Design of a Machine Learning Architecture for Hierarchical Vision Systems

Marcus A. Maloof

Department of Computer Science
Georgetown University
Washington, DC 20057-1232
maloof@cs.georgetown.edu
http://www.cs.georgetown.edu/~maloof

CS-99-01

July 1999

Abstract

In this report, we describe a machine learning architecture for hierarchical vision systems. These vision systems work by successively grouping visual constructs at one level, selecting the most promising, and passing them up to higher levels of processing. This continues from the pixel-level of the image to the object-model level. Traditionally, researchers have used static heuristics at each level to select the best constructs. In practice, this approach is brittle, because people have not been successful at surveying the evidence necessary for robust performance, and is static, because designers do not incorporate learning mechanisms that would let the system improve its performance with the aid of user feedback. The machine learning architecture proposed herein is an attempt to address both of these issues.

Key words: hierarchical vision systems, on-line learning, perceptual grouping

1 Introduction

We have devised a novel machine learning architecture that shows promise of significantly improving the accuracy of hierarchical machine vision systems. The proposed approach organizes learning and recognition modules into a hierarchy. Modules at the lower image level survey local information, such as pixel intensities and shape features, and form simple visual constructs, which they pass to higher, construct- and object-level modules. These modules higher in the hierarchy take into account more global types of information, such as spatial or geometric properties. In general, modules in the hierarchy take lower-level constructs and use perceptual grouping operators to form new constructs. To avoid computational bottlenecks, the modules evaluate new constructs and select only the most promising to pass to the next level of processing.

Traditionally, people have programmed heuristics for construct selection, and, as a result, these heuristics often lack robustness because of our inability to survey large amounts of evidence. A greater problem is that these mechanisms are static, meaning that they cannot adapt and improve once deployed in a system. People often interact directly with such systems, or at least supervise their operation, which suggests a role for interaction on the part of the user, and a role for adaptation and learning on the part of the intelligent vision system.

The proposed approach uses on-line learning algorithms, instead of static heuristics, to induce concept descriptions for selecting the most promising visual constructs at each level in the hierarchy. If a system based on this approach makes a mistake, then the operator can provide feedback interactively, which is used to update the concept descriptions for selecting constructs associated with each module. This presents a challenge because we must develop strategies for correctly and efficiently propagating feedback to the modules in the hierarchy that produced the error.

Ours is the first learning approach to take advantage of the decomposition of visual objects into constituent parts that is inherent to model-based and hierarchical vision systems. We anticipate that our approach will yield systems that attain higher accuracy than systems that use a single learning process at the top of a recognition hierarchy to map features to object classes, which is characteristic of many current approaches. This work also promises a methodology for managing user feedback in hierarchies of heterogeneous learning algorithms.

Our plan is to build an experimental vision system based on the proposed learning approach for the domain of detecting blasting caps in X-ray images of airport luggage. We intend to conduct experimental studies designed to measure the improvement in accuracy of our approach versus existing approaches, and to measure over time the affects of interaction and feedback on performance. The experimental studies will include cost-sensitive learning methods, since our data sets invariably will be skewed toward the class of lesser importance, and we do not have a precise cost analysis of errors for our domain. To assess the accuracy of the cost-sensitive learning methods, we will use Receiver Operating Characteristic (ROC) analysis and area under ROC curves as our performance metric. Statistical tests, such as Analysis of Variance and Duncan's test, will indicate whether the experimental results are statistically significant. We present preliminary results to illustrate our experimental methodology on the task of blob detection, which is the function of one of three modules in our experimental system.

2 Preliminaries

To focus and ground discussion, we will use the vision domain of blasting cap detection in X-ray images since blasting caps are representative of the class of visual objects we wish to recognize. They have compositional structure and recognition requires both local and global processing. Furthermore, we have chosen to build upon Nevatia's *perceptual grouping* approach (Mohan & Nevatia, 1989) for three-dimensional object recognition because it is mature and has been applied successfully to complex recognition problems, such as building detection in overhead images (Huertas & Nevatia, 1988; Lin & Nevatia, 1996). More importantly, the hierarchical nature of Nevatia's approach lends itself to a novel visual learning methodology, which is the topic of this report. In the next two sections, we provide relevant details of blasting cap detection and of Nevatia's perceptual grouping approach.

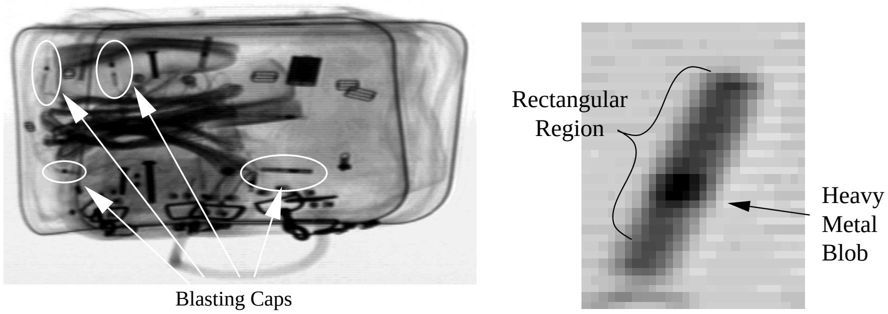

Figure 1: X-ray images of blasting caps in airport luggage. Left: A representative image from a collection of thirty. Right: A closeup of a region containing a blasting cap.

2.1 Blasting Cap Detection

When we X-ray blasting caps as they might occur in luggage in an airport scenario, they appear similar to the images in figure 1. One feature useful for recognition is the low-intensity blob near the center of the object, which is produced by a concentration of heavy metal explosive near the center of the blasting cap. Another is the rectangular region surrounding the blob, which is produced by the blasting cap's metal tube. However, the mere appearance of these two regions of interest is probably not sufficient for detection, so we must also ensure that the proper spatial relationships exist between a given blob and rectangular region (or "rectangle"). As we will see in the next section, Nevatia's perceptual grouping approach provides an elegant framework for solving this recognition problem.

2.2 Hierarchical Vision Systems

The main tenet of Nevatia's perceptual grouping approach is that machines can recognize objects by repeatedly grouping low-level constructs (e.g., lines) into higher-level ones (e.g., rectangles). At the lower levels of processing, the machine surveys local information, such as a region's area or pixel intensities. As recognition proceeds up the hierarchy, higher-level reasoning processes take into account global information, such as geometric and spatial relationships. The left diagram in figure 2 shows a schematic for a hierarchical vision system for recognizing blasting caps.

Inherent to this process are mechanisms at each level that select the most promising visual constructs for further processing at the higher levels, as shown in the right diagram of figure 2. Traditionally, people have manually programmed these heuristics for construct selection using methods such as constraint satisfaction networks (Mohan & Nevatia, 1989) and linear classifiers (Lin & Nevatia, 1996). We contend that using on-line machine learning techniques and user feedback to acquire the criteria for selecting the most promising constructs at each level will yield more robust vision systems capable of improving their accuracy[1] over time.

3 Statement of the Problem

Modern hierarchical machine vision systems are often brittle because, due to cognitive limitations, humans cannot adequately survey the evidence required for these systems to cope in their intended environments. Furthermore, the fact that these systems, once deployed, cannot adapt to changes in their environment contributes to their lack of robustness. Visual learning approaches (i.e., approaches that combine machine learning with computer vision) hold the potential for addressing both of these problems.

[1] The term *accuracy* is often used to mean "percent correct." However, there are different measurements of accuracy, percent correct being one (Swets, 1988). Hence, we will use the term *accuracy* in the general sense, and, as we will describe in section 6, use area under an ROC curve, instead of percent correct, as our measure of accuracy.

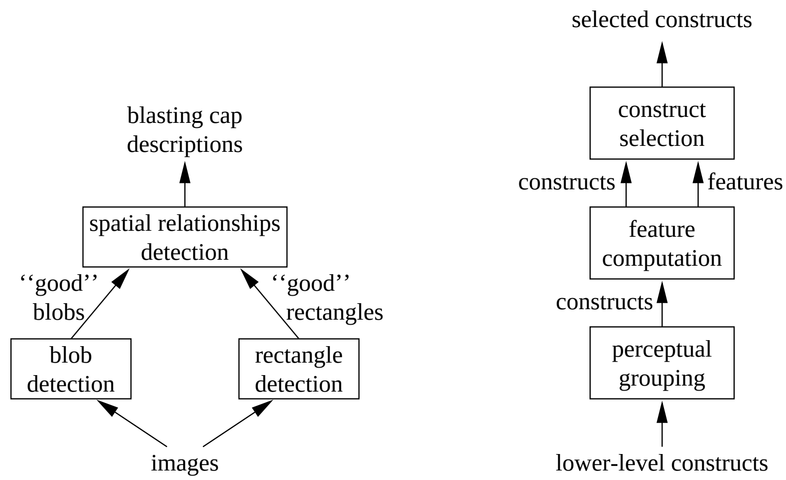

Figure 2: Hierarchical vision systems. Left: A hierarchy for recognizing blasting caps. Right: One level of the hierarchy.

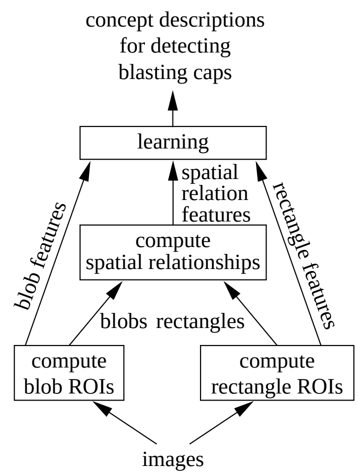

Figure 3: A depiction of current approaches to visual learning for the domain of blasting cap detection.

Recently, there has been considerable work on visual learning approaches; however, much of this research has concentrated on single-step learning and recognition schemes in which a learning technique is applied at one point in a recognition process to, for example, map features to object classes, as depicted in figure 3.

Although using a learning process at the end of a vision process for the task of mapping features to object classes may improve accuracy over traditional, hand-constructed classification methods, we contend that such an approach is undesirable for one primary reason: Current work in visual learning does not take advantage of the fact that the hierarchical approach decomposes objects into simpler parts, which may make learning easier and may result in vision systems with higher accuracy.

Consequently, we propose a new visual learning architecture that tightly integrates vision and learning processes, organizing them in a hierarchy, and addresses our criticism of existing approaches. As for our research hypothesis, we anticipate that the proposed visual learning architecture, which we discuss in the next section, will significantly improve accuracy as compared to existing architectures that use a single learning step at the top level of a recognition hierarchy.

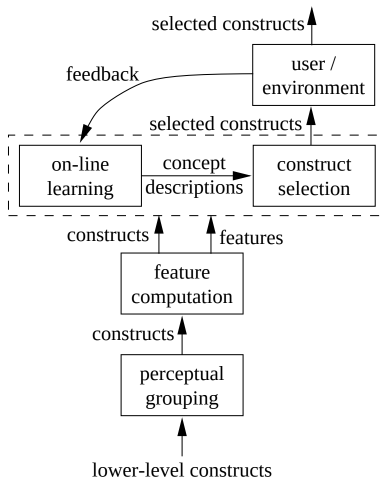

Figure 4: One level of the proposed approach to visual learning.

4 Proposed Approach

Our proposed approach involves organizing a set of learning and recognition modules into a hierarchy. The low-level modules in the hierarchy are responsible for detecting local features, such as blobs or straight lines, and then for passing these constructs to the next level the hierarchy. As we proceed to the topmost level of the hierarchy, the object level, the modules take into account more global types of information, such as spatial or geometric information.

Since each module of the hierarchy is similar in design, although each will be configured differently depending on its task, we will ground discussion by concentrating on one module in the hierarchy, which we present in figure 4. In section 5, we present the design of an experimental system for the domain of blasting cap detection that is based on our proposed approach.

The first step is a perceptual grouping process that takes lower-level constructs from modules lower in the hierarchy and groups them to form new constructs. For example, in one level of a building detection system, such a module would group linear features into parallelograms, which potentially correspond to rooftops. In our blasting cap domain, a module may group blob regions and rectangular regions based on spatial constraints.

These newly formed constructs are then passed to a feature computation component that computes a predetermined set of attribute values for the construct. Depending where the module resides in the hierarchy, these could be low-level attributes, such as statistics computed using the intensities of a region's pixels, or attributes that characterize the shape of the region, including area or measures of compactness. Conversely, higher-level modules would take into account more global types of information, such as geometric constraints. In our blasting cap domain, a module that groups the blob and rectangular regions might use as an attribute the distance between the centroids of the two regions.

Next, the construct selection process takes the current set of concept descriptions and uses the features computed in the previous step to classify each of the constructs. Those constructs that are labeled as positive (and thereby selected) are then passed to the next level of processing in the hierarchy. However, we assume that at some point in the recognition process, the user, or the environment, will provide feedback on the objects that were or were not correctly identified.

When provided, the positive or negative feedback will start on-line learning processes that will use the set of misclassified constructs and their features to modify each level's current set of concept descriptions. Learning and adaptation will occur throughout the life of the vision system.

Ideally, we would like for a user to provide feedback at the highest level, the object level. Assuming that the system draws a graphical representation of recognized objects on the image, to provide feedback for false positives, we would like for the user to simply click on the graphical representation of the misidentified object. Similarly, to provide feedback for false negatives, we would like for the user to select the image region containing the unidentified object. Unfortunately, this approach to feedback leads to problems with credit assignment because it is difficult at the top level of the hierarchy to pinpoint the individual construct(s) at lower levels that led to the misclassification. This is indeed a weighty issue, so in the proposed work, we are assuming that the user will identify the specific constructs at each level that led to the misclassification. However, we will postpone the specific details of this scheme until we describe our design for an experimental vision system, which we discuss in the next section.

5 Evaluation

To evaluate our approach, we propose to implement an experimental visual learning system for detecting blasting caps in X-ray images of luggage, which are representative of a class of objects that we wish to recognize. Furthermore, during and after its construction, we will conduct experiments in an effort to support our research hypothesis, which we stated previously. The following sections provide details about the image set, the implementation, and a fallback plan which we will pursue if we reach an impasse during the study.

5.1 Description of the Image Data

The image data that we will use for our inquiry consists of thirty X-ray images of suitcases containing blasting caps that appear much as they would in an airport scenario: flat with respect to the X-ray source, but rotated in the image plane. The images vary in the amount of clutter (e.g., clothes, shoes, batteries, pens) and in their planar orientation. There is enough variability in the position of the blasting caps within the bags to provide a range of difficult recognition problems. In some instances, the long axis of the blasting cap is perpendicular to the X-ray source. In others, the cap is behind opaque objects, partially occluded by various metal objects, and rotated out of the imaging plane. The left image in figure 1 shows a representative image from the collection. Each of the 8-bit grayscale images has dimensions of roughly 565×340.

As we have discussed previously, blasting caps are appropriate objects for this study because they require a hierarchical approach for recognition. The system must detect the blob and the rectangular region of the X-rayed cap, and then it must analyze the spatial relationship between these two regions, thus requiring, in some sense, a minimal hierarchy.[2] Furthermore, as compared to building detection in overhead imagery, this problem is relatively constrained. (Nevertheless, we are also considering applications to building detection.)

5.2 Implementation of an Experimental Visual Learning System

To validate our proposed approach for visual learning, we plan to use the approach to construct an experimental learning system for detecting blasting caps in X-ray images of luggage. The design for this system appears in figure 5.

The experimental system will consist of three primary modules: one for detecting the blob region in the center of the blasting cap, one for detecting the rectangular region, and one for detecting the appropriate spatial relationships between the blob and the rectangular region. We have already begun work on the blob detection module and present preliminary results in section 6.

Each module in the hierarchy will consist of the components pictured in figure 4: perceptual grouping, feature computation, on-line learning, and construct selection. The user will provide feedback at the top level of the hierarchy and, if necessary, at the lower levels as to the correctness of the selected constructs.

To implement feedback mechanisms, we envision the system presenting its results of processing as a graphical representation of the object detected in the image. To provide feedback for a false negative (i.e., an object erroneously identified as a blasting cap), the user would select the graphical representation of the mistaken object and indicate, by means of a menu or a button, that the selected object was misclassified.

The system would attempt to label all of the constructs involved in the formation of the object description as negative. However, a problem arises when a construct is used in both a true positive and a false negative

[2] Detecting rectangular regions will require processing that involves linear feature detection and grouping, but, for this study, we do not plan to use learning at these lower levels. Consequently, we are treating the detection of the rectangular region as a single vision process.

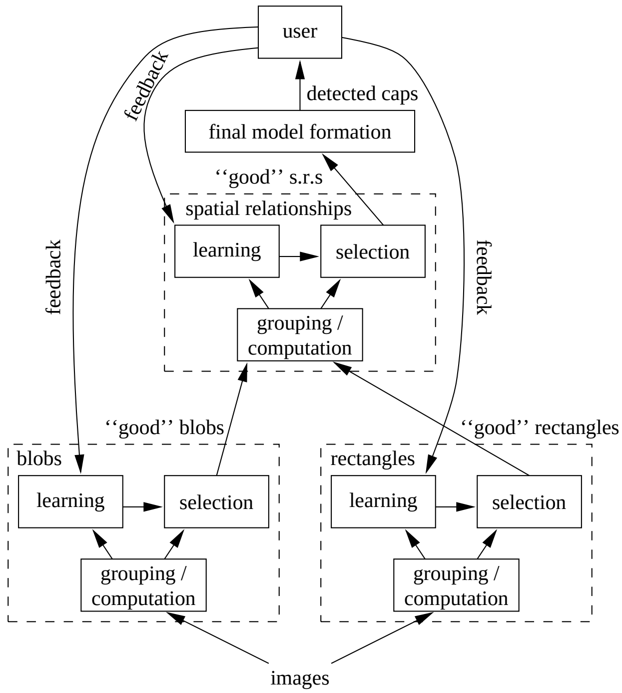

Figure 5: Design of an experimental visual learning system for blasting cap detection based on the proposed architecture.

description. For example, a linear construct could be part of a valid blasting cap description, and it could be part of an invalid description that corresponds to a rectangular region adjacent to the real object. In this situation, we cannot simply label the linear construct as a negative example because this action could reduce the system's ability to correctly identify blasting caps.

Therefore, as the system attempts to label constructs as negative, if it discovers conflicts, then it would present each of the constructs in question to the user. The user would decide which constructs should retain positive labels and which should get new negative labels. The process of presenting this information would start at the highest level of the hierarchy and descend through the lower levels. At each node of the hierarchy, the system would highlight all of the questionable constructs that are part of the object's final description. The user would review these constructs, identify any misclassified constructs, and proceed to the next module in the hierarchy. When the user completes this identification process, an on-line learning step will ensue that will use the newly labeled constructs as training examples to update the concept descriptions for selecting visual constructs at each level of the hierarchy. (Methods for reducing the amount of required interaction will be a fascinating area of future work.)

To provide feedback for false positives (i.e., blasting caps that were not identified), the user will use the mouse to identify the image region surrounding an unidentified blasting cap. The system will retrieve all of the visual constructs that fall within this region and present them to the user. As before, the system will present the constructs at each level in turn, and the user will identify the misclassified constructs. After completing this process, the newly labeled constructs will serve as training examples to the on-line learning process which will update the concept descriptions for selecting the most promising constructs.

The user would use these procedures any time the vision system makes a mistake. And as we have indicated previously, this process of feedback and adaptation continues throughout the life of the vision system.

5.3 Bootstrapping the System

A problem with the preceding discussion is that it assumes that the system already possesses a set of reliable concept descriptions for selecting visual constructs at each of the levels. When constructing the vision system, designers will configure each module in the hierarchy by starting at the image level and moving up the hierarchy to the object level. Once designers have created the routines at a given level for vision processing, perceptual grouping, feature computation, they will use batch learning to induce the concept descriptions necessary for selecting the most promising constructs.

When designers have configured the modules at one level and empirically validated performance, they can begin work at the next level, using the lower-level modules as input. Once designers complete the hierarchy, then they may need to further refine the system's accuracy before deployment in the intended environment. To accomplish this, they will employ the methodology described in the previous section that uses on-line learning and feedback to incrementally refine the concept descriptions for selecting promising visual constructs at each of the levels of the hierarchy until the system achieves an acceptable level of performance.

5.4 Fallback Plan

We have identified two possible points of failure and corresponding avenues for fallback:

1. There may be other vision domains that are appropriate for our proposed methodology. We have chosen the domain of blasting cap detection because it contains the salient characteristics of the class of objects we wish to recognize and requires a simple hierarchy for for recognition. During the period of investigation, we will examine other vision domains, both synthetic and real-world, as possible domains.

2. We have identified another possible learning approach that treats the recognition process from pixels to object descriptions as a grammar learning problem and uses a method, such as the inside-outside algorithm (Stolcke & Omohundro, 1994), to induce probabilistic context-free grammars (Langley, 1998). This approach would entail viewing the recognition hierarchy as a parse tree. The recognition problem then becomes determining whether the visual features in the hierarchy constitute a "legal sentence."

   We view this direction as interesting, but one that is inherently more complicated than the approach we have proposed, and, consequently, we leave it for future work. Furthermore, we have enough evidence from previous studies (Maloof, Duric, Michalski, & Rosenfeld, 1996; Maloof, Langley, Binford, & Sage, 1998; Maloof, Langley, Binford, & Nevatia, 1998) to suggest that the proposed approach has merits that should be explored before considering more complicated alternatives.

6 Preliminary Results

As mentioned previously, we have begun working on the blob detection module, so we will present preliminary results to further illustrate our approach. Our first task was to implement two additional vision routines that were needed but not present in the imgStar package (Winder, 1994), which is simply a library of image processing and vision routines. We implemented a simple region labeling routine and a routine that computes a variety of intensity and shape features, such as the average intensity of a region and the length of the major and minor axes of a fitted ellipse.

Using these routines and those present in the imgStar package, we thresholded five images at $T = 35$, extracted 424 regions from the images, and computed ten features for each: area, length of the major axis of a fitted ellipse, length of the minor axis of a fitted ellipse, eccentricity, and the minimum, maximum, average, mode, variance, and standard deviation of intensity values of the region. The image shown in figure 1 was one of the five that we used in this initial study.

We then presented each of the extracted regions to an expert for labeling using the visual interface shown in figure 6 which presents each region to the user by drawing a rectangle surrounding it. The expert clicks either the positive or negative button to label the candidate blob as either a positive or negative example. At the end of the labeling process, which took about 30 minutes, the expert had labeled 22 regions as positive examples of blobs and 402 regions as negative examples.

We suspected that not all of the features we computed would be relevant for classification, so we used a t-test to measure the relevance of each attribute for predicting each of the two classes: positive and negative, or blob, non-blob. We found that the statistics computed from a region's intensity values provided the

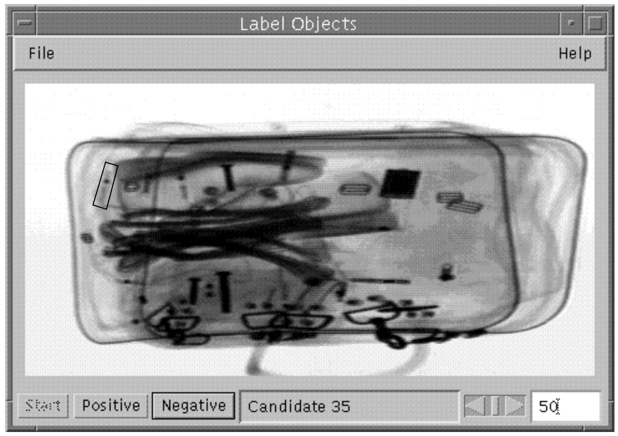

Figure 6: Visual interface for labeling blobs.

NN	C5.0	k-NN, $k=3$	k-NN, $k=5$	k-NN, $k=7$	k-NN, $k=11$	k-NN, $k=9$	Naive Bayes
0.7±0.01	0.72±0.03	0.74±0.01	0.81±0.02	0.82±0.02	0.82±0.0	0.83±0.02	0.86±0.01

Table 1: Rank ordered significant subgroups from Duncan's multiple-range test on the blob detection task ($p < .01$). Metrics are area under the ROC curve and 95% confidence intervals.

most power for discrimination, so we used these six features in an experiment designed to identify the best performing method on the blob detection task.

At this point, we took the expert-labeled data and randomly split it into ten sets of training data (60%) and testing data (40%). We ran three learning algorithms on each of the ten sets: naive Bayes, k-NN ($k = 1, 3, \ldots, 11$), and C5.0, the commercial successor of C4.5 (Quinlan, 1993), a program that learns decision trees.

To compensate for the data set's high skew toward the negative class, we ran these algorithms across a range of error costs and plotted the true positive and false positive rates for each cost as an Receiver Operating Characteristic (ROC) curve (Swets, 1988). Typically, we summarize these curves by computing the area under each, which we approximated using the trapezoid rule. These metrics and their 95% confidence intervals appear in table 1.

We prefer this evaluation methodology to traditional ones that use percent correct as the primary measure of accuracy because, with the latter, we make the assumption that error costs are equal between the classes (Provost, Fawcett, & Kohavi, 1998). For blasting cap detection, false negatives are much more costly than false positives. Skewed data sets complicate this issue because they have the effect of increasing the error cost of the majority class (Breiman, Friedman, Olshen, & Stone, 1984), thus biasing most learning algorithms toward this majority class, which, for most vision applications, is the class of lesser importance.

Returning to our preliminary results, we used two statistical methods to analyze our experimental results. An analysis of variance indicated that all of the differences in the means of the performances were statistically significant ($p < .01$).

Although we now can conclude that the performances (i.e., areas) of the learning algorithms are not equal, we do not know whether the *individual* performances of the learning algorithms differ. Therefore, we also used Duncan's multiple-range test to identify statistically significant subgroups of performances, also at $p < .01$. We follow tradition by drawing lines under these significant subgroups in table 1.

As we can see, a naive Bayesian classifier performed best on the task of blob detection with an area under the ROC curve of 0.86 ± 0.01. This result is significantly different from the next subgroup containing the k-NN classifiers, for $k = 5, \ldots, 11$. The third subgroup consists of k-NN, for $k = 3$, and C5.0. And finally, the fourth subgroup consists of C5.0 and Nearest Neighbor (i.e., $k = 1$). We present the ROC curves for naive Bayes, k-NN ($k = 9$)—the best performer—and C5.0 in figure 7. Notice that the ROC curve for naive Bayes dominates in the important region of the ROC space (i.e., true positive rate greater than 0.6).

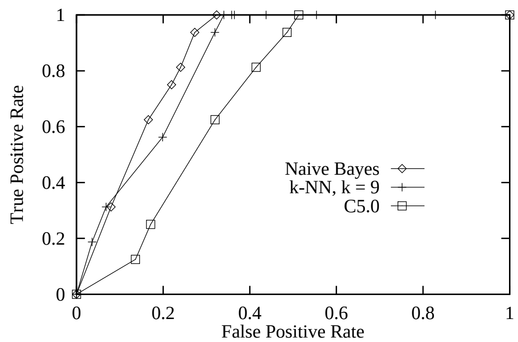

Figure 7: ROC curves for three selected learning methods on the blob detection task.

Although these results are preliminary, we included them to illustrate our approach for configuring the individual modules of our experimental vision system. Based on the results presented here, we would select the naive Bayesian classifier for our blob detection module. Then we would apply this same methodology to construct the module that detects rectangular regions. Once we have configured the lower level modules, we can begin to construct the modules at the higher levels, which, for our experimental system, is the module that detects valid spatial relationships between detected blob and rectangular regions.

7 Conclusion

The proposed work is the next logical step in a long-term investigation of mechanisms for interactive, adaptive software systems. Ours is a departure from current approaches that use a single learning algorithm at one point in a recognition process. As we have discussed, we propose to study multiple, possibly heterogeneous learning algorithms organized in a hierarchical manner. The need for such learning architectures arises in machine vision, especially for situations in which we must build systems to recognize 3-D objects that have compositional structure and that require both low-level, local processing and high-level, global processing for recognition. Building detection and blasting cap detection, our chosen domain, are two examples from this class of problems. And, as we saw in the previous sections, we have proposed a novel visual learning approach that stands to significantly improve the accuracy of hierarchical vision systems.

We anticipate that the success of this study will impact three fronts. Scientifically, this project seeks to tightly integrate vision and learning processes. If successful, the proposed research will yield vision systems with higher recognition rates and, in the longer term, will provide opportunities to adapt and improve the vision processes themselves. On a broader level, studying feedback mechanisms for learning algorithms organized in a hierarchy has scientific merit that extends beyond the context of vision problems and is important for a wide range of applications, such as computer intrusion detection and intelligent agents (e.g., agents for prioritizing one's e-mail queue).

Acknowledgements

Pat Langley, Tom Binford, and Ram Nevatia contributed greatly to the development of the ideas in this report. Dorothy Denning, Tim Snyder, and Cynthia Rais provided valuable suggestions for improving earlier drafts. This research was conducted in the Department of Computer Science at Georgetown University in Washington, DC. This work was supported by the department and by a grant from the Graduate School of Arts and Sciences at Georgetown.

References

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees.* Boca Raton, FL: CRC Press.

Huertas, A., & Nevatia, R. (1988). Detecting buildings in aerial images. *Computer Vision, Graphics, and Image Processing, 41*(2), 131–152.

Langley, P. (1998). *Personal communication.*

Lin, C., & Nevatia, R. (1996). Building detection and description from monocular aerial images. In *Proceedings of the Image Understanding Workshop* (pp. 461–468). San Francisco, CA: Morgan Kaufmann.

Maloof, M., Duric, Z., Michalski, R., & Rosenfeld, A. (1996). Recognizing blasting caps in X-ray images. In *Proceedings of the Image Understanding Workshop* (pp. 1257–1261). San Francisco, CA: Morgan Kaufmann.

Maloof, M., Langley, P., Binford, T., & Nevatia, R. (1998). Generalizing over aspect and location for rooftop detection. In *Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision (WACV '98)* (pp. 194–199). Los Alamitos, CA: IEEE Press.

Maloof, M., Langley, P., Binford, T., & Sage, S. (1998). *Learning to detect rooftops in overhead imagery* (Technical Report No. 98-1). Palo Alto, CA: Institute for the Study of Learning and Expertise.

Mohan, R., & Nevatia, R. (1989). Using perceptual organization to extract 3-D structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 11*(11), 1121–1139.

Provost, F., Fawcett, T., & Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 445–453). San Francisco, CA: Morgan Kaufmann.

Quinlan, J. (1993). *C4.5: Programs for machine learning.* San Francisco, CA: Morgan Kaufmann.

Stolcke, A., & Omohundro, S. (1994). Inducing probabilistic grammars by bayesian model merging. In *Proceedings of the Second International Conference on Grammatical Inference and Applications* (pp. 106–118). Berlin: Springer-Verlag.

Swets, J. (1988). Measuring the accuracy of diagnostic systems. *Science, 240*, 1285–1293.

Winder, S. (1994). *imgStar Image Processing Tools* (Software). Bath, UK: University of Bath. (http://www.bath.ac.uk/Centres/MEDIA/RESOURCES/resources.html)