2 until an accurate enough F is reached or until we have exhausted the data from U.

Figure 3.5: **Steps for Active Learning in Ranking**

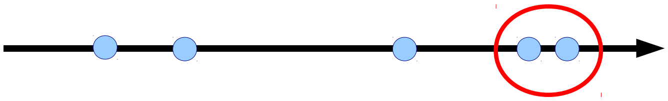


Figure 3.6: **Myopic Learner and Pair Selection.** The myopic learner, under the low-margin condition, selects the data pair with the lowest margin, or members that lie closest to each other.

details the steps of this process.

This learner is called the myopic learner, because it considers only the distances within, but not among, every data pair when selecting for image samples. This selection method is cost-efficient since it only requires computing the distance between each adjacent pair of data to determine a sample set. The image samples might also be easier for humans to rank since there may be sufficient visual dissimilarity (with regards to the attribute) between pairs. For instance, suppose $s = 4$, then the myopic learner may choose an image pair that both have a lot of some attribute and another image pair that both have very little of the same attribute. It would be easier to rank these four images as a whole than had they all been high-scoring candidates. Of course, since this learner does not minimize the *cumulative* distance among all s samples, even though it does select for an informative batch of image samples, it is not necessarily the *most* informative batch.

For that, we need to look beyond pairwise margins.

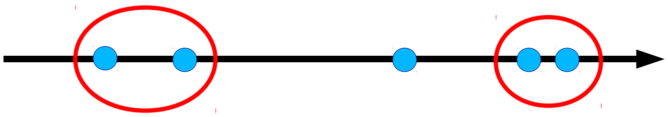


Figure 3.7: **Myopic Learner and Batch Selection.** For sample sizes $s > 2$, the myopic learner selects the $\frac{s}{2}$ data pairs with the lowest margins. Here, the $s = 4$, so the myopic learner selects two data pairs with the lowest margins.

1. Compute $|F(x_i - x_j)|$ for every pair of feature vectors in the unlabeled set U.
2. Sort the pairs in ascending order based on the value of $|F(x_i - x_j)|$ (e.g., $\{(x_i, x_j), \ldots\}$).
3. Take the top s unique feature vectors, or top $\frac{s}{2}$ pairs, from the pair list to build sample set S.

Figure 3.8: **Steps for Sample Selection - Myopic Learner**

3.3.2 Far-sighted Active Learner - A Listwise Low-Margin Approach

Unlike the myopic learner, the far-sighted active learner seeks samples with the mimimum *cumulative* margin distance among *all* data members, not just margin distance between independent pairs. In other words, after we project feature vectors from the unlabeled training set U onto weight vector, w, we aim to choose the closest s data members. For sample set S of size s, the objective is to minimize $|F(x_i - x_j)|$ for *all* $(x_i, x_j) \in S$, as follows.

$$S^* = \operatorname*{argmin}_{S \subseteq U} \sum_{\forall (x_i, x_j) \in S} |F(x_i - x_j)|, \tag{3.10}$$

for all possible S of size s that can be built from the unlabelled dataset U (see Figure 3.9).

To accomplish this using brute force, we would need to evaluate

$$M(S) = \sum_{\forall (x_i, x_j) \in S} |F(x_i - x_j)| \tag{3.11}$$

for every possible S, which is highly costly for large sizes of U. Yu [38] proposes a more efficient algorithm that involves sorting the feature vectors of U by the learned ranking function, then evaluating only the cumulative margin of each contiguous set of s sorted members in succession. Each time we shift one member up, we remove

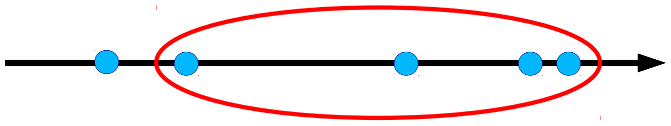


Figure 3.9: **Far-sighted Learner and Batch Selection.** For sample size $s = 4$, the far-sighted learner aims to select the set of samples with the lowest cumulative margin across *all* data members (compare to Figure 3.7, which only looks at margin distances between data pairs).

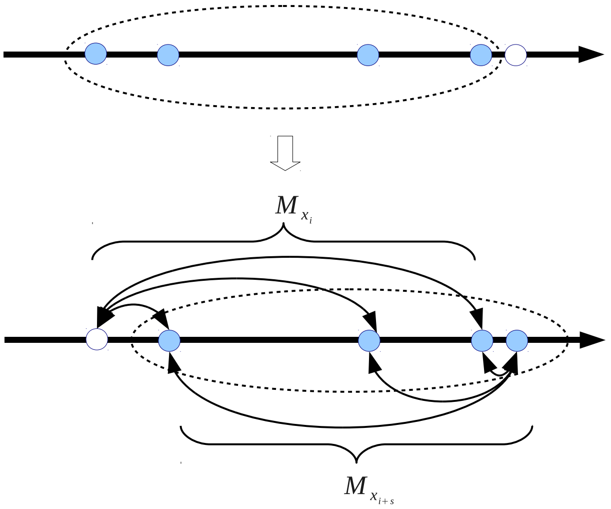


Figure 3.10: **Updating to the Next Sample Set.** This graphical depiction of step 6b of Figure 3.11 illustrates the process of updating the current sample set. We remove the oldest data member from the previous sample set and add in the next data member over from the sorted list to get the new sample set. Updating the cumulative margin involves subtracting the sum of the distances between the removed data member and the rest of the old sample members (M_{x_i}) and adding the sum of the distances between the newly added data member and the remaining sample members ($M_{x_{i+s}}$).

the oldest data member from the sample set and add in the next data member over from the sorted list. We also update the cumulative margin by subtracting from it the sum of the distances between the removed data member and the rest of the sample members (excluding the newly added data member), and we add to it the sum of the distances between the newly added data member and the remaining members of the sample set (see Figure 3.10). Figure 3.11 provides the steps in more detail. This method requires only ($O(|U|)$) evaluations, reducing the cost to linear time.

This learner is called the far-sighted learner, because it considers cumulative distance among all data members when selecting for image samples. Because the selected images have the lowest cumulative margins, they will provide the most information for learning a ranking function once annotated.

1. Build a set of rankings R by applying the previously learned ranking function F to the unlabelled dataset U.
2. Order the data members of U according to the corresponding rank values of R and write the newly ordered data to U_R.
3. Put the first s data members from U_R into temporary sample set S_T.
4. Compute $M(S_T)$ (defined in Equation 3.11) and assign it to M_1.
5. Set a M_{min} := A large number and $i := 1$.
6. Repeat the following until $i > |U_R| - s$
   (a) If $M_{min} > M_i$, then $M_{min} := M_i, I := i$
   (b) $M_{i+1} := M_i - M_{x_i} + M_{x_{(i+s)}}$ where
   $M_{x_i} = \sum_{i<j<i+s} |F(x_i - x_j)|$
   $M_{x_{(i+s)}} = \sum_{i<j<i+s} |F(x_j - x_{i+s})|$
   (See Figure 3.10)
   (c) $i := i + 1$
7. Final sample set $S = \{x_I, x_{I+1}, \ldots, x_{I+s-1}\}$.

Figure 3.11: **Steps for Sample Selection - Far-sighted Learner**

However, because the far-sighted learner by nature chooses samples whose rankings it has least confidence and most uncertainty in, it may select samples composed of all visually similar (with respect to the attribute) images, making them more difficult for a human to rank. This not only makes the decision process more time-consuming, but also increases the likelihood of annotators giving questionable or flat-out "wrong" rankings. In fact, one caveat of margin-based sample selection is that the image sample will only provide the most "good" information assuming the annotation is *correct*, but "correctness" can be highly subjective among similar-looking image samples whose rankings are ambiguous. To account for these problems, we introduce the constraint of visual diversity in the following section.

3.3.3 Diversity-Based Far-sighted Learner - A Visually-Sensitive Extension

The myopic and far-sighted learners both select image samples based solely on the low-margin condition. But what if by doing so, they select images that look

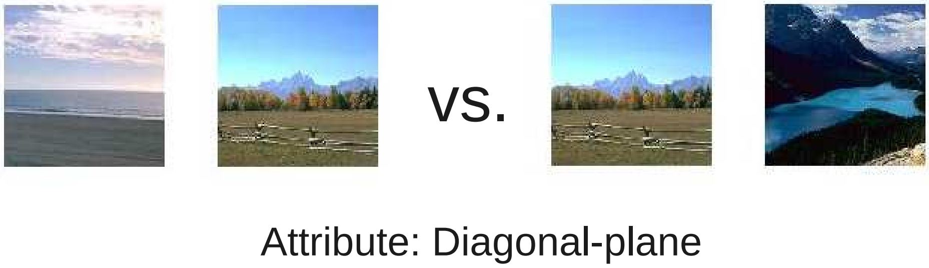


Figure 3.12: **Visually Similar vs Diverse Images** The image pair on the left is more visually similar than the image pair on the right. Over the relative attribute, *diagonal-plane*, the correct ranking for the rightmost pair is much more obvious and might even provide more information than the leftmost pair.

extremely similar such that their rankings are highly subjective and ambiguous? Should an annotator provide “wrong” or “questionable” rankings, the result would be counterproductive and perhaps even destructive for the learners (see Figure 3.12).

On the other hand, if the annotator cannot simply decide between any of the samples and in surrender ranks them all as equal, this may not harm the learning system, but it certainly would not provide much if any new information for the system.

Under the assumption that visually diverse images are more distinguishable, therefore easier and faster to rank, and more likely to take on different rank values with respect to each other, we propose a novel form of active learning to rank. This is a diversity-based extension of the far-sighted learner, named the “far-sighted-D” learner, that incorporates a “visual diversity” condition.

First, we need a evaluation metric for “visual diversity” as applied to images. Since we will be working with images’ feature vectors, we will consider two images to be “more diverse” if the Euclidean distance between their feature vectors is larger and “less diverse” if the Euclidean distance between their feature vectors is smaller.

Our diversity-based far-sighted-D learner will first divide the unlabeled set of training images U into k clusters using k-means clustering such that $k \geq s$ (recall that s is the size of the sample set of images selected for annotation) based on proximity of the feature vectors in Euclidean space. Then, it employs the same approach as the far-sighted learner described in Section 3.3.2 with one change: The far-sighted-D learner only selects images that belong to different clusters (see Figure 3.13). The goal is to select image samples that minimize the cumulative margin while simultaneously satisfying the visual-diversity constraint. The k in this case determines the tradeoff between lower margin and more visual diversity.

In order to accomplish this, we would once again face the problem of needing to evaluate every possible s combination of the data members in U in order to

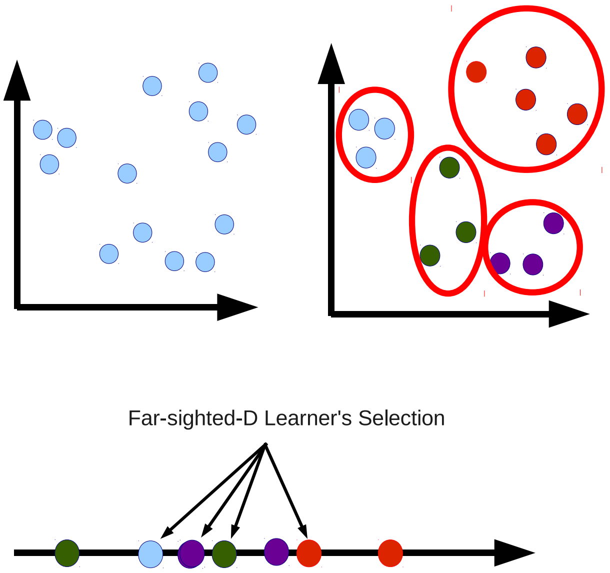


Figure 3.13: **Far-sighted-D Learner and Batch Selection.** For sample size $s = 4$ and cluster number $k = 4$, the far-sighted-D learner first uses k-means clustering to divide the data into k clusters. It then projects all data onto the weight vector, w, and from those, selects the closest s data members (i.e., ones with the lowest cumulative margin) such that each one belongs to a different cluster, enforcing diversity.

check that they satisfy the visual-diversity constraint, which would be far too expensive. Instead, we propose an alternative method that approximates the goal (i.e., selecting image samples that minimize the cumulative margin while satisfying the visual-diversity constraint). This approximation will satisfy the visual-diversity constraint (i.e., selecting samples whose members belong to different clusters), but it does not guarantee minimum cumulative margin given the constraint. In other words, there could exist other sample sets whose cumulative margins are still smaller even though they, too, satisfy the visual-diversity constraint. However, this approximation works in a much more manageable $O(|U|^2)$ time, and we find it to still be effective performance-wise, as later results will show (see Chapter 4).

The first few steps work exactly as steps 1-5 from Figure 3.11, so the setup is the same as that for the far-sighted learner: We first build a sorted list of data members by applying the previously learned ranking function to the unlabeled set, U. We then compute the cumulative margin of the first s sorted data members. From here, we iterate through each contiguous set of s sorted members in succession, checking if the current sample set members satisfy two constraints:

1. The current sample set yields the lowest cumulative margin seen so far.

2. The members of the current sample set satisfy the [**diversity condition**].

The “diversity condition” is that all members of the current sample set belong to different clusters, as discussed above. However, other metrics for evaluating diversity outside the scope of this paper could also be applied here.

- If the current sample set does not satisfy constraint 1, we simply examine the next sample set by removing the oldest data member and adding in the next data member over from the sorted list, just like we did before for the far-sighted learner as shown in Figure 3.10.
- If the current sample set satisfies both constraint 1 and 2, we record the new lowest cumulative margin seen so far and the samples belonging in the set that yielded this value. We then examine the next sample set over, again as shown in Figure 3.10.
- If the current sample set satisfies constraint 1 but not constraint 2, we start a subprocess. Prior to this, we make sure to store the current sample set for retrieval later. The subprocess:
  - We employ an “offender selection criterion” to identify one of the members in the sample set that is causing us to fail constraint 2. The offender selection criterion as follows: “Examine any members that belong to the same cluster within the sample set and of those, pick the one whose cumulative margin distance to the remaining sample members is the largest.” We throw out the chosen “offender” from the set and add in the next data member over from the sorted list to get a new sample set.
  - We continue checking for constraints 1 and 2 as before and updating the sample set like in the last step above until we either finally pass both constraints 1 and 2 or start to fail constraint 1. If we pass constraints 1 and 2, we record the new lowest cumulative margin and the samples belonging to the current set that yielded the value and terminate the subprocess. If we fail constraint 1, we immediately terminate the subprocess.

  When we are done with this subprocess, we retrieve the previously stored sample set, and examine the next sample set over, as shown in Figure 3.10.

Figure 3.14 presents a more detailed description of the steps.

One obvious concern for this learning approach is that it may still be computationally expensive, evaluating training candidates in $O(|U|^2)$ time. In a trial study, we ran this approach 20 times using $s = 4$ and $k = 10$ over 2688 images of scenes represented by 557-dimensional feature vectures. We found the average running time across all 20 runs to be 0.74 seconds (compared to 0.39 seconds by the far-sighted learner). Although this value may become more problematic to performance for much larger data sets, in practice, it may still be trivial compared to the time it takes for humans to annotate the images and effort saved by using a diversity-based learning method that selects for visually distinct images. Section 4.7.3 discusses more on performance and human effort saved.

```
1. Build a set of rankings $R$ by applying the previously learned ranking function $F$ to the unlabelled dataset $U$.
2. Order the data members of $U$ according to the corresponding rank values of $R$ and write the newly ordered data to $U_R$.
3. Put the first $s$ data members from $U_R$ into temporary sample set $S_T$.
4. Compute $M(S_T)$ (defined in Equation 3.11) and assign it to $M_1$.
5. Set a $M_{min} :=$ A large number and $i := 1$.
6. Repeat the following until $i > |U_R| - s$
   (a) Set vector $i_{vec} = \{i, i+1, \ldots, i+s-1\}$ and let $i_{vec}(j)$ denote "the $j^{th}$ member of $i_{vec}$"
   (b) $M_{prev} = M_i$
   (c) If $M_{min} \leq M_{prev}$, then skip to step (e)
   (d) Otherwise:
       i. If [diversity condition] is satisfied, then $M_{min} := M_{prev}, I_{vec} := i_{vec}$, then skip to step (e)
       ii. Otherwise:
           A. Identify one of the "offenders" from $i_{vec}$ using [offender selection criterion] and assign it to $o$
           B. Remove $o$ from $i_{vec}$
           C. n = $max(i_{vec})$+1 (i.e., index of the the next member to be added)
           D. $M_{prev} := M_{prev} - M_{x_o} + M_{x_n}$ where
              $M_{x_o} = \sum_{1<j<s-1} |F(x_o - x_{i_{vec}(j)})|$
              $M_{x_n} = \sum_{1<j<s-1} |F(x_n - x_{i_{vec}(j)})|$
           E. Add $n$ to $i_{vec}$
           F. Repeat from step (c)
   (e) $M_{i+1} := M_i - M_{x_i} + M_{x_{(i+s)}}$ where
       $M_{x_i} = \sum_{i<j<i+s} |F(x_i - x_j)|$
       $M_{x_{(i+s)}} = \sum_{i<j<i+s} |F(x_j - x_{i+s})|$
   (f) $i := i + 1$
7. Final sample set $S = \{x_{I_{vec}(1)}, x_{I_{vec}(2)}, \ldots, x_{I_{vec}(I+s-1)}\}$.
```

Figure 3.14: **Steps for Sample Selection - Far-sighted-D Learner**

Chapter 4

Experiments

The following experiments pit all three active sample selection methods against each other along with three baseline approaches (see Section 4.2) and evaluate their performance across three different data sets, the OSR [25], Pubfig [20], and Shoes [2] data sets, over 27 visual attributes total (see Section 4.1).

Section 4.3 explains the general setup of the experiments. In Section 4.5, we perform a sanity check by training the learners using artificially generated rank data. In Section 4.6, we train ranking functions using the different learners by annotating image samples with rank labels extrapolated from previously human-annotated, pairwise-ranked images. Finally, Section 4.7 puts the learners to practice by running the active learning loop in real time and collecting human annotation for each successive batch of learner-selected image samples live.

4.1 Description of Data Sets

For our experiments, we will be using three data sets.

The OSR data set contains 2,688 images of eight types of scenery (coast, forest, highway, inside city, mountain, open country, street, and tall building), represented by 512 low-level gist features and 45 global color features. We choose to use the following six visual attributes for ranking the scenic images: natural, open, perspective, large objects, diagonal plane, close depth (See Figure 4.1).

The Pubfig data set contains 772 images of eight celebrities' faces (Alexander Rodriquez, Clive Owen, Hugh Laurie, Jared Leto, Miley Cyrus, Scarlett Johansson, Viggo Mortensen, and Zac Efron), represented by 512 low-level gist features and 30 global color features. We choose to use the following 11 visual attributes for ranking the facial images: male, white, young, smiling, chubby, visible forehead, bushy eyebrows, narrow eyes, pointy nose, big lips, and round face (See Figure 4.2).

The Shoes data set contains 14,658 images of ten kinds of shoes (athletic shoes, boots, clogs, flats, high heels, pumps, rain boots, sneakers, stiletto, and wedding shoes), represented by 960 low-level gist features and 30 global color features. We choose to use the following ten visual attributes for ranking the shoe images: pointy at the front, open, bright in color, covered with ornaments, shiny, high at the heel, long on the leg, formal, sporty, and feminine (See Figure 4.3).

Each image is filtered and processed through several visual channels (texture, shape, intensity, etc) specific to each data set (OSR, Pubfig, Shoes) to produce a vector of view and scale-invariant features used to represent the image. Each data set has its own unique set of features that have been previously shown to be capable of summarizing multiple characteristics of an image [25, 32, 20, 2]. We also choose the image categories and visual attributes from each data set such that the attributes are distinct from one another, simple to identify, and show adequate variation across the categories. Because the three data sets are substantially diverse (i.e., Features making up a face are much different than those that make up scenery), the following experiments do well to comprehensively summarize active learning performance for image ranking throughout various visual contexts.

4.2 Baseline Approaches

This section introduces three baseline approaches: the passive learner, passive-D learner, and handicapped learner. These serve as points of comparison for our active learners. Like the active learners, the following methods differ in the way they select image samples at each iteration in the learning loop.

The *passive learner* does not process anything from the unlabeled training set and simply picks image samples at random. Because it performs sample selection at random, we expect this learner to perform worse than the far-sighted-D, far-sighted, and myopic learners.

The *passive-D learner* is based on “visual diversity” in the same way as the far-sighted-D learner in that it partitions the unlabeled training data into k clusters using k-means clustering and only selects s samples such that each sample member belongs to a unique cluster. The passive-D learner first uses k-means clustering to partition the data, where $k \geq s$, then randomly selects one sample from each cluster. Of the remaining k samples, it randomly selects s to be labeled. Since the passive-D learner complies to the visual-diversity condition but ignores the low-margin condition, it allows us to see the performance effect of utilizing the diversity condition alone. It can also help determine whether any performance increase in using the far-sighted-D learner is due to a combination of the visual-diversity and low-margin conditions, or soley the former. We expect the passive-D learner to perform better than the passive learner but worse than the far-sighted-D learner.

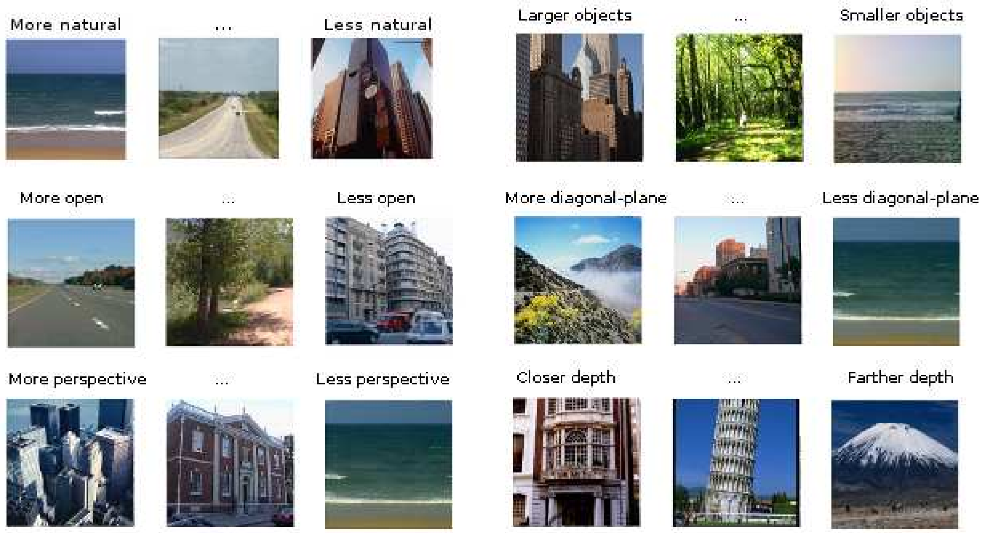


Figure 4.1: **OSR Attributes.** This figure provides visual examples of the different types of attributes from the OSR data set [25]: Natural, Open, Perspective, Large-Objects, Diagonal-Plane, and Close-Depth.

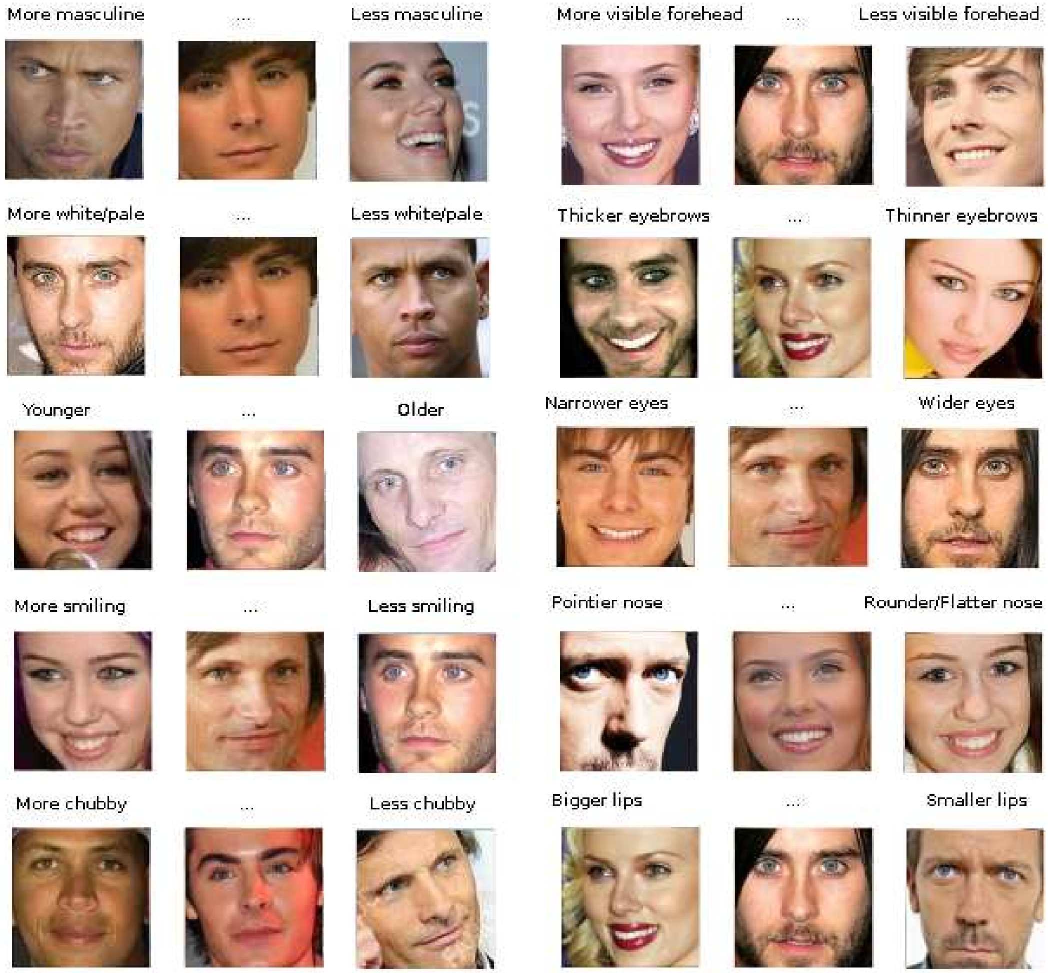


Figure 4.2: **Pubfig Attributes.** This figure provides visual examples of the different types of attributes from the Pubfig data set [20]: Male, White, Young, Smiling, Chubby, Visible Forehead, Bushy Eyebrows, Narrow Eyes, Pointy Nose, Big Lips.

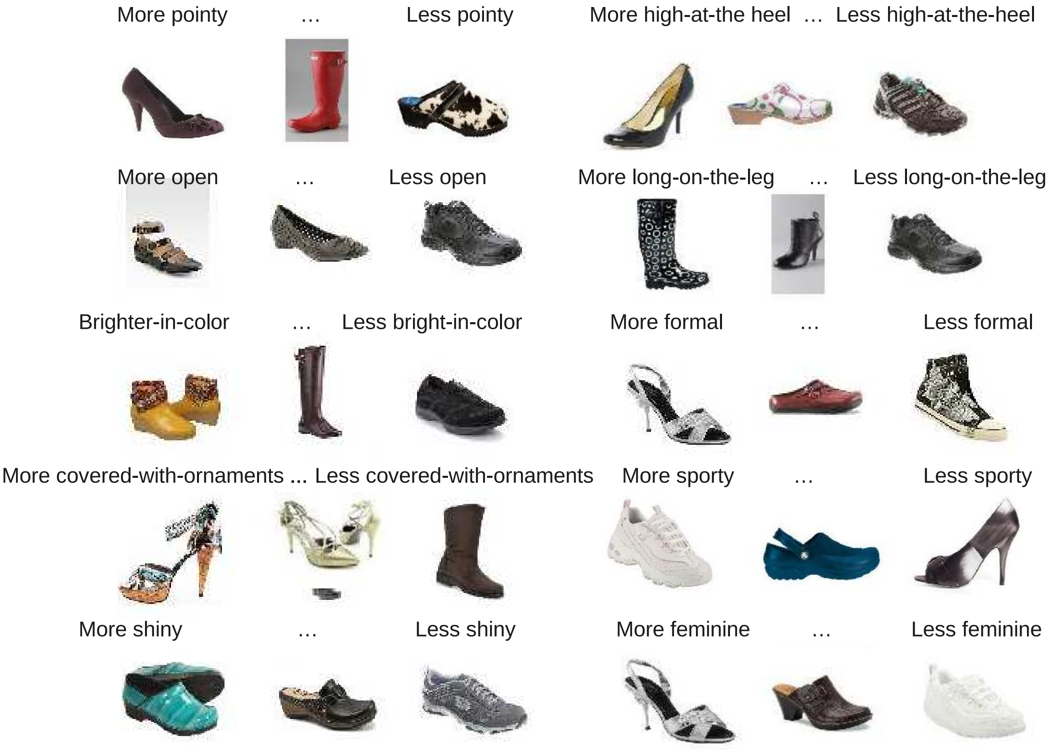


Figure 4.3: **Shoes Attributes.** This figure provides visual examples of the different types of attributes from the Shoes data set [20]: Pointy at the Front, Open, Bright in Color, Covered with Ornaments, Shiny, High at the Heel, Long on the Leg, Formal, Sporty, and Feminine.

The *handicapped learner* employs the opposite sample selection technique from that of the myopic learner. Instead of selecting the closest data pairs, or those with the smallest margins, the handicapped learner selects the farthest data pairs, or those with the largest margins. This means that the handicapped learner chooses image pairs whose rank values it already has much confidence in, so labeling these poorly-chosen samples would provide very little extra information for refining a ranking function. We expect the handicapped learner, as the "worst-choice" baseline approach, to perform even worse the passive learner.

4.3 General Experimental Setup

To run experiments over the different learners, we will divide the data sets into two groups, the unlabeled training set and the labeled test set. The training set provides the pool of unlabeled data from which learners will select image samples to be annotated. We will use the test set to evaluate the accuracy of the learned ranking functions by measuring the ordinal similarity between the function-generated and ground-truth rank values of the test set images with Kendall's tau (See Section 4.4), a rank-based correlation coefficient.

Experimentation over all six learners (three active and three baseline approaches) requires selecting images of sample size, $s > 2$ in order to differentiate the performance between the far-sighted and myopic learners. By nature of the myopic learner's pair-based approach, it is also necessary to have an even sample size. Therefore, we use a sample size of s=4 in the following experiments.

For the diversity-based learners (far-sighted-D and passive-D), we use $k = 10$ number of clusters in our experiments. Since the three data sets we will be working with contain images from eight to ten different categories, we feel that choosing ten clusters should sufficiently enforce diversity among the images of these data sets.

4.4 Accuracy Metric: Kendall's tau (τ)

To evaluate the accuracy of learned ranking functions in our experiments, we will use Kendall's tau (τ), a rank-based correlation coefficient that measures the degree of association between two sets of ordinal data, to measure the similarity between the function-calculated rankings and ground-truth rankings of our test data.

Below is an explanation of the metric's logistics.

Let $\{x_1, x_2, ..., x_n\}$ and $\{y_1, y_2, ..., y_n\}$ be rank values that comprise the set X of function-calculated rank values and set Y of ground-truth rank values, respectively. A rank value pair (x_i, y_i), (x_j, y_j) is concordant if $x_i > x_j$ and $y_i > y_j$ or if $x_i < x_j$ and $y_i < y_j$. A rank value pair (x_i, y_i), (x_j, y_j) is discordant if $x_i > x_j$ and $y_i < y_j$

or if $x_i > x_j$ and $y_i < y_j$. A rank value pair (x_i, y_i), (x_j, y_j) is neither concordant nor discordant if $x_i = x_j$ or $y_i = y_j$. In this way, we can accomodate partial orders.

Kendall's tau represents the number of concordant pairs less the number of discordant pairs divided by the total number of rank value pairs:

$$\tau = \frac{(\#\text{ concordant pairs}) - (\#\text{ discordant pairs})}{\frac{1}{2}n(n-1)} \tag{4.1}$$

In summary, $0 < \tau < 1$ indicates that there is much agreement over the rankings between the two sets, with $+1$ being a perfectly positive correlation, and $-1 < \tau < 0$ indicates that there is much disagreement over the rankings between the two sets, with -1 being a perfectly negative correlation. $\tau = 0$ indicates no correlation between the two sets of rankings. Using this metric, we can evaluate the accuracy of the learned ranking function.

4.5 Simulation Using Artificial Data

In this section, we test the learners against artificially generated rank data. During training, human uncertainty may arise over the labelling of ambiguous image samples. By using artificial data, this provides a sanity check and shows how learning to rank fares in a "blank-slate" situation where an unconditional, unambiguous "ground-truth" ranking exists over all data.

Section4.5.1 explains the experimental setup for this artificial simulation, and Section 4.5.2 details the results.

4.5.1 Experimental Setup

To achieve artificial ranking, we first produce an arbitrary ranking function, which is some randomly-generated weight vector, w. We then assign synthetic ground-truth rank values, derived from the arbitrary ranking function, to 700 randomly-generated 10-dimensional feature vectors. We set aside 670 of the features vectors, chosen at random, for the training set and remaining 30 for the test set. From the 670 of the training set, we randomly choose four samples as seeds to train an initial ranking function. The type of learner used determines subsequent samples. We run all six learners for 25 iterations, assigning the selected sample at each iteration the ground-truth rank value derived from the previously generated w. For the diversity-based learners, far-sighted-D and passive-D, we partition data from the training set into $k = 10$ clusters. At each iteration in the learning process, we compute the accuracy of the currently learned ranking function using Kendall's τ. We repeat the entire process 20 times, each time generating a new w and new set of feature vectors, and average the results across all 20 runs.

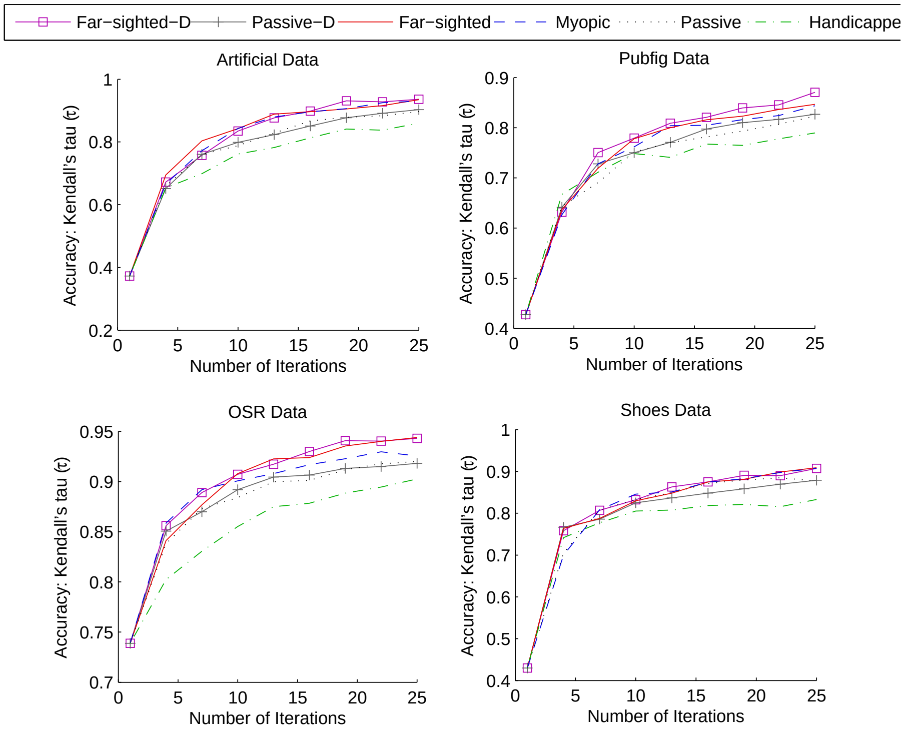


Figure 4.4: **Performance Results (Using Artificial Data)** These plots show the performance results of our experiments using artifically imposed rankings over synthetic data and images from the OSR, Pubfig, and Shoes data sets. We see that the active selection approaches generally outperform the passive ones, and the "handicapped" baseline is the weakest of all.

We conduct the same experiment three more times using actual image data from the OSR, Pubfig, and Shoes data sets. We employ identical constaints, except we use feature vectors of 700 randomly-selected images from the provided data sets for each of the 20 runs instead of randomly-generated feature vectors. The ground-truth rank values are still artificially imposed, but because the training and test data are of actual images, it provides a point of comparison for learning to rank images over learning to rank synthetic data.

4.5.2 Simulation Results

Figure 4.4 shows the results of the simulation experiments.

We see that there is definite performance differences between the active learners (far-sighted-D, far-sighted, and myopic), the passive-D and passive learners, and the handicapped learner. In all cases, the active learners outperform the passive-D and passive learners, and the handicapped learner perform the worst, which is the expected trend. The far-sighted learner outperforms the myopic learner in three of the experiments (artificial data, OSR, and Pubfig), and the far-sighted-D learner performs the best for the Pubfig data set and as well as, but no worse than, the other active learners for the rest of the experiments. These results suggest that active learning methods are definitely more advantageous over random sample selection, and that taking a both low-margin and diversity-based approach (i.e., the far-sighted-D learner) may be optimal for performance in certain cases, although it does not impair it in any way, keeping in mind that the rankings used in the experiments of this section are artificially imposed.

In order to generalize the performance capabilities of these learners to real-world applications, the next sections apply this same experimental setup to actual images.

4.6 Learning With Pairwise-Extrapolated Rank Labels

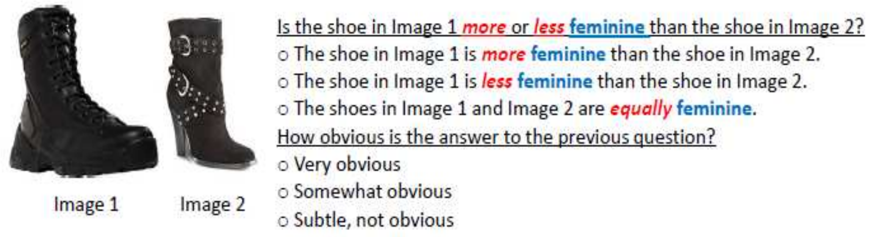
Artificially-generated rankings give us an idea of what to expect, so now we move on to working with attribute-specific rankings as applied to images. To our knowledge, visual attribute-specific rank data are not publicly available, so we must build our own global rank hierarchy over images from our data sets by collecting information from human annotators and extrapolating the rank labels. Section 4.6.1 details how this is done. Section 4.6.2 explains the experimental setup for each data set. Section 4.6.3 describes results from the OSR data set, Section 4.6.4 describes results from the Pubfig data set, and Section 4.6.5 describes results from the Shoes data set.

4.6.1 Data Collection and Rank Extrapolation Method

Using Amazon's Mechanical Turk (MTurk)[1], Kovashka, Parikh, and Grauman asked human participants to rank pairs of images from each data set [19], using questions in the form of "Is the [object] in Image 1 more or less [some attribute] than the [object] in Image 2?" Participants also specified a confidence level in the labels they gave, choosing among "very obvious," "somewhat obvious," and "subtle, not obvious." Figure 4.5 provides an example of the UI used.

From the OSR data set, they randomly selected 241 images and collected a total of 5,620 pairwise rankings over those images. For each attribute, they used 202-217 of the 241 images to collect 920-940 of the 5,620 pairwise rankings. From the Pubfig

[1] Amazon's Mechanical Turk (MTurk) is a crowdsourcing site that allows people to post Human Intelligence Tasks (HITs) for participants to complete for some amount of pay (https://www.mturk.com/).



Is the shoe in Image 1 *more* or *less* feminine than the shoe in Image 2?

○ The shoe in Image 1 is *more* **feminine** than the shoe in Image 2.

○ The shoe in Image 1 is *less* **feminine** than the shoe in Image 2.

○ The shoes in Image 1 and Image 2 are *equally* **feminine**.

How obvious is the answer to the previous question?

○ Very obvious

○ Somewhat obvious

○ Subtle, not obvious

Figure 4.5: **MTurk UI for Ranking Image Pairs.** This figure shows the MTurk UI used to collect pairwise rank data. The UI asks annotators questions in the form of "Is the [object] in Image 1 more or less [some attribute] than the [object] in Image 2?" Participants also specified a confidence level in the labels they gave, choosing among "very obvious," "somewhat obvious," and "subtle, not obvious."

data set, they randomly selected 242 images and collected a total of 10,860 pairwise rankings over those images. For each attribute, they used 200-211 of the 242 images to collect 947-1009 of the 10,860 pairwise rankings. From the Shoes data set, they randomly selected 240 images and collected a total of 7,100 pairwise rankings over those images. For each attribute, they used 161-172 of the 240 images to collect 700-725 of the 7,100 pairwise rankings. For more robust and reliable results, they queried multiple annotators on the same pair of images. We use this data here to impose a global, ground truth ranking over every image used.

Since the same pair of images received independent labels from multiple annotators, we must aggregate these labels. In order to do this, we first assign numerical rank scores to each pair of images. Within a pair, if Image 1 was ranked higher than Image 2, we assign the former a score of 2 and the latter a score of 1 and vice versa. If Image 1 was ranked the same as Image 2, both receive scores of 1.5. Then for each unique image pair, which may have multiple different rank scores, we compute a weighted average. Using each individual annotator's confidence level to determine weights, we give labels designated as "very obvious" twice the weight of labels designated as "somewhat obvious" and labels designated as "somewhat obvious" three times as much weight as labels designated as "subtle, not obvious." If all annotators designate an image pair as "subtle, not obvious," then we cast away that image pair as unreliable data. Remaining on the conservative side, we only consider averaged rank scores that differ by more than .3 to be "different." In other words, we are only confident that two images should be ranked differently if there is sufficient distance between their averaged rank scores (in this case, .3 units). For instance, within a unique image pair, if the averaged rank score is 1.2 for Image 1 and 1.4 for Image 2, we would consider Image 1 and Image 2 to be ranked equally. However, if the averaged rank score of is 1.2 for Image 1 but 1.6 for Image 2, then we would consider Image 2 to be ranked higher than Image 1. We chose this distance value of

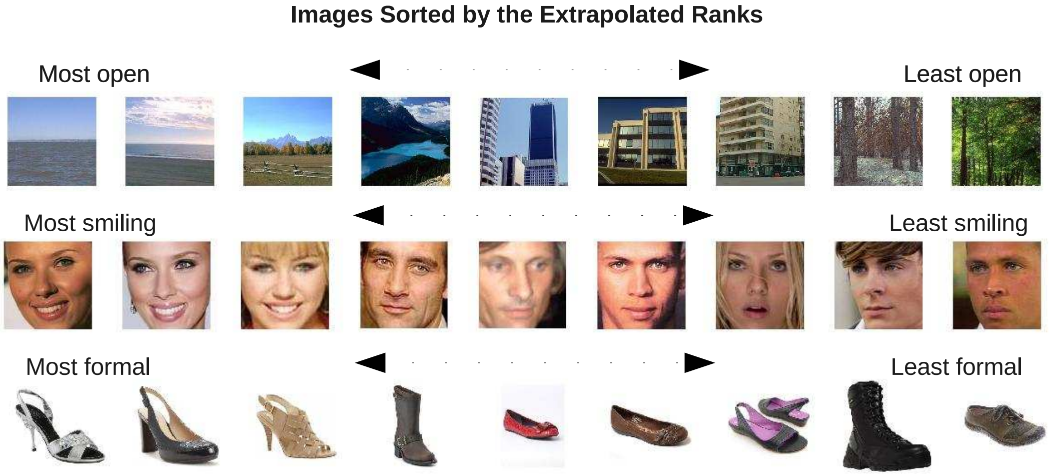


Figure 4.6: **OSR, Pubfig, and Shoes Images Sorted by Extrapolated Ranks.** This figure shows images from the OSR, Pubfig, and Shoes data sets sorted by the ground-truth rankings extrapolated from the pairwise-ranked data over the attributes open, smiling, and formal, respectively. We see that even though the extrapolated rankings arrived from calculation, they are fairly accurate.

.3 units based on exploratory trial-and-error runs over some test annotations. This safeguards against fluctuations in averages caused by uncertainty among annotators by essentially clustering close rank scores together.

Using the newly aggregated pairwise rankings for each attribute, we train a general ranking function. We then impose a final "ground truth" hierarchy by applying the ranking function to the same images used to train that ranking function in the first place. Although these ground-truth rank values arrive from computation, they are largely based on human-provided information and appear to be visually appropriate when the images are displayed in sorted order, as depicted in Figure 4.6.

Therefore, we use these extrapolated rankings for annotating samples in this section's experiments.

4.6.2 Experimental Setup

Of the 200-217 images previously ranked for each attribute of the OSR and Pubfig data sets, we set aside 40, chosen at random, to serve the test set and randomly select 4 more as seeds to train an initial ranking function. From the remaining held-out set of 156-173 images, we randomly choose 120 to serve as the training set. We run all six learners for 25 iterations, assigning the selected sample at each iteration the ground-truth rank value derived in Section 4.6.1. Like before, for the diversity-based learners, far-sighted-D and passive-D, we partition data from the training set into $k = 10$ clusters. At each iteration in the learning process, we compute the accuracy of the currently learned ranking function using Kendall's tau. We conduct the entire process 20 times, each time choosing at random 120 images from the held-out set of 160-177 but keeping everything else constant, and average the results across all 20 runs.

Since we have less images to work with for the Shoes data set (161-172 per attribute), we set aside 30 images to serve as the test set (as opposed to 40) and four as seeds to train an initial ranking function. Of the remaining held-out set of 127-139 images, we randomly choose 100 to serve as the training set (as opposed to 120). Insead of running the learners for 25 iterations as with the OSR and Pubfig data sets, we run the learners for 22 iterations with the Shoes data set. Other than these parameter adjustments, the experimental setup for the Shoes data set remain identical to that of the OSR and Pubfig data sets.

4.6.3 Results - OSR

Figure 4.7 shows the results for the OSR data set.

First, we examine the baseline approaches. Consistent with our predictions, the handicapped learner performs the worst out of all the learners. The passive learner performs worse than both the low-margin-based active learners and the diversity-based learners, indicating that selecting samples with low margins and selecting samples with diverse feature vectors both improve performance relative to random sample selection. Since in addition, the far-sighted-D learner outperforms the passive-D learner, this indicates that a combination of the low-margin condition and the visual-diversity condition fares better than merely the diversity condition alone.

In comparing the active learners against each other, the performance differences are not as radical, but we we do see that the far-sighted learner generally outperforms the myopic learning across the OSR attributes, and the far-sighted-D learner generally outperforms all of the other learners. However, for some of the attributes, it may take up to ten iterations before a performance difference is seen (e.g. *close-depth*). Past the tenth iteration, nearly all attributes show the performance trend of the far-sighted-D learner outperforming the far-sighted learner, and the far-sighted learner outperforming the myopic learner.

Although performance in the long run is meaningful, we are especially interested in learners that perform well early in the annotation process, since our motivation is to minimize human effort as much as possible. We do not see much of this happening among the active learners for the OSR data set except for one attribute, *open*. Notice that the learning curves for this attribute initially start out at an already-high τ

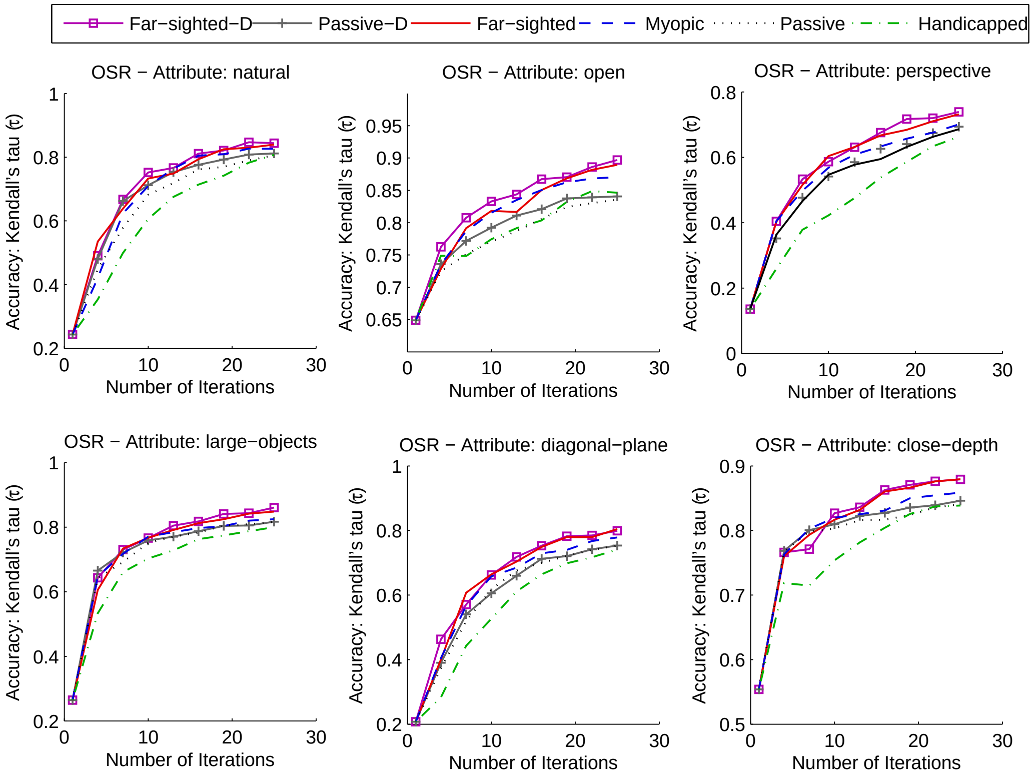


Figure 4.7: **Performance Results - OSR (Using Extrapolated Ranks).** Performance results after running the six learners over the six attributes of the OSR data set: natural, open, perspective, large-objects, diagonal-plane, and close-depth. From these results, we can see that in general, the far-sighted-D learner performs the best. The active learners generally outperform the passive learners, and the handicapped learner performs the worst, as expected.

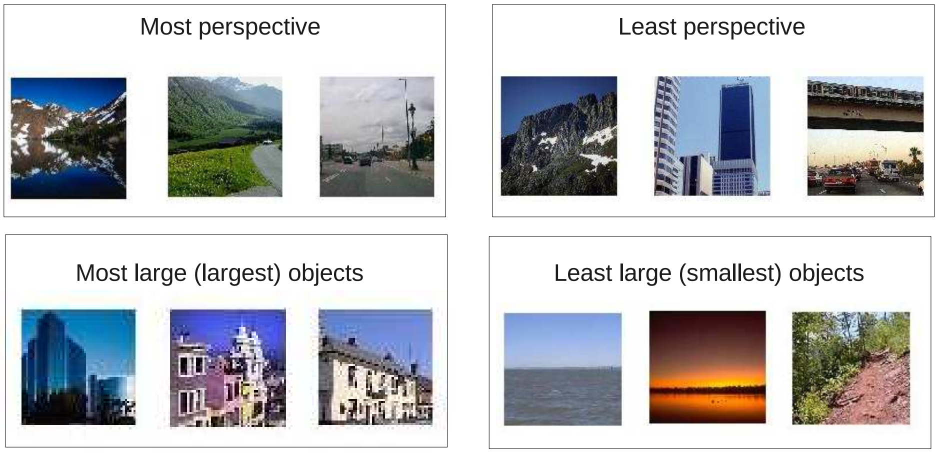


Figure 4.8: **Extremal Images of the Perspective and Large-Objects Attributes.** Here, we pick out three of the highest and lowest ranked images from the attributes, *perspective* and *large-objects*, based on the rankings extrapolated from the pairwise-ranked data in Section 4.6.1. Notice that the items making up the top and bottom ranks of the *perspective* attribute are much more varied than those of the *large-objects* attribute. Also notice that pictures of mountains and roads appear in both the top and bottom ranked items for the *perspective* attribute. This overlap does not show up for *large-objects*. Therefore, it's likely that it takes more of certain kinds of sample images to train a ranker for perspective scenes than for large-object scenes. This suggests why our active learners show more noticeable advantage against the passive learners on the challenging *perspective* attribute compared to some of the other attributes in the OSR data set.

value, suggesting that choosing a good seed is important, preferably one whose images show varying degrees of the attribute such that it will provide a fair amount of information to start the learning process on a good note.

Between the active learners and the baselines, however, we see this kind of "early performance differentiation" more commonly, such as with the attribute, *perspective*. Compared to attributes such as *large-objects* and *close-depth*, where there is not much early differentiation between the active learners and baselines, it is likely that the type of samples selected to train ranking functions for the *perspective* attribute matter more early on, because this attribute is more semantically complicated and therefore requires a certain variety of training samples to produce optimal results. On the other hand, what constitutes a larger object or how far away an object appears from the camera is more obvious and so benefits less from active learning(see Figure 4.8).

In summary, our OSR results suggest that the proposed far-sighted-D learner generally performs the best.

4.6.4 Results - Pubfig

Figure 4.7 shows the results for the Pubfig data set.

First, we analyze performance between the active learners and the baselines.

Like before, the handicapped learner performs the worst, indicating that margin space is important in sample selection. In most cases, the active learners outperform the passive and passive-D learners. Two major exceptions are the attributes, *white* and *round face*, where all learners (except for the handicapped learner on the attribute, *round face*) perform about equally well (although the far-sighted-D does slightly outperform the others for some of the iterations for the attribute, *white*).

A likely explanation is that in the case where an attribute is difficult to learn to rank because of ambiguity over how to annotate the samples, we expect to find both active and passive learners to perform about the same (See Figure 4.10). Low-margin based active learners are even more likely to select image samples whose rankings are ambiguous for human judges, since by definition they select image samples with the most uncertainty, which may be counterproductive to learning a ranking function. Diversity-based learners may not necessarily be immune either if the factors for ambiguity are tied to subjective uncertainty over the attribute's definition.

For instance, every image has different lighting that may effect how we perceive whiteness. Questions arise such as "Should we take lighting effects into account or should we simply ignore the lighting effects and rank the image based on what we *feel* the actual color of the face should be?" In the case of the *round-face* attribute, a large number of images are cropped at the sides of the face, making it difficult

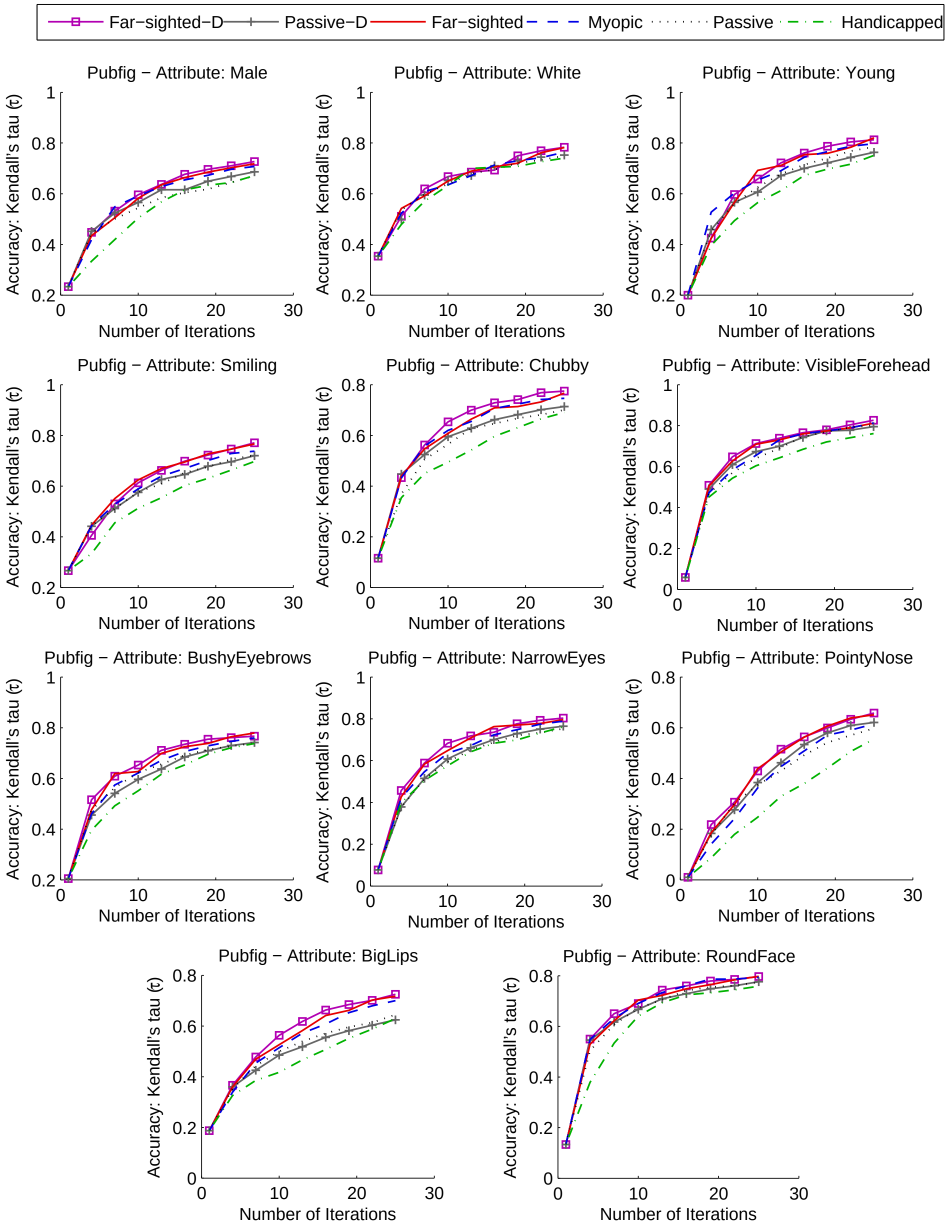


Figure 4.9: **Performance Results - Pubfig (Using Extrapolated Rank).** Performance results after running the six learners over the 11 attributes of the Pubfig data set: male, white, young, smiling, chubby, visible forehead, bushy eyebrows, narrow eyes, pointy nose, big lips, and round face. From these results, we can see that in general, the far-sighted-D learner performs the best except in cases where the attribute is highly localized (e.g., *smiling*, *pointy nose*). The active learners generally outperform the passive learners, and the handicapped learner performs the worst, as expected. An exceptional case is the attribute *white*, where all learners perform similarly, likely due to rank ambiguity.

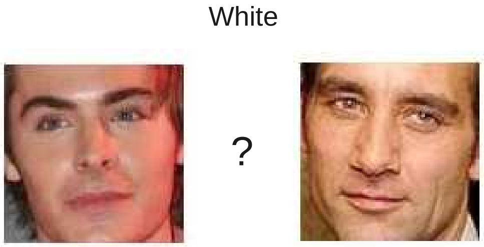


Figure 4.10: **Ambiguity in Ranking Two Faces Based on Whiteness** The faces of Zac Efron and Clive Owen in these pictures both have very different colors. Zac has a more pink undertone to his face and Clive has more gold undertone to his face. The question of "Which one is more white?" depends on whether you consider a pink undertone to be darker than a gold undertone or vice versa.

for human annotators to determine roundness. Indeed, there might not even exist a real, objective ground-truth ordering over some of the more exceptional cases, an issue that would be a good candidate for investigation in future works.

For six of the attributes (*smiling*, *visible forehead*, *bushy eyebrows*, *narrow eyes*, *pointy nose*, and *big lips*), the far-sighted learner outperforms the myopic learner. In the remaining cases, it generally performs as well as, but not considerably any worse than, the myopic learner, suggesting that a far-sighted approach by be better than a myopic one, but it certainly does not detract from performance in any way.

Finally, the far-sighted-D learner overall performs the best for all of the attributes except for four (*white*, *smiling*, *pointy nose*, and *round face*). However, even in these exceptional cases, it still performs as well and not considerably any worse than the other active learners. Also, in no case does the passive-D learner outperform the far-sighted-D learner, which suggests that it is a combination of both the low-margin and visual-diversity condition that gives the far-sighted-D learner the boost in performance over the rest.

We now examine the cases where the far-sighted-D learner does not outperform the far-sighted and myopic learners. The reason for the attributes *white* and *round face* is likely due to the large amount of ambiguity involved in ranking (also explained above), equalizing the playing field for all sample selection methods. As for the attributes *pointy nose* and *smiling*, we note that the far-sighted learner outperforms the myopic learner in both cases but the far-sighted-D learner only performs as well as the far-sighted learner and no better, indicating that selecting image samples with the lowest margins is still important, but selecting images with diverse feature vectors is not as important. One explanation is that the features involved in characterizing smiles and noses are too localized, such that it matters less that a learner selects faces that look different and more that a learner selects *mouths* or *noses* that look different. Because our image descriptors are global, these local

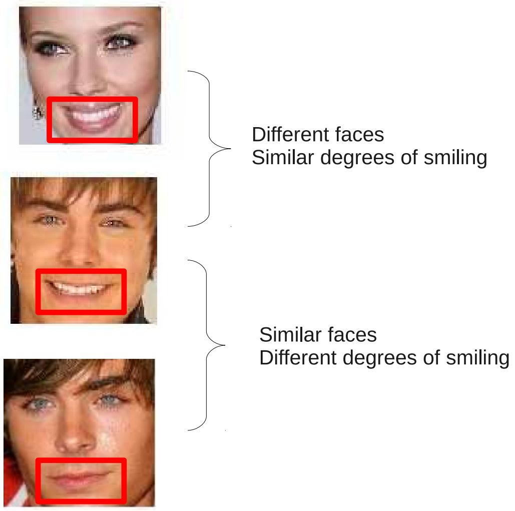


Figure 4.11: **Degree of Smiling vs Image Similarity** Although the bottom two images of Zac Efron are more similar, they have different degrees of smiling. While the top two images of Zac Efron and Scarlett Johansson are different, they have similar degrees of smiling. It is far better to focus only on the features defining the mouth (red box) than the all the features as a whole in this case.

parts cannot be completely isolated such that the local differences have sufficient influence (See Figure 4.11).

Note, however, that attributes such as *big lips* and *bushy eyebrows* show very good results with clear performance differentiation between the learners despite also being attributes largely characterized only a small part of the face. This can be explained by the observation that faces (from our Pubfig data set) that have bigger lips or bushier eyebrows tend to look more similar with respects to other facial features as well. For instance, pictures of people with bigger lips from our data set tend to have other, more feminine facial features as well. Despite the attribute itself applying to only a small portion of the face, it may not be completely independent of some of the other features of the face. Smiles, on the other hand, are almost entirely dependent on features defining the mouth. In the case of the pointy nose attribute, the degree of pointiness a nose *appears* to have can change even among images of the same person depending on the head rotation and lighting.

Future work may involve incorporating discovery of relevant features into active learning.

4.6.5 Results - Shoes

Figure 4.12 shows the results for the Shoes data set.

First, we analyze performance between the active learners and the baselines.

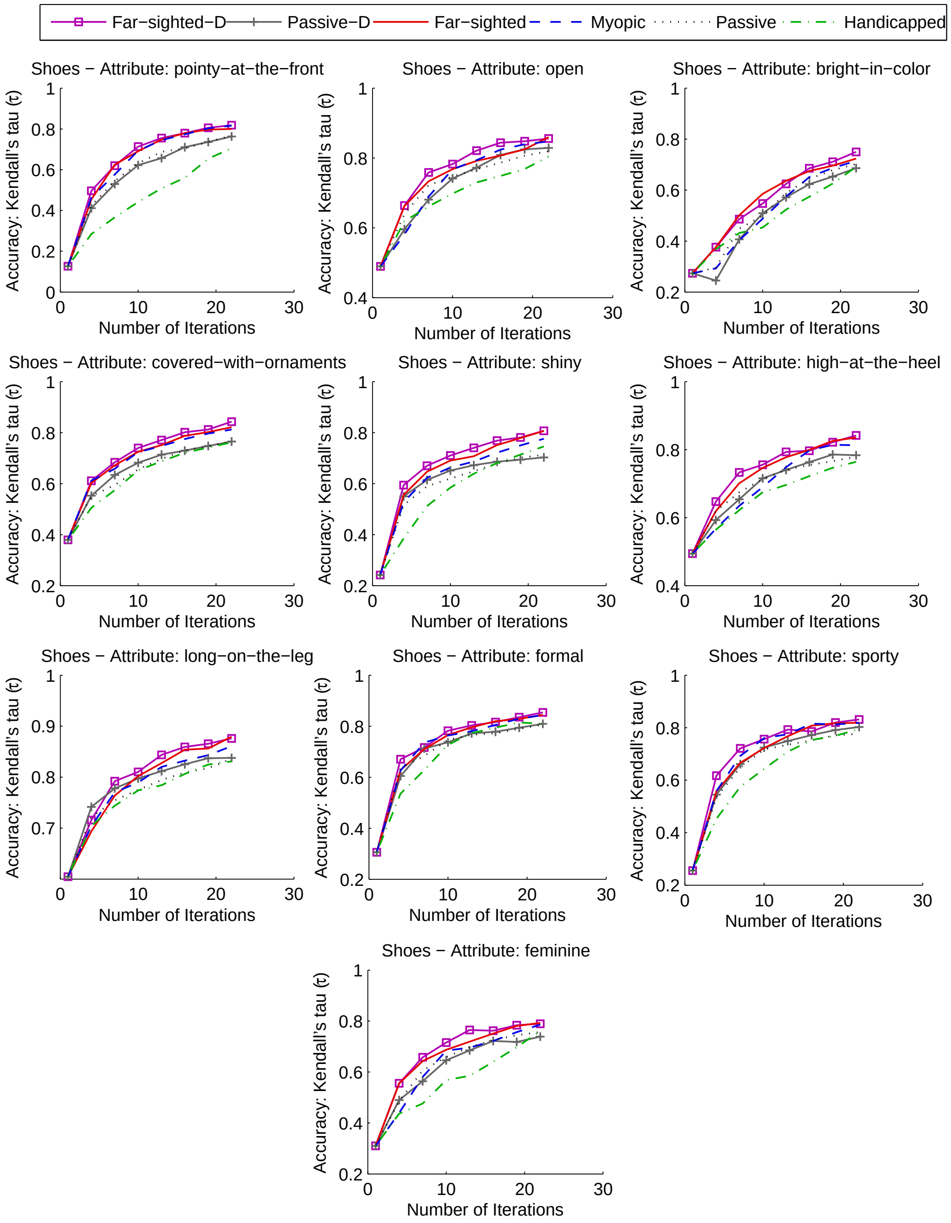


Figure 4.12: **Performance Results - Shoes (Using Extrapolated Ranks).** Performance results after running the six learners over the ten attributes of the Pubfig data set: pointy at the front, open, bright in color, covered with ornaments, shiny, high at the heel, long on the leg, formal, sporty, and feminine. From these results, we can see that in general, the far-sighted-D learner performs the best. The active learners generally outperform the passive learners, and the handicapped learner performs the worst, as expected. One exceptional case is the attribute *bright-in-color*, where all learners perform similarly, likely due to ambiguity in the attribute's definition.

Like above, the passive and handicapped learners generally perform worse than the active learners, with the handicapped learner performing the worst overall. The passive-D learner shows improvement in performance over the passive learner in some cases, but otherwise performs as well as and no worse the passive learner. This suggests that a visual-diversity condition is sometimes, but not always, advantageous in learning to rank images of shoes, though it does not appear to impair it.

In all except four cases, all active learners outperform both the passive-D and passive learners, although in those four exceptional cases, the passive-D and passive learner only perform as well as the far-sighted learner or the myopic learner for the first half of the iterations, after which they perform worse and continue doing so for the remaining iterations. In other words, given enough iterations (ten to twelve for the Shoes data set), all active learners consistently outperform the passive and passive-D learners. Note that in all cases, however, the far-sighted-D learner consistently outperforms both the passive-D and passive learners, emphasizing the importance of a sample selection method utilizing both low-margin and visual-diversity conditions.

We now take a look at performance among the active learners.

The far-sighted learner performs better than the myopic learner for four of the ten attributes and as well as, but not considerably worse than, the myopic learner for the rest, with the exception of the attribute, *sporty*, where the myopic learner outperforms the far-sighted learner. This indicates that a far-sighted approach for ranking shoes is sometimes, but not always, more advantageous over the myopic approach.

Finally, results indicate that in nearly all cases, far-sighted-D learner outperforms the rest of the learners. The one notable exception is the *bright-in-color* attribute, which is also interestingly the only case out of all our experiments so far where the far-sighted-D learner performs substantially worse than one of the active learners (far-sighted learner) for over half the iterations.

This could be due to the annotation ambiguity problem discussed in Section 4.6.4, since it also deals with an aspect of color (brightness) that could have multiple meanings. Indeed, taking a look at some of the higher and lower ranked shoes from the *bright-in-color* group, we see large disagreements between what people think brightness means. A few annotators ranked shoes as having higher bright-in-color values if they were simply more colorful (e.g., red, yellow) as opposed to white or black (see Figure 4.13). Others ranked shoes as having higher bright-in-color values if they were were more shiny. When we strip away such ambiguity by ranking shoes only based on the attribute *shiny*, we see that performance results are much better.

This raises one other interesting question. Why does the far-sighted-D learner initially perform worse than the far-sighted learner for the *bright-in-color* shoes attribute when it performed as well as and sometimes even better than the far-sighted

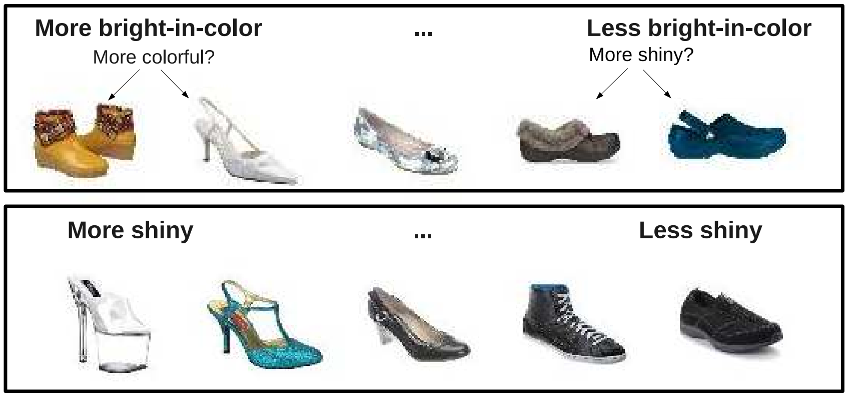


Figure 4.13: **Shoes Ranked on Bright-in-color and Shiny Attributes.** This figure shows two spectrums of shoes images sorted by the extrapolated ranks from Section 4.6.2 over the attributes, *bright-in-color* and *shiny*. There is some disagreement over what constitutes a image that's more bright-in-color. It appears that the leftmost two images of the top spectrum were ranked based on how colorful the shoes were (e.g., red, yellow as opposed to white, gray, black), whereas the rightmost two images appear to be ranked based on how shiny the shoes were. On the contrary, the shoes ranked by the less-ambiguous attribute, *shiny*, show much more agreement and produced better performance curves.

learner for the *white* attribute of the Pubfig data set in Section 4.6.4? A likely explanation is that the factors for ambiguity over what constitutes more "brightness" in shoes is so much more varied than those over what constitutes more "whiteness" in faces that selecting for more diverse image samples actually becomes counterproductive for learning. For instance, faces come in one general shade, but one shoe can possesss multiple colors. Uncertainty over which shades of colors (e.g., gold vs pink) more closely resemble "whiteness" is usually what causes ambiguity over ranking faces based on the *white* attribute, whereas factors such as *shininess* also play a role in determining a shoe's *brightness*. What if a sample contains a shoe that has vibrant colors but lacks luster and a shoe that has darker colors but reflects more light? In this case, adding the visual-diversity condition to the low-margin condition for active learning may actually adversely affect the learner's performance.

Overall, with the exception of the aforementioned special cases, our results continue to support the superiority of the proposed far-sighted-D learner over the other learners as applied to the average case.

4.7 Learning With Live Annotation

This section puts the learners into practice by running the active learning loop in real time over the OSR data set and asking humans to rank selected image samples at each iteration in the learning loop. Unlike the previous Section (4.6), where rank labels first were collected in bulk then processed, here we collect annotations and process them live. Using Amazon's MTurk, we ask participants to annotate successive batches of learner-selected image samples at each step in the learning process. Section 4.7.1 elaborates on the MTurk interface and the experimental setup. Section 4.7.2 discusses the results and Section 4.7.3 provides a brief analysis of the effort saved using the far-sighted-D learner.

4.7.1 User Interface and Experimental Setup

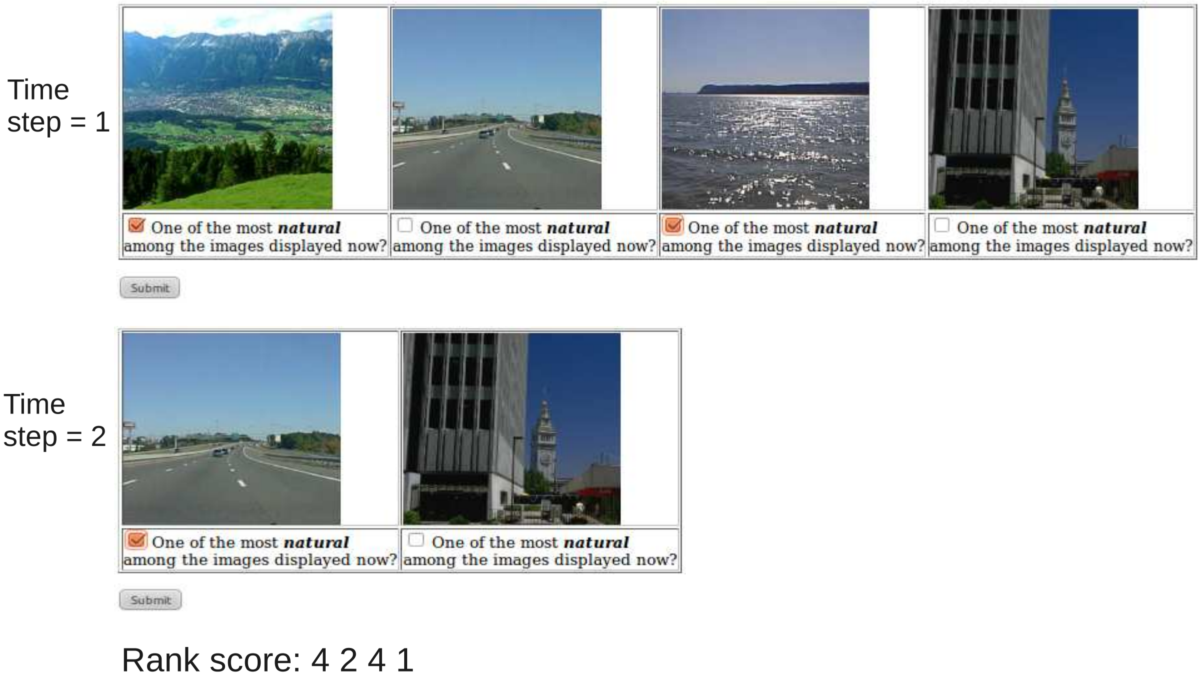
Using Amazon's MTurk, we ask each participant to rank six batches of four image samples, each batch selected by one of the six learners. In order to make the annotation process easy and intuitive, we ask participants to simply "select the image[s] with the most amount of [some attribute]" out of each batch of four images (since we are using a sample size of $s = 4$). Until all four images are accounted for, this process is repeated for the remaining, unselected images, terminating when there are no more images from the sample left to annotate. Figure 4.14 provides an example of the UI used. Once the participant finishes annotating all batches of samples, the interface turns in the labeled results for processing, and each learning algorithm selects a new batch of four samples for the participants to annotate.

To make the results more reliable, we ask five different participants to rank identical samples, and then we aggregate the result. The rank aggregation process is as follows:

First, we assign a numerical rank score from one to four to each image of the training sample based on the annotations provided by the human participants, with one being the lowest rank and four being the highest rank. Note that these scores indicate relative attribute strength only among the images contained in each sample set of four. Images may share the same rank score if they are ranked the same by the annotator.

We then weed out any outliers by comparing each annotation to the mean of the remaining four annotations using Kendall's τ. If $\tau < .7$ (a value that was chosen based on exploratory trial-and-error runs over some test annotations), then that annotation is thrown out as unreliable. After weeding out inconsistent data, we average the scores of the remaining annotations.

Finally, we assign images whose averaged ranked scores are relatively close to each other the same rank score using mean shift clustering. In other words, we are only confident that two images should be ranked differently if there is sufficient distance between their averaged rank scores. This last step is essentially the same as the corresponding step described in Section 4.6.1, where we combined rank values of data pairs that differed by less than 0.3 units. Like before, this step safeguards



☑ One of the most ***natural*** among the images displayed now? □ One of the most ***natural*** among the images displayed now? ☑ One of the most ***natural*** among the images displayed now? □ One of the most ***natural*** among the images displayed now?

☑ One of the most ***natural*** among the images displayed now? □ One of the most ***natural*** among the images displayed now?

Rank score: 4 2 4 1

Figure 4.14: **MTurk UI for Ranking Image Samples.** Our UI asks annotators to check the boxes under images with the “most amount of [some attribute]” starting with four images. After doing so, the participant is asked to repeat the process with the remaining, unchecked images until all have been labeled. From this, we can derive an ordering over these set of images and assign relative rank scores from 1-4 to each image, with four being the highest rank and one being the lowest. Note that these scores are not absolute rank values but are relative only among the set of images contained in this training sample.

against fluctuations in averages caused by minor uncertainty among annotators. Consider the following case: Suppose four annotators rank a sample of images in the same way such that the rank scores for each are $\{1, 2, 4, 4\}$ but one annotator ranks the same sample slightly differently such that the rank scores are $\{1, 2, 4, 3\}$. The averaged rank score will be $\{1, 2, 4, 3.8\}$, but since the averaged rank scores of the third and fourth element are relatively close together, it would be more sensible to consider them as sharing one rank.

The limitations of MTurk prevent us from ensuring that the same set of annotators rank every image sample at each iteration the learning process. Instead, one set of annotators in charge of labeling samples at one iteration is likely to be different from the set of annotators in charge of labeling samples at the next iteration. One issue is that the first set of annotators might have a very different “aggregated opinion” than the second set over what constitutes a stronger or weaker attribute. For instance, group A may be inclined to say a face is more masculine-looking if it has more facial hair while group B would make the same decision only if it is also more muscular. To mitigate this issue, we show each new set of annotators the results of previously ranked image samples prior to the annotation process so that their labels are more likely to be consistent with those of their predecessors.

We use the image pairs from Section 4.6.1 as the test set here and the aggregated pairwise rank values as the ground-truth values. To evaluate the accuracy of a learned ranking function, we apply it to each pair from the test set, compute a Kendall's τ value between the function-calculated and ground-truth pairwise rank values, then average all the τ's together. To ensure reliability of using these pairwise ranked images as test images, we also show the annotators a subset of the pairwise-ranked test images prior to the annotation prcoess so that their labels will likely be consistent with those of the test set.

The OSR data set contains 2688 images. Of those, 202-217 (depending on the attribute) were used in collecting the pairwise ranked data discussed in Section 4.6.1 and are therefore set aside here as the test set. The remaining 2486-2471 image serve as the training set. Of these, we randomly select four samples (for each attribute) as seeds to train an initial ranking function. We run all six learners for six to seven iterations, querying human annotators for rank labels at each iteration. Like before, for the diversity-based learners, far-sighted-D and passive-D, we partition data from the training set into $k = 10$ clusters.

4.7.2 Results

Figure 4.15 shows the results of the experiment. Due to time constraints, we were only able to perform the live annotation experiments over the attributes of the OSR data set.

First we look at the baseline approaches.

For all but one attribute (*perspective*), the handicapped learner performs the worst, as expected. For that one exception, the handicapped learner may have merely received a boost in performance at the very beginning from chance (i.e., just happened to choose an informative sample set without intending to), resulting only in a higher starting performance curve. Indeed, after the second iteration, the handicapped learner starts choosing uninformative samples again and no longer improves in performance. It even begins to decrease a bit in performance whereas the other learners continue to improve.

The passive learner performs worse than the active learners for three attributes (*perspective*, *large-objects*, and *close-depth*) and about as well but not necessarily better than the active learners for the remaining attributes (*natural*, *open*, and *diagonal-plane*). However, for two of the attributes, *natural* and *open*, the passive learner initially performs better than one of the active learners, but then falls below them

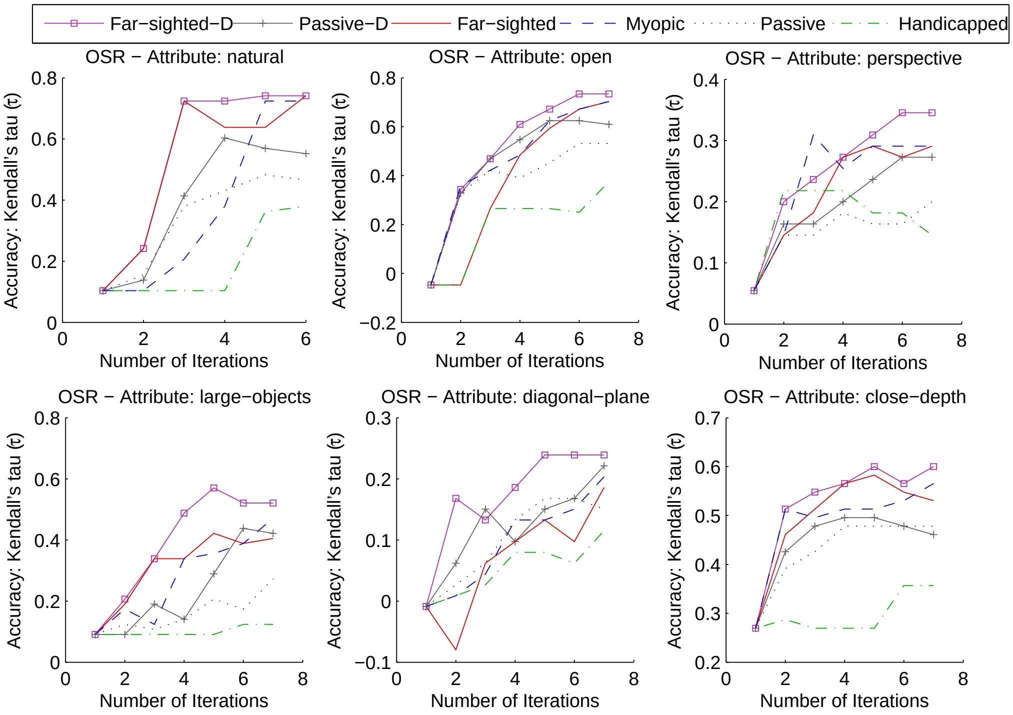


Figure 4.15: **Performance Results - OSR (Live Annotation)** Performance results after running the six learners over the six attributes of the OSR data set: natural, open, perspective, large-objects, diagonal-plane, and close-depth. From these results, we can see that overall, the far-sighted-D learner performs the best. The active learners generally outperform the passive learners, with the exception of the attribute *diagonal-plane*. The handicapped learner performs the worst, as expected.

and continues to perform worse than the active learners after the third iteration. In the case of the attribute, *diagonal-plane*, however, the passive learner does not perform much differently from most other active learners (with the exception of the far-sighted-D learner, which outperforms all other learners).

Note that out of all the attributes, *diagonal-plane* yields results with the lowest accuracy, topping at a measly $\tau < .3$. Taking a look at the types of samples selected in as early as the second iteration of the learning process (see Figure 4.16), we see that ambiguity over ranking similar-looking images might be a reason. In several instances, two or more images from a sample appear ambiguous in terms of the diagonal-plane attribute, making it difficult to label. However, note that the far-sighted-D learner fares very well in comparison, possibly because it aims to choose dissimilar-looking (and therefore less-ambiguous) samples while still selecting low-margin samples that will provide much information once annotated.

The passive-D learner performs worse than the far-sighted-D learner in all cases and performs better than the passive learner in all except for one case, the *diagonal-plane* attribute, likely for the same reason stated above. This shows that both the low-margin and the visual-diversity conditions are important for the best performance.

Next we examine the active learners.

With the exception of the attribute, *perspective*, the far-sighted-D learner consistently outperforms all other learners across the attributes. However, for the *perspective* attribute, the far-sighted-D learner does start to outperform the other learners at the fourth iteration and maintains a large performance gap afterwards. A reasonable explanation is that it is not the far-sighted-D learner that is behaving irregularly, but the myopic and handicapped learners (which are the only learners that outperform the far-sighted-D learner at the beginning). As explained above, the handicapped learner spikes in performance at the very beginning of the learning loop due to selecting unexpectedly informative samples by chance. It is likely that this is also what happens with the myopic learner, as we see that its performance curve gains a sharp spike for the first two iterations but then ceases to improve any more afterwards. The handicapped learner shows this same trend. On the other hand, all other learners, including the far-sighted-D learner, continue to show gradual improvement. Overall, there is still strong evidence in favor of the far-sighted-D learner.

The performance difference between the regular far-sighted learner and myopic learner is a bit more subtle. The far-sighted learner outperforms the myopic learner for most of the iterations in three cases (*natural*, *large-objects*, and *close-depth*), and the myopic learner outperforms the far-sighted learner for most of the iterations in the remaining three cases (*open*, *perspective*, and *diagonal-plane*). However, for nearly all of the attributes, one eventually catches up to and sometimes even

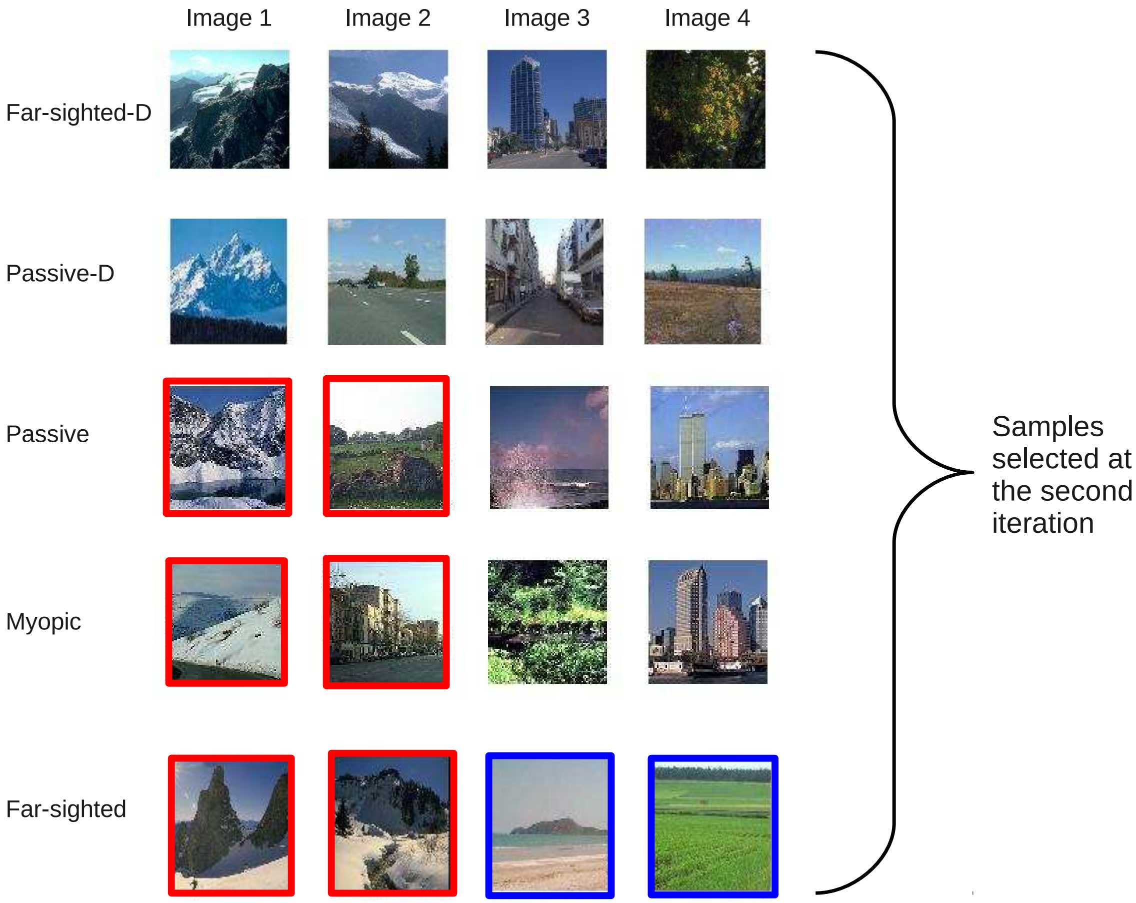


Figure 4.16: **Ambiguity over Ranking Scenes by the Attribute: Diagonal Plane.** This figure shows the image samples selected by each learner at the second iteration of learning loop for the attribute *diagonal-plane*. Images bounded by colored boxes on each row represent candidates for ambiguity. In the Passive row, if you factor the mountain slopes in Image 1 into the *diagonal-plane* attribute, then it would be ranked higher than Image 2. Otherwise, the ground in Image 1 is actually flatter than the ground in Image 2. For the Myopic row, if you only count the roads in Images 1 and 2 towards the *diagonal-plane* attribute, then they would be ranked equally. If you include the slopes from Image 1, then Image 1 would be ranked higher. For the Far-sighted row, there is even more confusion. Should ground in Image 4 be considered strongly slanting or very flat? The answer may vary depending on how you look at it. In this case, samples selected by the diversity-based learners would provide the clearest information whereas samples selected by the margin-based learners would provide the most confusing information.

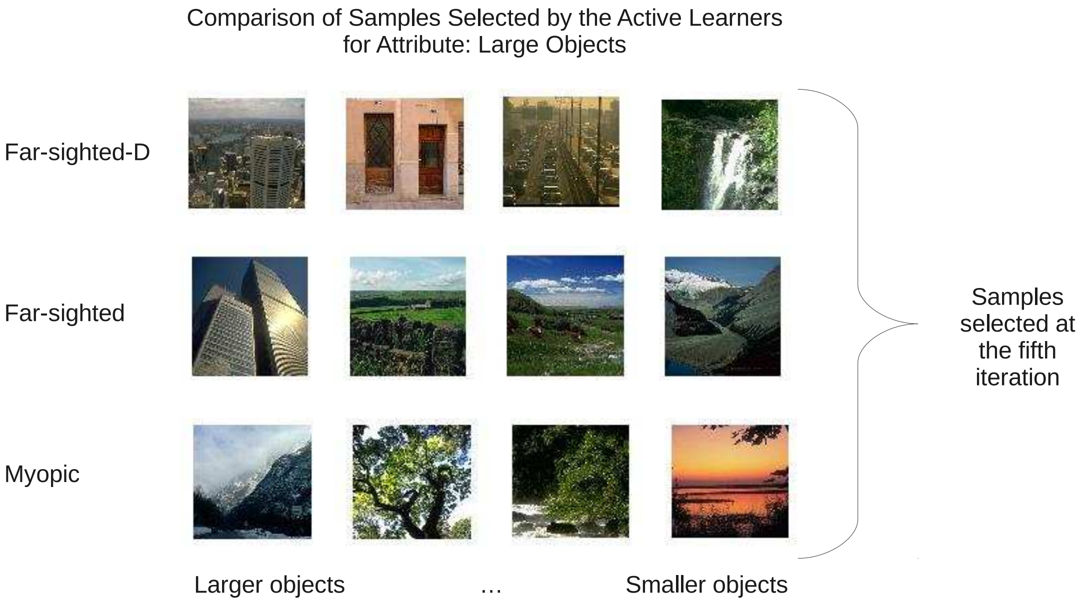

Comparison of Samples Selected by the Active Learners
for Attribute: Large Objects
Far-sighted-D
Far-sighted
Myopic
Samples selected at the fifth iteration
Larger objects
…
Smaller objects

Figure 4.17: **Comparison of Actively Selected Samples for Attribute: Large Objects.** Of the three active learners, the far-sighted-D learner performs better than the far-sighted learner, and the far-sighted learner performs better than the myopic learner when it comes to ranking by the attribute *large objects*. This figure shows the image samples selected by each active learner at the fifth iteration of learning loop sorted from highest to lowest rank. From the figure, we can see that the rightmost three images of the far-sighted-selected samples are similar-looking. Despite this, it's easy to see that the gate in the second image of the Far-sighted row is larger than any of the objects in the third image, but the third and fourth image rank about the same in terms of *large objects*. Because it is easy to decide upon the correct rankings for images over an attribute like *large objects*, we expect the far-sighted learner to perform better than the myopic-learner since it adheres to the low-margin condition more strictly. On the other hand, we can see that far-sighted-D selects a diverse set of images. This not only makes them easier to rank, but each image in this sample is diverse enough such that each receives a separate rank value. Since the far-sighted-D learner must also partially satisfy the low-margin condition of sample selection, its batch of samples will provide even more information, allowing it to outperform the regular far-sighted learner.

surpasses the other. Overall, both learners perform roughly the same, although the far-sighted learner tends to outperform the myopic learner in specific cases where images are easier to label (see Figure 4.17), whereas the myopic learner tends to outperform the far-sighted learner in specific cases where labelling is a bit more ambiguous in terms of the attribute (see Figure 4.16). Assuming no ambiguity, annotating samples selected by the far-sighted learner ideally yields the most information to train a ranking function, so in cases where assigning ranks is easy, the far-sighted learner is expected to outperform the myopic learner. However, when there is more ambiguity involved, the far-sighted learner is more likely than the myopic learner to select image samples that are all very similar to each other (as opposed to the myopic learner, which would only be expected to select similar *pairs* of images), thus making a good annotation difficult to come up with. Still, in all cases, the proposed far-sighted-D learner fares the best, as it strikes a balance between choosing samples that offer the most information, provided they are well-annotated, and choosing samples that are adequately diverse, such that it both offers substantial information once annotated as well as makes for easier annotation.

4.7.3 Discussion on Human Effort Saved

It took our annotators on average 4 minutes to complete a task where they had to label six batches of image samples (one selected by each type of learning algorithm). However, this is an overestimate since it includes the time it took for the annotators to read the instructions for completing the task as well as look over example annotations from previous annotators and the test set images. Therefore, we assume it took three minutes to actually perform the entire labeling task, or 30 seconds per sample set.

Now consider the attribute that yielded the worst performance across all learners, the *diagonal-plane* attribute. At the second iteration, the far-sighted-D learner has achieved an accuracy of $\tau = .18$. The next best learner does not reach that point until three iterations later. In this case, all else being equal, using the far-sighted-D learner saved us 1.5 minutes worth of human labeling effort. Even if we factor in the fact that it takes the far-sighted-D learner approximately 0.74 seconds to select four samples from 2688 OSR images (see last paragraph of Section 3.3.3), this value is negligible compared to the 1.5 minutes of human effort saved.

Next we take a look at an attribute that yielded one of the better performances across all the learners, the *close-depth* attribute. At the third iteration (which is the point where performance begins to differentiate for all learners), the far-sighted-D learner has achieved an accuracy of $\tau = .55$. One iteration afterwards, the far-sighted learner catches up, though the myopic learner does not hit that point until the sixth iteration. For ranking images over the *close-depth* attribute, all else being equal, we

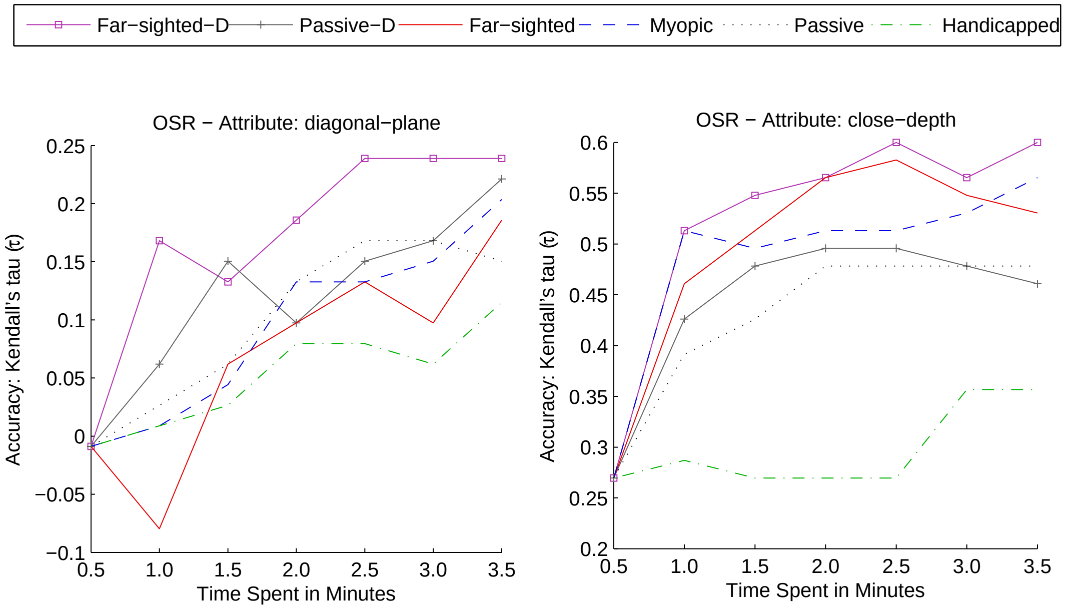


Figure 4.18: **OSR Performance Curves Plotted Against Time.** These performance curves for the *diagonal-plane* and *close-depth* attributes are like the ones from 4.15, except instead of being plotted against the number of iterations, these are plotted against human effort spent in terms if time (minutes). As we can see, using the far-sighted-D learner saves the most human effort in terms of time spent annotating.

spent 30 more seconds worth of human labeling effort using the far-sighted learner and 1.5 minutes more seconds worth of human labeling effort using the myopic learner compared to the far-sighted-D approach. As for the baseline approaches, not one of them even reach that point. In fact, in four of the six attributes, the passive learner never (within the six to seven iterations observed) performs as well as how the far-sighted-D learner performed at its third iteration, equating to some two minutes worth of lost time (possibly more, since we do not know how much longer it would have taken for the passive learner to finally catch up).

Keep in mind that since we do not have data on the time it took for the annotators to rank each individual learner-selected batch of samples, we assume here that it takes the same amount of time for the human to annotate all sample sets regardless of which learner selected it. Ideally, however, it should take less time to annotate samples selected by the diversity-based learners since they work to pick more visually diverse images. Knowing this, it's likely that in reality, we saved even more effort using the far-sighted-D learner than previously stated. While a few minutes may not seem like much at first, in seeking to learn ranking functions for multiple attributes across a number of data sets, the numbers can quickly add up.

In summary, although there is evidence to support that employing a sample selection method under the low-margin condition alone (e.g., the far-sighted and myopic learner) or the visual-diversity condition alone (e.g., the passive-D learner) yields better performance than a random sample selection method (i.e., the passive learner), it is clear that a combination of the two (e.g., the far-sighted-D learner) performs the best and saves the most human effort.

Chapter 5

Conclusion

In this work, we present a novel active learning model for image ranking over relative visual attributes. It not only selects images with the lowest rank margins as training samples, but also selects images that are visually diverse. Experimental results demonstrate the effectiveness of using this new model for training ranking functions on an assortment of images from data sets as diverse as faces, scenes, and shoes across as many as 27 distinct attributes using minimal human labeling effort. This suggests that our model can likely be applied to many other image classes and attributes as well, training ranking functions for a variety of attributes in less time than traditional methods. Since relative attributes allow us to capture semantic relationships between images as well as describe objects across multiple categories in an easier, more human-intuitive way as opposed to binary attributes, they are clearly useful in many computer vision applications such as recognition and image retrieval. Given the importance of relative visual attributes, the ability to derive ranking functions for more relative attributes in less time is highly valuable.

There is also potential for future work. In a majority of cases, our model learns an accurate ranking function in fewer iterations, using less training samples and human labeling effort. However, there are two exceptional cases: One case is where multiple, possibly conflicting factors define an attribute's relative strength such that selecting more visually diverse image samples leads to high uncertainty and ambiguity over how to rank them. The second case is where the attribute is highly localized such that only a few features matter in determining its strength, and therefore it provides little to no advantage to use the entire set of features for evaluating image diversity. This suggests possibilities for future work in this area, such as the weeding out or avoidance of ambiguous image cases, automatic discovery of features most relevant to the attribute, and other ways to evaluate visual diversity other than the Euclidean distance between feature vectors. Indeed, there is much promise in further enrichment of the active learning model for image ranking over relative visual attributes.

Bibliography

[1] M.-F. Balcan, A. Broder, and T. Zhang. Margin based active learning. *Lecture Notes in Computer Science*, 4539:35–50, 2007.

[2] T. Berg, A. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *European Conference on Computer Vision*, Sept 2010.

[3] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *European Conference on Computer Vision*, 2010.

[4] K. Brinker. Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th International Conference on Machine Learning*, 2003.

[5] K. Brinker. Active learning of label ranking functions. In *International conference on Machine learning (ICML)*, 2004.

[6] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon. Adapting ranking SVM to document retrieval. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2006.

[7] Z. Cao, T. Qin, Tie-Yan, L. Ming-Feng, and T. H. Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, 2007.

[8] O. Chapelle. Training a support vector machine in the primal. *Neural Computation*, 19(5):1155–1178, 2007.

[9] C. K. Dagli. Combining diversity-based active learning with discriminant analysis in image retrieval. In *Proceedings of the Third International Conference on Information Technology and Applications*, pages 173–178, 2005.

[10] P. Donmez and J. G. Carbonell. Active sampling for rank learning via optimizing the area under the roc curve. 5478:78–89, 2009.

[11] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1778–1785, June 2009.

[12] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *The Journal of Machine Learning Research*, 4:933–969, 2003.

[13] A. Frome. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[14] K.-S. Goh, E. Y. Chang, and W.-C. Lai. Multimodal concept-dependent active learning for image retrieval. In *Proceedings of the 12th annual ACM international conference on Multimedia*, 2004.

[15] S. C. H. Hoi, R. Jin, and M. R. Lyu. Large-scale text categorization by batch mode active learning. In *Proceedings of the 15th international conference on World Wide Web*, 2006.

[16] Y. Hu, M. Li, and N. Yu. Multiple-instance ranking: Learning to rank images for image retrieval. In *Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[17] V. Jain and M. Varma. Learning to re-rank: query-dependent image re-ranking using click data. In *Proceedings of the 20th international conference on World wide web*, 2011.

[18] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, pages 133–142, 2002.

[19] A. Kovashka, D. Parikh, and K. Grauman. Whittlesearch: Image search with relative attribute feedback. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[20] N. Kumar, A. Berg, P. Belhumeur, and S. Nayar. Attribute and simile classifiers for face verification. In *IEEE International Conference on Computer Vision*, pages 365–372, Sept 2009.

[21] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Computer Vision and Pattern Recognition (CVPR)*, pages 951–958, Jun 2009.

[22] T.-Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3), 2009.

[23] B. Long, O. Chapelle, Y. Zhang, Y. Chang, Z. Zheng, and B. Tseng. Active learning for ranking through expected loss optimization. In *International ACM SIGIR conference on Research and development in information retrieval*, 2010.

[24] P. Melville and R. J. Mooney. Constructing diverse classifier ensembles using artificial training examples. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 505–510, August 2003.

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

[26] T. Onoda, H. Murata, and S. Yamada. SVM-based interactive document retrieval with active learning. In *New Generation Computing*, 2003.

[27] D. Parikh and K. Grauman. Relative attributes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 503–510, Nov 2011.

[28] S. Rajaram, C. Dagli, N. Petrovic, and T. Huang. Diverse active ranking for multimedia search. In *Computer Vision and Pattern Recognition*, 2007.

[29] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where - and why? semantic relatedness for knowledge transfer. In *Computer Vision and Pattern Recognition (CVPR)*, pages 910–917, 2010.

[30] D. Roth and K. Small. Margin-based active learning for structured output spaces. *Lecture Notes in Computer Science*, 4212:413–424, 2006.

[31] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *International Conference on Machine Learning (ICML)*, pages 839–846, 2000.

[32] C. Siagian and L. Itti. Rapid biologically-inspired scene classification using features shared with visual attention. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 29, pages 300–312, 2007.

[33] X. Tian, D. Tao, X.-S. Hua, and X. Wu. Active reranking for web image search. In *IEEE Transactions on Image Processing*, 2010.

[34] S. Tong and E. Chang. Support vector machine active learning for image retrieval. 26:49–61, 2007.

[35] G. Wang, D. Forsyth, and D. Hoiem. Comparative object similarity for improved recognition with few or no examples. In *IEEE International Conference on Computer Vision*, pages 3525–3532, June 2010.

[36] J. Wang, K. Markert, and M. Everingham. Learning models for object recognition from natural language descriptions. In *British Machine Vision Conference*, Sept 2009.

[37] Z. Xu, R. Akella, and Y. Zhang. Incorporating diversity and density in active learning for relevance feedback. *Lecture Notes in Computer Science,*, 4425:246–257, 2007.

[38] H. Yu. SVM selective sampling for ranking with application to data retrieval. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, 2005.

[39] C. Zhang and T. Chen. An active learning framework for content-based information retrieval. 4:260–268, 2002.