Graph-Based Supervised Automatic Target Detection

Gal Mishne, Ronen Talmon, and Israel Cohen, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a detection method based on data-driven target modeling, which implicitly handles variations in the target appearance. Given a training set of images of the target, our approach constructs models based on local neighborhoods within the training set. We present a new metric using these models and show that, by controlling the notion of locality within the training set, this metric is invariant to perturbations in the appearance of the target. Using this metric in a supervised graph framework, we construct a low-dimensional embedding of test images. Then, a detection score based on the embedding determines the presence of a target in each image. The method is applied to a data set of side-scan sonar images and achieves impressive results in the detection of sea mines. The proposed framework is general and can be applied to different target detection problems in a broad range of signals.**

***Index Terms*—Automated mine detection, automatic target detection, nonlinear-dimensionality reduction, side-scan sonar.**

I. Introduction

TARGET detection in images is important in military applications and various imaging systems such as hyperspectral [1], [2], synthetic aperture radar [3], [4], ground-penetrating radar [5], and side-scan sonar [6], [7]. The goal is to detect the target, usually man-made structures, vehicles, or devices, in a cluttered background. Automatic target detection is important for practical reasons, given the large amount of images produced in such applications. A supervised approach is useful in target detection when training images exist or prior knowledge exists regarding the target (e.g., its size and appearance). This prior knowledge can be used for modeling the target, feature selection, training a classifier, rejecting false alarms (FAs), etc., using various methods [2], [7]–[10].

Automatic detection of sea mines in side-scan sonar imagery is a challenging task due to the high variability in the appearance of the target and seabed reverberations (background clutter). Objects in side-scan sonar appear as a strong bright region (highlight) aside a dark region (shadow). The shadow is due to the object blocking the sonar waves from reaching the seabed. This paired highlight–shadow region is the primary feature for detection of sea mines [11]. Research in this field focuses on three aspects of the problem: detection of minelike objects (MLOs) in the image, classification of these objects as mine or nonmine, and identification of the sea-mine type [12], [13]. In this paper, we propose a new detection method and demonstrate its application in extracting MLOs from the cluttered seabed.

Algorithms proposed for MLO detection include the Markov random field (MRF) models [12], [14], a 2-D multiscale Gauss Markov random field (GMRF) with matched subspace detector (MSD) [15], a multidimensional generalized autoregressive conditional heteroscedasticity (GARCH) model with MSD [10], nonlinear matched filters [6], [8], morphological filters [16], etc. The detection is sometimes accompanied by extraction of the shadow, for example, using co-operating statistical snakes [12], [17] or deformable templates [18]. Following the detection of MLOs, a classification and identification procedure is applied to determine whether the objects are a mine or not, usually focusing on the shape of the shadow region [7], [11], [13], [17]–[22].

In target detection, the appearance of the target is usually known in advance, and reference images may also be available or simulated. In side-scan sonar, for example, augmented reality simulators have been proposed to embed synthetic target models on a real image of the seafloor [7], [20]. Many algorithms for sea-mine classification make use of training data. Reed *et al.* use the Hausdorff distance to compare test objects to a synthetic training set of MLO shadow regions produced by a sonar simulator [17]. Quidu *et al.* compare the Fourier descriptors of the contour of a tested shadow region to the Fourier descriptors of an initial set of prototype shadows [19]. Myers and Fawcett propose matching an object's signature image with a number of computer-generated templates using a generalized cross-correlation measure for template matching [21].

MLO detection algorithms, on the other hand, usually take advantage of prior information by applying a statistical model that is appropriate for the sonar acquisition scenario and/or searching for a joint signature of highlight and shadow. Dobeck *et al.* designed a nonlinear matched filter for MLO detection, which contains four distinct regions, namely, pretarget, highlight, dead zone, and shadow, based on the expected size of the sea mine [8]. Lange and Vincent propose using grayscale morphological filters to extract bright and dark regions from the image, expecting these to be highlight and shadow regions. These filters impose geometric constraints on shape, size, and area, determined by prior information on the expected size of the sea mines in the images [16]. Coiras *et al.* presented a special set of spatial filters, termed central filters, specifically designed for detection of MLOs. Their design ensures object presence and a highlight–shadow dichotomy [7]. Reed *et al.* [12] and Mignotte *et al.* [14] incorporate the prior knowledge on the spatial dependence between highlight and shadow regions into an MRF model, each proposing different distributions

Manuscript received April 15, 2014; revised August 5, 2014; accepted September 18, 2014. This work was supported by the Israel Science Foundation under Grant 1130/11. The work of R. Talmon, a Horev Fellow, was supported by the Taub Foundations.

The authors are with the Department of Electrical Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel (e-mail: galga@tx.technion.ac.il; ronen@ee.technion.ac.il; icohen@ee.technion.ac.il).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2014.2364333

for the seabed-reverberation and shadow regions. Noiboar and Cohen present an anomaly-detection-based approach, where the anomaly subspace for the MSD incorporates available *a priori* information about the target using a few real sea-mine images [10].

Most target detection methods require statistical modeling or heuristic filter design using prior knowledge on the appearance (size and geometry) of the expected target and the image formation process. When using a training set, typically, many images are included in the training set, in order to account for variability of the target appearance. In this paper, we propose a data-driven detection method to model the target, which implicitly handles variations in the target appearance, allowing for a small-sized training set.

Recently, Talmon *et al.* presented diffusion-graph-based filters for supervised speech enhancement [23]. A similar framework was proposed by Haddad *et al.* for filtering a known pattern in an image [24]. Both papers propose a principal component analysis (PCA)-based metric for constructing local models of the signal, using a training set.

We propose a new local metric for supervised target detection. This metric, as opposed to the PCA-based metric, is invariant to perturbations in the appearance of the target, as defined by the training set. Our approach is supervised to the extent that the user needs to input an appropriate training set and a notion of similarity between patches within the training set. No other *a priori* information is required, i.e., this approach does not rely on statistical modeling or imposing typical shape parameters. The paired appearance of the highlight–shadow region arises implicitly from the calculated metric and does not need to be imposed as prior information.

Consider that several training images of the target are available, either real or simulated, that may differ in their appearance, for example, in size, orientation, contrast, etc. Extracting overlapping patches from these images provides a training set of image patches containing the target. Our approach constructs a model for each training patch based on its local neighborhood within the training set: other training patches which are similar to the given patch. The main contribution of our approach is that, by controlling the notion of locality, i.e., how the neighborhood of each training patch is chosen, we effectively construct a metric which emphasizes similarities within the local neighborhood while allowing for a desired invariance to other dissimilarities. These similarities and dissimilarities are learned from the variability of the target in each local neighborhood of patches. This metric, therefore, enables to compare test patches containing the target to the training set, while repressing the differences due to slight changes in the target appearance. On the other hand, the metric emphasizes differences from the training set to which we want to be sensitive and penalizes them heavily. Thus, this metric does not penalize variability in the appearance of the target in the test image as compared to the training set, in contrast to other metrics such as the Euclidean distance.

Assume that there is an intrinsic set of parameters governing the appearance of the image patches that contain a target, such as shape parameters, textures, and lighting conditions. The proposed metric enables to design an invariance to certain intrinsic parameters, while emphasizing the similarity in other parameters. We show that this can be done in a data-driven manner, without explicitly modeling and calculating the intrinsic parameters. Calculating the element-wise empirical mean and variance of the local neighborhood provides a model for each training patch, with the desired invariant properties.

The proposed invariant metric is used to define an affinity kernel between the training set and the test set. In [23] and [24], an affinity kernel is used in a supervised graph-based algorithm to construct a filter which extracts the desired pattern from the input signal. In our approach, we use the supervised graph framework; however, we do not use the graph filter to detect the target in the image. Instead, we construct an embedding of the high-dimensional image patches into a low-dimensional space, which separates the patches containing the target from the patches that contain the background. We propose a new detection score in the embedding space, based on the structure of the affinity kernel, that determines the presence of a target in the image. The framework that we present is general and can be applied to different target detection problems in a broad range of signals, e.g., audio signals, hyperspectral images, and videos.

This paper is organized as follows. In Section II, we propose a metric for comparing training and test patches which enables to implicitly design an invariance to perturbations in the target model. In Section III, this metric is inserted in a supervised graph-based framework which provides a low-dimensional embedding of the data. Section IV presents a target detection score in the low-dimensional embedding. Section V reviews related work in which a different approach to target modeling is used, based on a PCA approach, and in Section VI, we analyze the advantages and disadvantages of both methods. Finally, Section VII presents experimental results in a 1-D toy problem and the real-world problem of sea-mine detection in side-scan sonar images. Using a training set consisting of merely five images, we demonstrate the success of our method compared to other supervised methods.

II. Local Neighborhood Modeling

In this section, we formulate the problem and present a new metric for comparing image patches based on local neighborhoods of patches in the training set. We show that, by controlling how these neighborhoods are defined, we can efficiently construct a metric that emphasizes similarities within the local neighborhood, while allowing for a desired invariance to other dissimilarities. We demonstrate our method in the application of side-scan sonar images.

A. Problem Formulation

In target detection applications, images of the target can be acquired or simulated in advance. Given a new test image, the goal is to determine whether a target exists in the image, based on prior information available from the training set. High-dimensional features are commonly used for image representation. In our approach, we describe the images using overlapping patches extracted from the training set and test image. Some approaches model both background and target [2], [9]; however, in our approach, we model only the target.

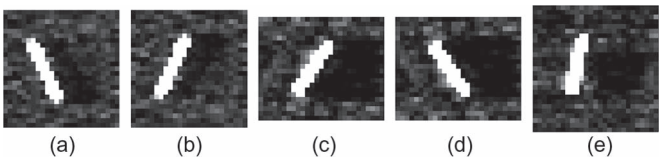


Fig. 1. Training set of sea mines in side-scan sonar images. Three images were used (a), (c), and (e). The images (b) and (d) are vertical reflections of (a) and (c), respectively, added to the set to increase variability. The pixels on the sea-mine highlights were saturated in order to diminish variability of target intensity in the training set, which is due to noisy acquisition.

Given a test image, typically, most patches belong to the background. In side-scan sonar, the appearance of background patches is determined by the backscattered energy from the seabed, which follows Rayleigh distributions for isotropic regions of the seafloor. In areas with more complex seafloor topography or backscatter from sand ripples, more complex models are required [25], [26].

A patch containing the target, an MLO, will typically include a small bright highlight, accompanied by a shadow region to the right or left of the highlight, dependent on the acquisition of the image. The shadow region is due to the MLO effectively blocking the sonar waves from reaching the region of the seabed adjacent to the sea mine [12]. The shadow region is usually larger than the highlight region in the image. Examples of a few sea mines, composing our training set, are presented in Fig. 1. Note that the pixels on the highlight of the sea mines in our training set were saturated to diminish variability in the highlight intensity due to noisy acquisition and differences in the reflectivity of the objects. This was done so that perturbations in the target model would result from differences in orientation and size and not from intensity.

The appearance of a patch containing a sea mine is determined by several parameters of the sea mine: the location of its center in the patch, its orientation in regard to the sensor, its size (length and width), its reflectivity, and the length of the shadow (determined by the height of the mine protruding above the seabed and the grazing angle). One could explicitly calculate these parameters for a test patch using shape analysis and compare them to the typical values learned from the patches in the training set to determine the existence of a target. The expected geometry of the target could also be imposed as prior information in a statistical model or heuristic filter. Our approach, on the other hand, compares the intrinsic parameters of the sea-mine appearance between patches, using the patches directly, without performing explicit shape analysis.

Given a set of training patches containing the target, we want to compare patches extracted from a test image to the training set. If a test patch is similar to the training set, we determine that a target has been detected in this patch. Our focus in this work is to define this notion of similarity between the test and training sets. We make two observations regarding the comparison of two patches containing targets. First, a target patch probably does not contain only pixels belonging to the signal of interest. The patch will usually also contain pixels belonging to the background, which are not of interest for determining whether the patch contains a target. Second, similar patches can be considered different realizations of the same scene with slight variations so that many of the pixels containing the target are identical and some of the pixels are different due to these variations. If the Euclidean distance is used to compare patches, all pixels in the patch are weighted evenly. Yet, it is desirable to ignore differences due to comparing background pixels in both patches. In addition, we want to put less emphasis on target pixels who are different due to slight variations in the specific realization of the given patch.

These goals can be achieved by associating a weight with each pixel in the patch which determines how important it is in terms of its signal content. Thus, we ensure that, when calculating the distance between patches, we are comparing only the relevant pixels. Obviously, it is tedious and inefficient to set such a weight vector manually for each and every patch in the training set. In the next section, we present a method to calculate the weight vector for each patch based on its local neighborhood in the training set. The variance of each pixel in the patch, estimated using a local neighborhood of training patches, yields an automatic method to obtain weight vectors with the desired properties.

Myers and Fawcett have addressed a similar problem when using the cross-correlation measure for template matching [21]. They propose using complementary templates, which are an inverse binary mask of a template model, to penalize areas of echoes or shadows that fall outside the ideal templates. However, their approach does not enable perturbations in the templates as the mask is binary, whereas we propose a weighted metric. Thus, to achieve a variation of orientations, they require a large number of templates of sea mines at varying aspects.

B. Invariant Metric

We denote by $Z_i \in \mathbb{R}^N$ the column stacked version of the $\sqrt{N} \times \sqrt{N}$ patch centered at the pixel i in the image. Let $\boldsymbol{\theta}$ be a vector of intrinsic parameters which determines the appearance of the sea mine, for example, the location of its center in the patch, its orientation in regard to the sensor, its size (length and width), its reflectivity, and the length of the shadow (which depends on the height of the mine protruding above the seabed and the grazing angle). The parameters in $\boldsymbol{\theta}$ are unknown and will be inferred by our method from the training data. We consider each sea-mine patch a sonar measurement of a sea mine, with the realization of the measurement dependent on the parameter vector and measurement noise. We assume that sea-mine patches with similar appearance have similar parameter vectors and are realizations of the same scene with slight perturbations.

We assume that the column stack of the patch Z_i is a vector of N nonlinear noisy measurements of the unknown intrinsic parameters

$$Z_i(x) = f(x; \boldsymbol{\theta}_i) + \eta_i(x), \qquad x \in \{1, \ldots, N\} \tag{1}$$

where $f(x; \boldsymbol{\theta})$ is a smooth nonlinear function mapping the parameter vector $\boldsymbol{\theta}$ to the x pixel in the patch and η is a zero-mean measurement noise with variance σ_η^2 independent of $\boldsymbol{\theta}$ and x. The specific pixel within the patch is denoted by x, which ranges from one to N. Note that this measurement model neglects explicit interactions between pixels in the patch. However, all pixels within the patch share the same mapping

f and the same parameter vector $\boldsymbol{\theta}$, providing an implicit connection between pixels.

Given a training set $\{\overline{Z}_i\}_{i=1}^{\overline{M}}$ of $\overline{M}$ patches, we calculate the local statistics of each patch, using its k nearest neighbors within the training set, denoted by $\mathcal{N}_i$. These nearest neighbors, which are similar in appearance to the given patch, can be seen as perturbations of the given patch. These perturbation are due to slight variations of a subset of parameters in $\boldsymbol{\theta}$, depending on the choice of the local neighborhood. Consider several training patches belonging to the same neighborhood, which all have the same center-of-mass location, yet differ in the orientation of the sea mine. The sense of similarity within this neighborhood is defined by the location, and we want to place a large weight on this location. The perturbations within the neighborhood are defined by changes in orientation, and we want to assign low weights to such differences in the orientation. In a similar manner, we can choose to collect together patches with similar orientations but different lighting conditions, yielding a different weight vector. Note that we are addressing *slight* perturbations and not the range of all possible values.

As this is a supervised approach, the user has control over how a local neighborhood is defined: what similarities determine these nearest neighbors. If all nearest neighbor patches of a training patch have a consistent value in a certain parameter, then similarity of that parameter is important when comparing other patches to the training patch. On the other hand, differences between the nearest neighbors determine what variations are allowed in the local model of the training patch. Perturbations of these parameters when comparing other patches to the training patch should be ignored or repressed in comparison to the consistent parameters. For a given neighborhood, we can separate the parameter vector into two sets: $\boldsymbol{\theta} = (\boldsymbol{\theta}^c, \boldsymbol{\theta}^v)^{\mathrm{T}}$. The parameters included in $\boldsymbol{\theta}^c$ are consistent within the neighborhood, while $\boldsymbol{\theta}^v$ contains parameters that have variability within the neighborhood. Controlling the definition of the local neighborhood determines to which parameters the model will be sensitive and to which parameters it will be invariant. Given a small set of close neighbors, we can model each pixel as

$$Z_i(x) = f\left(x; \boldsymbol{\theta}_i^c, \boldsymbol{\theta}_i^v\right) + \eta_i(x) \tag{2}$$

where $\boldsymbol{\theta}_i^c$ and $\boldsymbol{\theta}_i^v$ relate to the local neighborhood of Z_i.

Our goal is to empirically infer a model for each training patch, which will allow an invariance to the inconsistent parameters. However, we do not want to learn a shape model for our target and perform shape analysis for every patch to retrieve its parameter vector. Instead, we propose a data-driven approach which is based on the patch pixels and presents an implicit method of achieving this invariance. This is done via the empirical local variance vector of the pixels, calculated in the local neighborhood of patches.

The empirical local variance for each pixel in the training patch is estimated using

$$\hat{\sigma}_i^2(x) = \frac{1}{k} \sum_{\overline{Z}_j \in \mathcal{N}_i} \left(\overline{Z}_j(x) - \hat{\mu}_i(x)\right)^2 \tag{3}$$

where $\hat{\mu}_i(x)$ is the empirical local mean of pixel x and $k = |\mathcal{N}_i|$ is the number of nearest neighbors used in the empirical estimations. Note that this variance is local in the sense that it is calculated for a given pixel x based on the values $\overline{Z}_j(x)$, $\overline{Z}_j \in \mathcal{N}_i$ and is not a spatial variance within the patch $\overline{Z}_i$. Following our assumption, the set of nearest neighbors is such that some of the parameters are identical, i.e., the empirical local variance of these parameters among the neighbors is zero, whereas other parameters have high empirical variance.

Within the local neighborhood of a given patch, the intrinsic parameters defining the appearance of the patches are close. The differences between a given patch $\overline{Z}_i$ and one of its nearest neighbors $\overline{Z}_j \in \mathcal{N}_i$ can be seen as a perturbation of the given patch due to slight variations in the parameter vector. Assuming a locally linear model in the parameter space, a nearest neighbor patch $\overline{Z}_j$ can be written as

$$\begin{aligned} \overline{Z}_j(x) &= f(x; \boldsymbol{\theta}_j) + \eta_j(x) \\ &= f(x; \boldsymbol{\theta}_i) + \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i)(\boldsymbol{\theta}_j - \boldsymbol{\theta}_i) + \eta_j(x) \end{aligned} \tag{4}$$

where we have neglected higher order terms, θ_i is the corresponding parameter vector of $\overline{Z}_i$, and $\nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i)$ is the gradient of $f(x; \boldsymbol{\theta}_i)$. Denoting the entries of the parameter vector as $\boldsymbol{\theta} = (\boldsymbol{\theta}(1), \boldsymbol{\theta}(2), \ldots)^{\mathrm{T}}$, the gradient is given by the partial derivatives

$$\nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i) = \left(\frac{\partial f(x; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}(1)}, \frac{\partial f(x; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}(2)}, \ldots \right) \Big|_{\boldsymbol{\theta} = \boldsymbol{\theta}_i} \tag{5}$$

computed at $\boldsymbol{\theta} = \boldsymbol{\theta}_i$.

We now present the results of using this linear model in the empirical estimation of the mean and variance of $\overline{Z}_i(x)$. The full derivation is provided in Appendix I.

The empirical local mean is given by

$$\hat{\mu}_i(x) = \frac{1}{k} \sum_{\overline{Z}_j \in \mathcal{N}_i} f(x; \boldsymbol{\theta_j}) \approx f(x; \hat{m}_{\boldsymbol{\theta}_i}) \tag{6}$$

where we used the assumption that the noise has zero mean and $\hat{m}_{\boldsymbol{\theta}_i}$ is the empirical mean of the parameter vector in $\mathcal{N}_i$.

In a similar manner, plugging the linear model given in (4) and the empirical local mean (6) into (3) yields

$$\begin{aligned} \hat{\sigma}_i^2(x; \boldsymbol{\theta_i}) &= \frac{1}{k} \sum_{Z_j \in \mathcal{N}_i} \left(f(x; \boldsymbol{\theta_j}) + \eta_j(x) - f(x; \hat{m}_{\boldsymbol{\theta}_i})\right)^2 \\ &= \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i) \mathrm{Cov}(\boldsymbol{\theta}_i) \nabla_{\boldsymbol{\theta}} f(x; \boldsymbol{\theta}_i) + \sigma_\eta^2 \end{aligned} \tag{7}$$

where we used the assumption that the noise is independent of the signal and $\mathrm{Cov}(\boldsymbol{\theta}_i)$ depends on the empirical covariance of the parameter vector $\boldsymbol{\theta}$ within the local neighborhood of Z_i. Assuming that the parameters in $\boldsymbol{\theta}$ are independent of each other, the estimated covariance matrix $\mathrm{Cov}(\boldsymbol{\theta}_i)$ is diagonal with the empirical variance of each parameter as an element on the diagonal. We denote the diagonal as the vector $\sigma_{\boldsymbol{\theta}}^2$, i.e., a vector containing the empirical variances of each parameter $\theta \in \boldsymbol{\theta}$, and the diagonal matrix with $\sigma_{\boldsymbol{\theta}}^2$ as its diagonal by Ω. Therefore, the left term in the right-hand side of (7) can be rewritten as

$$\begin{aligned} \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i) \Omega_i \nabla_{\boldsymbol{\theta}} f(x; \boldsymbol{\theta}_i) &= \nabla_{\boldsymbol{\theta}^c} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i) \Omega_i^c \nabla_{\boldsymbol{\theta}^c} f(x; \boldsymbol{\theta}_i) \\ &\quad + \nabla_{\boldsymbol{\theta}^v} f^{\mathrm{T}}(x; \boldsymbol{\theta}_i) \Omega_i^v \nabla_{\boldsymbol{\theta}^v} f(x; \boldsymbol{\theta}_i) \end{aligned} \tag{8}$$

where Ω_i^c and Ω_i^v are diagonal matrices with the empirical variance vectors of the parameter sets $\boldsymbol{\theta}_i^c$ and $\boldsymbol{\theta}_i^v$ as their diagonals, respectively. Since we defined $\boldsymbol{\theta}_i^c$ as the parameters which are consistent within the neighborhood, the empirical variances $\sigma^2_{\boldsymbol{\theta}_i^c} \to 0$. Finally, the empirical local variance of Z_i at pixel x is given by

$$\hat{\sigma}_i^2(x) = \hat{\sigma}_v^2(x) + \sigma_\eta^2(x) \tag{9}$$

where $\hat{\sigma}_v^2(x) = \nabla_{\boldsymbol{\theta}^v} f^{\mathrm{T}}(x;\boldsymbol{\theta}_i)\Omega_i^v \nabla_{\boldsymbol{\theta}^v} f(x;\boldsymbol{\theta}_i)$. We have obtained that the local empirical variance of the pixel depends on perturbations in the parameter vector within the local neighborhood of the training patch. Since the measure of similarity used to define the neighborhood essentially defines how the parameters are divided between $\boldsymbol{\theta}^c$ and $\boldsymbol{\theta}^v$, it effectively controls the variance. Note that, although we are calculating empirical estimations of the statistics for each pixel in the patch independently, there is an implicit dependence between pixels, as the neighborhood used to calculate these statistics depends on the appearance of the entire patch and not just the pixel.

If we set the weight associated with each pixel to be the inverse local variance, we obtain a weight vector with the desired properties. If a pixel x has consistent values among the patches belonging to the local neighborhood, then $\hat{\sigma}_v(x) \to 0$, and the pixel $\overline{Z}(x)$ is associated with weight $1/\sigma_\eta^2$. If a pixel x has inconsistent values among the patches belonging to the local neighborhood, then $\hat{\sigma}_v^2(x) > 0$, and the pixel $\overline{Z}(x)$ is associated with weight $1/(\sigma_\eta^2 + \hat{\sigma}_v^2(x)) < 1/\sigma_\eta^2$. Such a pixel has high local variance either due to variability in the model, if it is a pixel belonging to the target, or due to differences in the background, if it belongs to the background. Thus, pixels which can be consistently associated with the signal have a larger weight than pixels which account for perturbations in the signal, or background pixels which are not part of the desired signal, yet belong to the patch. For example, the pixels which contain the central body of the target are assigned high values in the corresponding elements of the weight vector, which is desirable as we want to penalize patches which differ in the values of these pixels.

This property is demonstrated for the case of sea mines in Fig. 2. Fig. 2(a)–(c) displays two patches, each containing a sea mine. The sea mine is composed of a bright highlight and a dark shadow to its right. Fig. 2(b)–(d) displays the inverse variance vector calculated for each patch, reshaped as $\sqrt{N} \times \sqrt{N}$ patch. The pixels of the central parts of the highlight and shadow are heavily weighted, whereas there is a low weight surrounding the outline of the sea mine. These weights force a small distance between pixels in the highlight and shadow while enabling small variations in the appearance of the sea mine in regard to its orientation and position. In addition, the pixels which are background pixels have a lower weight than those belonging to the sea mine. This demonstrates that the inverse local variance vector realizes the desired properties. Fig. 2(e) displays three patches from the local neighborhood of the patch shown in (c). These patches have similar center-of-mass positions and length, yet differ, for example, in orientation and background. These similarities and perturbations account for the low and high weights in the inverse local variance vector.

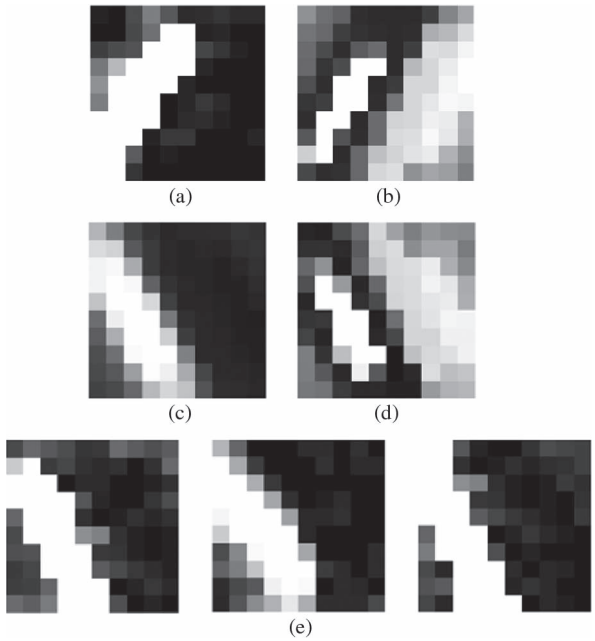


Fig. 2. (a) and (c) Two training patches containing sea mines with a bright highlight and a dark shadow to the right. (b) and (d) Inverse local variance vector of each patch, reshaped as $\sqrt{N} \times \sqrt{N}$ patch. White corresponds to a high weight, and black corresponds to a low weight. The elements corresponding to the pixels in the central parts of the highlight and shadow are heavily weighted, whereas there is a low weight surrounding the outline of the sea mine and the background. (e) Three patches from the local neighborhood of patch (c). These patches have similar center-of-mass positions and length, yet differ, for example, in orientation and background. These perturbations are accounted for in the weight vector (d).

We associate each training patch with the estimated local statistical model composed of its local empirical mean $\hat{\mu}_i(x), x \in 1, \ldots, N$ and the local empirical variances of each pixel in the patch $\hat{\sigma}_i^2(x)$. The mean is used to represent the patch, and the variances are used to weight each pixel by its importance. To facilitate the desired weighting, we propose the following squared weighted distance between pairs of patches

$$d^2(\overline{Z}_i, \overline{Z}_j) = \sum_{x=1}^{N} \frac{\left(\hat{\mu}_i(x) - \hat{\mu}_j(x)\right)^2}{\hat{\sigma}_i^2(x) + \hat{\sigma}_j^2(x)} \tag{10}$$

where $\hat{\mu}_k$ and $\hat{\sigma}_k$ are the empirical local mean vector and local variance vectors of the patch $\overline{Z}_k, k \in \{1, \ldots, \overline{M}\}$, respectively. Thus, the patches are compared via their local model, and the pixels are weighted according to their combined importance in both patches. Pixels with high local variance in either patch are assigned a low weight whereas low variance in both patches corresponds to a high weight.

Note that the number of neighbors used in defining the local neighborhood should be limited or restricted by computing an error threshold between the given patch and a neighbor candidate to ensure that patches are similar enough to be used in the empirical calculations. This threshold should be application dependent and determined empirically, based on the size of the patches used, the typical intensities of the target, and the variability within the training set. In addition, in the specific case of sea mines, since the spatial support of the highlight is small, averaging too many possible perturbations of the orientation and position of the sea mine will attenuate

the highlight in the empirical mean. In our experiments, using $k = 16$ neighbors for each training patch yielded good results. However, increasing the number to $k = 32$ resulted in a blurred target model.

C. Controlling the Invariants via the Training Set

The proposed distance defined in (10) provides a metric with "soft" invariance to certain properties of the signal. This invariance can be controlled and provides the user with a method to define characteristics of the target to which they want to be invariant. This is done by the choice of the training set and definition of the similarities which determine the local neighborhoods within the training set. In Section VII-A, we provide a numerical example for 1-D signals on creating shift and scale-invariant metrics, by controlling the parameters of the training set.

In the case of sea mines, it is desirable to enable an invariance to slight differences in the rotation and position of the sea mine and the intensity and size of the highlight and shadow, in comparison to the sea mines in the training set. For example, this metric allows one to consider sea mines for a small range of orientations as similar to a given patch, without having to explicitly define this range or calculate it. If the local neighborhoods are determined such that they include patches at slightly varying orientations as in Fig. 2(e), this creates an invariance to slight variation in the rotation parameter. The invariance is achieved implicitly via the weight vector since low values are assigned to the pixels which correspond to slight rotations of the sea mine in the training patch. In terms of the implicit parameter vector $\boldsymbol{\theta}$, the local variability in the rotation parameter effectively sets it in $\boldsymbol{\theta}^v$. This enables one to limit the size of the training set and not require a training example for every configuration of the parameter vector.

In addition, the proposed weighted distance can be used to enhance the target while repressing the background. If all sea mines in the training set will have similar background values thus that the empirical value of the background pixels in the patch is on the same order of that of the highlight or shadow, then the weighted distance will not be invariant to the background. This should be taken into consideration so that the training set will have varying backgrounds. For example, if the training set is created using synthetic sea mines placed on real or simulated seabed backgrounds, then a different background should be used for each sea mine. If real sea-mine examples are used for the training set, then it is preferable to use sea mines on different types of backgrounds. Thus, the calculated distance will reduce the affinity between a training patch and a test patch due to their having similar background values, providing an invariance to the background.

In related work in the field of sea-mine detection, the authors in [17], [20], and [26] propose the use of simulators to create images of synthetic sea mines on real or simulated seabed backgrounds for use as training data. In [26], Coiras *et al.* introduce a multiresolution statistical approach to seabed reconstruction from side-scan sonar. In the paper, they present an application of this procedure in which synthetic objects are artificially embedded into a side-scan image. This can be used to produce a training set of a realistic environment. Such simulators enable one to directly control the parameters of sea mines and the seabed appearance in a training set. Therefore, they can be used to determine local neighborhoods based on similarity or dissimilarity of given model parameters. This enables one to create a training set with local neighborhoods capturing the desired perturbations and thus determining the desired invariance: to the background, orientation, or highlight intensity, for example.

III. Graph-Based Intrinsic Embedding

Given a test image, all its overlapping patches are extracted, providing a test set of M image patches $\{Z_i\}_{i=1}^{M} \in \mathbb{R}^N$. A graph-based algorithm is used to embed the high-dimensional image patches in a low-dimensional space $\mathbb{R}^d$, $d < N$. As proposed in [23], [24], and [27], we define a nonsymmetric weighted square distance between a training patch $\overline{Z}_j$ and a test patch Z_i, using the new metric, as

$$a^2(Z_i, \overline{Z}_j) = \sum_{x=1}^{N} \left(Z_i(x) - \hat{\mu}_j(x)\right)^2 / \hat{\sigma}_j^2(x). \tag{11}$$

An affinity matrix, based on this distance, is defined between the data set of all image patches $\{Z_i\}_{i=1}^{M}$ and the training set $\{\overline{Z}_j\}_{j=1}^{\overline{M}}$

$$\mathbf{A}[i,j] = \exp\left\{-a^2(Z_i, \overline{Z}_j)/\epsilon^2\right\} \tag{12}$$

where ϵ is a scale factor. The Gaussian function further enhances the notion of locality as defined by the proposed metric, as patches with a distance larger than ϵ have a negligible affinity. The scale ϵ is set to be on the order of the median distance within the training set $\{\overline{Z}_j\}_j$, as is common practice. This parameter can be fine tuned to obtain optimal results. Note that setting ϵ to be too large will result in *all* test patches being similar to the training set and the target detection will fail. On the other hand, setting ϵ to be too small will result in none of the test patches, including those containing targets, to be similar to the training set, and the target detection will also fail. Also, although ϵ is a global scale, the proposed metric is adaptive to each training patch $\overline{Z}_j$ via the local empirical variance of each patch $\hat{\sigma}_j^2(x), x \in \{1, \ldots, N\}$.

The matrix $\mathbf{A}$ is an $M \times \overline{M}$ affinity matrix, and we assume that $M > \overline{M}$. We define the symmetric kernel $\mathbf{W} = \mathbf{A}^{\mathrm{T}}\mathbf{A}$, which is an $\overline{M} \times \overline{M}$ matrix

$$\mathbf{W}[i,j] = \sum_{l=1}^{M} \mathbf{A}[l,i]\mathbf{A}[l,j]. \tag{13}$$

This kernel can be interpreted as an affinity metric between any two training patches via all patches in the data set [23], [24]. Following [27], this kernel can be rewritten as the convolution of two Gaussians, and using the convolution theorem, the result is proportional to a symmetric affinity matrix given by

$$\mathbf{W}^{\mathrm{sym}}[i,j] = \exp\left\{-\sum_{x=1}^{N} \left(\overline{\mu}_i(x) - \overline{\mu}_j(x)\right)^2 / \left(\overline{\sigma}_i^2(x) + \overline{\sigma}_j^2(x)\right)\epsilon^2\right\}. \tag{14}$$

The proof is provided in Appendix II. The matrix is a symmetric affinity matrix based on the distance defined in (10). Thus, the symmetric kernel on the training set, defined via the data set, approximates the direct affinity between the training set patches.

The eigendecomposition of the matrix $\mathbf{W}$ (13) yields a set of decreasing eigenvalues $\{\lambda_l\}$ and eigenvectors $\{\phi_l\} \in \mathbb{R}^{\overline{M}}$. The spectrum of affinity matrices such as $\mathbf{W}$ exhibits a spectral gap, with only a few eigenvalues close to one and all of the rest quickly tending to zero. Thus, the leading d eigenvectors $\{\phi_l\}_{l=1}^{d}$, corresponding to the d largest eigenvalues $\{\lambda_l\}_{l=1}^{d}$, provide a lower dimensional embedding of the training set $\{\overline{Z}_j\}_{j=1}^{\overline{M}}$, as seen via the data set $\{Z_i\}_{i=1}^{M}$. The dimension d can be determined by retaining only the eigenvalues for which $\lambda_l > \delta\lambda_1$, where a typical value for δ is 0.1. Note that d is an estimate of the intrinsic dimensionality of the data and does not depend on the dimension of the representation, i.e., the patch size N. These eigenvectors are also the singular right vectors of $\mathbf{A}$ and can be used to calculate the singular left eigenvectors $\{\psi_l\} \in \mathbb{R}^M$ of $\mathbf{A}$ by [27]

$$\psi_l = \frac{1}{\sqrt{\lambda_l}}\mathbf{A}\phi_l. \tag{15}$$

Thus, an eigendecomposition of $\mathbf{W}$ provides an efficient manner in which to calculate the singular left eigenvectors of $\mathbf{A}$, which are used for low-dimensional embedding of the data set $\{Z_i\}_{i=1}^{M}$. This embedding is expected to reveal which patches in the image are similar to the reference set.

Following [24], instead of calculating the eigenvectors of the Markov operator, we calculate the eigenvectors of the normalized graph Laplacian, which converges to the continuous Laplace–Beltrami operator on the manifold [27], [28]. This normalization handles nonuniform sampling of the measurements so that the embedding does not depend on the density of the data points [29], [30]. First, the kernel $\mathbf{W}$ is normalized by its density

$$\widetilde{\mathbf{W}} = \mathbf{Q}^{-1}\mathbf{W}\mathbf{Q}^{-1} \tag{16}$$

where the elements of the diagonal matrix $\mathbf{Q}$ are the sum of the rows of $\mathbf{W}$: $\mathbf{Q}[i,i] = \sum_j \mathbf{W}[i,j]$.

The normalized graph Laplacian is then constructed for this kernel, yielding an anisotropic kernel

$$\widetilde{\mathbf{P}} = \widetilde{\mathbf{D}}^{-1}\widetilde{\mathbf{W}} \tag{17}$$

where $\widetilde{\mathbf{D}}$ is a diagonal matrix whose elements are $\widetilde{\mathbf{D}}[i,i] = \sum_j \widetilde{\mathbf{W}}[i,j]$. The spectral decomposition of $\widetilde{\mathbf{P}}$ yields the set of eigenvalues $\{\tilde{\lambda}_l\}$ and eigenvectors $\{\widetilde{\phi}_l\}$, which we assume to be of unit norm. The eigenvalues of $\widetilde{\mathbf{P}}$ are all nonnegative and bounded by one, sorted in decreasing order with $\tilde{\lambda}_0 = 1$. The first eigenvector $\widetilde{\phi}_0$ is a uniform column vector. The eigenvectors $\{\widetilde{\phi}_l\}$ are discrete approximations of the eigenfunctions of the Laplace–Beltrami operator on the manifold of the training set.

Now, we can calculate the eigenvectors $\{\widetilde{\psi}_l\} \in \mathbb{R}^M$ for the data set, as an out-of-sample extension of the eigenvectors $\{\widetilde{\phi}_l\} \in \mathbb{R}^{\overline{M}}$, $\overline{M} < M$

$$\widetilde{\psi}_l = \frac{1}{\sqrt{\tilde{\lambda}_l}}\widetilde{\mathbf{A}}\widetilde{\phi}_l. \tag{18}$$

The matrix $\widetilde{\mathbf{A}}$ is given by

$$\widetilde{\mathbf{A}} = \mathbf{D}^{-1}\mathbf{A}\mathbf{Q}^{-1} \tag{19}$$

where $\mathbf{D}$ is a diagonal matrix whose elements are

$$\mathbf{D}[i,i] = \sum_j (\mathbf{A}\mathbf{Q}^{-1})[i,j]. \tag{20}$$

The matrix $\widetilde{\mathbf{A}}$ provides an efficient out-of-sample extension from the embedding of the training set to the embedding of the data set, by a weighted mean of the eigenvectors $\{\widetilde{\phi}_l\}$.

The supervised graph yields a lower dimensional representation of the image, using the eigenvector entries as new coordinates for each pixel in the image. Using the first d eigenvectors, excluding the first trivial eigenvector, we embed the M pixels onto the eigenvectors $\widetilde{\psi}_l$

$$\widetilde{\Psi}_d : Z_i \to \left(\widetilde{\psi}_1(i), \widetilde{\psi}_2(i), \ldots, \widetilde{\psi}_d(i)\right). \tag{21}$$

IV. Target Detection

The low-dimensional representation is expected to separate the target from the background clutter. A construction of the embedding also provides a detection score. Calculation of the eigenvector $\widetilde{\psi}_l$ via the affinity matrix $\widetilde{\mathbf{A}}$ shows that each element in $\widetilde{\psi}_l$ is proportional to a weighted mean of the elements of $\widetilde{\phi}_l$, which are the embedding of the training set. Consider a background patch Z_i which is equally distant from all training patches. The elements of the corresponding row vector $\widetilde{\mathbf{A}}[i,:]$ are uniform, all equaling $1/\overline{M}$, so that this vector equals $\widetilde{\phi}_0^{\mathrm{T}}$. Since the eigenvectors are orthonormal, the embedding of an ideal background patch is given by

$$\widetilde{\psi}_l(i) = \frac{1}{\sqrt{\tilde{\lambda}_l}}\widetilde{\mathbf{A}}[i,:]\,\widetilde{\phi}_l = \frac{1}{\sqrt{\tilde{\lambda}_l}}\,\widetilde{\phi}_0^{\mathrm{T}}\,\widetilde{\phi}_l = 0. \tag{22}$$

Thus, an ideal background patch is embedded at the origin. To measure how close a patch is to being an ideal background patch, calculating the distance in the embedding space between the image patches and the ideal background patch is essentially calculating the norm of the embedding. Therefore, calculation of the eigenvectors $\{\widetilde{\psi}_l\}_{l=1}^{d}$ via the affinity matrix results in all background patches being clustered in a d-dimensional ball around the origin. On the other hand, the patches which contain a target have high affinity to the patches in the training set to which they are similar, under the weighted distance. Therefore, their embedding is meaningful and removed from the origin.

This is demonstrated in Fig. 3. In Fig. 3(b), the first three coordinates of the embedding are displayed for all patches extracted from the side-scan sonar image in Fig. 3(a). The

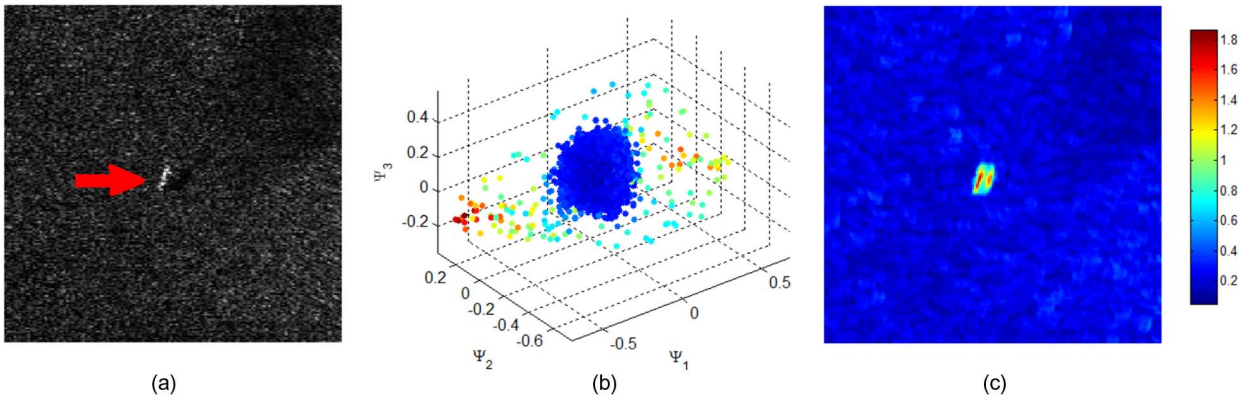


Fig. 3. (a) Side-scan sonar image containing a sea mine indicated by the red arrow. (b) First three coordinates of the embedding $\widetilde{\Psi}$, calculated for the image, using the training set in Fig. 1. Each data point i is colored according to the embedding norm $\|\widetilde{\Psi}_d(i)\|^2$. (c) Image in (a) with each pixel colored according to the embedding norm.

points are colored according to the embedding norm $\|\widetilde{\Psi}_d(i)\|^2$. Fig. 3(c) displays the image in Fig. 3(a) with each pixel colored according to the embedding norm. The target is easily distinguishable from the background according to this score. This demonstrates the property that the embedding coordinates of most of the patches are scattered in a ball around the origin while the few patches corresponding to the sea mine are embedded distantly from the origin.

Therefore, calculating the norm of the embedding

$$\left\|\widetilde{\Psi}_d(i)\right\|^2 = \sum_{l=1}^{d} \widetilde{\psi}_l^2(i) \tag{23}$$

for every pixel i can be used as a target detection score. The background patches will have a norm close to zero, whereas the target will have a meaningful norm. Depending on the application, the score can be thresholded to produce a binary map of detection, or the patches with top-ranking scores can be outputted to be inspected by the user.

It should be noted that, in simple images, the row sum of the affinity matrix (11), given by $S(i) = \sum_{j=1}^{M} \mathbf{A}[i,j]$, is a reasonable indicator of whether a target is present in the image. Summing the proposed affinity between a given patch and all training patches could be used for detection. This can be seen in Fig. 4. The first column displays the original side-scan sonar images, the second column displays the affinity sum S for each pixel in the image, and the third column displays the embedding norm $\|\widetilde{\Psi}_d(i)\|^2$ for each pixel in the image. For the simple case of the sea mine in Fig. 4(a), it is easily seen that thresholding S results in a detection of the sea mine.

However, for more complex images, such as Fig. 4(d), the affinity in itself does not give a good enough indication of whether a patch contains a sea mine. In Fig. 4(e), it is difficult to determine whether a patch containing a sea mine exists, and if so, which patches contain a sea mine as the S values have a high variance and, in addition, many patches scattered throughout the image have a high affinity sum. However, in Fig. 4(f), the norm of the embedding clearly separates the sea mine from the background. This demonstrates that the proposed affinity in itself is insufficient to determine the existence of a sea mine and the embedding is required to provide a meaningful representation of the data. In the next section, we show that, for the embedding to provide a meaningful representation, the metric used in the graph construction needs to be appropriate to the application.

V. Affinity Measure Using PCA-Based Local Model

We compare the affinity defined using the proposed weighted distance to the affinity proposed in [23] and [24]. There, the affinity kernel is defined by means of a linear projection operator onto local models of the training set. This local data-driven model for the training set is used to enhance the connection between nodes that correspond to the same training model.

First, an affinity measure is defined to measure the similarity between two data points, for example, using a Gaussian kernel

$$\mathbf{A}[i,j] = \exp\left\{-\|Z_i - \overline{Z}_j\|^2/\epsilon_{\text{euc}}^2\right\} \tag{24}$$

where $\epsilon_{\text{euc}} > 0$ is a scale parameter.

As in the affinity proposed in Section II, the local neighborhoods of each training patch are used to define a local model. Each training patch is represented by its mean (6) and its tangent space, calculated using PCA. Using the local covariance matrix for each patch C_j, which characterizes the tangent space at μ_j, the first few principal components define a model for each patch. The local covariance matrix is estimated using the local neighborhood by

$$\hat{C}_j = \frac{1}{k} \sum_{\overline{Z}_i \in \mathcal{N}_j} (\overline{Z}_i - \hat{\mu}_j)^{\mathrm{T}} (\overline{Z}_i - \hat{\mu}_j). \tag{25}$$

Let $\{\overline{v}_{j,l}\}_{l=1}^{L}$ be the set of L normalized principal components of the training patch $\overline{Z}_j$. A linear projection operator onto the local PCA model of the $\overline{Z}_j$ is defined by

$$P_j(Z_i) = \hat{\mu}_j + \sum_{l=1}^{L} \langle Z_i - \hat{\mu}_j, \overline{v}_{j,l} \rangle \overline{v}_{j,l}. \tag{26}$$

This projection is used in the graph construction by defining a pairwise metric between the image patches and the training

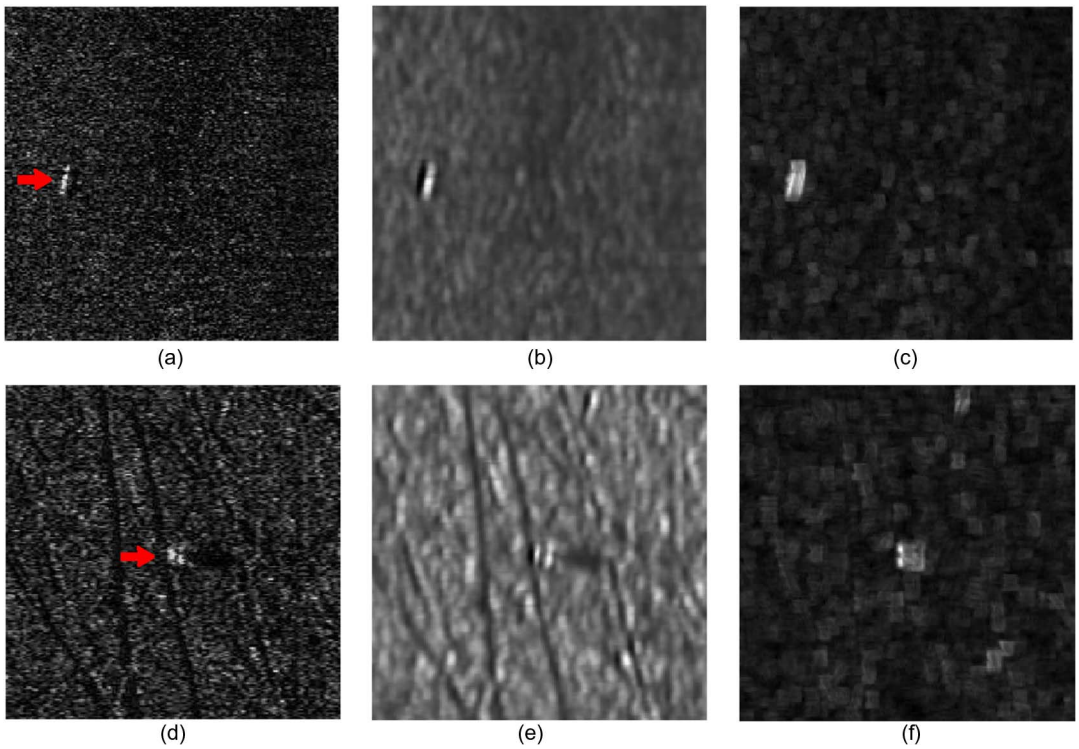


Fig. 4. (a) and (d) Side-scan-sonar image containing a sea mine indicated by the red arrow. (b) and (e) Each pixel associated with the sum of the affinity between its patch and the training patches S_i. (c) and (f) Each pixel associated with the norm of the low-dimensional embedding $\|\widetilde{\Psi}_d(i)\|^2$. For (top row) a simple image, both measures are suitable for separating the target from the background. For (bottom row) a more complex image, the affinity in itself is insufficient to separate the target from the background, yet the low-dimensional embedding provides meaningful representation.

patches, based on the linear projection onto the local models. The metric is given by

$$a^2_{P_j}(Z_i, \overline{Z}_j) = \|P_j(Z_i) - \hat{\mu}_j\|^2 = \sum_{l=1}^{L} \left(\langle Z_i - \hat{\mu}_j, \overline{v}_{j,l}\rangle\right)^2 \quad (27)$$

where we used the fact that the principal components are orthonormal.

Following [23] and [24], a nonsymmetric kernel is defined between the training set and all patches of a test image as

$$\mathbf{A}[i,j] = \exp\left\{-\frac{\|Z_i - \overline{Z}_j\|^2}{\epsilon^2_{\text{euc}}} - \frac{\|P_j(Z_i) - \hat{\mu}_j\|^2}{\epsilon^2_{\text{PCA}}}\right\} \quad (28)$$

where ϵ_{euc} and ϵ_{PCA} are scale parameters, which we set based on the training set. The scale ϵ_{euc} was set as the mean of the Euclidean distances between each patch and its 32 closest neighbors. Similarly, ϵ_{PCA} was set as the mean of the projection distances between each patch and its 32 closest neighbors. This enabled an automatic method to set the scale parameters. Given the PCA-based affinity, the graph-based embedding and detection are carried out as explained in Sections III and IV, respectively. In the following section, we discuss the advantages and disadvantages of both this approach and our approach. In Section VII-B, both methods are applied to the real-world task of sea-mine detection in side-scan sonar images.

VI. Discussion

In related work, two metrics have been proposed in constructing the affinity between data points in supervised graph-based frameworks. The first, reviewed in the previous section, proposes constructing a projection-based metric between the training and test sets [23], [24]. In this approach, the principal components are calculated in local neighborhoods of the training sets. The metric consists of projecting the difference between the test patch and the empirical mean of the training patch onto the principal components. Effectively, this means weighting the pixel differences $Z(x) - \hat{\mu}(x)$ by the values of the entries of the principal components $\{v_l\}_{l=1}^{L}$. Yet, in the application of target detection, the principal components of a given local neighborhood in the training set correspond to the factors which vary the most in the neighborhood. These tend to be the outline of the target or areas in the background, both of which are less important in terms of signal content than the main body of the target. Thus, instead of penalizing differences due to dissimilarity in the central regions of the target, this metric penalizes for differences due to perturbations between the model and the test patch.

This is shown in Fig. 5, which displays the first two principal components $\overline{v}_1$ and $\overline{v}_2$ of the training patch in Fig. 2(c). Both principal components correspond to perturbations in the orientation and shape of the target. On the other hand, the entries of the principal components corresponding to pixels on the central parts of the highlight and shadow equal zero. Thus, differences in the important part of the signal, the central regions of the target, will be weighted by zero, whereas differences on the outlines will be enhanced.

The use of the projection operator is appropriate for comparing distances within the same model. It is also useful in

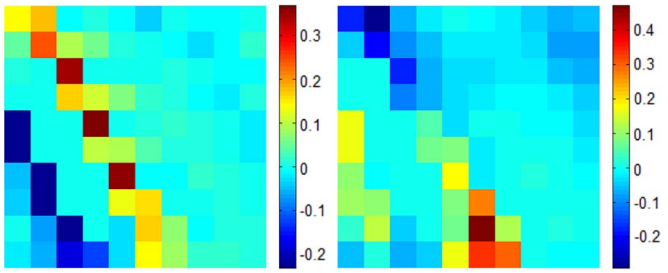


Fig. 5. First two principal components calculated for the patch shown in Fig. 2(c) based on its local neighborhood. The dominant values in the principal components correspond to the perturbations in the orientation and shape of the target. The pixels corresponding to the central parts of the target equal zero.

canceling the orthogonal component to a training patch when comparing between patches, as in the problem of intersecting textures described in [24]. However, for the purpose of target detection, we want to determine whether a given patch belongs to the training model. Yet, this projection enhances the difference between a test patch and a training patch due to the variability of the appearance of the model. Consider a test patch which belongs to the model described by the jth training patch, for example, $Z_i = \hat{\mu}_j + \overline{v}_{j,1}$. Then, $a^2_{P_j}(Z_i) = 1$, which results in lowering the affinity between this patch and the training patch, and that is the opposite of our purpose. For patches which do not belong to the model, such as background patches, the result of the projection is arbitrary, so that different background patches will receive a range of values in the projection distance, regardless of their true association with the model. Thus, this operator is also not useful in separating the background from the target. On the other hand, our proposed affinity represses the difference between the patches, arising from the variability in the appearance of the test patch as compared to $\hat{\mu}_j$. This is demonstrated in our results in Section VII-B.

A second metric used in graph-based processing is a Mahalanobis-based metric [27], [28], [31]

$$a^2_{C_j}(Z, \overline{Z}_j) = (Z - \hat{\mu}_j)^{\mathrm{T}} C_j^{-1} (Z - \hat{\mu}_j) \tag{29}$$

where the covariance matrix C_j is calculated using the local neighborhood of the training point $\overline{Z}_j$ as in (25). The covariance matrix has low rank, and therefore, the inverse is typically calculated via the principal components

$$C_j^{-1} = \sum_{l=1}^{L} \gamma_{j,l}^{-1} \overline{v}_{j,l} \overline{v}_{j,l}^{\mathrm{T}} \tag{30}$$

where $\gamma_{j,l}$ denotes the eigenvalues of the covariance matrix. Thus, in this metric, as opposed to the PCA-based metric, the principal components are weighted inversely so that the dominant principal components have the lowest weight. This counteracts the disadvantage of PCA as the components that account for the most variability in the covariance matrix are assigned low weights, essentially repressing them. This is similar to our method, where the pixels with the highest variance are assigned the lowest weights. However, our main requirement was to assign high weights to the pixels that have low variance or do not vary at all. The disadvantage of using the Mahalanobis metric is that the factors representing very low variability in the data are not evident in the principal components. There is essentially no way to distinguish these components which are meaningful from the components which are due to the covariance matrix having low rank. This was the motivation for the metric that we proposed, which was inspired by the Mahalanobis metric.

To summarize, when applying manifold learning in a supervised framework, the choice of metric should be appropriate to the application and expected measurements. One needs to decide, for example, whether it is important to repress the tangent space of the training points or to allow for perturbations in the data in comparison to the training set.

In this paper, we have focused on local models for metric learning, whose common property is to construct a metric which is invariant to certain properties of the target in the training set. We remark that there are other transform-based methods that construct invariants, such as the scale-invariant feature transform [32] and histogram of oriented gradients [33]. The recently introduced scattering transform, computed with a deep convolutional network [34], provides a stable translation-invariant representation and has achieved state-of-the-art results in texture classification [35], [36]. These transforms provide predefined invariance to certain properties such as dilation, orientation, changes in illumination, and translations. Our approach, on the other hand, builds a data-adaptive invariant metric, where the invariance implicitly arises by the notion of similarity within the training set. Regarding target detection, this enables one to suppress the background pixels when comparing patches and compare only the relevant signal content. On the other hand, a general predefined invariant transform will incorporate the background pixels into its feature vector and not weight them differently than the target pixels. In the specific case of sea mines in side-scan sonar, it should be noted that orientation-invariant features are problematic. Only the highlight appears at different orientations, whereas due to the acquisition process, the accompanying shadow is always along the range direction, regardless of the orientation angle of the sea mine. In light of the results of our research, an interesting future direction is to explore the scattering transform and features learned by convolutional networks for the purpose of sea-mine detection.

VII. Experimental Results

A. Toy Problem

We demonstrate our method on the following toy problem. Consider a family of exponential 1-D signals: $f(x) = \exp\{-(x-b)^2/2a^2\}$. Two parameters control the signal, the location of its center of mass b and its scale a; thus, $\boldsymbol{\theta} = (a\ b)^{\mathrm{T}}$. The signal is measured by 20 sensors located at $x = \{1, 2, \ldots, 20\}$, and each sensor has an independent noise with a standard deviation that is equal to 0.001. Such a model is similar to the one presented in [37] for a biological target acquired by a 1-D sensor array. We design two invariant distances by controlling the local neighborhood of the training measurements. An analysis of these distances is provided in Appendix III.

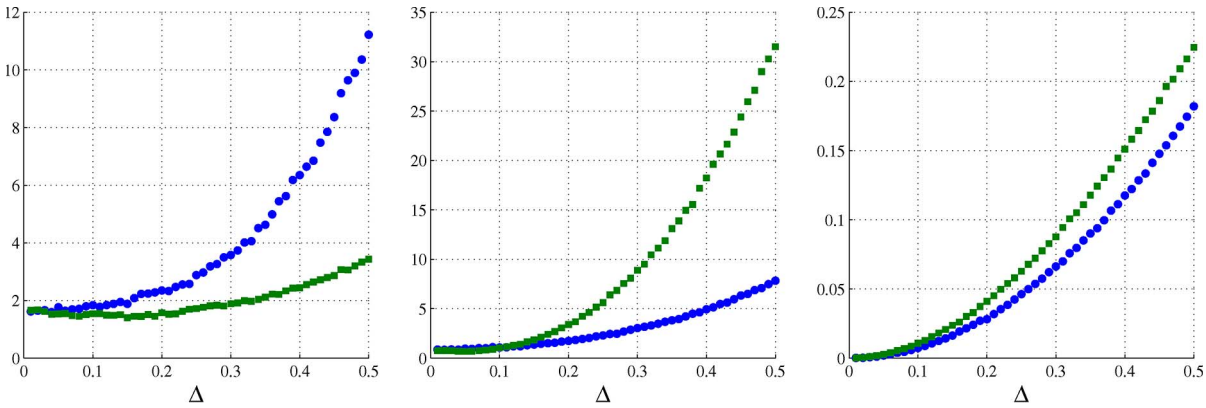


Fig. 6. Comparison of (left) the scale-invariant distance, (center) the shift-invariant distance, and (right) the Euclidean distance for perturbations in (green squares) scale and (blue circles) shift. The figure plots the distance between the training measurement and test measurements, where (green squares) we set the shift and vary the scale ($\Delta b = 0$, $\Delta a = \Delta > 0$) or (blue circles) set the scale and vary the shifts ($\Delta a = 0$, $\Delta b = \Delta > 0$).

We examine a training measurement with $b = 10$ and $a = 1.2$, and we design two invariant distances for this training measurement. The first is a scale-invariant distance. We set $b = 10$ and take ten other measurements with various scale parameters $a \in [0.58, 2.45]$. Calculating the model following (3) and (6) and plugging them into (12) yield a scale-invariant distance. The second distance that we design is a shift-invariant distance. We set $a = 1.2$ and take ten other measurements with various scale parameters $b \in [9.3, 10.8]$. This yields a shift-invariant distance.

Fig. 6 displays the distances calculated from the training measurement to two sets of test measurements. In the first set, we set the shift parameter $\Delta b = 0$ and vary the scale $\Delta a = \Delta > 0$ (green squares). In the second set, we set the scale parameter $\Delta a = 0$ and vary the shift parameter $\Delta b = \Delta > 0$ (blue circles). In Fig. 6 (left), the distances are calculated using the scale-invariant metric. Measurements corresponding to differences in the shift parameter have a greater distance from the training measurement than measurements corresponding to differences in the scale parameter. Thus, this distance is indeed scale invariant, penalizing differences in shifts while repressing differences in scale. In Fig. 6 (center), the distances are calculated using the shift-invariant metric. Here, we see the inverse trend: Measurements corresponding to differences in the scale parameter have a greater distance from the training measurement than measurements corresponding to differences in the shift parameter. Thus, we have indeed obtained a shift-invariant distance, as intended. In Fig. 6 (right), the distances are calculated using the Euclidean distance between measurements. Here, the sensitivity to differences in the parameters is similar, with differences in scale having a slightly larger impact on the distance. This follows the result that we obtained in our analytical derivation in Appendix III: The Euclidean distance is more similar to the shift-invariant distance.

B. Side-Scan Sonar

We demonstrate the proposed method for sea-mine detection in real side-scan sonar images, achieving a high detection rate. The sea mines in the images are the required targets, and the reflections from the seabed are considered normal background clutter.

We evaluated our method on a set of 44 side-scan sonar images with sea mines, where we cropped the image to size 200 (range) $\times$ 200 (cross-range) cells with a region containing a sea mine. The ratio of a cell's range dimension to cross-range dimension is 15:15 (cm), and the images were encoded in 8-b gray scale. Typical dimensions of a sea mine in these images are approximately 15 pixels by 3 pixels for the highlight, and the length of the shadow in the range direction is roughly about 15 pixels. These images were collected by the Naval Surface Warfare Center Coastal System Station (Panama City, FL, USA) and exhibit drastic changes in background clutter.

The size of the patch N in the algorithm should be determined by prior knowledge on the expected typical size of the target and the sonar resolution. The patch size should be such that it covers a significant portion of the target but does not necessarily have to contain the entire target. Based on the expected size of the target in our experiments, we used patches of size 10×10. Using small patches of size 5×5 did not properly capture the joint "signature" of the highlight and shadow, resulting in a high FA rate. Using a larger patch size results in longer running time.

The images that we used for our training set are shown in Fig. 1. Three images were used, and two of the images were flipped vertically and also added to the set, to gain more variation in the possible orientations of the sea mine in the image. More variation could be achieved by adding more images, if available. The training examples mostly differ in orientation and size of the shadow. Note that the size of the training set should be application dependent, as it depends on the expected variability of the appearance of the target and the parameters one wants to be invariant to. The size of the sea-mine images was roughly 25×25 pixels. All overlapping patches were extracted from these images; however, not all patches contain a significant portion of the sea mine. The patches extracted from the borders of the images, for example, contain mostly background pixels. Thus, the relevant signal content in such patches is low. After discarding the irrelevant patches, we obtained a training set $\{\overline{Z}_i\}$ of $\overline{M} = 277$ patches of size 10×10 pixels.

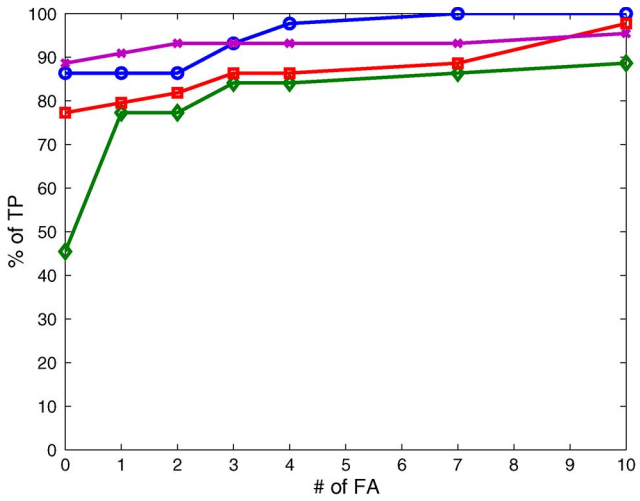


Fig. 7. TP percentage versus FA rate. (Blue, "circle") Proposed method. (Green, "diamond") Local PCA. (Red, "square") Euclidean-distance-based affinity. (Purple, "x") Anomaly detection.

The number of overlapping patches for each test image is $M = 36\,481$. A great advantage of the supervised graph is that the eigenfunctions for the test image $\{\widetilde{\psi}_j\}$ can be efficiently calculated using the eigenfunctions $\{\widetilde{\phi}_j\}$ obtained using the affinity between the training and test sets, averting the need to perform an eigendecomposition of an $M \times M$ matrix. The dimension for the low-dimensional embedding was set to $d = 9$. We set this value empirically based on typical values of the spectral gap in the given images.

We compared the performance of the proposed method with those of three competing approaches:

1) the local PCA-based method described in Section V;
2) a graph-based approach in which the affinity kernel between patches is based on the Euclidean distance as in (24);
3) an anomaly detection algorithm presented in a previous work [38], [39].

We calculate a receiver operating characteristic (ROC) curve for each method to analyze their performances. Detections are found by assigning each pixel the norm of its embedding coordinates (23) and spatially smoothing the detection score image to repress small detections which are due to noise, using a Gaussian filter of size 3×3 and standard deviation of 0.5. The detection score is then thresholded, resulting in a binary image. A detection on the sea mine is considered to be a true positive (TP) for a given image, and any other detections are FAs. Thus, there may be more than one FA per image, but only one TP. Each threshold gives us a (TP, FA) pair plotted in the ROC curve. For each method, we plot the percentage of TPs per number of FAs.

Results are shown in the graph in Fig. 7. The graph shows that the proposed approach (blue-circle plot) is superior to calculating the affinity using the Euclidean distance between patches (red-square plot). This demonstrates that the affinity defined by our weighted distance is better at comparing the test and training sets and separating the target from the background. In addition, our method is superior to the local PCA method (green-diamond plot) described in Section V, particularly for a low FA rate. Comparing the Euclidean distance affinity to the local PCA method, it is shown that adding the projection operator to the affinity actually hinders the performance of the algorithm when applied to target detection. This result affirms our analysis in Section V that the projection operator used in the PCA method enhances the difference between a test patch and a training patch due to the variability of the appearance of the model, effectively lowering the affinity between them.

In comparison to the anomaly detection algorithm (purple-x plot), it shows better results for the number of FAs greater than three, and then, it gives slightly poorer results. For zero FAs, the difference is 2% in favor of the anomaly detection algorithm. Overall, the algorithms are very similar in their performances, with a difference of at most 7%. Note that the results of a supervised method can be improved by extending the training set, as the set that we used was rather limited (based on five images). In addition, the advantage of a supervised approach is that the detections found by the algorithm will necessarily be similar to the required target. On the other hand, the anomaly detection approach, which is unsupervised, will output anomalous objects which may have no resemblance to the required target.

The computational complexity of the detection process is as follows. Calculation of the matrix $\widetilde{\mathbf{A}}$ (19) requires $O(M\overline{M}N + \overline{M}^2 M)$ operations. The complexity of the eigendecomposition of the matrix $\widetilde{\mathbf{P}}$ (13) is $O(\overline{M}^3)$ but, in practice, depends on the algorithm used and the structure of the matrix and its sparsity. The complexity of the out-of-sample extension used to calculate the embedding $\widetilde{\Psi}_d$ (21) is $O(M\overline{M}d)$ operations. Calculation of the detection score requires $O(Md)$ operations. Thus, the overall computational complexity of the detection process is $O(M\overline{M}(N + \overline{M} + d) + \overline{M}^3)$.

We compared the average running times of the four algorithms: our metric—13.93 s/image, Euclidean affinity—7.77 s/image, PCA-based affinity—15.50 s/image, and anomaly detection—33.88 s/image. The four algorithms have been implemented in Matlab, and the numerical experiments have been carried out on a Lenovo ThinkCentre M series desktop, with an Intel Core i5-3570 QuadCore CPU 3.40 GHz and 4.0-GB RAM. It should be noted that these are Matlab implementations and have not been optimized for running time. The target detection approach is computationally more efficient than the anomaly detection approach for several reasons. First, the anomaly detection algorithm is a multiscale algorithm which performs an embedding and detection process for several scales of the image. The supervised approach presented here uses a single scale. Second, the calculation of the embedding is faster in the supervised approach. The anomaly detection algorithm employs an out-of-sample extension method [40] to calculate the embedding for the entire image. This method is computationally more intensive than the extension from the training set to the image in the proposed approach, which is based on a simple matrix multiplication (18). Third, calculation of the anomaly detection score requires finding nearest neighbors in the embedding coordinates for each pixel, whereas the target detection score is a norm calculation, which is a much simpler operation.

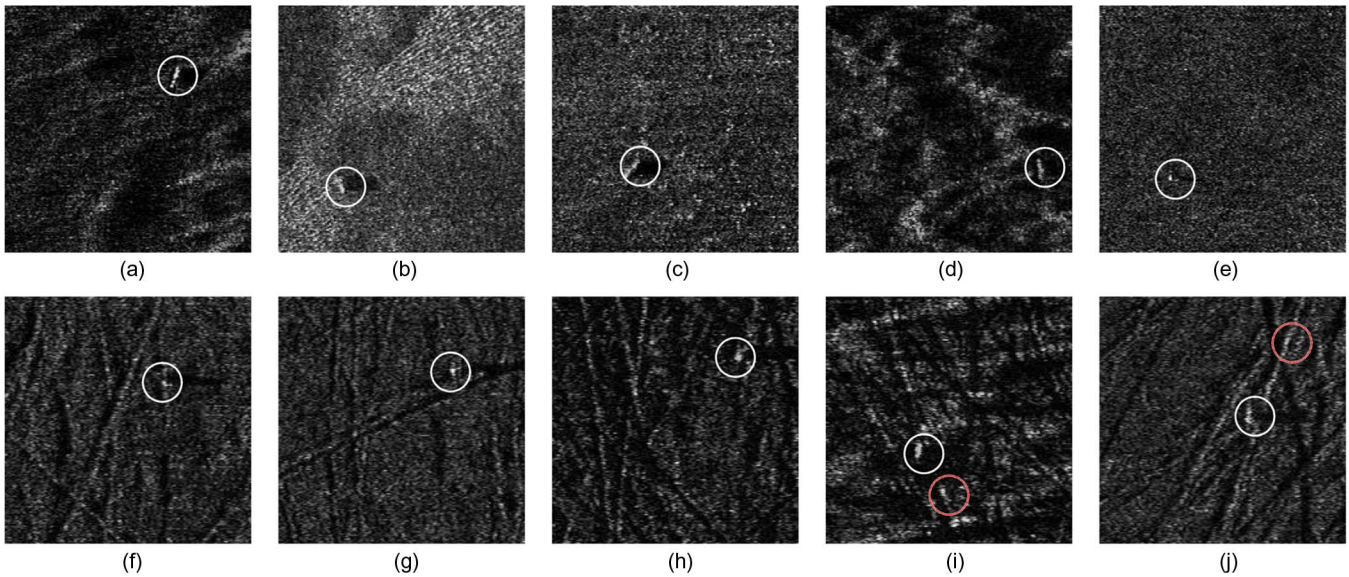


Fig. 8. Results of target detection applied to eight side-scan sonar images containing sea mines. Thresholding the target detection score by 0.76 gives the detections indicated by the circles. All sea mines were detected successfully, indicated by the white circles, and two FAs are indicated by red circles.

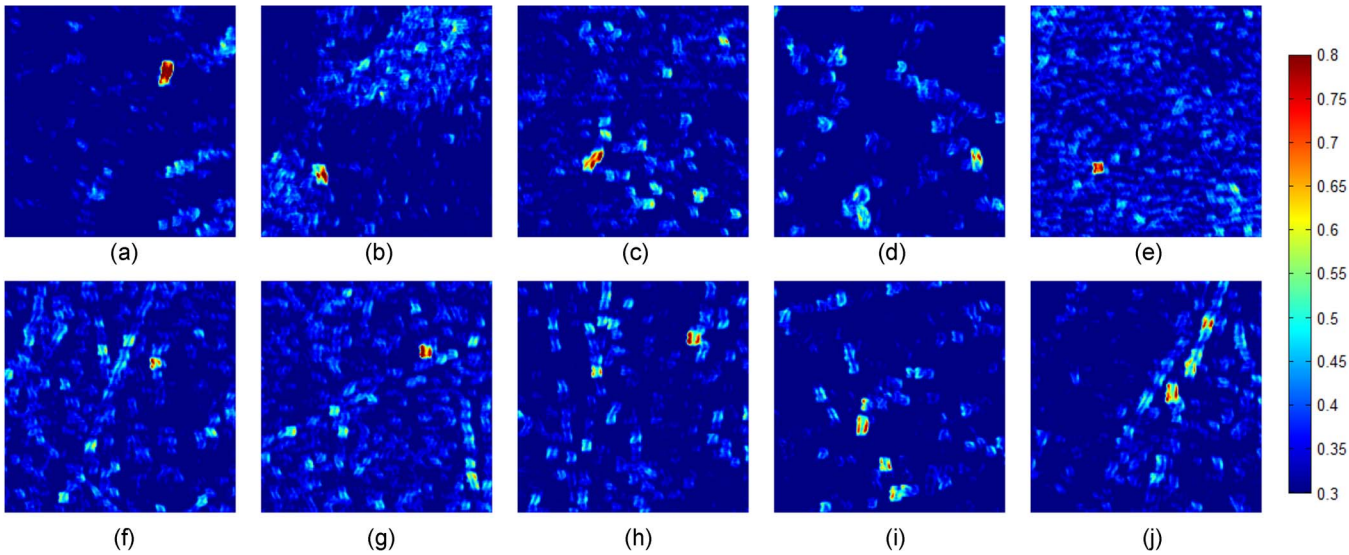


Fig. 9. Detection score corresponding to the images displayed in Fig. 8. Each pixel is colored according to the embedding norm.

Note that, typically in the detection stage of sea-mine hunting, a TP is any MLO, whether it is a mine or not, and FAs are noise or seabed scattering [13]. Here, we treat only the sea mines as TPs, and all other detections are FAs, as the purpose of this experiment is to evaluate the performance of the three supervised metrics and the unsupervised method in their ability to extract the target from the cluttered background. We intend to examine the potential of our approach to produce a smaller number of MLOs for classification than other detection methods, which are not data adaptive. Since the application of our method should be complemented by an appropriate classification algorithm, we note that the new embedding provides a data-driven invariant set of features based on the intrinsic parameters of the data, which may prove to be useful in the classification procedure.

Fig. 8 shows eight side-scan sonar images with sea mines. Each image contains one sea mine on a highly cluttered seabottom background. The background patterns are diverse. Some appear as noise [Fig. 8(c)–(e)], whereas others contain relatively slow changing backgrounds [Fig. 8(a)]. Images with a rapidly changing background [Fig. 8(b), (d), (h), and (j)] or images that contain many shadows from seabed reflections [Fig. 8(f), (g), and (i)] are particularly difficult. Also, the size of the sea mine and its shadow differ from one image to another. For example, in Fig. 8(a), the sea mine is quite large, whereas in Fig. 8(e) and (f), the sea mine is small. The orientation of the sea mine is also subject to variation [Fig. 8(b)–(d)].

Fig. 9 displays the detection score of each of the eight side-scan sonar images given in Fig. 8. The sea mines in all images receive a high detection score. There are areas in the background which have a nonnegligible score and can therefore be detected if the threshold is too low.

The detection results indicated by a white circle in Fig. 8 are achieved by applying a threshold of 0.76 to the detection score

in Fig. 9. This threshold corresponds to 93% percent TP with a total of three FAs. In Fig. 9(i) and (j), two FAs are indicated by a red circle. Although using a limited training set of five images, a positive detection of the sea mines is achieved in all displayed images. The algorithm was able to detect the sea mine even when there was a large difference in the size and orientation of the target compared to the training set and under variable background clutter.

VIII. Conclusion

We have introduced a new metric for constructing local models for supervised target detection. The proposed method enables the user to design a local metric between a training set of target patches and patches from a test image. We show that this metric has an intuitive meaning in the patch space: Determining a weight vector for the pixels in the patch enables one to emphasize certain similarities to the constructed model while also allowing for perturbations in its appearance. We also show that, by controlling the notion of locality within the training set, this procedure creates invariant metrics to certain implicit factors in the parameter space, such as the orientation of the target and its background. Thus, this metric enables correct target detection despite variations in the target appearance.

The metric is used to define an affinity kernel between the given training set and the test set. A graph-based framework based on this kernel is used for dimensionality reduction. We have also proposed a detection score in the reduced dimensionality based on the properties of the affinity kernel. We demonstrate that both the newly proposed metric and the graph-based embedding are required for successful target detection. Experimental results for MLO detection in a set of real side-scan sonar images demonstrated the successful performance of the algorithm, in comparison to competing methods. The results show the capability of the proposed model and algorithm to cope with a variety of targets and background clutter patterns.

Appendix I
Local Statistics in the Parameter Space

The measurement Z at a sensor x is a scalar function of the parameter vector $\boldsymbol{\theta}$. Ignoring the measurement noise η

$$Z(x) = f(x;\boldsymbol{\theta}). \tag{31}$$

Writing a neighboring point using the Taylor expansion

$$\begin{aligned} Z'(x) &= f(x;\boldsymbol{\theta}') \\ &= f(x;\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta})(\boldsymbol{\theta}'-\boldsymbol{\theta}) + \mathcal{O}\left(\|\boldsymbol{\theta}'-\boldsymbol{\theta}\|^2\right) \end{aligned} \tag{32}$$

yields that the Euclidean distance between the two measurements, calculated over all sensors, is

$$\begin{aligned} (Z'-Z)^2 = &\sum_x (\boldsymbol{\theta}'-\boldsymbol{\theta})^{\mathrm{T}} \nabla_{\boldsymbol{\theta}} f(x;\boldsymbol{\theta}) \\ &\times \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta})(\boldsymbol{\theta}'-\boldsymbol{\theta}) + \mathcal{O}(\|\boldsymbol{\theta}'-\boldsymbol{\theta}\|^3). \end{aligned} \tag{33}$$

The empirical mean of $Z(x)$ in (6) can be written in terms of $\hat{m}_{\boldsymbol{\theta}}$, the empirical mean of $\boldsymbol{\theta}$

$$\begin{aligned} \hat{\mu}_Z(x) &= \frac{1}{k}\sum_{Z'\in\mathcal{N}_Z} Z'(x) \\ &= f(x;\boldsymbol{\theta}) + \frac{1}{k}\sum \left(\nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta})(\boldsymbol{\theta}'-\boldsymbol{\theta}) + \mathcal{O}(\|\boldsymbol{\theta}'-\boldsymbol{\theta}\|^2)\right) \\ &= f(x;\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta})(\hat{m}_{\boldsymbol{\theta}} - \boldsymbol{\theta}) + \mathcal{O}(\|\boldsymbol{\theta}'-\boldsymbol{\theta}\|^2) \\ &\approx f(x;\hat{m}_{\boldsymbol{\theta}}). \end{aligned} \tag{34}$$

The empirical variance of $Z(x)$ in (3) can be written in terms of the empirical covariance of $\boldsymbol{\theta}$

$$\begin{aligned} \hat{\sigma}_Z(x)^2 &= \frac{1}{k}\sum_{Z'\in\mathcal{N}_Z} \left(Z'(x) - \hat{\mu}_Z(x)\right)^2 \\ &= \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta}) \frac{1}{k}\sum (\boldsymbol{\theta}' - \hat{m}_{\boldsymbol{\theta}})(\boldsymbol{\theta}' - \hat{m}_{\boldsymbol{\theta}})^{\mathrm{T}} \nabla_{\boldsymbol{\theta}} f(x;\boldsymbol{\theta}) \\ &= \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta}) \mathrm{Cov}(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} f(x;\boldsymbol{\theta}) \end{aligned} \tag{35}$$

where $\mathrm{Cov}(\boldsymbol{\theta})$ is the empirical covariance of $\boldsymbol{\theta}$ in the neighborhood of Z. Assuming that the parameters are independent, this matrix is diagonal with the empirical variances on the diagonal. We denote the diagonal as the vector $\sigma^2_{\boldsymbol{\theta}}$, i.e., a vector containing the variances of each parameter θ. Therefore, (35) can be rewritten as

$$\hat{\sigma}^2_Z(x) = \nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta}) \mathrm{diag}(\sigma^2_{\boldsymbol{\theta}}) \nabla_{\boldsymbol{\theta}} f(x;\boldsymbol{\theta}). \tag{36}$$

Finally, the distance in (11) can be written in terms of the parameter vector

$$a^2(Z', Z) = \sum_x \frac{\left(\nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta})(\Delta\boldsymbol{\theta})\right)^2}{\nabla_{\boldsymbol{\theta}} f^{\mathrm{T}}(x;\boldsymbol{\theta}) \mathrm{diag}(\sigma^2_{\boldsymbol{\theta}}) \nabla_{\boldsymbol{\theta}} f(x;\boldsymbol{\theta})} \tag{37}$$

where $\Delta\boldsymbol{\theta} = \boldsymbol{\theta}' - \hat{m}_{\boldsymbol{\theta}}$. Note that controlling the local neighborhood of a training point effectively controls the empirical variance of the parameter vector, which, in turn, enables us to create invariants to certain perturbations of these parameters. To create an invariance to a certain parameter, the neighborhood should be determined such that the variance of all other parameters approaches zero, and the only variability is in the required parameter. We demonstrate this for 1-D functions in Appendix III.

Appendix II
Proof of (14)

In (13), we have a discrete sum over all points in the test set

$$\mathbf{W}[i,j] = \sum_{l=1}^{M} \exp\left\{-a^2(Z_l, \overline{Z}_i)\right\} \exp\left\{-a^2(Z_l, \overline{Z}_j)\right\} \tag{38}$$

where we omit the parameter ϵ for compactness sake. For a large enough test set, summing over all points is equivalent to

summing over all possible values of $\mathbf{Z}$. Therefore, this sum can be replaced by an integral over all possible values of $\mathbf{Z} \in \mathbb{R}^N$

$$W(\overline{\mathbf{Z}}_i, \overline{\mathbf{Z}}_j) = \int_{\mathbb{R}^N} \exp\left\{-\sum_{x=1}^{N} \frac{(Z(x)-\overline{\mu}_i(x))^2}{\overline{\sigma}_i^2(x)} + \frac{\left(Z(x)-\overline{\mu}_j(x)\right)^2}{\overline{\sigma}_j^2(x)}\right\} d\mathbf{Z}. \tag{39}$$

Using the separability of the integral and the exponential functions, this can be rewritten as a product of 1-D integrals

$$W(\overline{\mathbf{Z}}_i, \overline{\mathbf{Z}}_j) = \prod_{x=1}^{N} \int_{\mathbb{R}} \exp\left\{-\frac{(Z(x)-\overline{\mu}_i(x))^2}{\overline{\sigma}_i^2(x)} - \frac{\left(Z(x)-\overline{\mu}_j(x)\right)^2}{\overline{\sigma}_j^2(x)}\right\} dZ. \tag{40}$$

For compactness sake, we neglect the notation of pixel x in the calculation of the 1-D integral.

The integral can be rewritten as a convolution of Gaussians using the change of variables $\hat{Z} = Z - \mu_i$

$$\left(g(\overline{\sigma}_i) * g(\overline{\sigma}_j)\right)(\overline{\mu}_i - \overline{\mu}_j) = \int_{\mathbb{R}} \exp\left\{-\frac{\hat{Z}^2}{\overline{\sigma}_i^2}\right\} \exp\left\{-\frac{\left(\hat{Z}-(\overline{\mu}_j-\overline{\mu}_i)\right)^2}{\overline{\sigma}_j^2}\right\} dZ \tag{41}$$

where $g(\sigma) = \exp\{-x^2/\sigma^2\}$. Using the convolution theorem and the Fourier transform of the Gaussian function

$$\mathcal{F}\left\{g(\overline{\sigma}_i) * g(\overline{\sigma}_j)\right\} = \pi \overline{\sigma}_i \overline{\sigma}_j \exp\left\{-\pi^2 k^2 \left(\overline{\sigma}_i^2 + \overline{\sigma}_j^2\right)\right\}. \tag{42}$$

Applying the inverse transform yields

$$\left(g(\overline{\sigma}_i) * g(\overline{\sigma}_j)\right)(\overline{\mu}_i - \overline{\mu}_j) = \sqrt{\frac{\pi \overline{\sigma}_i^2 \overline{\sigma}_j^2}{\overline{\sigma}_i^2 + \overline{\sigma}_j^2}} \exp\left\{-(\overline{\mu}_i - \overline{\mu}_j)^2 / \left(\overline{\sigma}_i^2 + \overline{\sigma}_j^2\right)\right\}. \tag{43}$$

Plugging this back into (40) yields

$$W(\overline{\mathbf{Z}}_i, \overline{\mathbf{Z}}_j) \propto \exp\left\{-\sum_{x=1}^{N} \frac{\left(\overline{\mu}_i(x)-\overline{\mu}_j(x)\right)^2}{\left(\overline{\sigma}_i^2(x)+\overline{\sigma}_j^2(x)\right)}\right\} \tag{44}$$

which is the symmetric matrix given in (14).

Appendix III
Designing Shift- and Scale-Invariant Distances

Consider a family of 1-D functions that are given by dilations and shifts of one another

$$Z(x) = f(x;\boldsymbol{\theta}) = f\left(\frac{x-b}{a}\right) \tag{45}$$

so the parameter vector is $\boldsymbol{\theta} = (a \;\; b)^{\mathrm{T}}$. A first-order Taylor expansion of a neighbor point as in (32), using the chain rule, yields

$$\begin{aligned} Z'(x) &= f(x;\boldsymbol{\theta}) + \left.\frac{df}{du}\right|_{u=\frac{x-b}{a}} \nabla_{\boldsymbol{\theta}} u^{\mathrm{T}} \Delta\boldsymbol{\theta} \\ &= f(x;a,b) - \left.\frac{df}{du}\right|_{u=\frac{x-b}{a}} \left(\frac{x-b}{a^2}\Delta a + \frac{1}{a}\Delta b\right). \end{aligned} \tag{46}$$

Plugging this into (37) yields

$$\begin{aligned} a^2(Z', Z) &= \int_{b-L}^{b+L} \frac{\left(\left.\frac{df}{du}\right|_{u=\frac{x-b}{a}} \left(\frac{x-b}{a^2}\Delta a + \frac{1}{a}\Delta b\right)\right)^2}{\left(\left.\frac{df}{du}\right|_{u=\frac{x-b}{a}}\right)^2 \left(\left(\frac{x-b}{a^2}\right)^2 \sigma_a^2 + \left(\frac{1}{a}\right)^2 \sigma_b^2\right)} dx \\ &= \int_{-L}^{L} \frac{(x\Delta a + a\Delta b)^2}{x^2\sigma_a^2 + a^2\sigma_b^2} dx. \end{aligned} \tag{47}$$

We replace the discrete sum with an integral in the continuous domain and integrate over a symmetric interval of length $2L$ surrounding the shift parameter b. We compare this distance to the Euclidean distance in (33)

$$\|Z' - Z\|^2 = \int_{-L}^{L} \left(\left.\frac{df}{du}\right|_{u=\frac{x}{a}}\right)^2 \left(\frac{x}{a^2}\Delta a + \frac{1}{a}\Delta b\right)^2 dx \tag{48}$$

where, again, we used a change of variables $x = x - b$ in evaluating the integral. Note that the proposed distance (47) calculated for two close measurement points does not depend on the mapping f between the parameter vector and the measurements Z and Z'. Thus, calculating the distance between two measurements is essentially a calculation in the parameter space, i.e., this distance depends only on the unknown parameters. This holds for any family of 1-D functions which represent a change of variables. The Euclidean distance, on the other hand, does depend on the mapping f.

To obtain a scale-invariant distance a, we set $\sigma_b = 0$

$$\begin{aligned} a^2(Z', Z) &= \frac{1}{\sigma_a^2} \int_{-L}^{L} \left((\Delta a)^2 + \frac{(a\Delta b)^2}{x^2}\right) dx \\ &= \frac{2L}{\sigma_a^2}(\Delta a)^2 + \frac{a\Delta b^2}{\sigma_a^2} \int_{-L}^{L} \frac{1}{x^2} dx. \end{aligned} \tag{49}$$

The second term is an integral which approaches infinity. Therefore, for $\Delta b = 0$, $|\Delta a| > 0$, we have finite distances, whereas a slight perturbation in Δb yields infinite distances. Thus, we have achieved an invariance to dilations Δa in comparison to shifts Δb.

To obtain a shift-invariant distance, we set $\sigma_a = 0$

$$\begin{aligned} a^2(Z', Z) &= \frac{1}{\sigma_b^2} \int_{-L}^{L} \left(x \frac{\Delta a}{a} + \Delta b \right)^2 dx \\ &= \frac{1}{\sigma_b^2} \left(\frac{2L^3}{3a^2} (\Delta a)^2 + 2L (\Delta b)^2 \right) \\ &= \frac{2L^3}{\sigma_b^2} \left(\frac{(\Delta a)^2}{3a^2} + \frac{(\Delta b)^2}{L^2} \right) \end{aligned} \tag{50}$$

where the cross-terms are canceled out due to the symmetry of the integral interval. This result shows that, if the interval of the integral $2L$ follows $2L > \sqrt{(12)}a$, then a perturbation in dilations $\Delta a = \Delta$ causes a larger increase in distance than an identical perturbation in shifts $\Delta b = \Delta$ providing a certain shift invariance

$$a^2(Z', Z)\Big|_{\substack{\Delta a = \Delta > 0 \\ \Delta b = 0}} \propto \frac{L^2}{a^2} a^2(Z', Z)\Big|_{\substack{\Delta a = 0 \\ \Delta b = \Delta > 0}}. \tag{51}$$

If the variance of the shift parameter in the local neighborhood of the measurement Z, $\sigma_b \to \infty$, then $a^2(Z', Z) \to 0$. This means that this distance is meaningful for a reasonable limited variance, which depends on the integral interval. Comparing this result to the scale-invariant distance, the shift invariance achieved via this metric is less efficient than the scale invariance.

Compared to the Euclidean distance, we can see that, if $f(u)$ is a polynomial in u, the ratio between distances due to a perturbation in dilations to distances due to a perturbation in shifts is proportional to L^2/a^2, as in the shift-invariant case. In such cases, the distances are similar in terms of their shift invariance, and other methods, specifically designed for shift invariance, might achieve better performance. In the toy example presented in Section VII-A, $f(u)$ is an exponential function, and the designed shift-invariant distance has meaningful shift invariance compared to the Euclidean distance.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments and useful suggestions.

REFERENCES

[1] Q. Du and I. Kopriva, "Automated target detection and discrimination using constrained kurtosis maximization," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 1, pp. 38–42, Jan. 2008.

[2] Y. Chen, N. Nasrabadi, and T. Tran, "Sparse representation for target detection in hyperspectral imagery," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 629–640, Jun. 2011.

[3] B. Bhanu and Y. Lin, "Genetic algorithm based feature selection for target detection in SAR images," *Image Vis. Comput.*, vol. 21, no. 7, pp. 591–608, Jul. 2003.

[4] G. Mercier and F. Girard-Ardhuin, "Partially supervised oil-slick detection by SAR imagery using kernel expansion," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2839–2846, Oct. 2006.

[5] P. Torrione, K. Morton, R. Sakaguchi, and L. Collins, "Histograms of oriented gradients for landmine detection in ground-penetrating radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 3, pp. 1539–1550, Mar. 2014.

[6] G. J. Dobeck, "Algorithm fusion for automated sea mine detection and classification," in *Proc. MTS/IEEE OCEANS Conf. Exhibition*, 2001, vol. 1, pp. 130–134, Marine Technol. Soc.

[7] E. Coiras, P.-Y. Mignotte, Y. Petillot, J. Bell, and K. Lebart, "Supervised target detection and classification by training on augmented reality data," *IET Radar Sonar Navigat.*, vol. 1, no. 1, pp. 83–90, Feb. 2007.

[8] G. J. Dobeck, J. C. Hyland, and L. Smedley, "Automated detection and classification of sea mines in sonar imagery," in *Proc. SPIE*, Jul. 1997, vol. 3079, pp. 90–110.

[9] C. Spence, L. Parra, and P. Sajda, "Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model," in *Proc. IEEE Workshop Math. Methods Biomed. Image Anal.*, 2001, pp. 3–10.

[10] A. Noiboar and I. Cohen, "Anomaly detection based on wavelet domain GARCH random field modeling," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 5, pp. 1361–1373, May 2007.

[11] 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 J. Ocean. Eng*, vol. 30, no. 2, pp. 360–371, Apr. 2005.

[12] S. Reed, Y. Petillot, and J. Bell, "An automatic approach to the detection and extraction of mine features in sidescan sonar," *IEEE J. Ocean. Eng*, vol. 28, no. 1, pp. 90–105, Jan. 2003.

[13] F. Florin, F. Van Zeebroeck, I. Quidu, and N. Le Bouffant, "Classification performances of mine hunting sonar: Theory, practical results and operational applications," in *Proc. UDT Europe*, Jun. 2003, pp. 1–11.

[14] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy, "Three-class Markovian segmentation of high resolution sonar images," *Comput. Vis. Image Understanding*, vol. 76, no. 3, pp. 191–204, Dec. 1999.

[15] A. Goldman and I. Cohen, "Anomaly subspace detection based on a multi-scale Markov random field model," *Signal Process.*, vol. 85, no. 3, pp. 463–479, Mar. 2005.

[16] H. Lange and L. M. Vincent, "Advanced gray-scale morphological filters for the detection of sea mines in side-scan sonar imagery," in *Proc. SPIE*, 2000, vol. 4038, pp. 362–372.

[17] S. Reed, Y. Petillot, and J. Bell, "Automated approach to classification of mine-like objects in sidescan sonar using highlight and shadow information," *Proc. Inst. Elect. Eng.—Radar, Sonar Navigat.*, vol. 151, no. 1, pp. 48–56, Feb. 2004.

[18] M. Mignotte and C. Collet, "Hybrid genetic optimization and statistical model based approach for the classification of shadow shapes in sonar imagery," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 2, pp. 129–141, Feb. 2000.

[19] I. Quidu, J. Malkass, G. Burel, and P. Vilbé, "Mine classification based on raw sonar data: An approach combining Fourier descriptors, statistical models and genetic algorithms," in *Proc. OCEANS MTS/IEEE Conf.*, Providence, RI, USA, Sep. 2000, pp. 285–290.

[20] Y. Petillot, Y. Pailhas, and J. Sawas, "Target recognition in synthetic aperture and high resolution side-scan sonar," in *Proc. Eur. Conf. Underwater Acoust.*, 2010, pp. 99–106.

[21] V. Myers and J. A. Fawcett, "A template matching procedure for automatic target recognition in synthetic aperture sonar imagery," *IEEE Signal Process. Lett.*, vol. 17, no. 7, pp. 683–686, Jul. 2010.

[22] A. El Bergui, I. Quidu, B. Zerr, and B. Solaiman, "Model based classification of mine-like objects in sidescan sonar using the highlight information," in *Proc. 11th ECUA*, 2012, vol. 17, pp. 1158–1165.

[23] R. Talmon, I. Cohen, S. Gannot, and R. Coifman, "Supervised graph-based processing for sequential transient interference suppression," *IEEE Trans. Audio, Speech Language Process.*, vol. 20, no. 9, pp. 2528–2538, Nov. 2012.

[24] A. Haddad, D. Kushnir, and R. R. Coifman, "Texture separation via a reference set," *Appl. Comput. Harmonic Anal.*, vol. 36, no. 2, pp. 335–347, Mar. 2014.

[25] J. Bell and L. Linnett, "Simulation and analysis of synthetic sidescan sonar images," *Proc. Inst. Elect. Eng.—Radar, Sonar Navigat.*, vol. 144, no. 4, pp. 219–226, Aug. 1997.

[26] E. Coiras, Y. Petillot, and D. Lane, "Multiresolution 3-D reconstruction from side-scan sonar images," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 382–390, Feb. 2007.

[27] D. Kushnir, A. Haddad, and R. R. Coifman, "Anisotropic diffusion on sub-manifolds with application to earth structure classification," *Appl. Comput. Harmonic Anal.*, vol. 32, no. 2, pp. 280–294, Mar. 2012.

[28] A. Singer and R. R. Coifman, "Nonlinear independent component analysis with diffusion maps," *Appl. Comput. Harmonic Anal.*, vol. 25, no. 2, pp. 226–239, Sep. 2008.

[29] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmonic Anal.*, vol. 21, no. 1, pp. 5–30, Jul. 2006.

[30] S. Lafon, Y. Keller, and R. R. Coifman, "Data fusion and multicue data matching by diffusion maps," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1784–1797, Nov. 2006.
[31] R. Talmon and R. R. Coifman, "Empirical intrinsic geometry for nonlinear modeling and time series filtering," *Proc. Nat. Acad. Sci.*, vol. 110, no. 31, pp. 12535–12540, Jul. 2013.
[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
[33] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE CVPR*, 2005, pp. 886–893.
[34] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *Proc. IEEE ISCAS*, May 2010, pp. 253–256.
[35] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1872–1886, Aug. 2013.
[36] L. Sifre and S. Mallat, "Rotation, scaling and deformation invariant scattering for texture discrimination," in *Proc. IEEE CVPR*, Jun. 2013, pp. 1233–1240.
[37] R. Talmon, Y. Shkolnisky, and R. R. Coifman, "Nonlinear modeling and processing using empirical intrinsic geometry with application to biomedical imaging," in *Geometric Science of Information*. New York, NY, USA: Springer-Verlag, 2013, pp. 441–448.
[38] G. Mishne and I. Cohen, "Multiscale anomaly detection using diffusion maps," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 1, pp. 111–123, Feb. 2013.
[39] G. Mishne and I. Cohen, "Multiscale anomaly detection using diffusion maps and saliency score," in *Proc. IEEE ICASSP*, 2014, pp. 2823–2827.
[40] N. Rabin and R. R. Coifman, "Heterogeneous datasets representation and learning using diffusion maps and Laplacian pyramids," in *Proc. 12th SIAM Int. Conf. Data Mining*, 2012, pp. 189–199.

Gal Mishne received the B.Sc. degree (*summa cum laude*) in electrical engineering and physics from Technion–Israel Institute of Technology, Haifa, Israel, in 2009, where she is currently working toward the Ph.D. degree in electrical engineering.

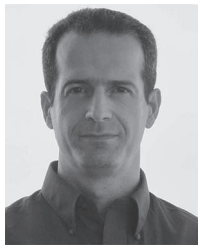
From 2008 to 2013, she was an Image Processing Engineer with the Israeli defense industry. Her main areas of interest include signal processing, image processing, and geometric methods for data analysis.

Ms. Mishne is a recipient of the Ollendorff Fellowship for 2014 and was a recipient of the Wilk Family Award from the Signal and Image Processing Laboratory for 2009.

Ronen Talmon received the B.A. degree (*cum laude*) in mathematics and computer science from The Open University, Ra'anana, Israel, in 2005 and the Ph.D. degree in electrical engineering from Technion–Israel Institute of Technology, Haifa, Israel, in 2011.

From 2000 to 2005, he was a Software Developer and Researcher at a technological unit of the Israeli Defense Forces. From 2005 to 2011, he was a Teaching Assistant with the Department of Electrical Engineering, Technion–Israel Institute of Technology. From 2011 to 2013, he was a Gibbs Assistant Professor with the Mathematics Department, Yale University, New Haven, CT, USA. In 2013, he joined the Department of Electrical Engineering, Technion–Israel Institute of Technology, where he is currently an Assistant Professor of electrical engineering. His research interests are statistical signal processing, analysis and modeling of signals, speech enhancement, biomedical signal processing, applied harmonic analysis, and diffusion geometry.

Dr. Talmon was the recipient of the Irwin and Joan Jacobs Fellowship, the Andrew and Erna Fince Viterbi Fellowship, and the Horev Fellowship.



Israel Cohen (M'01–SM'03) received the B.Sc. (*summa cum laude*), M.Sc., and Ph.D. degrees in electrical engineering from Technion–Israel Institute of Technology, Haifa, Israel, in 1990, 1993, and 1998, respectively.

From 1990 to 1998, he was a Research Scientist with RAFAEL Research Laboratories, Haifa, Israel Ministry of Defense. From 1998 to 2001, he was a Postdoctoral Research Associate with the Computer Science Department, Yale University, New Haven, CT, USA. In 2001, he joined the Department of Electrical Engineering, Technion–Israel Institute of Technology, where he is currently a Professor of electrical engineering. He is a Coeditor of the Multichannel Speech Processing Section of the *Springer Handbook of Speech Processing* (Springer, 2008), a coauthor of *Noise Reduction in Speech Processing* (Springer, 2009), a Coeditor of *Speech Processing in Modern Communication: Challenges and Perspectives* (Springer, 2010), and a General Cochair of the 2010 International Workshop on Acoustic Echo and Noise Control. He served as Guest Editor of the *European Association for Signal Processing Journal on Advances in Signal Processing* Special Issue on Advances in Multimicrophone Speech Processing and the *Elsevier Speech Communication Journal* Special Issue on Speech Enhancement. His research interests are statistical signal processing, analysis and modeling of acoustic signals, speech enhancement, noise estimation, microphone arrays, source localization, blind source separation, system identification, and adaptive filtering.

Dr. Cohen was a recipient of the Alexander Goldberg Prize for Excellence in Research, and the Muriel and David Jacknow Award for Excellence in Teaching. He serves as a member of the IEEE Audio and Acoustic Signal Processing Technical Committee and the IEEE Speech and Language Processing Technical Committee. He served as Associate Editor of the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING and the IEEE SIGNAL PROCESSING LETTERS.