# Multiple-instance learning-based sonar image classification

J. Tory Cobb[a], Xiaoxiao Du[b], Alina Zare[c], and Matthew Emigh[a]

[a]Naval Surface Warfare Center Panama City Division, Panama City, FL USA
[b]Electrical & Computer Engineering Dept., University of Missouri, Columbia, MO USA
[c]Electrical & Computer Engineering Dept., University of Florida, Gainesville, FL USA

## ABSTRACT

An approach to image labeling by seabed context based on multiple-instance learning via embedded instance selection (MILES) is presented. Sonar images are first segmented into superpixels with associated intensity and texture feature distributions. These superpixels are defined as the "instances" and the sonar images are defined as the "bags" within the MILES classification framework. The intensity feature distributions are discrete while the texture feature distributions are continuous, thus the Cauchy-Schwarz divergence metric is used to embed the instances in a higher-dimensional discriminatory space. Results are given for labeled synthetic aperture sonar (SAS) image database containing images with a variety of seabed textures.

**Keywords:** Multiple-instance learning, synthetic aperture sonar image, Cauchy-Schwarz divergence, superpixel

## 1. INTRODUCTION

High-frequency synthetic aperture sonar (SAS) systems produce high-resolution seabed imagery of near photographic quality. This sensing medium has enabled recent advanced computer vision analysis techniques that were previously not well-suited to lower resolution real aperture sonar. Several recent approaches to SAS image seabed texture segmentation have been applied using these advanced techniques with some measure of success. As with any natural scene imagery, SAS image texture boundaries may be diffuse and the texture content may be mixed or confused. A recently published SAS image boundary detection algorithm[1] sought to oversegment SAS imagery into superpixels. In this research we will use these superpixels as elemental quantities in an algorithm that tags images with context labels based on their seafloor composition.

Several previous attempts to segment high-resolution sonar imagery ignored the confused content of segmented areas.[2–4] In addition to confused content, the boundaries between seabed contexts are usually gradual with wide regions of transition. Thus it is difficult, if not impossible, to specify mutually-exclusive contextual labels corresponding to each pixel in a sonar image as required by standard supervised learning. Previously, we employed Multiple Instance Learning (MIL) techniques to address the problem of mixed labels in classification exemplars.[5] MIL has been widely used for supervised classification where training class labels are associated with a set (bag) of training sample points (instances), rather than the individual instance labeling. In MIL training, a bag is labeled as positive if it contains at least one instance of a target class, and negative if all the instances in the bag are negative. In our MIL application here, an image will be considered a bag and the superpixels will be instances. An image may contain multiple labels depending on what seabed contexts are present. Here the MIL approach addresses the common situation where a sonar image contains more than one context. In Fig. 1, the four sample SAS images demonstrate the varying distinctions between different seabed types and illustrate that many images contain more than one context.

Among the large variety of MIL approaches, the MILES (Multiple-Instance Learning via Embedded Instance Selection) approach is well suited to the goal of seabed context-based image labeling.[6–8] To improve classification performance from earlier work by Du,[5] we encode instances with more descriptive intensity and texture-based features.[9] The classification features are derived from both pixel intensity and texture information generated by the original superpixel formation algorithm. In order to use the high-dimensional continuous texture pdfs in the embedded bag-space, we employ a Cauchy-Schwarz divergence metric to measure distances between instances and bags.

Approved for Public Release; distribution is unlimited.

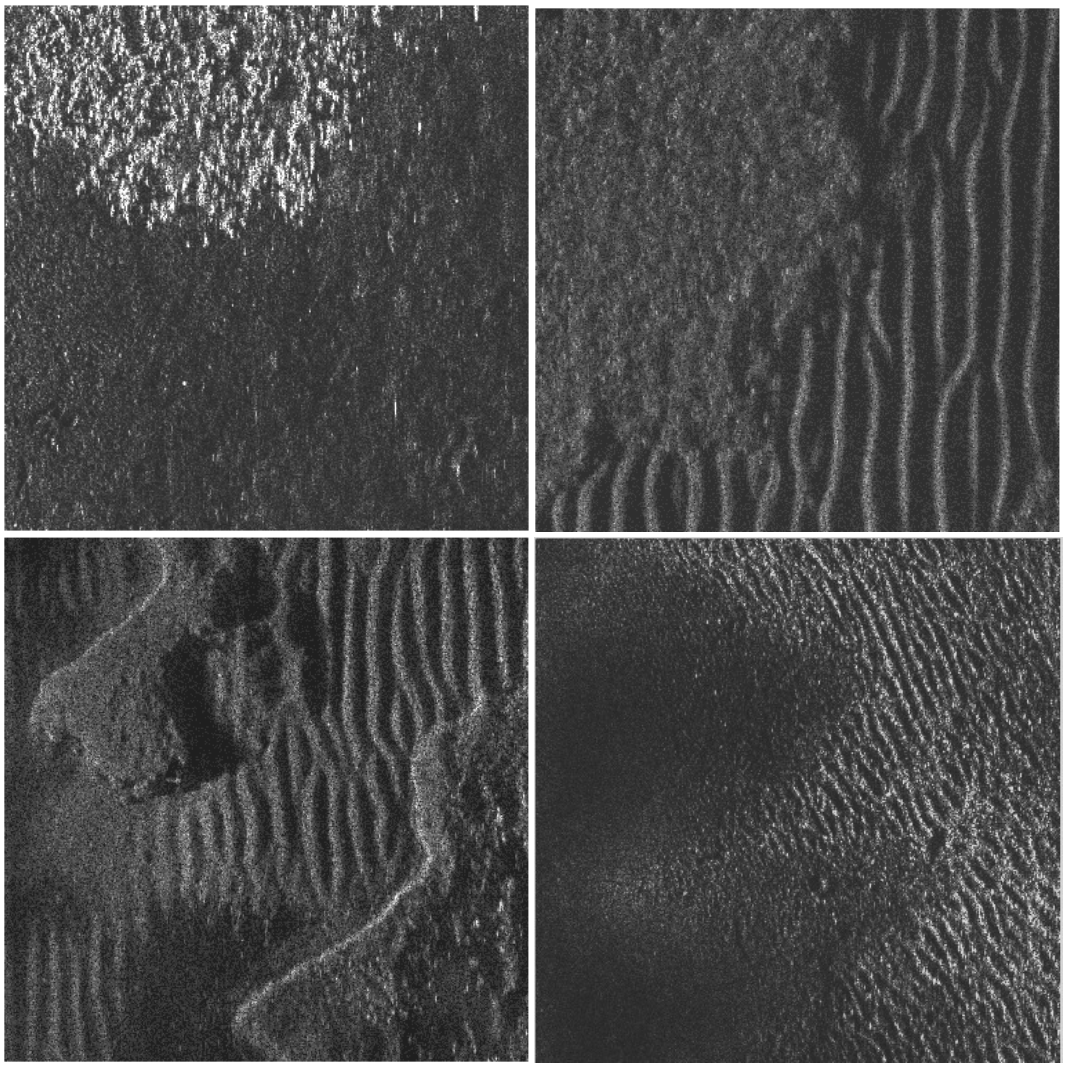


Figure 1. Montage of SAS sample images with various textures. Sand ripples of various sizes and periodicities, sea grass, hard-packed sand, and rock beds are all visible in these images.

## 2. SUPERPIXEL FORMATION

Cobb, et. al presented a boundary detection algorithm using superpixel information from intensity and textural features.[1] Drawing on previous work from Malik, Ren, and Martin, sonar image pixels were clustered into distinct regions by encoding intensity and textural distances into a similarity matrix and performing multi-class spectral clustering.[9–11] The final clustered regions typically have homogeneous texture and intensity distributions and are useful elemental quantities for large-scale image segmentation or seafloor classification. In this paper we begin with the SAS superpixel generated from this prior work as the lowest-level input to the MILES classification algorithm.

### 2.1 Superpixel Feature Vector - Intensity

The $k^{th}$ superpixel in an image set is denoted $s_k$. Associated with each superpixel are the probability density distributions (pdf) of both intensity and textural features. The length $N_{\mathbf{x}_k}$ vector $\mathbf{x}_k$ is composed of pixel values

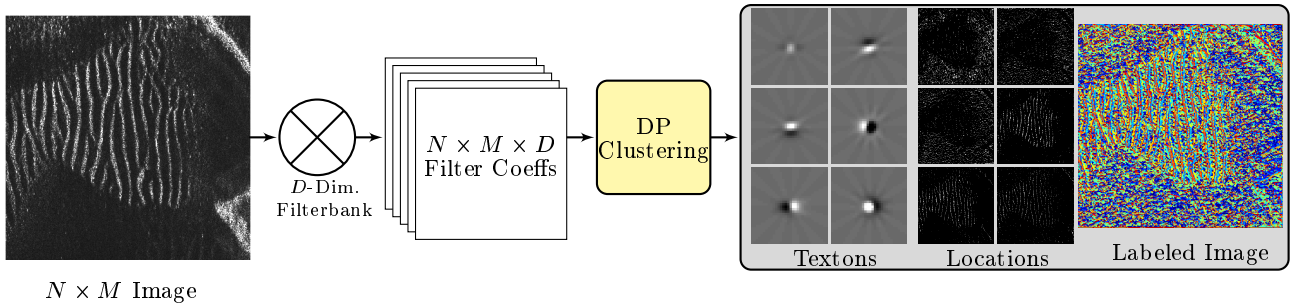



Figure 2. Processing steps for texton generation and labeling. From left to right, the original image $N \times M$ is convolved with a $D$-dimensional filterbank. After filtering, the $N \times M \times D$ filter coefficients are clustered using a DP clustering algorithm. The centroids from the clustering algorithm form texture primitives called textons. The texton map pixel values are encoded with the texton label that is closest to the filter coefficient vector at that pixel location. The final texton map on the right side of the processing stream depicts the spatial regularity of the texton labels for the different textures in the original image.

from superpixel $s_k$. The intensity pdf $p_b(s_k)$ is computed using a Parzen window estimator[12]

$$p_b(s_k) = \frac{1}{N_s} \sum_{j=1}^{N_{\mathbf{x}_k}} \psi(\mathbf{x} - \mathbf{x}_k(j); h_\psi), \tag{1}$$

where $N_s$ is a suitable normalizing factor such that $p_b(s_k)$ is a valid pdf and $\psi(\mathbf{x}; h_\psi)$ is a Parzen window of width $h_\psi$. For this application, we assume pixel intensity values vary between 0 and 1 or $\mathbf{x}_k \in [0,1]^{N_{\mathbf{x}_k}}$ and we set $h_\psi = 1$. Since the pixel intensity is bounded, we discretize the continuous pdf into 32 regularly-spaced intervals to speed computation prior to computing MILES bag-instance similarity.

### 2.2 Superpixel Feature Vector - Texture

Textons are texture primitives formed from a superposition of Gabor filter responses.[9,13] After filtering each normalized $M \times N$ sonar image with a $D$-sized filterbank of various orientations, the $M \times N \times D$ filter responses are quantized using a Dirichlet process clustering algorithm. This algorithm is designed to function like the $K$-means algorithm except that the number of clusters or the $K$ value is determined automatically. The centroids of these clusters are the linear superposition of filter responses and can be visualized as a texture primitive or texton. Following texton generation, each pixel is assigned the label of the closest texton by minimum Euclidean distance between the filter response at the pixel value and the cluster centroid. Fig. 2 summarizes the processing steps for texton generation and labeling. Readers are encouraged to read Cobb's paper for a detailed review of the DP clustering algorithm and the texton generation approach.[1]

Texton labels are usually collected into a histogram for single image segmentation tasks. In this multi-image segmentation algorithm we seek to compare textures across several sonar images and must recover a universal representation from the original single-image texton histograms. Recall that textons are formed from the linear superposition of Gabor filter coefficients. This superposition is encoded by the centroids from the DP clustering algorithm.[1] In single-image segmentation, these centroids and their associated labels are referenced to a particular image. However, the centroid location in the wavelet filterbank space is universally referenced provided we use the same filterbank for all the images in the set. Using this idea of universal representation, we can assume the collection of centroids assigned to the pixels in the superpixel encode a mixture of multivariate Gaussian (MoG) pdf of unit variance located in the filterbank space. [Note: the unit variance MoG assumption follows from the DP clustering algorithm parameters.[1]]

The texton labeling procedure for each image produces a finite number of textons or MoG centroids. Each centroid can be considered a component mixture in the MoG formulation. Assuming we have $M$ unique texton labels for the pixels forming a superpixel we can denote a MoG texture pdf $p_t(s_k)$ for superpixel $s_k$ as

$$p_t(s_k) = \sum_{m=1}^{M} \pi_m \, \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_m, \mathbf{I}), \tag{2}$$

where $\mathbf{z}$ is the $D$-dimensional variable covering filterbank space and $\boldsymbol{\mu}_m$ is the $m^{th}$ centroid location in filterbank space. The mixture weight for the $m^{th}$ component, $\pi_m$, is the occurrence frequency of the $m^{th}$ centroid label in superpixel $s_k$ or

$$\pi_m = \frac{|\mathcal{M}|}{N_{\mathbf{x}_k}}, \tag{3}$$

where $\mathcal{M}$ is the set of pixels assigned centroid label $m$.

## 3. MILES CLASSIFICATION

In MILES classification, discriminating features are found by mapping similarity between instances and bags into a higher-dimensional feature space.[8] Here we create classification features by mapping each bag (image) to a new feature space based on its similarity to each instance (superpixel). In our prior work we created a high-dimensional bag feature vector by finding the distance between each training instance and the closest instance in that particular bag.[5] Here we perform a similar mapping but modify the distance metric to accommodate features described by continuous pdfs.

### 3.1 Feature Vector Formation

We denote the total number instances in the training set as $N$. For each bag $\mathbf{B}_i$, similarity is computed between every instance $s_k$ and the instances in the $i^{th}$ bag. The smallest distance is picked as the $k^{th}$ element in the bag feature vector. In other words, for each bag in a data set, one $N$-dimensional feature vector is computed. Since the superpixel texture features are represented by continuous pdfs of different dimensions, simple Euclidean distances are not adequate to measure similarities between the instances and bags.

To adequately capture similarity our mapping in this application is based on the Cauchy-Schwarz divergence ($D_{CS}$) between two pdfs.[14] $D_{CS}$ between pdfs $q(x)$ and $p(x)$ is defined as

$$D_{CS} \triangleq -\log\left(\frac{\int q(x)p(x)dx}{\sqrt{\int q(x)^2 \int p(x)^2 dx}}\right). \tag{4}$$

For our discrete intensity feature pdfs, the $D_{CS}$ calculation between superpixels $s_k$ and $s_j$ is straightforward

$$D_{CS}\left(p_b(s_k) \,||\, p_b(s_j)\right) = -\log\left(\sum_{i=1}^{32} p_b^i(s_k) p_b^i(s_j)\right) + \frac{1}{2}\log\left(\sum_{i=1}^{32}\left(p_b^i(s_k)\right)^2\right) + \frac{1}{2}\log\left(\sum_{i=1}^{32}\left(p_b^i(s_j)\right)^2\right), \tag{5}$$

where $p_b^i(\cdot)$ is the $i^{th}$ element of the discretized intensity pdf $p_b$. The $D_{CS}$ calculation for the texture divergence is more complicated, but can be computed in closed form by invoking the property that the integral of a product of Gaussians remains a Guassian. Kampa provides a detailed derivation of the $D_{CS}$ for the divergence between two general MoGs.[15] Here we show the $D_{CS}$ for the texture MoG pdfs which are restricted to uncorrelated, unit variance:

$$D_{CS}\left(p_t(s_k) \,||\, p_t(s_j)\right) = -\log\left(\sum_{u=1}^{U}\sum_{v=1}^{V} \pi_u \tau_v z_{uv}\right) + \frac{1}{2}\log\left(\sum_{u=1}^{U}\frac{\pi_u^2}{(2\pi)^{D/2}} + 2\sum_{u=1}^{U}\sum_{u'<u} \pi_u \pi_{u'} z_{uu'}\right) + \frac{1}{2}\log\left(\sum_{v=1}^{V}\frac{\tau_v^2}{(2\pi)^{D/2}} + 2\sum_{v=1}^{V}\sum_{v'<v} \tau_v \tau_{v'} z_{vv'}\right), \tag{6}$$

where

$$p_t(s_k) = \sum_{u=1}^{U} \pi_u \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_u, \mathbf{I}), \tag{7}$$

$$p_t(s_j) = \sum_{v=1}^{V} \tau_v \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_v, \mathbf{I}), \tag{8}$$

$$z_{uv} = \mathcal{N}(\boldsymbol{\mu}_u; \boldsymbol{\mu}_v, 2\mathbf{I}), \tag{9}$$

$$z_{uu'} = \mathcal{N}(\boldsymbol{\mu}_u; \boldsymbol{\mu}_{u'}, 2\mathbf{I}), \text{ and} \tag{10}$$

$$z_{vv'} = \mathcal{N}(\boldsymbol{\mu}_v; \boldsymbol{\mu}_{v'}, 2\mathbf{I}). \tag{11}$$

To calculate the $k^{th}$ element of the $N$-dimensional bag feature vector $\mathbf{m}(\mathbf{B}_i)$ we find the minimum sum of intensity and texture divergences between every instance in the dataset and the instances contained in bag $\mathbf{B}_i$

$$S(s_k \mid \mathbf{B}_i) = \max_{j \in \mathcal{I}(\mathbf{B}_i)} \exp\{-(D_{CS}(p_b(s_k)||p_b(s_j)) + D_{CS}(p_t(s_k)||p_t(s_j)))\}, \tag{12}$$

where the index $k$ varies over all the $N$ instances in the training set and $\mathcal{I}(\mathbf{B}_i)$ are the indices of the instances in $\mathbf{B}_i$. The MILES feature vector $\mathbf{m}(\mathbf{B}_i)$ is thus populated

$$\mathbf{m}(\mathbf{B}_i) = [S(s_1 \mid \mathbf{B}_i), S(s_2 \mid \mathbf{B}_i), \ldots, S(s_N \mid \mathbf{B}_i)]. \tag{13}$$

### 3.2 1-Norm Support Vector Machine Classification

We formulate our classification function using the 1-Norm or Sparse Support Vector Machine (SVM)[8,16]

$$y(\mathbf{B}_i) = sign\left(\mathbf{w}^T \mathbf{m}(\mathbf{B}_i) + b\right), \tag{14}$$

where $\mathbf{w}$ and $b$ are found by minimizing the objective function

$$\Phi(\mathbf{w}, b) = \lambda \sum_{k=1}^{N} |w_k| + \beta \sum_{i=1}^{l^+} \zeta_i + (1-\beta) \sum_{j=1}^{l^-} \eta_i, \tag{15}$$

such that

$$(\mathbf{w}^T \mathbf{m}_i^+ + b) + \zeta_i \geq 1, \qquad i = 1, \ldots, l^+, \tag{16}$$

$$-(\mathbf{w}^T \mathbf{m}_j^- + b) + \eta_i \geq 1, \qquad j = 1, \ldots, l^-, \tag{17}$$

$$\boldsymbol{\zeta}, \boldsymbol{\eta} \geq \mathbf{0}, \tag{18}$$

where $\lambda$ controls the sparsity of $\mathbf{w}$, superscripted $\mathbf{m}_i^+$ and $\mathbf{m}_j^-$ are feature vectors of positive and negative bag samples, $\boldsymbol{\zeta}$ and $\boldsymbol{\eta}$ are hinge losses, $l^+$ and $l^-$ are the numbers of positive and negative bag examples respectively, and $\beta$ is a weighting applied to positive and negative exemplars. Eqns. (15-18) can be manipulated to efficiently solve a linear program for $\mathbf{w}$ and $b$.[8]

Following training, a test sample $i$ is classified by the equation

$$y(\mathbf{B}_i) = \sum_{k \in K^+} w_k S(s_k|\mathbf{B}_i) + b, \tag{19}$$

where $K^+$ are the indices for which $|w_k| > 0$. The one-norm penalty sparsifies $\mathbf{w}$, effectively limiting the number of features that contribute to the classification of bag $\mathbf{B}_i$ and simultaneous performs feature (or instance) selection with classification function optimization. Note that the classic two-norm SVM classifer uses a quadratic regularization term for $\mathbf{w}$ that sparsifies the number of support vectors,[17] whereas the one-norm SVM sparsifies the effective dimensionality of feature space. The one-norm SVM effectively limits the number of instances (or superpixels) that contribute to the classification of a new bag (or image). These instance prototypes should capture the salient characteristics of the context label.

## 4. EXPERIMENT AND RESULTS

The MILES supervised classification algorithm was tested against a high-frequency SAS database of 125 labeled $1000 \times 1000$ images containing the following four seabed context labels: 1) dark (mud, shadow), 2) bright (sand, seagrass), 3) rock and 4) sand ripple. If an image contained a discernable seabed context example, it was assigned that label. Each image could have multiple labels. The MILES one-norm SVM algorithms were trained and tested (one versus all) with labeled instances from the database using two-fold cross-validation. The cross-validation was conducted using random draws of positive and negative examples. Positive and negative samples were equally-weighted by setting the $\beta$ variable in Equation (15) equal to the fraction of the samples from the negative class.

To compare our performance against a similar algorithm, we also trained the MILES classifier from our previous work (labeled Algorithm I) on the same data.[5] Algorithm I uses an instance feature triplet of mean, variance, and the variance of the output of a $128 \times 128$ Laplacian of Gaussian (LoG) filter ($\sigma = 4$) over the image region bounded by the superpixel. Additionally, Algorithm I uses the similarity mapping $S(s_k \mid \mathbf{B}_i)$ as described by Chen rather than the mapping in Equation (12).[8] We denote our approach in this paper as Algorithm II.

Tables 1 and 2 list the training and test error for the 100 trial 2-fold validation of the database for Algorithms I and II respectively. Algorithm II performs significantly better for all seabed contexts. The more descriptive feature set, notably full pdfs of intensity values and more texture information, contributes additional discriminatory information not captured by the feature triplet of Algorithm I.

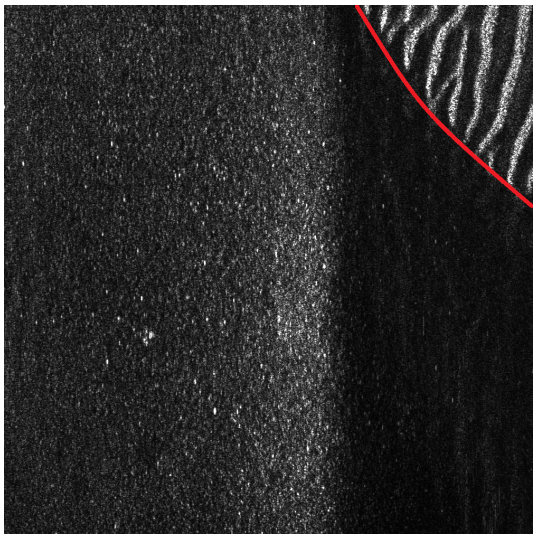
Table 1. SAS Image Labeling Error Results (Algorithm I)

| | Algorithm I (Du, et. al 2014) $\lambda = 2.5$ | |
|---|---|---|
| Seabed Label | Training Error (mean ± std) | Testing Error (mean ± std) |
| Dark | 0.1519 ± 0.0502 | 0.2566 ± 0.0486 |
| Bright | 0.1027 ± 0.0393 | 0.1971 ± 0.0701 |
| Rock | 0.2175 ± 0.0804 | 0.2645 ± 0.0778 |
| Ripple | 0.0695 ± 0.0027 | 0.1307 ± 0.0391 |

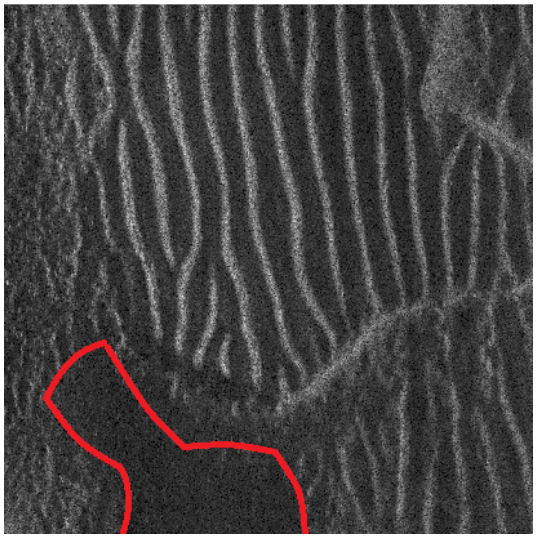
Table 2. SAS Image Labeling Error Results (Algorithm II)

| | Algorithm II (current approach) $\lambda = 0.75$ | |
|---|---|---|
| Seabed Label | Training Error (mean ± std) | Testing Error (mean ± std) |
| Dark | **0.1030** ± 0.0337 | **0.1603** ± 0.0394 |
| Bright | **0.0957** ± 0.0275 | **0.1603** ± 0.0527 |
| Rock | **0.0252** ± 0.0182 | **0.0444** ± 0.0125 |
| Ripple | **0.0586** ± 0.0267 | **0.1150** ± 0.0320 |

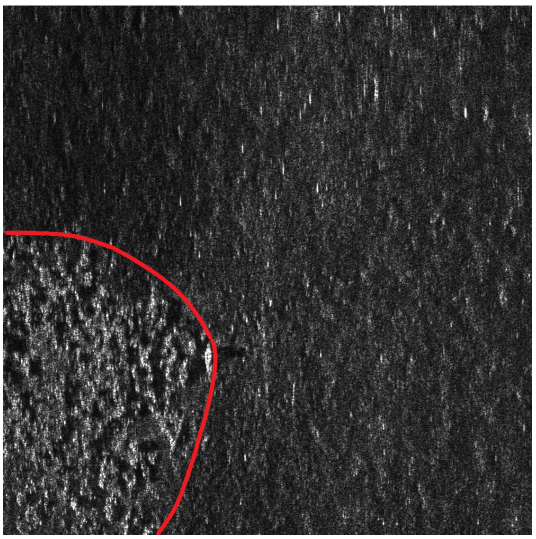
Recall that in the MIL context images (bags) may contain more than one seabed context label depending upon their composition. Correct classifications depend on the training instances matching some portion of the instances in the test bag. The fewer the number of labeled instances for a given class in the bag, the less likely there is some match between the training and test set. Misclassifications using Algorithm II are usually due to an image containing only a small labeled area of the seabed context in question. Fig. 3 depicts examples where



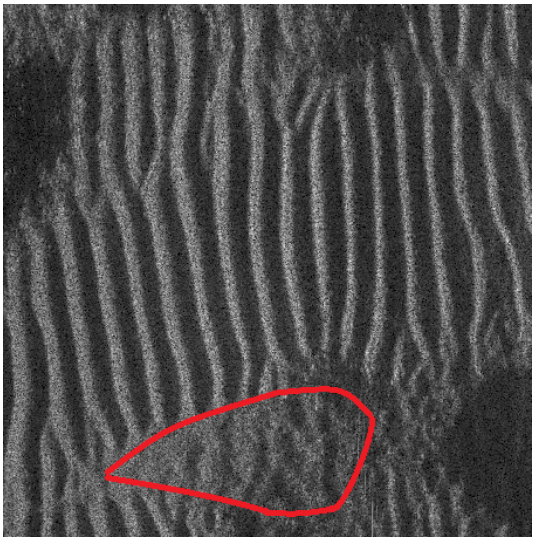
(a) Test Error Rate = 37%



(b) Test Error Rate = 44%



(c) Test Error Rate = 27%



(d) Test Error Rate = 29%

Figure 3. Sample images that were misclassified at higher rates than average for each label group. The misclassified labels outlined in red are (a) ripple, (b) dark, (c) rock, and (d) bright. In (a), (b), and (d), the labeled regions are small and less likely to match a significant portion of the training instances. In (c) the labeled rock region is blurry and its instances are not a good match between the sharply-focused rock training instances.

a context has a higher than average misclassification rate. In Figs. 3(a), 3(b), and 3(d) the labeled regions (in red) are very small compared to the other dominant contexts in the image. In Fig. 3(c) the labeled rock area in the lower left is blurry and likely contributes to poorer performance since the test instances do not match the rock training instances that are sharply in focus.

## 5. CONCLUSION

In this paper we describe an approach to SAS image labeling based on MILES classification. The MILES classification approach fits well with the paradigm of image tagging because an image is not assumed to be a "pure" representation of a context class, but merely contain some constituent parts from the class in question.

The MILES approach with more descriptive intensity and texture features outperforms our first attempt at MILES classification of SAS images which used a less descriptive feature set.

Here we also describe a universal representation for texton labels across an image set. The incorporation of continuous pdfs into the feature representation is made possible by using the Cauchy-Schwarz divergence metric between MoGs. This texton labeling approach and associated C-S divergence metric may also be applicable to more general image co-segmentation tasks and we intend to look into these applications in the future.

## ACKNOWLEDGMENTS

This research was funded by the Office of Naval Research Code 32OE.

## REFERENCES

[1] Cobb, J. T. and Zare, A., "Superpixel Formation and Boundary Detection in Synthetic Aperture Sonar Imagery," in [*3rd Intl. Conf. SAS and SAR*], (2014).

[2] M. Mignotte, C. Collet, P. P. and Boutheny, P., "Sonar Image Segmentation Using an Unsupervised Hierarchical MRF Model," *IEEE Trans. Image Process.* **9**(7), 1216–1231 (2000).

[3] Celik, T. and Tjahjadi, T., "A Novel Method for Sidescan Sonar Image Segmentation," *IEEE J. Ocean. Eng.* **36**(2), 186–194 (2011).

[4] Cobb, J. T. and Zare, A., "Multi-Image Texton Selection for Sonar Image Seabed Co-segmentation," in [*Proc. SPIE, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XVIII*], **8709**(87090H) (2013).

[5] Du, X., Zare, A., and Cobb, J., "Possibilistic Context Identification for SAS Imagery," in [*Proc. SPIE, Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XX*], **9454** (2014).

[6] Maron, O. and Ratan, A., "Multiple-Instance Learning for Natural Scene Classification," in [*Proc. 15th Int'l Conf. Machine Learning*], 341–349 (1998).

[7] Zhang, Q., Goldman, S., Yu, W., and Fritts, J., "Content-Based Image Retrieval Using Multiple-Instance Learning," in [*Proc. 19th Int'l Conf. Machine Learning*], 682–689 (2002).

[8] Chen, Y., Bi, J., and Wang, J., "MILES: Multiple-Instance Learning via Embedded Instance Selection," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(12), 1931–1947 (2006).

[9] Malik, J., Belongie, S., Leung, T., and Shi, J., "Contour and Texture Analysis for Image Segmentation," *Int'l J. Computer Vision* **43**, 7–27 (Jun. 2001).

[10] Ren, X. and Malik, J., "Learning a Classification Model for Segmentation," in [*9th IEEE Int'l Conf. on Comp. Vision (ICCV '03)*], **1**, 10–17 (2003).

[11] Martin, D., Fowlkes, C., and Malik, J., "Learning to Detect Natural Image Boundaries using Local Brightness, Color, and Texture Cues," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(5), 530–548 (2004).

[12] Bishop, C. M., [*Pattern Recognition and Machine Learning*], Springer-Verlag New York, Inc. (2006).

[13] Malik, J. and Perona, P., "Contour and Texture Analysis for Image Segmentation," *Int'l J. Computer Vision* **43**, 7–27 (Jun. 2001).

[14] Jenssen, R., Erdogmus, D., Hild, K., Principe, J., and Eltoft, T., "Optimizing the Cauchy-Schwartz PDF Distance for Information-Theoretic, Non-Parametric Clustering," in [*Energy Minimization Methods in Computer Vision and Pattern Recognition*], **1**, 1–6 (2005).

[15] Kampa, K., Hasanbelliu, E., and Principe, J., "Closed-Form Cauchy-Schwartz PDF Divergence for Mixture of Gaussians," in [*Proc. Int'l Joint Conf. on Neural Networks (IJCNN)*], 2578–2585 (2011).

[16] Bi, J., Bennett, K. P., Embrechts, M., Breneman, C., and Song, M., "Dimensionality Reduction via Sparse Support Vector Machines," *J. Machine Learning Research* **3**, 1229–1243 (2003).

[17] Scholkopf, B. and Smola, A., [*Learning with Kernels*], MIT Press (2002).