Feature Extraction and Target Classification of Side-Scan Sonar Images

Jason Rhinelander, PhD
Division of Engineering
Saint Mary's University
Halifax, Nova Scotia, Canada
Email: jason.rhinelander@smu.ca

***Abstract*—Side-scan sonar technology has been used over the last three decades for underwater surveying and imaging. Application areas of side-scan sonar include archaeology, security and defence, seabed classification, and environmental surveying. In recent years the use of autonomous underwater systems has allowed for automatic collection of data. Along with automatic collection of data comes the need to automatically detect what information is important. Automatic target recognition can allow for efficient task planning and autonomous system deployment for security and defence applications.**

Support Vector Machines (SVMs) are proven general purpose methods for pattern classification. They provide maximum margin classification that does not over fit to training data. It is generally accepted that the choice of kernel function allows for domain specific information to be leveraged in the classification system. In this paper it is shown that for target classification in side-scan sonar, extra feature extraction and data engineering can result in better classification performance compared to parameter optimization alone.

I. Introduction

Side-scan sonar technology has been used over the last three decades for underwater surveying and imaging. Application areas of side-scan sonar include archaeology, security and defense, seabed classification, and environmental surveying. In recent years, the use of autonomous under water systems has allowed for automatic collection of data. However, along with automatic collection of data comes the need for the system to determine what data is important.

A. Side-scan Sonar

Side-scan sonar is an active acoustic range detection system that is often used for underwater surveying, archaeology, and Automated Target Recognition (ATR). Side-scan sonar utilizes an acoustic emitter and two acoustic receivers, one on the port and starboard side on the hull of vessel, on a towed "fish", or mounted on an underwater vehicle. The return signal from the emitter is measured and an image is composed by the forward motion of the emitter/receiver. An example of such an image is shown in Figure 1.

Computational Intelligence (CI) approaches are necessary when side-scan sonar is combined with autonomous systems. An autonomous system can be operated within a remote environment, capture data through side scan-sonar, and then return the captured data for analysis. Communications between

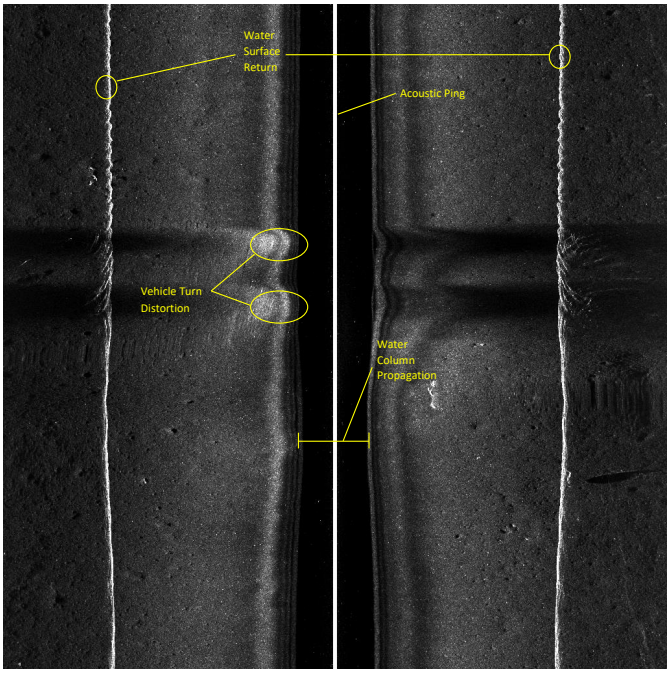

Fig. 1. An example of some of the unique features and challenges associated with side-scan sonar images.

an autonomous system and the shore/vessel is often too restrictive or non-existent for real-time processing. It is therefore advantageous for the autonomous system to be able to make complex decisions while deployed in the field. For example, when surveying for targets of interest it may be important for the underwater vehicle to surface and transmit its data versus finishing its planned route. This is especially the case for security related operations. In the work of Quintal et al [1], they state "current side-scan data processing techniques are largely manual, highly time-consuming, and prone to operator error. Availability of well-trained analysts is also a challenge". These are the motivating factors for research into machine learning and automatic target recognition in side-scan sonar.

The task of image classification has been extensively studied. There are some unique challenges in the case of side-scan sonar images. Since the forward movement of the vehicle "scans" the sea floor, if there is a change in heading

there are non-linear distortions created in the imagery. Other distortions and interference are caused by sudden changes in depth, schools of fish, and vegetation such as seaweed. There are existing approaches to object recognition in side-scan sonar images. In the work of Wang et al. [2], the authors were successful in multi-level object segmentation in side-scan sonar images using adaptive approaches. Pinto et al. [3] examine the segmentation, skeleton extraction and autonomous system requirements in side-scan sonar imagery. Chang et al. [4] investigate the use of fuzzy C-means based clustering for shadow removal in side-scan sonar images. A survey of analysis techniques for side-scan sonar can be found in the work of Xiaojun et al [5]. In the work of Dura et al. [6] the authors propose a novel active learning technique based on kernel classifiers with the purpose of enhancing the detection of underwater objects without a priori knowledge. In the work of Wang et al. [7], [8] the authors investigate the use of multi-aspect classification of sonar imagery and also the use of SVMs for imbalanced class labels respectively. This paper focuses on the benfit of pre-filtering of images before classification using SVMs.

Before describing the classification system presented in the paper, both the mathematical notation used, and a brief description of SVMs will be given.

1) Mathematical Notation: Scalar variables are represented by *italicized* lowercase letters, and vector variables by **bold** lowercase letters. When referring to a component dimension of a vector, round bracketing is used. For example, the ith dimension of the vector $\mathbf{x}$, is stated as $\mathbf{x}(i)$. For matrices and gray-scale images, row-column addressing is used. Given image I, a pixel in the ith row and jth column is described as $I(i,j)$. Finally, matrices and images are addressed row-wise from top to bottom, and column-wise from left to right.

The following is a description of some of the functions and variables used in this paper:

- $\mathbf{x} \in \mathbb{R}^d$: A vector of dimensionality d. $\mathbf{x}$ is used as an input vector for the classification process.
- $y \in \pm 1$: A classification label associated with a sonar image. The +1 and -1 are targets of interest and non-targets respectively.
- $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$: Defines a kernel function that represents a dot product in Reproducing Kernel Hilbert Space (RKHS), between the two input vectors, $\mathbf{x}_1$ and $\mathbf{x}_2$. Throughout this paper, a Gaussian kernel function is used, $k(\mathbf{x}_1, \mathbf{x}_2) = e^{-\gamma \|\mathbf{x}_1 - \mathbf{x}_2\|^2}$
- γ : Width parameter of the Gaussian kernel function.
- $f : \mathbb{R}^d \rightarrow \pm 1$: A classification function that predicts the corresponding label of an unobserved input vector.

2) Support Vector Machines (SVMs): SVMs are general purpose algorithms that can be trained with any set of input vector and label pairs $(\mathbf{x}_i, y_i)$. To use SVMs for image classification, an input vector is created based on pixel values, an appropriate kernel function must be chosen, and hyper-parameters are optimized. The training algorithm solves a constrained Quadratic Programming (QP) problem that minimizes the classification error while also maximizing the margin of separation between +1 and -1 class labels in the set. Domain specific information can be used to improve the performance of SVMs for image classification in two ways. First, the images maybe pre-processed to extract useful information. Second, the choice of kernel function can have a dramatic influence on performance and accuracy. In a previous work, using online kernel algorithms for processing image data was investigated [9].

There are two types of SVMs or Support Vector Classifiers (SVCs) that are used in this paper; the C-SVC algorithm, and the ν-SVC algorithm. The hyper-parameters C and ν control the behavior of the training algorithm. With both SVCs, once the QP problem has been solved, a subset of training vectors are retained by assigning values to the α_i variables. Any non-zero α_i means that the corresponding $(\mathbf{x}_i, y_i)$ becomes a Support Vector (SV) that contributes to the classification of unobserved data. The parameter C is used to apply regularization to the resulting classification function, with a lower value of C possibly resulting in more support vectors. In the case of ν-SVC, the ν parameter describes a upper bound on the proportion of training errors, and a lower bound on the proportion of SVs within the training set. From a practitioner's perspective, the ν parameter is bounded between 0 and 1.0, corresponding to 0% and 100% of the training set respectively. The training QP problem for both C-SVC and ν-SVC are given in equations 1 and 2. Once the QP problem has been solved the classification function is defined in equation 3. A Gaussian kernel function was chosen because it is a universal kernel that can be applied to many machine learning problems with little or no domain specific knowledge. An interested reader can refer to [10], [11] for more details on SVMs.

$$\begin{aligned} \underset{\boldsymbol{\alpha} \in \mathbb{R}^m}{\text{maximize}} \quad & W(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \tfrac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to:} \quad & 0 \leq \alpha_i \leq \tfrac{C}{m}, \forall i = 1 \ldots m \\ & \sum_{i=1}^{m} \alpha_i y_i = 0 \end{aligned} \tag{1}$$

$$\begin{aligned} \underset{\boldsymbol{\alpha} \in \mathbb{R}^m}{\text{maximize}} \quad & W(\boldsymbol{\alpha}) = -\tfrac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to:} \quad & 0 \leq \alpha_i \leq \tfrac{C}{m}, \forall i = 1 \ldots m \\ & \sum_{i=1}^{m} \alpha_i y_i = 0 \\ & \sum_{i=1}^{m} \alpha_i \geq \upsilon \end{aligned} \tag{2}$$

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{m} \alpha_i y_i k(\mathbf{x}, \mathbf{x}_i) + b \right) \tag{3}$$

II. Proposed Classification System

The MNIST handwritten digit data set [12] is currently one of the most popular benchmark data sets for image classification. This study proceeds in a similar fashion in that there is a collection of sonar images of objects of interest

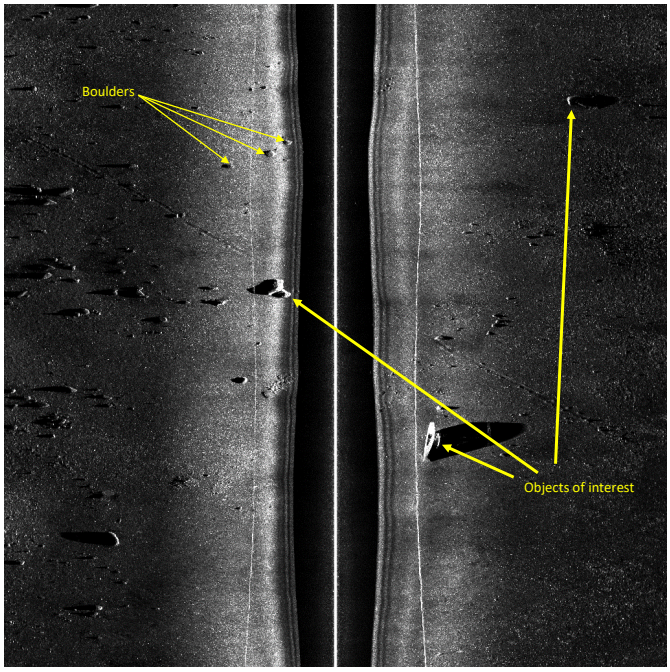

Fig. 2. A sample side-scan sonar image with some objects of interest illustrated.

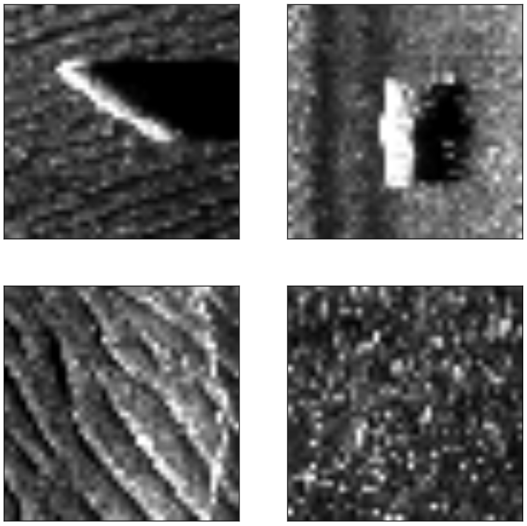

Fig. 3. The top row shows images of targets identified by a side-scan sonar operator. The bottom row shows examples of background images (seabed with sand and debris) where there is no target present.

(+1 class label) and randomly selected sonar images which do not contain targets of interest (-1 class label). Side-scan sonar images were captured by a REMUS-100 underwater vehicle and targets of interest were labeled by trained sonar technicians, as illustrated by Figure 2. For each target of interest, a 41 pixel by 41 pixel image was extracted from the side-scan sonar image. The sonar targets were identified by a human operator and are within 20 pixels of the the actual target's center. Therefore a 41 by 41 pixel image size was chosen to ensure that most of the target pixels were captured in the sample. The original data set of targets was from the DSTO ATR challenge in 2013, but in this study, the primary objective is to examine the effect of image pre-processing on SVM performance as applied to side-scan sonar data. The classifier training data contains 654 training images that contain 202 targets of interest identified by a sonar operator. Figure 3 illustrates examples of target objects and non-target images (background). The classification system works in three stages.

The first stage expands the data set from 202, 41x41 pixel target image patches to 1404 image patches. For every target image patch another is created by transposing the column addresses of the pixels within the patch. This operation is performed since a target image on one side of the vehicle would produce a mirrored image if it were on the opposite side. This increases the number of targets to 404 which are assigned +1 class labels. SVMs are binary classifiers so a second pass is made through the data set and for any training image that contains no targets, a random 41x41 pixel patch was extracted with a -1 class label. There were a total of 1000 random image patches without targets.

The second stage of image processing involves the use of either a Sobel or median filter, or no filtering (raw pixel values are used). Figure 4 illustrates the image processing that is explored in this paper. The first column illustrates three original target images in the sonar data. The second column shows the results of applying a Sobel filter to each image and each pixel value is the magnitude of the Sobel response in the x and y directions. The third column shows the application of a median filter (using a 3x3 pixel window size) to each corresponding image. The Sobel filter enhances the edges of each sonar image, while the median filter has a smoothing effect on neighboring pixels within the image. Three separate image sets are created in this stage.

The third stage takes the pixels in each image and then creates a feature vector, $(\mathbf{x})$, by copying pixel values into the feature vector in a row by row fashion. Each image patch label is added to the label vector, $\mathbf{y}$. The final step is to normalize each feature across all feature vectors. SVMs are sensitive to differing scales of magnitude as the Gaussian kernel function involves the L2-norm. All feature vectors are compiled into a feature matrix, $\mathbf{X}$. Rows represent individual feature vectors from each image and columns represent specific pixel locations across the entire data set. Each column was normalized by subtracting the minimum and dividing by the maximum in each column. This results in all features being on the scale of 0.0 to 1.0. The resulting dimensions are $\mathbf{X} \in \mathbb{R}^{1404} \times \mathbb{R}^{1681}$, and $\mathbf{y} \in \mathbb{R}^{1681}$. Figure 5 illustrates the data flow through the three stages in the classification system.

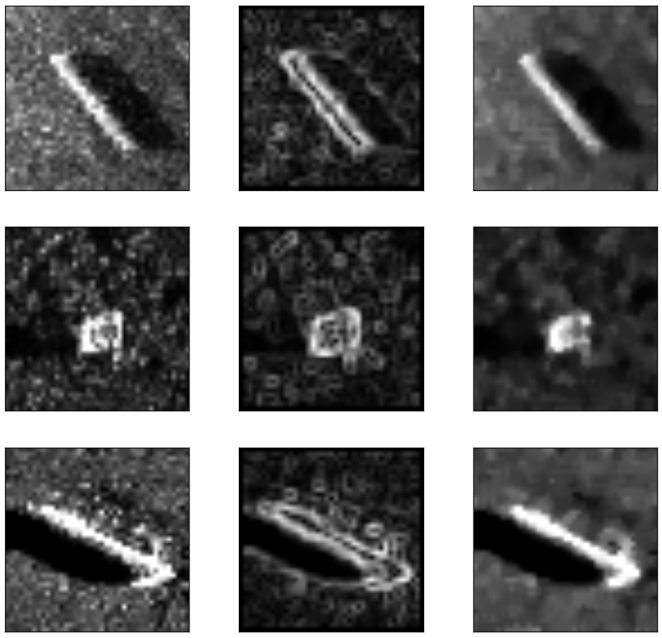

Fig. 4. Examples of raw image data and image data after Sobel and median filters are applied. The first column shows three separate raw target images. The second and third column show the Sobel and median filtering result respectively.

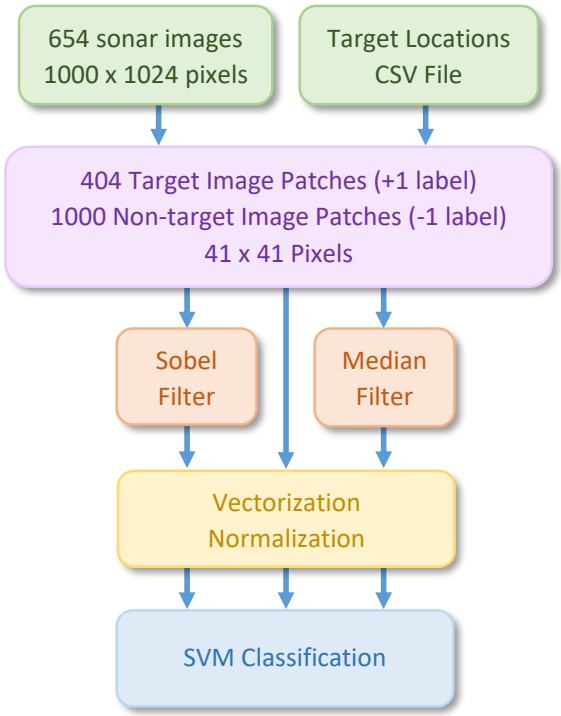

Fig. 5. The flow of processing used in this paper. It should be noted that the final two stages, vector normalization and SVM application result in three separate normalized datasets and three separate SVM classifiers.

III. Experimental Results

The three data sets that were created in the previous section were used as training data sets for both C-SVC and ν-SVC. There is no separate validation test data set in this case, therefore stratified 10-fold cross validation was performed. The scikit-learn machine learning library was used to implement the cross validation and LIBSVM (included with scikit-learn) was the SVM library employed. The choice to use stratified 10-fold cross validation was made because of the imbalance of +1 to -1 target labels (+1 has 404 samples while -1 has 1000 samples). Stratified cross validation maintains the proportion of class labels between folds.

K-fold cross validation takes the data set and divides it into K equally sized data sets. The K-1 smaller data sets are then used for training the classification function and the final K-th data set is used for testing. The process is repeated until all K subsets have been used as the testing data set and the accuracy results are then averaged across the K test sets, for this paper K$=10$ was used.

The most significant problem with machine learning using SVMs is optimizing hyper-parameters and kernel parameters. The most commonly used method to find optimal parameters is to employ K-fold cross validation in combination with a grid search method across parameters. The parameters that produce the classifier with the highest accuracy are chosen. In this paper, multiple rounds of grid search were used to narrow down the optimal parameters for each classifier. This procedure was used to optimize the hyper-parameters C and ν for C-SVC and ν-SVC respectively, and also to optimize the Gaussian kernel parameter, γ. Table I gives the maximum accuracy obtained on the three data sets described in figure 5. The results show that using the Sobel filter with both C-SVC and ν-SVC have the highest accuracy rate.

TABLE I
10-fold cross validation results.

Classifier	Filter Type	Accuracy	Parameters	
C-SVC	Sobel	**92.7%**	$C = 28$	$\gamma = 0.036$
ν-SVC	Sobel	**92.8%**	$\nu = 0.225$	$\gamma = 0.05$
C-SVC	none	84.0%	$C = 5$	$\gamma = 0.02$
ν-SVC	none	84.1%	$\nu = 0.235$	$\gamma = 0.017$
C-SVC	median	86.5%	$C = 50$	$\gamma = 0.05$
ν-SVC	median	86.0%	$\nu = 0.27$	$\gamma = 0.03$

Results are averaged across 10-folds of stratified cross validation. Figure 6 shows all Receiver Operating Characteristic (ROC) curves for each experiment performed. When viewing the ROC curves, the dashed line represents the expected true positive and false positive classification rate of randomly guessing the classification result. Each fold of cross validation is plotted on the ROC curve with a thinner line, and the mean result (averaging of all 10 folds of cross validation) is given by the dark curve. The accuracy of the classifier is given by the area under the mean ROC curve. It is interesting to note that the Sobel filtered dataset results in a "tighter" distribution of single cross-validation ROC results about the mean ROC curve. The accuracy values vary slightly with every simulation run because sample index values are randomly shuffled each time the dataset is loaded.

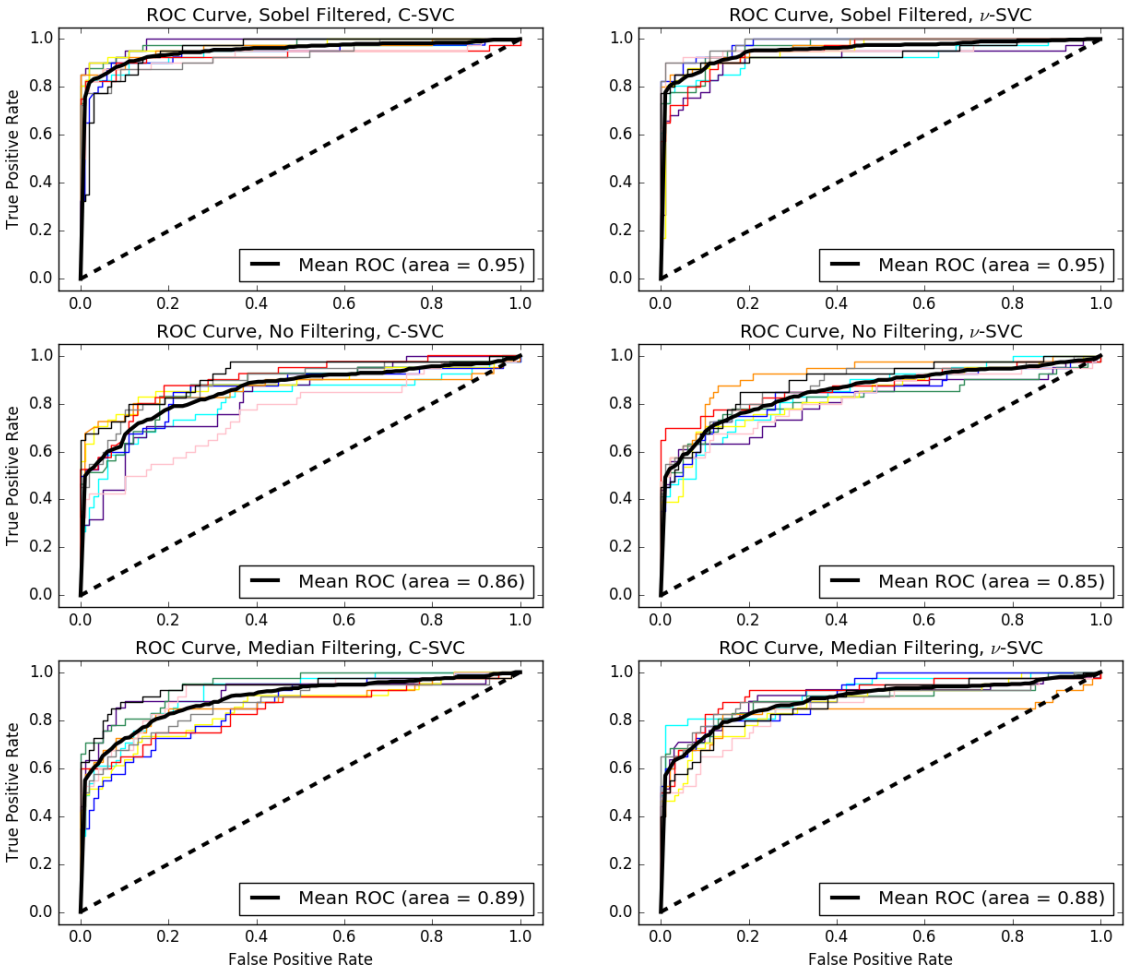

Fig. 6. ROC curves for the described experiments.

IV. Conclusions and Future Work

The use of filtering has a positive effect on classifier accuracy, and the accuracy is increased most significantly when a Sobel filter is used. The Sobel filter is often used in the first stages of edge detection in image analysis. SVMs are similar to template methods in that they use a linear combination of kernel function evaluations that involve both support vectors (SVs) and unobserved data.

Two main conclusions can be drawn from this paper. First, the largest distinguishing characteristic for identified targets is geometrical uniqueness amongst surrounding topology. Objects of interest mostly consist of longer cylindrical, box like, or sphere like shapes. These shapes consist of longer edges and therefore a Sobel filter will respond with high spatial gradient values, resulting in better pre-processing of training data. The second conclusion that can be drawn is that a median filter will filter out extreme values, such as noise, resulting in a slightly better classifier than with raw data alone.

Previous research into the application of SVMs has held the common view that a proper kernel function allows a data engineer to apply domain specific knowledge to a classification system. The experiments conducted in this paper illustrate that the accuracy of an SVM for sonar image classification can be improved using pre-filtering. Future work will be conducted with regard to the integration of feature extraction for the optimization of SVMs and other kernel based methods.

Acknowledgments

This work was funded by Saint Mary's University. I would also like to acknowledge Defence Research and Development Canada (DRDC) Atlantic and Defence Science and Technology (DST) Group, formerly the Defence Science and

Technology Organization (DSTO) for access to the side-scan sonar data.

REFERENCES

[1] R. T. Quintal, J. E. Kiernan, J. Shannon, and P. S. Dysart, "Automatic contact detection in side-scan sonar data," in *IEEE International Conference on Technologies for Homeland Security (HST), 2010*, Nov 2010, pp. 270–275.

[2] X. Wang, H. Wang, X. Ye, L. Zhao, and K. Wang, "A novel segmentation algorithm for side-scan sonar imagery with multi-object," in *IEEE International Conference on Robotics and Biomimetics, 2007.*, Dec 2007, pp. 2110–2114.

[3] M. Pinto, B. Ferreira, A. Matos, and N. Cruz, "Side scan sonar image segmentation and feature extraction," in *OCEANS 2009*, Oct 2009, pp. 1–9.

[4] R. Chang, Y. Wang, J. Hou, S. Qiu, R. Nian, B. He, and A. Lendasse, "Underwater object detection with efficient shadow-removal for side scan sonar images," in *OCEANS 2016 - Shanghai*, April 2016, pp. 1–5.

[5] J. Xiaojun, L. Nan, L. Renyi, and Y. Tianhe, "Study and implementation of the methods of the side-scan sonar image processing," in *2008 International Conference on Computer Science and Software Engineering*, vol. 6, Dec 2008, pp. 109–112.

[6] E. Dura, Y. Zhang, X. Liao, G. J. Dobeck, and L. Carin, "Active learning for detection of mine-like objects in side-scan sonar imagery," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 360–371, April 2005.

[7] X. Wang, X. Liu, N. Japkowicz, S. Matwin, and B. Nguyen, "Automatic target recognition using multiple-aspect sonar images," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, July 2014, pp. 2330–2337.

[8] X. Wang, H. Shao, N. Japkowicz, S. Matwin, X. Liu, A. Bourque, and B. Nguyen, "Using svm with adaptively asymmetric misclassification costs for mine-like objects detection," in *11th International Conference on Machine Learning and Applications (ICMLA), 2012*, vol. 2, Dec 2012, pp. 78–82.

[9] J. Rhinelander and P. X. Liu, "Tracking a moving hypothesis for visual data with explicit switch detection," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, July 2009, pp. 1–8.

[10] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[11] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer-Verlag New York, Inc., 1995.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov 1998.