Optimization of support vector machine (SVM) for object classification

Matthew Scholten[a], Neil Dhingra[b], Thomas T. Lu[1c], Tien-Hsin Chao[c]

[a]California State University, Long Beach, CA, USA; [b]University of Michigan, MI, USA;
[c]Jet Propulsion Laboratory/California Institute of Technology, Pasadena, CA, USA

ABSTRACT

The Support Vector Machine (SVM) is a powerful algorithm, useful in classifying data in to species. The SVMs implemented in this research were used as classifiers for the final stage in a Multistage Automatic Target Recognition (ATR) system. A single kernel SVM known as SVMlight, and a modified version known as a SVM with K-Means Clustering were used. These SVM algorithms were tested as classifiers under varying conditions. Image noise levels varied, and the orientation of the targets changed. The classifiers were then optimized to demonstrate their maximum potential as classifiers. Results demonstrate the reliability of SVM as a method for classification. From trial to trial, SVM produces consistent results.

Keywords: Support Vector Machine (SVM); Automatic Target Recognition (ATR); neural network; false alarm reduction.

1. INTRODUCTION

Computer vision is a unique field of study pertaining to the development of systems that automatically draw useful information from images. The information encoded in an image can be valuable in many applications, allowing a computer to understand an environment on a visual level, to recognize objects, and to perform operations autonomously. Applications of computer vision range from medical image analysis to autonomous guidance and maneuvering of spacecraft. Automatic Target Recognition (ATR), the ability of a computer vision system to recognize, and identify specific objects is of particular importance, and of substantial difficulty. Targets must be identified under varying conditions, and in various orientations with minimum false alarms. Furthermore, ATR algorithms must be computationally efficient if they are to operate in real time.

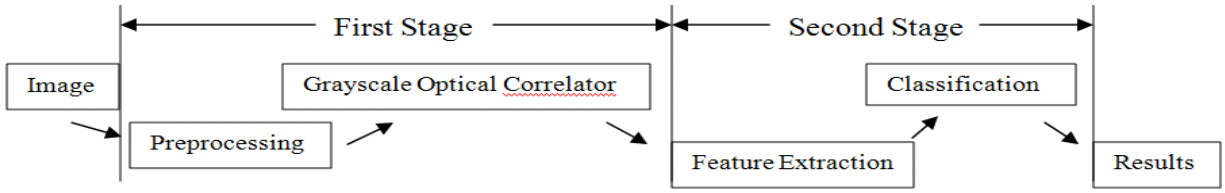

Figure 1: System block diagram of a two-stage ATR system: First stage: pre-processing and target detection using correlations; Second stage: Feature extraction and classification.

[1] e-mail: Thomas.T.Lu@jpl.nasa.gov , Tel: (818) 354-9513, Fax: (818) 393-4272

To address the issue of speed versus accuracy in the ATR system, Jet Propulsion Laboratory has developed a multistage ATR system that quickly scans an image for potential targets, and subsequently weeds out misclassified targets [1-3]. As illustrated in Figure 1, the input image is first preprocessed to enhance the image quality.

To quickly identify subsets of the image that contain targets, the ATR system uses a Grayscale Optical Correlator (GOC) in combination with an Optimum Trade-off Maximum Average Correlation Height (OT-MACH) filter [4-5]. These subsets are considered Regions of Interest (ROIs) and serve to reduce the amount of data that is passed to the final stage of the ATR system. To reduce the dimensionality of the data being classified, the Principal Component Analysis (PCA) is used to further reduce the ROI into 18 features. The final stage of the ATR system uses a machine learner to classify each previously extracted feature as a target, or non-target. This paper documents the use of two types of Support Vector Machines that are used as final classifiers in an automatic target recognition system [6].

2. METHODOLOGY

In an ATR application, a classifier is very important in ensuring the correct identification of the true targets while rejecting false positives. We discuss the optimization of the SVM as a classifier. In our experiment, a grid-search was performed on an interval of possible parameter values to find the optimal setting for the SVM kernel parameter γ, and the regularization parameter C. The interval [a,b] was divided evenly into n steps for each of the SVM parameters. At each of the subsequent steps, parameter values were set, and the SVM classifier was tested. Once optimal parameters were selected, the process was repeated with the grid-search interval centered at the previously selected best value. This time the search interval was reduced, and the resolution of the grid search was increased. This sweep search was performed on SVM, and K-means Clustering with SVM (K-Means SVM) classifiers to determine the best possible fit during classification.

Once optimal parameters were found, testing was done to see how precise the parameter selection needed to be. The optimal parameters were tested and performance scores were recorded. The highest performing parameters were used as baseline values to measure the degree of precision necessary when setting C, and γ. To test nessecary precision, the optimal C, and γ were set, and alternately, each parameter was modified. This process was repeated multiple times, modifying the parameters each time by a factor of two.

2.1 Support Vector Machine Classification

In order to distinguish classes in highly enmeshed data, as can be the case with ATR, Support Vector Machines examine a set of labeled instance pairs in a multidimensional feature-space. SVM attempts to separate the data in to classes by creating a separating hyperplane along the maximal margin of class separation [7], as illustrated in Fig. 2.

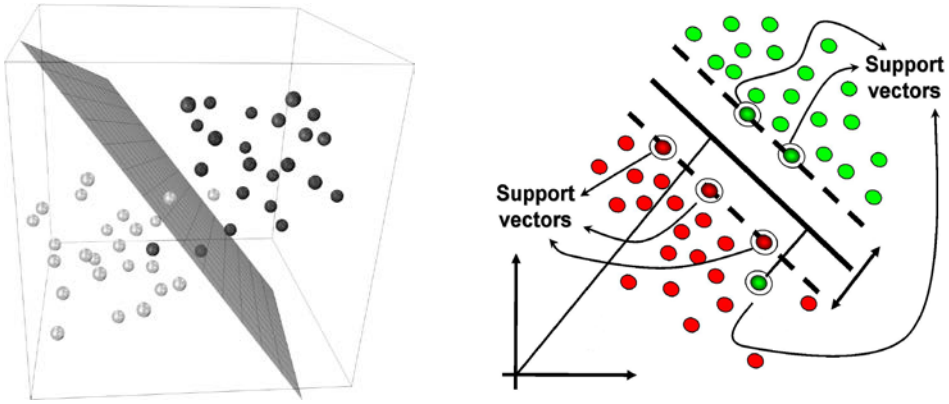

Figure 2: Illustration of the principles of SVM: SVM separates classes along their maximal margin of separation. Support vectors are selected to set the maximal margin between class clusters. [8,9]

The number of features representing each labeled data point determines the dimensionality of feature-space. In our

sonar objects ID case, 18 feature vectors were selected through PCA. Thus an 18 dimensional feature-space is used. Mapping of feature-space is determined by the SVM kernel, and is optimized by maximizing the class separation of data [2]. The kernel is selected based on the conditions of input data. Kernel parameters are optimized for the best 'fit'. In this case, a Gaussian Radial Basis Function was used as the SVM kernel function.

Given a training set of instance-label pairs (x_i , y_i), i = 1, . . ., l , where x_i □R^n is the set of training vectors, and y □ R such that y_i □$[1,-1]^l$, are the two-class labels , SVM requires solving the following optimization problem [10]:

$$\min_{\omega, b, \xi} \left(\frac{1}{2}\omega^T \omega + C \sum_{i=1}^{l} \xi \right) \tag{1}$$

subject to

$$y_i (\omega^T \varphi(x_i) + b) \geq 1 - \xi_i , \tag{2}$$

and

$$\xi_i \geq 0, \ \ i = 1, \ldots, l, \tag{3}$$

where ω is the weight parameter, b is the bias, and the kernel function $\varphi(x_i)$ maps x_i into a higher-dimensional space. ξ is the error penalty, C > 0 is the regularization parameter for the error penalty, and the decision function is

$$sgn(\omega^T \phi(x) + b) = sgn(\sum_{i=1}^{l} y_i \alpha_{iK}(x_i, x_j) + b) \tag{4}$$

The kernel function:

$$K(x_i, x_j) \equiv \varphi(x_i)^T \varphi(x_j). \tag{5}$$

Optimizing the Radial Basis Function:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \ \ \gamma > 0. \tag{6}$$

With RBF kernel parameter γ optimized to set the kernel width. Tightening the spread of instances in regions of dense species population, and expanding the spread where population is sparse. [10]

2.2 k-Means Clustering with SVM Classification

k-Means clustering with SVM (k-SVM) classification places the instance-pairs of each class into sub-classes depending on the distance between them in feature-space. The user specifies a number of classes, and the algorithm attempts to form the best 'clusters'. The algorithm *uses a point as a 'mean' to define each class. Each point in the data set is classified according to which data point it is closest too. k-Means clustering* randomly places means in the feature space and then sorts the data into clusters based on distance from the mean. Each cluster's mean is then moved to the average of all the points in that cluster. This is repeated until the means no longer move. Because of the random initial placement of means, the algorithm can converge to different clusters. To counteract this, the process is repeated several times and the clustering of data which best reduces entropy in the clusters is chosen. [11]

2.3 SVM Parameters

To control the fit of data. In the case of the Gaussian RBF kernel used here, the optimization of the kernel parameter γ is required. The SVM classifier also requires the optimization of the parameter 'C' which sets the penalty for

misclassifying a point. The parameter C relocates a sample point closer to the center of its class. The particular C, and γ values that are best for a given application are not known, so they must be set prior to operation. The common approach for SVM parameter optimization is for arbitrary parameter values to be tested, and the best of the set to be used during runtime. [8]

3. EXPERIMENTAL RESULTS

We used a set of sonar images as our test images. Performance was measured using a Free -Response Operating Characteristics (FROC) curve. The FROC curve displays the relationship between percent accuracy, and total number of false positive target detections per image, as shown in Figure 3. When selecting the optimal C, and γ parameters, the grid search algorithm tested the SVM classifiers at each iteration, and displayed their FROC curves. This method reavealed the performance of the SVM Classifiers along the range of possible parameter values. Increasing the number of iterations, and decreasing the search interval determined the nessecary resolution of a grid search for parameters.

This testing illustrates the ability of SVM, and K-Means SVM to detect objects in sonar images. The necessary range of a search for SVM parameters was demonstrated. Necessary precision during the parameter search was shown to be fairly low. Both SVM classifiers proved to be fairly robust when C, and γ were set to within 10% of their optimal value. During K-Means testing, there was some degree of fluctuation in performance which can be attributed to the random method of locating K-Means between sub-classes. During each iteration, performance of the K-Means SVM classifier fluctuated by up to 25%.

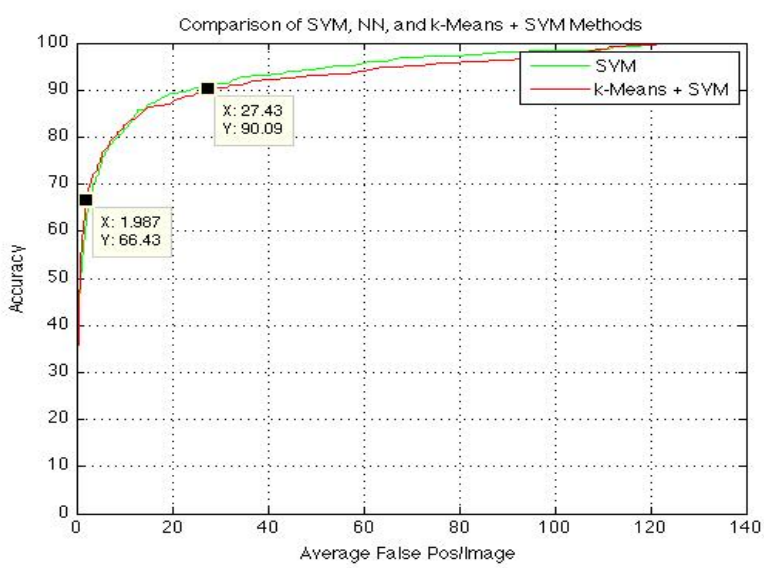

Figure 3: Example of the the Free-Response Operating Characteristics (FROC) curve. Above, displaying the detection accuracy vs. average false positives per image of the SVM and k-Means SVM in classifying objects.

3.1 Support Vector Machine - Classification Performance

We first perform a coarse grid search of the C, and γ values. Initial C, and γ values for the SVM classifier were set according to Table I, and baseline performance measurements were obtained to compare subsequent optimized parameters. These values were obtained by performing an initial search for the parameter's value. When C = 4.72 and γ = 0.96, the system performance is 61.04% accuracy of correct object detection at 2 false positives per image level.

Table I: Initial SVM parameters – baseline values, and performance

C	γ	Percent Accuracy @ 2 False-Positives per image	Number of False-positive per image @ 90% accuracy
4.72	0.96	61.04	24

As shown in the experiment in Figure 4, we sweep the γ value between [0, 30]. The optimal γ parameter was obtained around 1.9. The system performace improved from the inital 61.04% to ~66% at 2 false positives per image. Figure 4 shows that a larger than nessecary γ causes diminished classifier performance.

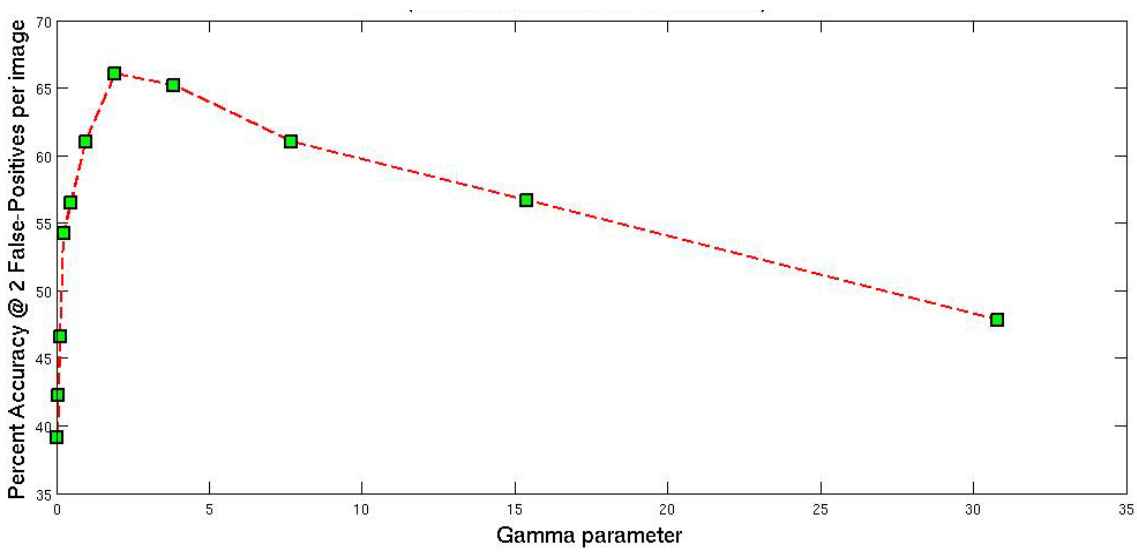

Figure 4: As the Kernel Parameter γ is modified, a change in performance is noted. The optimal γ value is ~2.3. The performance improved from 61% to 66%. The performace degraded when γ value deveated from the optimum value

We then optimized the C parameter for error penalty. As shown in Figure 5, the performance of the SVM classifier improved when we optimized the C parameter. The performace reached a threshold around 70% accuracy when the C parameter reached its optimal value ~ 20. Performance stayed fairly consistantly with an increasing parameter value, but choosing a higher than necessary C value reduced the classifier's ability to generalize to new data. This result illustrates the required grid search range when sweeping for SVM parameters. A search for C parameter values larger than 20 were shown not to be benificial.

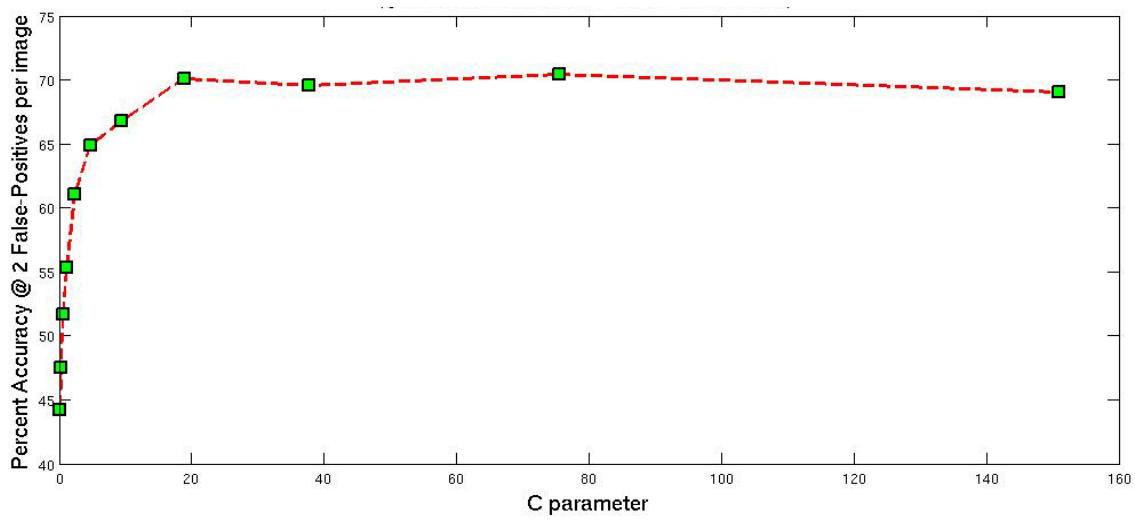

Figure 5: Optimizing the performance as the Regularization Parameter C is modified, an improvement in performance to 70% accuracy is noted. The C value is optimum around 20.

Once the optimal values for C, and γ were obtained, the grid search resolution was increased substantially around the optimal values. Testing on our sonar images showed that the modification of each SVM parameter of up to 10% could cause a change in performance of 1% to 2% on this data set.

3.2 K-Means Clustering With Support Vector Machine - Classification Performance

The K-Means SVM classifier uses two SVM classifiers together to attempt to refine the placement of the separating hyperplane between classes of instance pairs. Because there are two classifiers operating, there are a total of four SVM parameters to be set. Figure 6 shows the optimization of γ value in the range of [0, 30]. The optimal value is γ = 0.5 at ~ 65% accuracy.

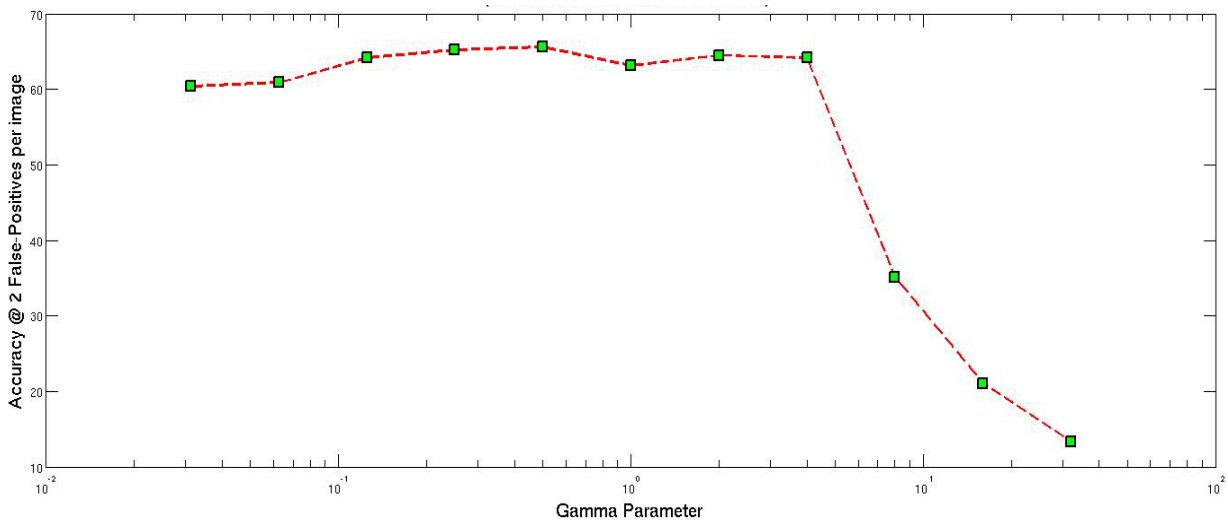

Figure 6: Optimizing the K-means SVM: As the Kernel Parameter γ is modified, a change in performance is noted. The accuracy reaches ~65%.

As shown in Figure 7, when the C parameter was changed for the K-Means classifier, the overall performance

of the classifier was recorded. There was some degree of variation in performance with each iteration. Shown below in Figure 7 are performance recordings that fluctuate between 58 and 68 percent accuracy with 2 false positives per image. The optimal C value is C = 12 at 67.3% accuracy. We can see the fluctuation of the performance when C parameter is changed.

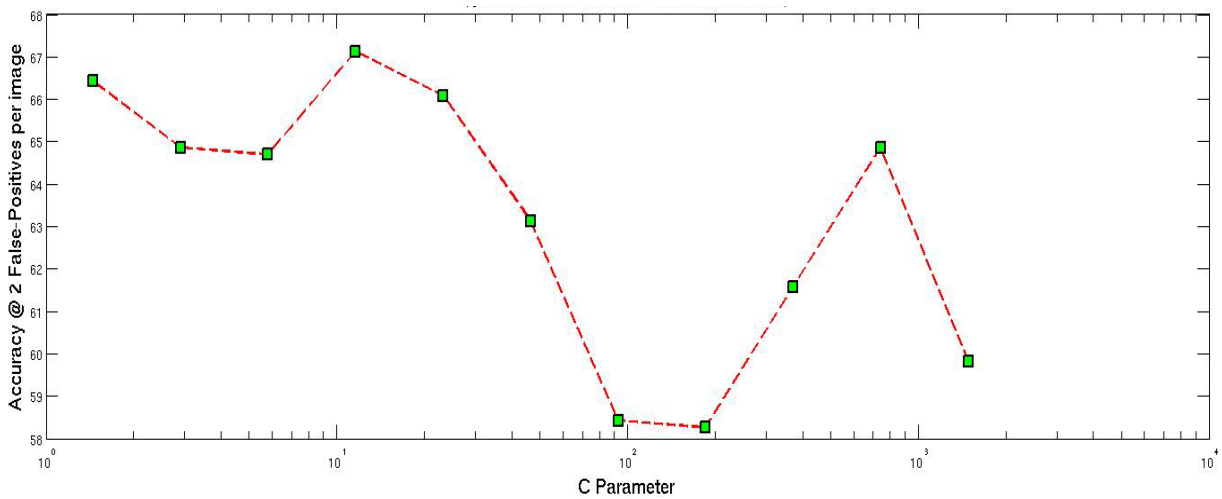

Figure 7: As the Regularization Parameter C is modified, a change in performance is noted.

We also optimize the second SVM parameters in the K-Means SVM. The C and γ parameters are searched to find the optimal performance. Figures 8 and 9 show the optimization of the accuracy through grid searches of the C and γ parameters.

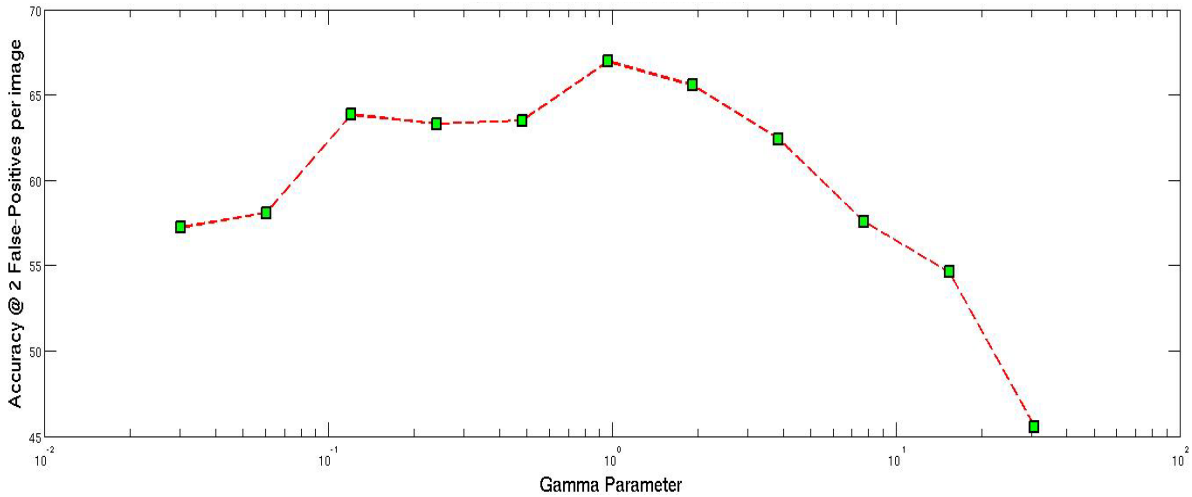

Figure 8: Optimization of the γ parameters. The best performance is achieved when γ = 0.96 at ~66% accuracy.

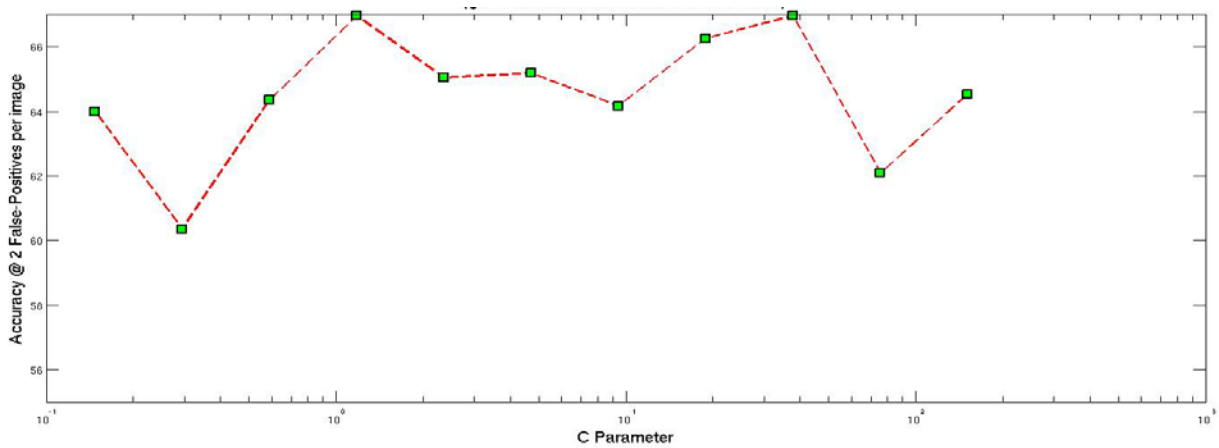

Figure 9: As the Regularization Parameter C is modified, we see multiple optimal C values. A smaller C paarmeter is preferred, C = 1.2 at ~ 7=67% accuracy.

The K-Means SVM seemed to benefit from a grid search on the C and γ parameters for each of its classifiers. The charts above demonstrate the performance of the classifier over all. Notice that a γ that is too high reduces performance substantially.

4. CONCLUSION

The SVM classifier produces consistent results, and is easily optimized. Optimization of SVM parameters increases performance, and is efficiently done using a grid search. When tested multiple times with the same kernel parameters SVM produced identical classification accuracy, while K-Means SVM results tend to vary. K-Means SVM seems to return useful performance measures when sweeping for the γ parameter. This suggests that a grid search for the K-Means SVM γ parameter is ueful. Sweeping for the C parameter returned varied performance results, all of which were between 58 and 68 percent accuracy with an average of 2 false positives per image. This complexity made optimization of K-Means SVM difficult to recognize. Testing showed that a grid search for the SVM parameters is a significant improvement to the random parameter selection technique for γ, the same is probably true for K-Means SVM C parameters.

5. ACKNOWLEDGMENTS

This research was carried out at the Jet Propulsion Laboratory (JPL), California Institute of Technology under a contract with the National Aeronautics and Space Administration (NASA) and was under the sponsorship of the NASA Undergraduate Student Research Project (USRP) program through JPL.

REFERENCES

1. T. Lu, C. L. Hughlett, H. Zhou, T-H. Chao, J. C. Hanan, "Neural network post-processing of grayscale optical correlator," Proc. *SPIE* **5908**, *Optical Information Processing III*, 2005.
2. Tsung Han (Hank) Lin, T. Lu, Henry Braun, Western Edens, Yuhan Zhang, T-H. Chao, Christopher Assad, and Terrance Huntsberger, " Optimization of a multi-stage ATR system for small target identification," Proceedings of SPIE Vol. 7696C, 2010.

3. W. Nicholas Greene, Yuhan Zhang, T. Lu , T-H. Chao, “Feature extraction and selection strategies for automated target recognition**,”** SPIE Symposium on Defense, Security & Sensing Conference, Independent Component Analyses, Wavelets, Neural Networks, Biosystems, and Nanoengineering, Proceedings of SPIE Vol. 7703, 2010.
4. Ye, D., Edens, W., Lu, T., Chao, T., “Neural Network target identification system for false alarm reduction,” Proc. SPIE Vol. 7340, 2009.
5. Johnson, O., Edens, W., Lu, T., Chao, T. "Optimization of OT-MACH filter generation for target recognition," Optical Pattern Recognition XX. Proceedings of the SPIE, Volume 7340, 2009.
6. Chang, C. C., Lin, C. J., “LIBSVM : a library for support vector machines” ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011.
7. B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, ACM Press, pages 144-152, 1992.
8. Üstün, B,. 2010, “support Vector Machines” from www.cac.science.ru.nl/people/ustun/SVM.JPG, Retrieved 2011.
9. Maven,. “Lib Linear” From: www.bwaldvogel.de/liblinear-java/svm3d_big.png, Retrieved 2011.
10. Qun C., Qingcai C. Xiaolong W., “Scaling Gaussian RBF kernel width to improve SVM Classifications” IEEE 2005.
11. Hsu C., Chang C., Lin C., "A Practical Guide to Support Vector Classification," http://www.csie.ntu.edu.tw/~cjlinhttp://www.csie.ntu.edu.tw/~cjlinhttp://www.csie.ntu.edu.tw/~cjlin , 04/15/2010.