XCS for Robust Automatic Target Recognition

B. Ravichandran and Avinash Gandhe

Scientific Systems Company Inc.
500 West Cummings Park, Suite 3000
Woburn, Massachusetts 01801
+1 (781) 933-5355

{ravi,avinash}@ssci.com

R. E. Smith

Faculty of Computing, Engineering, and Mathematical Sciences
University of The West of England
Bristol, United Kingdom
+44 117 330 6224

robert.smith@uwe.ac.uk

ABSTRACT

A primary strength of the XCS approach is its ability to create maximally accurate general rules. In automatic target recognition (ATR) there is a need for robust performance beyond so-called *standard operating conditions* (SOCs, those conditions for which training data is available) to *extended operating conditions* (EOCs, conditions of known targets that cannot be foreseen and trained for). EOCs include things like vehicle-specific variations, environmental effects (mud, etc.), unanticipated viewing angles, and articulation of components of the target (hatches, turrets, etc.). This paper presents experiments where XCS addresses structural generalization over global and local features normally used in ATR classification. In many SOCs, these features are adequate for target recognition. Our goal with XCS is to form generalized rules that utilize these features for effective ATR in EOCs. Results show that XCS is effective in this generalization task. Conclusions and future directions for research are discussed.

Categories and Subject Descriptors

J.7 [**COMPUTERS IN OTHER SYSTEMS**] *Military*.

General Terms

Algorithms, Design, Experimentation.

Keywords:

Automatic target recognition, extended operating conditions, structural learning, generalization

1. INTRODUCTION

Automatic Target Recognition (ATR) spans a large space that includes *targets*, *sensors*, and the *environment*. Progress in the field can be sampled in review articles by Nagy [15] in 1968, Kanal [10] in 1974, Bhanu [2], and Jain [9] in 2000. There are mny successful applications of pattern recognition and ATR. However, as evidenced by recent technology demonstrations such as the DARPA MSTAR program, and as is often the case in military applications, the conditions of operation differ from those

from training and are not anticipated. Subsequently there is dramatic reduction in the related ATR performance which does not degrade gracefully. These are referred to as *Extended Operating Conditions* (EOC) [11] and the related technical challenge is the development of *robust* ATR systems [16][21].

Addressing this technical challenge, as shown in Figure 1, the development of a robust ATR system for EOCs has become a confluence of techniques that include (1) data fusion (2) performance evaluation (3) modeling and (4) learning. *Data Fusion* arises from the premise that more than one type of sensor or information can be used for robust ATR. Related challenges include sensor management, data registration etc. *Performance Evaluation* assesses ATR performance independent of specific target, sensor, and environmental conditions and to predict ATR performance or establish performance bounds. *Modeling* is a complement to field data and a surrogate to data that cannot be collected and is based on the rendering of targets, sensors, and the environment. *Learning* addresses the challenge of finding a composite or generalized global model(s) that can account for the variation between the training and testing sets in EOCs.
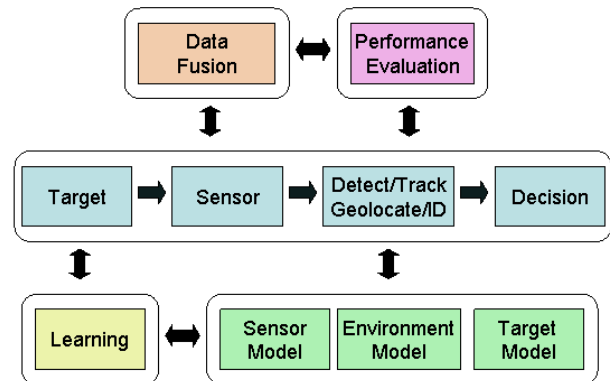

Figure 1: A System for Robust ATR

The focus of this work is the development and demonstration of *learning* for Robust ATR. In this context, learning is the acquisition of general rules that extend to unforeseen EOCs.

In order to put our ideas and work in context, consider:

- The acquisition of rules and heuristics, either offline on online. We will refer to this as the *structural learning* problem.

- The adjustment of action selection parameters, thresholds, and other parameters in the reasoning and

ATR system. We will refer to this as the *parameter learning* problem.

These two complementary problems are discussed as general aspects of any machine learning system and then specifically addressed as part of a robust ATR system.

Within the context of ATR, learning [4] that only considers the parameter learning problem (adaptive adjustment of utility parameters) will be inherently burdened by failure to effectively address the structural learning problem. However, if the structural learning problem is adequately addressed, one can employ multiple models and features in generalized contexts for ATR across multiple EOCs, extending to unforeseen situations. We believe this is the most promising direction for using machine learning to enhance ATR robustness.

The primary innovation of this work is the development of an automated way of developing (heuristic) inference rules that can draw on multiple models and multiple feature types to make more robust ATR decisions.

The key realization is that this "meta learning" problem is one of structural learning, that can be conducted independently of parameter learning associated with each model and feature based technique, and more effectively draw on the strengths of all such techniques, and even information from unforeseen techniques. We accomplish this by using robust, genetics-based machine learning for the ill-conditioned combinatorial problem of structural rule learning, while using statistical and mathematical techniques for parameter learning.

In the sections that follow, Section 2 presents an overview of machine learning. In particular it highlights two interrelated learning problems: structural learning and parameter learning. It then describes *Learning Classifier Systems* (LCS) as an approach for the structural learning problem and then describes XCS as a method for implementing the LCS.

Section 3 describes the LCS-based ATR system developed during the course of this project.

Section 4 describes the results of applying the LCS based ATR system to a number of standard operating and extended operating conditions.

Section 5 provides a summary and conclusion of this work.

2. LEARNING FOR EOCs in ATR

This section presents an overview of machine learning. In particular it highlights two interrelated learning problems: structural learning and parameter learning. It then describes *Learning Classifier Systems* (LCS) as an approach for the structural learning problem and then describes XCS as a method for implementing the LCS.

2.1 A Categorization of Machine Learning

Machine learning is categorized by two interrelated problems, *structural learning problem* and *parameter learning problem*. While these terms are most common in the discussion of Bayesian networks, these two problems are present in all typical machine learning approaches. However they are seldom recognized as ubiquitous.

The *structural learning problem* profoundly impacts the complexity of the parameter learning problem. The number of parameters to tune typically expands exponentially with the addition of the structural elements.

The *parameter learning problem* is typically well treated by straightforward update techniques founded in statistics and the calculus. The structural learning problem is, in general, an ill-conditioned, combinatorial optimization problem. However, the computational burden of overcoming the structural learning problem is often great: an appropriate structure can lead to a computationally efficient implementation of complex, effective knowledge that can be re-used. Moreover, the selection of an appropriate structure can greatly increase the speed and effectiveness of parameter learning.

In all training-data-driven machine learning approaches, some form of the following procedure is used:

1. Initialize structural elements of the knowledge representation (randomly, or based on a priori information)
2. Initialize parameters of those elements (randomly, or based on a priori information)
3. Present one of more items of training data to the algorithm at a time. For each such set:
   (a) Identify a subset of structural elements that are *active* for this training case.
   (b) Update the parameters of the active elements, typically by:
      i. Determining a candidate solution based on the active elements
      ii. Receiving feedback on the quality of this candidate solution (supervision or reinforcement)
      iii. Updating parameters based on this feedback
      iv. Repeating until some criteria is met
   (c) Update structural elements
   (d) Repeat until some criteria is met

Table 1 summarizes the nature of these procedures for typical machine learning approaches. Note that the LCS approach is the only scheme that typically treats the structural learning problem, but also note that the genetic learning paradigm can and has been applied for structural learning in each of the other listed schemes.

Approach	Structural Elements	Parameters	Active Structural Elements	Determination of Candidate Solution	Parameter Update	Structural Update
Neural Networks	Nodes and Connections	Weights	Hidden layer nodes with significant output	Summation at output nodes	Backpropagation	None
Bayesian Networks	Node per event, connections for dependent events	Conditional Probabilities	Events with significant a posteriori probability	Maximum Likelihood Estimation	Conditional probability update from data	None
Rule-based systems	Rules	Conflict Resolution Parameters	Matched Rules	Conflict resolution	Conflict resolution parameters for each rule	None
Decision Trees	Nodes	Thresholds for splitting on data attributes	Active node	Class at active node	Threshold update	Greedy addition of nodes
Function approximation (kernel methods, etc.)	Basis set	Coefficients	Basis functions with significant output	Weighted Summation	Weight update	Arbitrary truncation
Learning Classifier Systems	Generalized Rules	Performance Parameters	Matched Rules	Conflict resolution (or summation) on performance parameters	Performance Parameter Updates	Genetic Learning

Table 1: Typical structural and parameter learning step details in typical, training-data-driven schemes.

3. LCS Based ATR

This section describes the LCS based ATR system developed during the course of this work. It also describes the features used by the LCS. An overview of the classification algorithm is shown in Figure 2. The training data, represented as <condition> <action> pairs are used by the XCS to generate *classification rules*. These rules are then used to match testing data and classify them accordingly into one of the candidate targets.
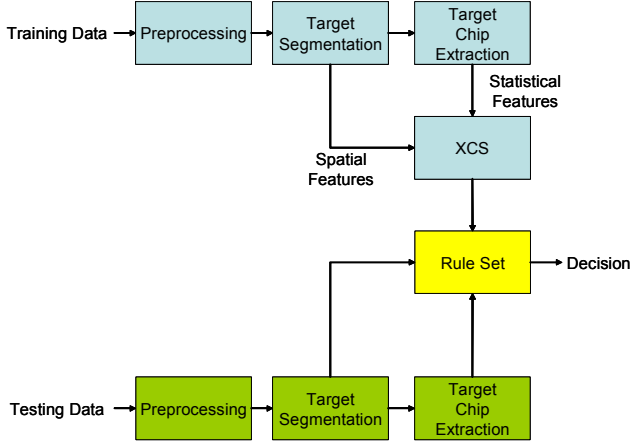

Figure 2: Overview of classification algorithm using XCS

In our work we have used the freely available XCS software from the Illinois Genetic Algorithms Laboratory [20] and tailored it to our particular application. Examples using the XCS to generate classification rules are shown in the following section.

3.1 Feature Representation

In this section we describe the features we use to describe the objects of interest within our Learning Classifier System. We want these features to not only represent the objects of interest concisely, but also to possess sufficient descriptive capabilities to allow the LCS to generalize over this feature space. We chose three types of features, a combination of both global and local features, including features based on the coefficients from Principal Component Analysis (PCA), region-based spatial features, and scattering-center-based features.

3.1.1 Global Features

PCA-based coefficients were chosen as global features to capture information from across the whole image. PCA is a well documented technique [5][6] that draws from the Karhunen-Loève expansion and is based on the concept that high-dimensional data can be succinctly described with a low-dimensional feature vector by projecting the data onto its principal components. For classification, PCA computes a projection matrix based on all the training data. This projection matrix is then used in the testing phase to compute a set of coefficients used for classification in conjunction with a selected distance metric. In our work, when using PCA alone for classification, these coefficients are used with the Mahalanobis distance metric [5,6] defined as:

$$d_k = (c - c_k)^T \mathbf{S}_K^{-1} (c - c_k)$$

where $\mathbf{S}_K$ is the covariance of the weights from class k.. When using XCS as a classifier, the inputs are the differences, d_{ij}, between the distance of a test sample from the training classes, i and j, rather than the coefficients themselves.

Figure 3 shows the discrimination obtained using PCA between two classes, SAM and tank.
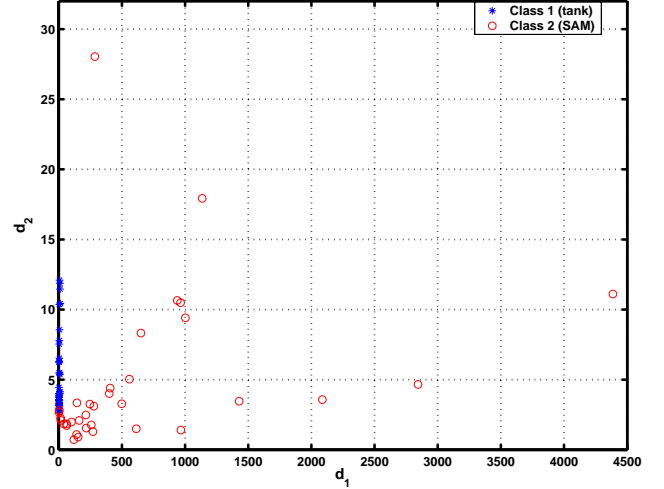

Figure 3: Scatter plot of two targets in PCA-based, distance-from-class space.

The feature space shown is the distance of the test samples from each of the classes based on their PCA coefficients. One can see that the distance, d_1, of the tank testing samples to the tank training class is, in general, small when compared to the same distance computed for the SAM testing samples. Similarly, the distance d_2 is, in general, smaller for the SAM testing samples than for the tank testing samples.

3.1.2 Spatial features

Spatial features capture information pertinent to the object. We derive parameters for the spatial features by segmenting the image and then computing the area, bounding box axis ratio, and extent. Examples of the segmentation for two target chips, a T-72 and a ZSU-23-4, are shown in Figure 4.
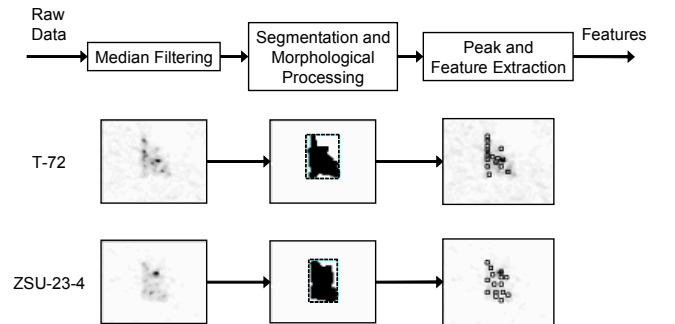

Figure 4: Segmentation and extraction of dominant scatterers in SAR image. Examples are shown for a T-72 tank and ZSU-23-4 anti-aircraft gun.

Both these images are collected at a depression angle of 15 degrees. The figures also show bounding boxes around the targets.

Figure 5 shows a scatter plot for three targets, the T-72 tank, ZSU-23-4 anti-aircraft gun and the BRDM-2 recon vehicle.
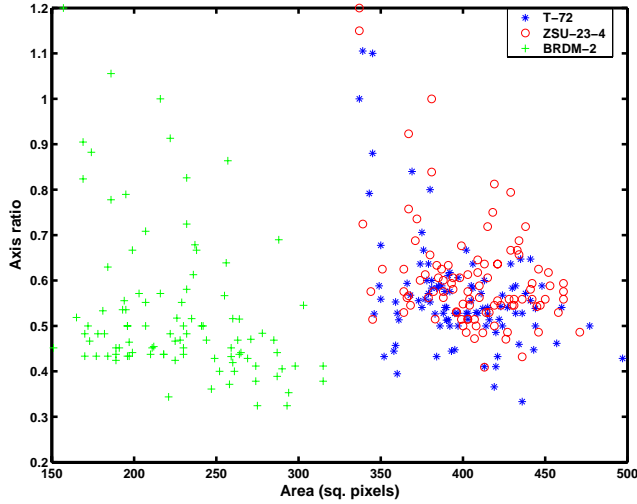

Figure 5: Scatter plot of three targets in Area-Axis Ratio feature space.

The data samples are plotted in (Area, Axis Ratio) feature space. Although there is not much separation between the T-72 and ZSU-23-4, the BRDM-2 is well separated from both the T-72 and ZSU-23-4. This suggests the benefits of using these spatial features in the classifier system.

3.1.3 Intensity Features
Whereas spatial features captured the spatial extent of the object, intensity features capture the localized intensity as represented by the dominant scatterers in the SAR image.

The properties that we encode are the relative location of the dominant scatters in different targets, that is, we encode whether the most dominant scatterers are "close together" or "far apart". We also encode the relative magnitudes of the scatterers for various targets.

The steps followed in extracting this information are shown in Figure 4. The first significant step is segmentation of the image chip (as described in Section 3.1.2) in order to determine the extent of the potential target. A peak extraction algorithm is then employed (only on the target region of the image) in order to extract the dominant scatterers. Examples of this processing shown in Figure 3 for the T-72 and ZSU-23-4 target chips. The scattering center peaks found from the peak extraction algorithm are indicated by the white '□'s overlaid on the original SAR image. Statistics are then obtained for these scatterers. We are currently using the following statistics: The mean (μ_a) and variance (σ_a) of the peak amplitudes, and the mean (μ_{dist}) and variance (σ_{dist}) of the distance of the peak locations from the center of the target.

Although Figure 4 shows the dominant scatterers for two targets, these images by themselves do not indicate any discrimination between the targets based on the peak statistics. Figure 6 shows a scatter plot for three targets, the T-72 tank, ZSU-23-4 anti-aircraft gun and the BRDM-2 recon vehicle.
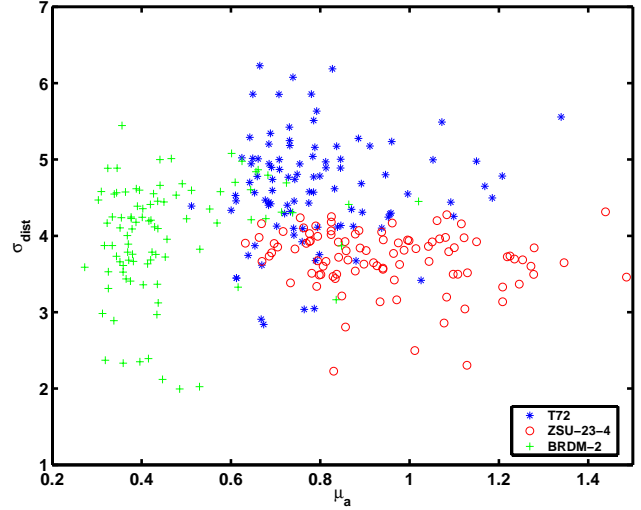

Figure 6: Scatter plot of three targets from Figure 5 in (μ_a, σ_{dist}) feature space

The targets are shown in a two-dimensional feature space corresponding to the (i) mean of the magnitude of the peak (μ_a) and (ii) the variance of the location of the peak (σ_{dist}). The T-72's are represented by the "*", the ZSU-23-4's by the "o" and the BRDM-2's by the "+". One can clearly see that these features do provide a significant level of discrimination. Note that the T-72 and ZSU-23-4 that were not separated by the spatial features in Figure 5 are now separated to a large extent. This shows that the spatial features and scatterer-based features have the potential to compliment each other.

3.2 Baseline for Comparison
We compare our LCS-ATR results to two baselines. One is based on the MSE classifier the other is a "distance"-based classifier.

3.2.1 MSE Classifier
One of the baseline classification algorithms used to assess the performance of XCS is based on the mean square error (MSE) metric [14]. The MSE classifier metric is used to calculate distances to a template of a known target and a testing sample is then assigned to the class represented by the template to which the test sample has minimum distance. The MSE metric is computed as follows:

$$\rho_n = \frac{\vec{x}^t \vec{\mu}_n}{\sqrt{\vec{x}^t \vec{x} \vec{\mu}_n^t \vec{\mu}_n}}$$

$$\mathrm{MSE}_n = \frac{1}{\rho_n^2} - 1$$

where $\vec{x}$ is the "vectorized test chip" (image), $\vec{\mu}_n$ is the vectorized mean template for the class n target, ρ_n is the correlation between $\vec{x}$ and $\vec{\mu}_n$, and MSE_n is a measure of distance to class n.

3.2.2 Distance Classifier

The PCA results in coefficients as features sets. Once we compute these features sets, we need tools to classify them. Typically, various distance measures are used. We use the Mahalanobis Distance.

As a preliminary illustration of the structural generalization that XCS can afford over-and-above the features made available by distance classification, see Figure 7.
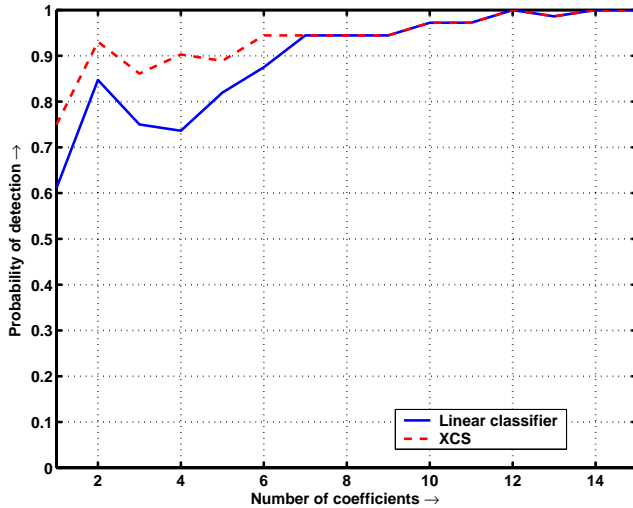

Figure 7: Overall classification rate for two simulated SAR targets versus number of PCA coefficients for a Mahalanobis distance based classifier and an XCS that generalizes over distances from PCA prototypes.

In this case, distances from the two PCA prototypes are used as a feature for the LCS, encoded as six bits for each distance. Results show that the generalization ability of the XCS sustains classification performance as the dimensionality of the prototype space is lowered. It is this generalization ability that we hope to exploit in larger feature spaces, to extend normal classification performance with these features beyond SOCs into EOCs. EOC results are presented in the following sections.

4. Test and Evaluation

This section describes the results of applying the LCS-based ATR system as shown in Figure 2 to one standard operating condition (SOC) and a number of extended operating conditions (EOCs). Note that there is an intentional departure from typical training and testing procedures. In the EOC cases, we are intentionally attempting to generalize for *unforeseen* cases. Therefore, we will be testing on conditions that are not in the training set. This places a premium on obtaining structural generalizations from the training data that transfer themselves effectively to the unforeseen testing conditions. In general, a SOC refers to training sets and testing sets that are similar, and an EOC refers to training sets and testing sets that are variant.

In context of SAR ATR, the EOCs arise from target, sensor, and environmental variations. Target related variations include serial number, version, configuration, articulation, sensor or acquisition

related variations include depression angle and aspect angle, and environment related variations include background types, atmospheric effects, obscuration, camouflage, deception [11][14].

The SAR data used in this project is from the DARPA-sponsored MSTAR program's public distribution. This is real SAR data. For the results presented in this paper, the EOCs were chosen depending on those available in the MSTAR Public Release target set. These included target-related variations (serial number, articulation) and sensor or acquisition-related variations (depression angle, aspect angle).

In the results that follow, we begin with performance for nominal conditions (Section 4.1) and then describe the results for a number of EOCs that include serial number (Section 4.2), aspect angle (Section 4.3), depression angle (Section 4.4), and articulation (Section 4.5).

For each case, we first describe the SAR data used for training and testing and then present the classification results using a confusion matrix. Along with the classification results for the machine learning based ATR system, we also present the results for MSE classifier (Section 3.2.1) and a distance classifier that uses PCA coefficients as features (Section 3.2.2) as baselines for comparison. In describing the results, we will comment on the *accuracy*, the performance on the training set and the *robustness* the performance on the testing set.

4.1 Nominal conditions

In this section we investigate the performance of the XCS under nominal conditions with multiple targets. In order to do so, we design an experiment with 4 targets. The targets under training and testing scenarios are shown in Table 2. One can see that in this case, the training data is very similar to the testing data, in that they are both acquired at depression angles of 15 degrees, contain data samples at all aspect angles, and have the same articulation and serial numbers.

Target	Training Data		Testing Data	
	Depression Angle	**No. of samples**	**Depression Angle**	**No. of samples**
T-72	15 degrees	100	15 degrees	100
ZSU-23-4	15 degrees	100	15 degrees	100
2S1	15 degrees	100	15 degrees	100
BRDM-2	15 degrees	100	15 degrees	100

Table 2: Training and Testing dataset for nominal conditions.

The percentage of correctly classified targets obtained using the MSE classifier, PCA distance classifier and XCS are shown in Table 3. We use the MSE classifier and PCA distance classifier as baselines and compare the XCS results to them. The results are shown in the form of a confusion matrix with each entry showing how many of the samples from the target indicated by row label are classified as the target indicated by the column label.

	MSE training				MSE testing			
	T-72	ZSU	2S1	BRDM	T-72	ZSU	2S1	BRDM
T-72	82	2	12	4	79	2	10	9
ZSU	12	86	1	1	24	74	1	1
2S1	5	10	67	18	6	18	58	18
BRDM	0	0	24	76	1	1	27	71
	PCA training				PCA testing			
T-72	94	1	5	0	86	8	6	0
ZSU	1	99	0	0	1	99	0	0
2S1	2	3	95	0	0	14	86	0
BRDM	2	7	11	80	5	11	23	61
	XCS training				XCS testing			
T-72	100	0	0	0	92	3	5	0
ZSU	3	96	1	0	5	88	6	1
2S1	0	0	100	0	3	1	94	2
BRDM	0	0	0	100	2	1	9	88

Table 3: Classification rates (as % of total samples) under nominal conditions

As expected, the PCA and XCS classifiers perform well on the training set. We will mainly discuss the results on the testing set, since these results are true indicators of accuracy and robustness of the classifier. For the testing set, one can see that most of the targets are correctly classified by the PCA and XCS, which both outperform MSE. However, XCS outperforms PCA by a significant margin on three of the four targets, most noticeably on the BRDM-2. The overall rates of correct classification are: MSE = 70.5%, PCA = 83% and XCS = 90%.

4.2 EOC: Serial number

The MSTAR dataset includes the same type of target with data collected from different serial numbers. In this experiment, we are attempting to make a first generalization over the serial number EOC. The, we train the XCS using one serial number from each target and test on a different serial number. The training and testing data are described in Table 4.

	Training Data			Testing Data		
Target	Serial No.	Dep. Angle	Samples	Serial No.	Dep. Angle	Samples
T-72	Sn132	15 degrees	200	Sn812	15 degrees	200
BMP-2	Sn9563	15 degrees	200	Sn9566	15 degrees	200

Table 4: Training and Testing dataset for serial number EOC condition

Table 5 shows the classification results using both PCA and XCS. PCA seems to have trouble correctly classifying the BMP-2 even in the training data, while XCS perfectly classifies all the training samples. On the testing set, while the XCS and PCA each seem to favor different targets, overall the XCS performs better than PCA. Both these classifiers outperform the baseline MSE classifier. Overall, the rates of correct classification are: MSE = 69.75%, PCA = 81% and XCS = 85.5%.

	MSE training		PCA training		XCS training	
	T-72	BMP-2	T-72	BMP-2	T-72	BMP-2
T-72	83	17	99	1	100	0
BMP-2	22.5	77.5	17	83	0	100
	MSE testing		PCA testing		XCS testing	
T-72	65	35	89.5	10.5	80.5	19.5
BMP-2	25.5	74.5	27.5	72.5	9.5	90.5

Table 5: Classification rates (as % of total samples) under the serial number EOC

4.3 EOC: Aspect angle

In this experiment we test the robustness of the classifiers to varying target acquisition aspect angle. In all data files in the MSTAR database, a target is acquired from a number of aspect angles from 0 degrees to 360 degrees. As shown in Table 6, in this experiment we train the classifiers on data collected with aspect angles between 0 degrees and 180 degrees and test the classifiers on aspect angles between 180 degrees and 360 degrees. Four targets are used, namely, T-72, ZSU-23-4, 2S1 and BRDM-2.

	Training Data			Testing Data		
Target	Aspects	Dep. Angle	Samples	Aspects	Dep. Angle	Samples
T-72	0 degrees - 180 degrees	15 degrees	100	180 degrees - 360 degrees	15 degrees	100
ZSU-23-4	0 degrees - 180 degrees	15 degrees	100	180 degrees - 360 degrees	15 degrees	100
2S1	0 degrees - 180 degrees	15 degrees	100	180 degrees - 360 degrees	15 degrees	100
BRDM-2	0 degrees - 180 degrees	15 degrees	100	180 degrees - 360 degrees	15[degrees	100

Table 6: Training and Testing dataset for the aspect angle EOC

Table 7 shows the classification results. The MSE baseline seems to perform particularly poorly on most targets in this EOC. On the testing data, the performance of PCA and XCS is similar on the T-72, ZSU-23-4 and 2S1 targets. However, on the BRDM-2, XCS has a significantly better classification rate. Overall, XCS's classification rate of 85% compares favorably with PCA's classification rate of 80.75 %.

	MSE training				PCA training				XCS training			
	T-72	ZSU	2S1	BRDM	T-72	ZSU	2S1	BRDM	T72	ZSU	2S1	BRDM
T-72	86	1	13	0	97	2	1	0	99	0	1	0
ZSU	14	85	0	1	0	100	0	0	0	97	0	3
2S1	2	3	87	8	0	1	99	0	0	0	100	0
BRDM	1	0	20	79	1	2	8	89	0	0	0	100
	MSE testing				PCA testing				XCS testing			
T-72	36	1	0	63	87	5	8	0	89	3	8	0
ZSU	26	52	16	6	0	100	0	0	1	97	2	0
2S1	17	16	38	29	11	8	81	0	14	7	77	2
BRDM	16	3	1	80	7	17	21	55	3	7	11	77

Table 7: Classification rates (as % of total samples) under the aspect angle EOC.

4.4 EOC: Depression angle

In this experiment we consider another acquisition-related EOC - the depression angle of target with respect to the sensor. The MSTAR database contains data from five targets at depression angles of 15 degrees, 17 degrees, 30 degrees and 45 degrees. In this section, four targets are considered as shown in Table 8. The data at 15 degrees is used to train the classifiers and testing is carried out on the data collected at 30 degrees.

Target	Training Data		Testing Data	
	Dep. Angle	Samples	Depression Angle	Samples
T-72	15 degrees	100	30 degrees	100
ZSU-23-4	15 degrees	100	30 degrees	100
2S1	15 degrees	100	30 degrees	100
BRDM-2	15 degrees	100	30 degrees	100

Table 8: Training and testing dataset for depression angle EOC

The classification results of Table 9 show that PCA outperforms the XCS on the two targets - the ZSU-23-4 and the 2S1. However, the difference in classification rates for the BRDM-2 is quite large and is in favor of XCS. This in fact leads to an overall higher classification rate for XCS by 1.5%. Overall, the rates of correct classification are: MSE = 52.25%, PCA = 71.5% and XCS = 73%.

	MSE training				PCA training				XCS training			
	T-72	ZSU	2S1	BRDM	T-72	ZSU	2S1	BRDM	T72	ZSU	2S1	BRDM
T-72	82	2	12	4	94	1	5	0	100	0	0	0
ZSU	12	86	1	1	1	99	0	0	2	98	0	0
2S1	5	10	67	18	2	3	95	0	0	0	100	0
BRDM	0	0	24	76	2	7	11	80	0	0	0	100
	MSE testing				PCA testing				XCS testing			
T-72	53	2	18	27	71	9	20	0	77	6	13	4
ZSU	14	56	17	13	7	71	22	0	23	57	16	4
2S1	1	10	59	30	1	7	92	0	8	2	81	9
BRDM	0	0	19	41	10	2	36	52	9	0	14	77

Table 9: Classification rates (as % of total samples) for the depression angle EOC.

4.5 EOC: Articulation

The MSTAR database has data for three targets under nominal and articulated conditions. In this experiment, we train the classifiers under nominal conditions and test under articulated conditions (Table 10).

Target	Training Data			Testing Data		
	Articulation	Dep. Angle	Samples	Articulation	Dep. Angle	Samples
T-72	nominal	30 degrees	100	articulated	30 degrees	100
ZSU-23-4	nominal	30 degrees	100	articulated	30 degrees	100
BRDM-2	nominal	30 degrees	100	articulated	30 degrees	100

Table 10: Training and Testing dataset for the aspect articulated EOC.

The classification results, presented in Table 11, show that the PCA almost entirely mis-classifies the BRDM-2 samples. The XCS however shows considerable improvement over the PCA and correctly classifies 40 % of the BRDM-2 samples in testing set. However, for this particular target, the MSE outperforms both the XCS and PCA. MSE, in fact, performs surprisingly well in this experiment. However, XCS does significantly outperform the MSE on T-72 and ZSU-23-4, which is encouraging. For the BRDM-2, we believe the classification results for the XCS are affected by its use of the PCA-based features. This is indicated by the complete misclassification of the BRDM-2 by the PCA classifier. Although XCS turns off some of these features, it is unable to completely ignore them. The overall classification rates are : MSE = 76 %, PCA = 61.33 % and XCS = 74.67%.

	MSE training			PCA training			XCS training		
	T-72	ZSU	BRDM	T-72	ZSU	BRDM	T-72	ZSU	BRDM
T-72	91	8	1	98	2	0	100	0	0
ZSU	19	81	0	0	100	0	0	99	1
BRDM	1	2	97	6	25	69	0	0	100
	MSE testing			PCA testing			XCS testing		
T-72	74	25	1	85	15	0	87	13	0
ZSU	20	80	0	2	98	0	3	97	0
BRDM	9	17	74	14	85	1	6	54	40

Table 11: Classification rates (as % of total samples) for the articulation EOC.

4.6 Results Summary

The overall results for each ATR system (MSE,PCA, and XCS) applied to nominal conditions and EOCs are shown in Table 12. Results show that the XCS's generalizations are consistently effective in extending the utilization of the features we have used to the EOCs. Since strong generalization is the only hope for effectively coping with EOCs (which are unforeseen, by definition), this indicates the promise of XCS in this ATR domain.

	MSE	PCA	XCS
Nominal	70.5	83	90
Serial EOC	69.75	81	85.5
Aspect EOC	51.5	80.75	85
Depression EOC	52.25	71.5	73
Articulation EOC	76	61.33	74.67

Table 12: Summary of Testing Classification Rates.

5. Summary, Conclusions, and Future Work

The primary innovation of this work is the development of an automated way of developing heuristic inference rules that can draw on multiple models and multiple feature types to make more robust ATR decisions. This system was tested on MSTAR Public Release SAR data using nominal and extended operation conditions. These results were also compared against two baseline classifiers, a PCA-based distance classifier and a MSE classifier.

In the experiments presented here, the LCS-based ATR system generally exhibited performance with higher accuracy (in most

cases 99%) and better robustness (in most cases over 80%). By exploiting XCS's ability to generalize, we are able to show results that effective on SOCs (for which the system was trained) and EOCs (for which no training data was presented). This generalization is key for robust ATR in the field.

Future areas for investigation include further test and evaluation, particularly with more targets and EOCs. This work presented results based on the MSTAR public release data. Our next step is to expand this to include all the MSTAR data, resulting in a larger number of targets, target types, and EOCs.

Acknowledgements

References

[1] T. Back, D. B. Fogel, and Z. Michalewicz, Z. (Editors) *The Handbook of Evolutionary Computation*, Oxford University Press, 1997

[2] B. Bhanu, "Automatic Target Recognition: State-of-the-Art Survey", *IEEE Transactions on Aerospace and Electronic Systems*, AES Vol. 22, No. 4, pp. 364-379, 1986.

[3] K. A. De Jong, W. M. Spears, and D. F. Gordon, "Using Genetic Algorithms for Concept Learning", *Machine Learning* (13), 1993.

[4] J. Diemunsch, J. Wissinger, R. Ristroph, E. Fruedenthal,W. Severson, *MSTAR's Extensible Search Engine and Model-based Inferencing Toolkit* Technical Report Number ASC-99-0746. Air Force Research Laboratory, 1999.

[5] R. Duda, P. Hart, D. Stork, *Pattern Classification*, 2nd Edition, Wiley, 2001.

[6] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press Inc., New York NY, 1972.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[8] J. H. Holland, *Adaptation in Natural and Artificial Systems* MIT Press, 1992

[9] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical Pattern Recognition: A Review", *IEEE Transaction on Pattern Analysis and Machine Intelligence* 22, 4-37, 2000.

[10] L. Kanal, "Patterns in Pattern Recognition" *IEEE Transactions Information Theory* IT-20, November, 1974.

[11] E. R. Keydel, S.W. Lee, J.T. Moore, "MSTAR Extended Operating Conditions: A Tutorial", In *Algorithms for synthetic Aperture Radar Imagery IV*, SPIE Volume 2757, April 1997.

[12] P. L. Lanzi, W. Stolzmann, and S. W. Wilson (Editors) *Advances in Learning Classifier Systems* Lecture Notes in Artificial Intelligence, pp. 103-113. Springer Verlag, 1996

[13] P. L. Lanzi, W. Stolzmann, and S. W. Wilson (Editors) *Learning Classifier Systems: From Foundations to Applications*, pp., 63-81, Springer Verlag, 2000.

[14] J.C. Mossing and T.D. Ross, "An evaluation of SAR ATR algorithm performance sensitivity to MSTAR Extended Operating Conditions," Proc. of SPIE, Algorithms for Synthetic Aperture Radar Imagery V, Vol. 3370, pp.554-565, April 1998, Orlando, FL.

[15] G. Nagy, "State of the Art in Pattern Recognition", *Proceedings of the IEEE*, Vol 56 pp. 836–862, 1968

[16] T. Ross and and L. Westerkamp and E. Zelnio and T. J. Burns "Extensibility and Other Model-Based ATR Evaluation Concepts" *Algorithms for Synthetic Aperture Radar Imagery IV* SPIE 3070, 1997

[17] R. E. Smith, B. A. Dike, B. Ravichandran, A. El-Fallah, and R. K. Mehra, "Discovering Novel Fighter Combat Maneuvers in Simulation: Simulating Test Pilot Creativity" In P. J. Bentley and D. W. Corne (Editors), *Creative Evolutionary Systems* pp. 467-486. Morgan Kaufmann, 2001.

[18] S. W. Wilson, "ZCS: A Zeroth-Level Classifier System", *Evolutionary Computation* 2(1). pp. 1-18, 1994.

[19] S. W. Wilson, "Classifier Fitness based on Accuracy", *Evolutionary Computation* 3(2). pp. 149-175, 1994.

[20] ftp://ftp-illigal.ge.uiuc.edue/pub/src/XCS/XCS1.2.tar.Z

[21] E. Zelnio and F. Garber and L. Westerkamp and S. Worrell and J . Westerkamp and M. Jarratt and C. Deardork and P. Ryan "Characterization of ATR Systems" *Algorithms for Synthetic Aperture Radar Imagery IV* SPIE 3070, 1997