Learning Descriptions of 2D Blob-Like Shapes for Object Recognition in X-Ray Images: An Initial Study

M. A. Maloof and R. S. Michalski

MLI 94-4

August 1994

Learning Descriptions of 2D Blob-Like Shapes for Object Recognition in X-Ray Images: An Initial Study

Abstract

This paper describes a method for applying AQ15c to learning shape descriptions of 2D blob-like objects in x-ray images. The methodology and initial experimental results are discussed, along with comparisons to *k*-nearest neighbor and to feed-forward neural networks. The AQ15c learning method is shown to have distinct advantages over the aforementioned techniques in terms of higher or comprable classification accuracy, learning and recognition time, and understandability of learned concepts. This approach is well-suited for recognizing objects that can be isolated in the image using histogram and thresholding techniques and that have little internal structure.

Key words: machine learning, machine vision, shape recognition, concept learning.

Acknowledgments

The authors would like to thank the many people whose useful discussions contributed to this work: Jerzy Bala, Eric Bloedorn, Ibrahim Imam, and Ali Hadjarian. The authors wish to also thank those who read preliminary drafts of this paper and provided valued comments: Peter Pachowicz, Jerzy Bala, Janusz Wnek, Eric Bloedorn, and Ibrahim Imam.

This research was conducted in the Center for Machine Learning and Inference at George Mason University. The Center's research is supported in part by the Advanced Research Projects Agency under Grant No. N00014-91-J-1854, administered by the Office of Naval Research, and the Grant No. F49620-92-J-0549, administered by the Air Force Office of Scientific Research, in part by the Office of Naval Research under Grant No. N00014-91-J-1351, and in part by the National Science Foundation under Grants No. IRI-9020266 and DMI-9496192.

1 Introduction

In this paper, AQ15c (Wnek 1994) symbolic learning is used to learn shape descriptions of 2D blob-like objects in x-ray images. The class of objects is a set of x-rayed blasting caps from an image database. The methodology and experimental results are presented, along with comparisons to *k*-nearest neighbor (Weiss and Kulikowski 1992), a statistical pattern recognition technique, and to a feed-forward neural network (Zurada 1992), which a nonsymbolic learning method. Symbolic learning methods have some distinct advantages over the aforementioned methods, and in this case, provided higher or comprable classification accuracy. This approach is well-suited for recognizing objects that can be isolated in the image using histogram and thresholding techniques and that have little internal structure.

Briefly, the paper's organization is as follows. Section 2 provides background and a relevant review of applications of machine learning techniques to problems in machine vision. Section 3 discusses our methodology. Section 4 details our results, while in Section 5, we discuss the contribution and significance of these results and plans for future research.

2 Background

A new and emerging research area, called Machine Learning and Vision (MLV) applies machine learning techniques to problems in machine vision (Michalski et al. 1994). From a vision standpoint, this research stands to quicken vision system development, and in the deployment phases, will add flexibility and adaptability. For learning, this research will inevitably produce powerful techniques and methodologies for dealing with the complexities of vision data. Although various machine learning techniques can potentially be used in a variety of contexts within a vision system, here we concentrate on how learning can be used to induce concept descriptions of shape for simple 2D objects.

2.1 Representation Language and AQ Learning

In the approach presented, a concept description is induced from a set of pre-classified training examples within the context of background knowledge. The induced concept description can then be used to classify or recognize unknown objects. By using this approach, a few epistemological assumptions are made. First, that each training example is descriptive of the concept to be learned. Second, that given training examples for a concept are sufficient to learn a concept description. Sufficiency is either assumed or guaranteed by an expert. Third and finally, future unknowns will be from the same classes from which the training examples were drawn.

Examples, background knowledge, and concept descriptions must be expressed using a representation, preferably a symbolic representation. Symbolic representations are stressed because of the importance of human understandability of learned concepts. Each potential representation carries along with it a representational bias, which is a bias introduced by the representation chosen to express knowledge that ultimately affects how and what knowledge are and can be represented. The AQ15c concept learning system (Wnek 1994) uses an attributional representation language based on Variable-Valued Logic, or VL_1 (Michalski 1973), to learn decision rules from training examples. We begin by defining a finite set of relevant attributes and finite attribute domains for a given problem. Each training example is expressed as a vector of attribute values. Hypotheses are induced by AQ and have the form:

D1 <:: *C1*

where

C1 is a concept description or cover consisting of a disjunction of complexes,
D1 is the decision class assigned by *C1*, and
<:: denotes a class assignment operator.

Complexes or rules are conjunctions of selectors or conditions, each having the form:

[<*referee*> <*relation*> <*referent*>]

where

<*referee*> is a member of the finite set of relevant domain attributes,
<*relation*> is a relational operator (=, <>, >, <, >=, <=), and
<*referent*> is a subset of the finite domain for <*referee*>.

Under strict matching conventions, a decision class is assigned if one of the rules in its cover is true. A rule is true if all of its selectors are true.

The objective is to induce a set of decision rules that describes the training examples in a maximally general way. The difficulty arises with the term *maximally general*. When phrased in this manner, the problem of finding a maximally general description becomes an instance of the set covering problem. That is, find a minimal set of attribute-value pairs that covers (or explains) all the positive concept examples while not covering any of the negative concept examples. Negative concept examples either are explicitly labeled as such, or in a multiple concept learning context, are formed by the remaining training examples for all other concepts.

The set covering problem is known to be NP-complete, but the AQ algorithm (Michalski 1969) solves the set covering problem in a quasi-optimal manner. Briefly, the AQ algorithm randomly selects one of the positive training examples (referred to as the *seed*) and uses this example as the basis to compute a maximally general description with respect to the negative examples (referred to as the *bounded star*). A preference criterion is used to select the most preferable description from the bounded star. If this description results in all positive examples being covered, then the final concept description is the disjunction of all descriptions selected from bounded stars. If positive events remain uncovered by the description, then the algorithm is repeated until all positive examples are covered. AQ15c (Wnek 1994) is the most recent implementation of the AQ algorithm.

2.2 Related MLV Work

Shepherd (1983) used a decision tree learning algorithm to classify shapes of chocolates for an industrial vision system. Using feature vectors to represent examples, Shepherd compared a decision tree algorithm, *k*-nn, and a minimum distance classifier using classification accuracy. Classification accuracies for these classifiers were comprable with the minimum distance classifier producing the best accuracy at 82%.

Cromwell and Kak (1991) also characterize object shapes using feature vectors for images containing electrical components such as resistors, capacitors, and transistors. Concepts are learned by applying generalization rules and selecting the concept that explains the most examples from the training set. Their induction methodology is based on Michalski (1980). The average classification accuracy for their system was 72%. No comparisons were made to other classifiers.

The foundations of applying AQ learning to recognition problems in vision were laid by Michalski (1972, 1973). In those seminal papers, AQ was used to learn and discriminate among classes of textures. These ideas were further developed by Bala (1993). In this work, eight Laws' masks were used to filter structural (e.g., vertical edges) and surface characteristics (e.g., average pixel value in a neighborhood) from a variety of different textures. The eight extracted values from the Laws' masks taken from a region of the texture served as a training example for the AQ algorithm. Pachowicz (1989) compared AQ and *k*-nn on similar data using average classification accuracy. AQ averaged 80% accuracy, while *k*-nn averaged 70%.

3 Methodology

The problem to be solved is classifying or recognizing blasting caps in x-ray images. A vision system performing this task could be used as a focus-of-attention system for airport security personnel who are tasked with searching luggage for potentially dangerous objects.

The symbolic learning methodology used here closely parallels Michalski's (1973) and Bala's (1993) and proceeds through a five step process: (1) Data Reduction, (2) Blob Isolation, (3) Discretization, (4) Learning, and (5) Recognition. (See Figure 1.) The following subsections describe the methodology in greater detail.

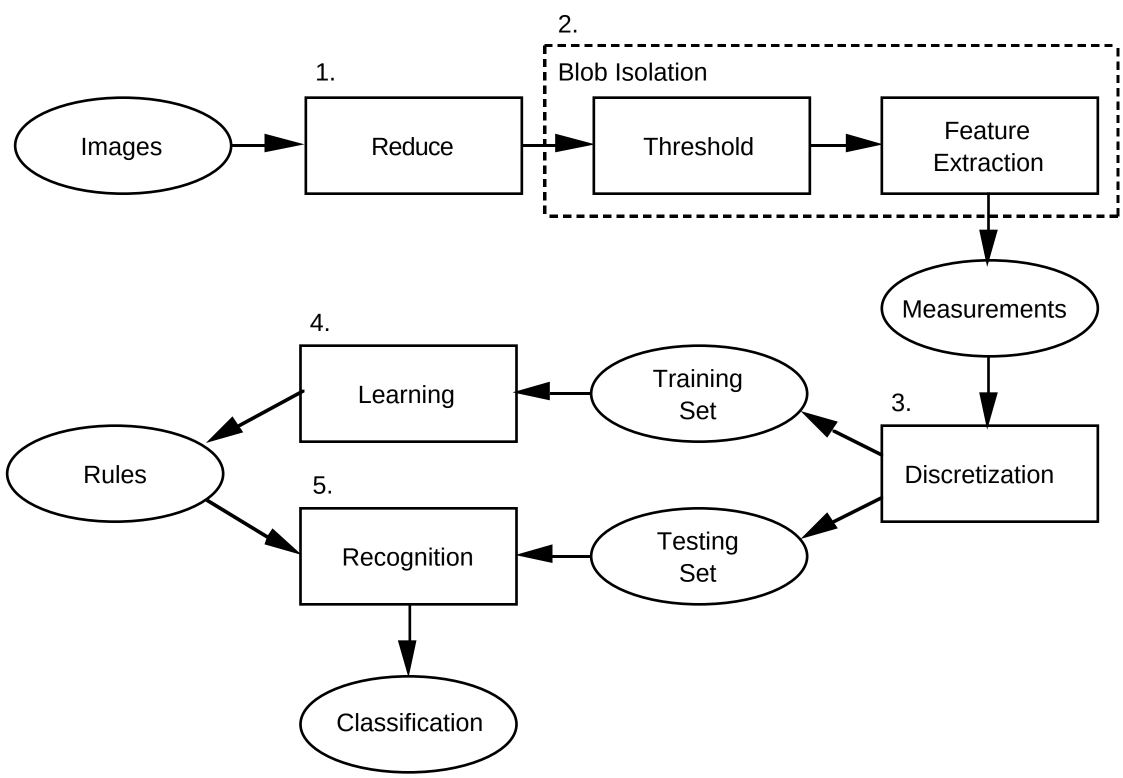

Figure 1: Five step learning and recognition methodology.

3.1 Data Reduction

The first step in pre-processing vision data for machine learning algorithms is data reduction, which is designed to decrease the volume of the vision data without significantly or adversely affecting the image content (i.e., the objects to be recognized). The explicit steps involved are dependent on the images being used and the types of features to be extracted. Examples of data reduction operations include reducing the size of the images or quantizing a 255 gray level images to 4 gray level images. For this application, we scaled the image and quantized image gray level values. The details of this reduction are discussed in section 3.4.

3.2 Blob Isolation

After reducing the image data, our objective was to isolate pertinent blobs present in the cap images and compute a suite of statistics that characterize the blobs. A similar approach was taken by Sydow and Cooper (1992).

Using a threshold operation, we were able to isolate 1–3 blobs in each image (see Figure 2b). The blobs corresponded to the plug area (see Figure 2c, area 1), the primary high explosive (Figure 2c, area 2), which contains heavy metals, and the base of the cap (Figure 2c, area 3). Depending on the quality of the x-ray and the type of cap, the blobs corresponding to the base and the primary high explosive were either not present or integrated with the plug area. The threshold level was set based on the image histogram.

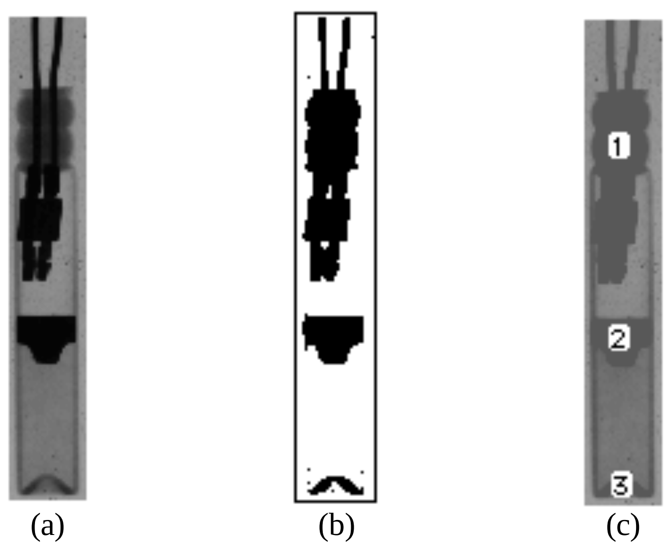

Figure 2: Sample images; (a) Original image; (b) Original image after thresholding; (c) Detected blobs.

After isolating the blobs using the threshold operation, fourteen statistics were computed on blobs consisting of more than 25 pixels (see Table 1). The statistics were based on the blob's gray level values above the threshold level. Table 2 shows an example of the statistics computed for the images in Figure 2.

3.3 Discretization

AQ15c (Wnek 1994) is an attributional learning system that requires discrete attribute values. Many of the statistics computed, such as `area` and `mean` (see Table 2), were real-valued attributes and therefore required discretization.

Several techniques exist for discretizing real-valued attributes (Kerber 1992), including equal-width-intervals and equal-frequency-intervals. With equal-width-intervals, the real range is divided into *n* equal-sized intervals and real values are mapped into the first *n* integers. A problem with this approach is that if the classification algorithm needs to discriminate between two real values and these values are mapped into the same range, any basis for discrimination is destroyed. In other words, the scaling procedure excessively abstracts the data.

Statistic	Description
`area`	Area of blob
`mean`	Average gray level intensity
`sd`	Standard deviation of gray level intensity
`x`	x-coordinate of ellipse fitted to blob
`y`	y-coordinate of ellipse fitted to blob
`mode`	Gray level value occurring most frequently
`perimeter`	Perimeter of blob
`major`	Length of major axis of fitted ellipse
`minor`	Length of minor axis of fitted ellipse
`angle`	Angle between major elliptic axis and the x-axis of the image plane.
`back`	Sum of gray level values minus the background
`intden`	Modal density of smoothed histogram
`min`	Minimum Pixel Value
`max`	Maximum Pixel Value

Table 1: Blob statistics and their description.

Attribute	Value
`area`	0.0127
`mean`	219.5070
`sd`	1.3403
`x`	0.4403
`y`	3.3263
`mode`	219
`perimeter`	0.5554
`major`	0.1980
`minor`	0.0818
`angle`	95.0737
`intden`	0.0065
`back`	219
`min`	219
`max`	223

Table 2: Example of computed statistics.

Equal-frequency-intervals involves discretizing based on the frequency distribution of attribute values over the real-valued range. A problem associated with this technique is that a small group of important outliers could be grouped with a larger cluster of attribute values. Conversely, an important grouping of attribute values could be divided and mapped into different intervals because of outliers.

The ChiMerge algorithm is a discretization algorithm, but unlike the naive techniques discussed previously, uses the χ^2 statistic to merge real attribute values into statistically relevant intervals. In other words, the algorithm groups and separates real-valued attributes into intervals based on a statistical measure of uniformity.

AQ15c, using the ChiMerge algorithm, discretized all attribute values into at least 10 intervals using a 99% significance level. ChiMerge has the freedom to construct any number of necessary intervals; however, one of the parameters to the algorithm is a lower bound on number of intervals. The significance level determines how parsimonious ChiMerge behaves when grouping real-valued attributes. For higher significance levels (e.g., 99%), ChiMerge tends to construct a small number of large intervals (Kerber 1992).

For these experiments, a variety of lower bounds and significance levels were tried, but the best result were achieved with a lower bound of 10 or fewer intervals with a 99% significance level. Most attributes were scaled to 10 discrete intervals, however mean, standard deviation, and angle required 15, 12, and 11 intervals, respectively. Table 3 illustrates the real ranges ChiMerge chose for discretization of the `mean` attribute. Notice that ChiMerge partitioned attribute ranges into differing widths for the attribute (e.g., scaled intervals 5 and 6). ChiMerge constructed discretization ranges, similar to those in Table 3, for each of the 14 attributes. Table 4 shows the statistics from Table 2 after discretization.

Unscaled Attribute Range	Scaled Interval
0.04...214.36	0
214.37...219.41	1
219.42...222.09	2
222.10...225.35	3
225.36...231.65	4
231.66...232.04	5
232.05...242.98	6
242.99...250.77	7
250.78...251.92	8
251.93...254.20	9

Table 3: Discretized attribute intervals for the `mean` attribute.

3.4 Optimization of the Representation Space

As stated in previous sections, initial data preparation steps for AQ15c involve reducing image volume, computing statistics, and quantizing the real-valued statistics into discrete intervals to form linear attributes which are then used to induce an object description. The parameters involved in reducing image volume are scaling values in the x and y dimensions and thresholding image gray levels. To guide our selection of image reduction parameters, information theoretic measures were used to investigate how image content is affected by various settings of these parameters. It is crucial to reduce as much of the image data as possible while removing only a minimum of image content.

Attribute	Scaled Value
area	0
mean	2
sd	1
x	7
y	9
mode	1
perimeter	1
major	0
minor	1
angle	7
intden	1
back	5
min	4
max	2

Table 4: Example of discretized statistics.

Obviously, the space of all possible *x* and *y* scalings and image quantizations for each of the 25 images is enormous. Exhaustively conducting experiments over this space is prohibitive, so we chose a set of 5 representative images and extracted features for various gray level values (256, 64, 16, and 4 levels) and scaling values (100, 80, 60, 40, and 20% of the original dimension), while maintaining a pre-determined aspect ratio between the *x* and *y* dimensions ($x - y <= 40\%$).

For each set of parameters (i.e., x-scale value, y-scale value, and gray level), we extracted 14 statistics or attributes for each of the five images. Both the entropy measure of information content (Quinlan 1983) and the Promise score (Baim 1984) were computed for each of the 14 attributes for each data file. To measure the "goodness" of a data file, and hence the goodness of the image reduction parameters, the sum of the respective information theoretic metrics for each attribute was used.

Images reduced by 40% in both *x* and *y* dimensions yielded the highest information content. Quantization level did not affect information content for the levels considered here. Consequently, subsequent event extraction and learning was conducted on images reduced by 40% in the *x* and *y* dimensions for 256 gray levels.

3.5 Learning

Each sequence of 14 measurements serves as an example of a particular blob. Thus, measurements can be grouped into three distinct classes: blob1, blob2, and blob3. These classifications correspond to the image labelings in Figure 2c. After extracting data from 25 x-ray images, the blob1 class contained 25 examples, the blob2 class contained 21 examples, and the blob3 class contained 20 examples. Each example consisted of 14 linear attributes, corresponding to the 14 computed statistics, while each attribute had between 10 and 15 value levels, a result of the ChiMerge discretization.

Learning was conducted for 100 trials using a 2-fold cross validation methodology (Weiss and Kulikowski 1992). For each trial, the complete set of pre-classified examples was divided evenly into training and testing sets. These sets were scaled and given to AQ15c, which learned a set of decision rules for the training examples. Following the learning step, the decision rules were used to classify the examples in the testing set, which produced a classification rate for the trial. The overall recognition rate was the average recognition rate for the 100 trails. This testing and validation methodology was also used for experiments involving other classification methods. AQ15c learning parameters were held constant for the 100 trials. The best performance resulted when AQ15c was set to generate general rules; maxstar was set to 10; covers were allowed to intersect; and ambiguous examples were considered positive examples. Figure 3 presents one of the rules learned by AQ15c.

```
blob1 ← [y=2..5] [minor=0..3] [angle=1..6]
        (t:17, u:17)

blob2 ← [sd=2..10] [y=1..3]
        (t:12, u:12)

blob3 ← [y=0]
        (t:10, u:10)
```

Figure 3: Example of rules induced by AQ15c.

For the first rule presented in Figure 3, the decision class blob1 is assigned to an unknown example if the example's `y` value is between 2 and 5 inclusively, its `minor` value is between 0 and 3 inclusively, and its `angle` value is between 1 and 6 inclusively. The t-weight (total weight) and u-weight (unique weight) values indicate the strength of rules. In this case, the rule covered a total of 17 examples, while uniquely covering also 17 examples. The fact that the rule's t-weight and u-weight are equal indicates that all the training examples for the blob1 class were covered by this single rule. Most of the rules AQ generated covered all training examples for a given class. This particular ruleset was taken from a learning and recognition trial having an overall classification accuracy of 100%.

3.6 Recognition

After learning VL_1 descriptions for the examples of the three classes, these descriptions were used to classify blobs in the testing sets. Recognition is accomplished by a flexible match algorithm (Reinke 1984). Each rule or VL_1 expression is applied to a testing example yielding a real number called the *degree of consonance*. The degree of consonance, which ranges from 0.0 (no match) to 1.0 (complete match), is a measure of closeness between the testing example and a concept description.

Although there are several parameters that affect how the degree of consonance is evaluated when assigning a decision class (see Reinke 1984 for more details), we set the parameters such that the decision class chosen was the one with the maximum degree of consonance. Ties involving the correct decision class were ruled in favor of AQ15c.

4 Results

For learning and recognition, the AQ15c inductive learning system was used. Results were compared with a *k*-nearest neighbor classifier (Weiss and Kulikowski 1992), a classical statistical pattern recognition technique, and a feed-forward neural network (Zurada 1992), a non-symbolic empirical induction system.

The *k*-nearest neighbor classifier (*k*-nn) works by taking a testing example and finding its *k* closest neighbors in the sample space using some distance measure. Typically *k* is odd to prevent ties, so the class with the most closest neighbors is assigned as the decision class for the testing example. Several runs were made using 100 trial, 2 fold cross-validation, with k = 1, 3, 5, 7, 9, and 15 using a Euclidean distance measure. k = 3 produced the best average classification accuracy of 65%, while $k = 5$ produced the single best classifier with an accuracy of 86%.

The second experiment was with the Quickprop implementation of a feed-forward neural network (Fahlman 1988). An artificial neural network (ANN) is a non-symbolic learning model inspired by the neuronal architecture of the human brain. The class of multilayer feed-forward networks is capable of learning non-linear statistical regularities from pre-classified examples. These models are considered non-symbolic since learned concepts are represented as real-valued weights distributed throughout the network's connections.

The network architecture chosen was a 1 hidden layer network with 14 input, 8 hidden, and 3 output units. Training and testing data were mapped into a continuous real range of [0, 1]. Output patterns were encoded as a linear representation of each of the three possible classes of blobs. Learning parameters were set at 0.55 for epsilon and 0.9 for momentum. Several runs were made using 100 trial, 2 fold cross-validation after the network's learning parameters had been determined. The best average classification rate was 96%, while the single best network achieved a classification accuracy of 100%.

The final experiment involved AQ15c which is a symbolic rule induction learning system that generates maximally general descriptions by heuristically solving the set covering problem using the AQ algorithm. The learning parameters used were intersecting covers to generate rules, a maxstar of 10, the trim parameter was set to induce general rules, and ambiguous examples were regarded as positive. As before, several runs were made using 100 trial, 2 fold cross-validation. The best average classification rate was 92%, while the single best classification rate was 100%. Figure 3 shows an induced ruleset that achieved 100% classification accuracy on testing data. Table 5 summarizes the performance of the three classification methods. Figure 4 contains learning curves for the three methods. This graph represents how the classification accuracy increases with respect to increasing amounts of training data.

Classification Technique	Average Classification Accuracy (%)	Best Single Performance (%)	Average Learning and Recognition Time (seconds)
k-nn	65	86	0.8957
ANN	96	100	1.6974
AQ15c	92	100	0.0411

Table 5: Performance summary for classification technique.

5 Discussion

The primary contribution this works makes is that it demonstrates one way in which machine learning can be applied to a recognition problem in machine vision. One of the main advantages symbolic methods have over non-symbolic and statistical methods is comprehensibility of what was learned. With AQ-type learning we know precisely which attributes and value ranges will lead to what decisions. The learned symbolic rules also suggest which attributes possess the most discriminating power. Such information can lead to optimizations of the feature extraction procedure, which is especially important when acquiring costly or dangerous features.

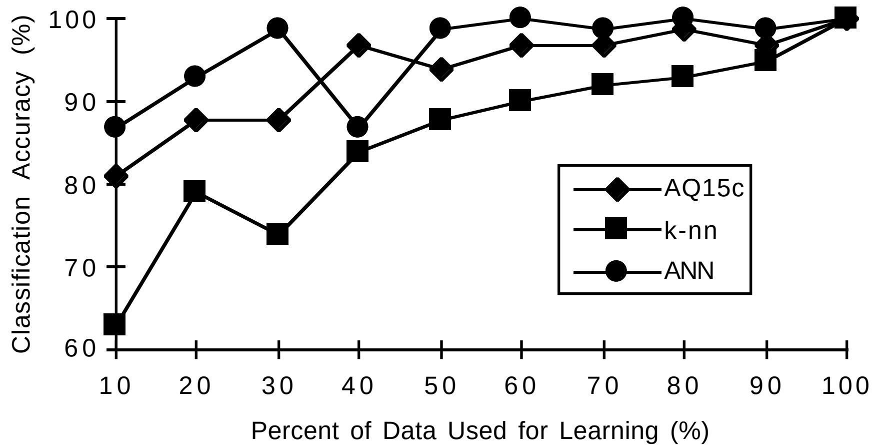

Figure 4: Learning curves for the three classification methods.

Certainly, additional data analysis on the data can be performed to determine which features or attributes are relevant. With AQ, this analysis is a byproduct of learning. Unless additional analysis is performed, we cannot be sure which attributes are relevant for *k*-nn since the relationship between a testing example and a class example is represented as a distance measure. And we cannot be sure with a neural network, since the learned representation is distributed throughout the network connections as real-valued weights.

Future work will investigate how to apply learning to the acquisition of spatial relationships between shapes. Experimentation will investigate how orientation affects the extracted features and how learning copes with such variation. Finally, additional x-ray images have been acquired of blasting caps in luggage.

6 Conclusions

To a large extent, the work presented in this paper is a natural progression of previous research on applying symbolic machine learning to texture recognition. Indeed, the application of machine learning to problems in machine vision can occur at differing levels of representation, and can be used in conjunction with a variety of vision processes, such as model formulation, pose estimation, and segmentation. This paper has demonstrated one such application of machine learning for acquiring symbolic descriptions of 2D shapes for object recognition. Although performance for the neural network model was slightly better than AQ15c, the obvious and intuitive comprehensibility of the learned symbolic descriptions is an important factor that cannot be considered lightly.

References

Baim, P. W. (1984) Automated acquisition of decision rules: the problems of attribute construction and selection. Master's Thesis, University of Illinois, Urbana, IL.

Bala, J. W. (1993) Learning to recognize visual concepts: development and implementation of a method for texture concept acquisition through inductive learning. Ph.D. Dissertation, George Mason University, Fairfax, VA.

Cromwell, R. L., and Kak, A. C. (1991) Automatic generation of object class descriptions using symbolic learning techniques. *AAAI-91*, 710–717.

Fahlman, S. E. (1988) *An empirical study of learning speed in back-propagation networks*. Technical Report CMU-CS-88-182. Department of Computer Science, Carnegie-Mellon University, Pittsburg, PA.

Kerber, R. (1992) ChiMerge: discretization of numeric attributes. *AAAI-92*. 123–128.

Michalski, R. S. (1969) On the quasi-minimal solution of the general covering problem. *Fifth International Symposium on Information Processing*, A3: 125–128.

Michalski, R. S. (1972) A variable-valued logic system as applied to picture description and recognition. *IFIP Working Conference on Graphic Languages*, 21–47.

Michalski, R. S. (1973) AQVAL/1 — computer implementation of a variable-valued logic system VL_1 and examples of its application to pattern recognition. *First International Joint Conference on Pattern Recognition*, 3–17.

Michalski, R. S. (1980) Pattern recognition as rule-guided inductive inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2.4:349–361.

Michalski, R. S.; Rosenfeld, A.; and Aloimonos, Y. (1994) Machine vision and learning: research issues and directions. *Reports of the Machine Learning and Inference Laboratory*, MLI 94–6. Center for Machine Learning and Inference, George Mason University, Fairfax, VA.

Pachowicz, P. W. (1989) Low-level numerical characteristics and inductive learning methodology in texture recognition. *Proceedings of the IEEE Workshop on Tools for Artificial Intelligence*, 91–98. Fairfax, VA.

Quinlan, J. R. (1983) Learning efficient classification procedures and their application to chess end games. In Michalski, R. S.; Carbonell, J. G.; and Mitchell, T. M., eds., *Machine Learning: An Artificial Intelligence Approach*, Vol. 1, 463–482. Palo Alto, CA: Tioga Publishing.

Reinke, R. E. (1984) Knowledge acquisition and refinement tools for the ADVISE META-EXPERT System. Master's Thesis, University of Illinois, Urbana, IL.

Shepherd, B. A. (1983) An appraisal of a decision tree appoach to image classification. *IJCAI-83*, 473–475.

Sydow, P. D., and Cooper, E. G. (1992) Development of a machine vision system for automated structural assembly. *NASA Technical Memorandum* 4366. Langley Research Center, Hampton, VA.

Weiss, S. M. and Kulikowski, C. A. (1992) *Computer systems that learn: classification and prediction methods from statistics, neural nets, machine learning and expert systems*. San Mateo, CA: Morgan Kaufmann.

Wnek, J. (1994) A fast implementation of the AQ-based inductive learning program for large datasets: AQ15c. *Reports of the Machine Learning and Inference Laboratory*, MLI 94–3. Center for Machine Learning and Inference, George Mason University, Fairfax, VA.

Zurada, J. M. (1992) *Introduction to artificial neural systems*. St. Paul, MN: West Publishing.