Diffusion Features for Target Specific Recognition with Synthetic Aperture Sonar Raw Signals and Acoustic Color

Jason C. Isaacs
Naval Surface Warfare Center
Panama City, FL 32407
jason.c.isaacs1@navy.mil

James D. Tucker
Naval Surface Warfare Center
Panama City, FL 32407
james.d.tucker@navy.mil

Abstract

Given a high dimensional dataset, one would like to be able to represent this data using fewer parameters while preserving relevant signal information. If we assume the original data actually exists on a lower dimensional manifold embedded in a high dimensional feature space, then recently popularized approaches based in graph-theory and differential geometry allow us to learn the underlying manifold that generates the data. One such technique, called Diffusion Maps, is said to preserve the local proximity between data points by first constructing a representation for the underlying manifold. This work examines target specific classification problems using Diffusion Maps to embed inverse imaged synthetic aperture sonar signal data for automatic target recognition. The data set contains six target types. Results demonstrate that the diffusion features capture suitable discriminating information from the raw signals and acoustic color to improve target specific recognition with a lower false alarm rate. However, fusion performance is degraded.

1. Introduction

The central problem in high-dimensional data analysis is the trade-off between computational complexity and the resolution gained with either more features or pixels. Therefore, a typical first step in analyzing high-dimensional data is to find a lower-dimensional representation and the concise description of its underlying geometry and density. This is usually done however, with global dimension reducing techniques such as principal component analysis, and Multidimensional Scaling. These techniques in general work well with well behaved maximally variant data. What if the data is only locally correlated? Then these techniques do not provide informative embedded data. Alternatively, graph based manifold learning techniques offer to embed the data based on local relationship preservation, i.e., they generally preserve the neighborhood structure. Such techniques are Diffusion Maps [2] and [3], Local linear Embedding [12], Laplacian Eigenmaps [1], Hessian Eigenmaps [5], and Local Tangent Space Alignment [14].

In this paper we consider the manifold learning technique Diffusion Maps of Coifman et al. [2], [3] and analyze the raw signal and acoustic colors features generated by Diffusion Maps. This is done by looking at the classification results for one-vs.-all for each target type within the sonar dataset using a boosted decision tree.

It should be clarified that, eventually, in a scene-based recognition system, we envision the use of shapes in conjunction with, acoustic and image analysis based features [4], [7], [8], [9], and [10].

2. Synthetic Aperture Sonar

Synthetic aperture sonar (SAS) [11], is similar to SAR in that the aperture is formed artificially from received signals to give the appearance of a real aperture several times the size of the transmit/receive pair. SAS is performed by collecting a set of time domain signals and match filtering the signals to eliminate any coherence with the transmitted pulse. SAS images are generated by beamforming the time domain signals using various techniques, e.g. time-delay, chirp scaling, or ω-k beamforming [11]. In this work the wave-number technique ω-k is used. The goal here is to classify specific targets within objects that have been detected in sonar imagery. The explicit data that will be analyzed is the inverse imaging raw signals and the acoustic color of an extracted region of interest around a detected object. For a given a sonar image snippet this involves inverse beamforming the snippet to resolve the target area time-domain, or raw signal data. This method allows for the elimination of interference by using only the contributing complex-image target information to derive the time-domain data, see Figure 1.

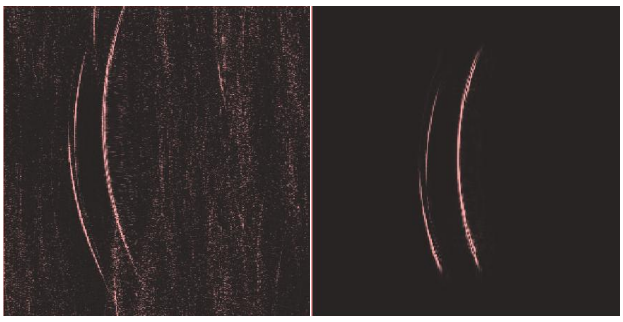

Figure 1. Images of a sequence of sonar pings that comprise time-domain signals about a target object. The original stave data is on the left and the inverse image data is on the right

As can be seen in Figure 1, the inverse image data is devoid of the interference that is seen in the original stave data. The task at hand after detection is to classify these

objects appropriately as targets, specific targets, or non-targets. This work follows the work in [8] with new frequency-aspect information fused with raw signal data to focus on target specific discrimination. Example target ROIs are shown in Figure 2 below. This figure shows four targets and one non-target. Figure 3 below shows the inverse image processes as used for the experimental data in this work. Inverse imaging should be thought of as an inverse Stolt mapping of the beamformed image data back to the time-domain.

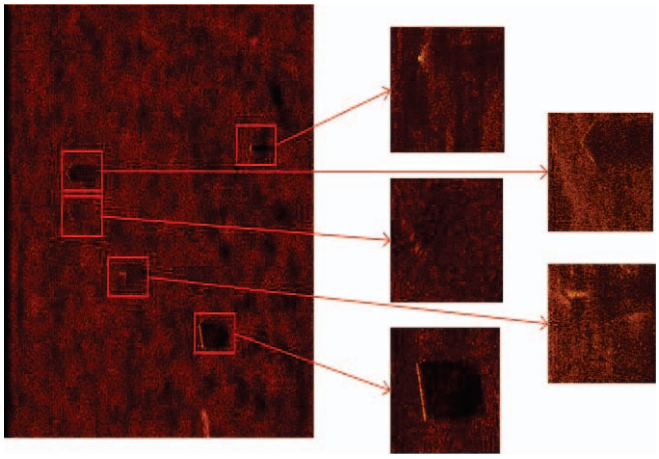

Figure 2. SWAT SAS image with target and non-target ROIs.

After detection, the ROI is inverted back to the time-domain through an inverse imaging process as shown in image of Figure 3. The raw signals are aligned in time as objects are seen with a moving sonar beam that contacts the targets at delayed intervals. These raw signals are then used to classify the object.

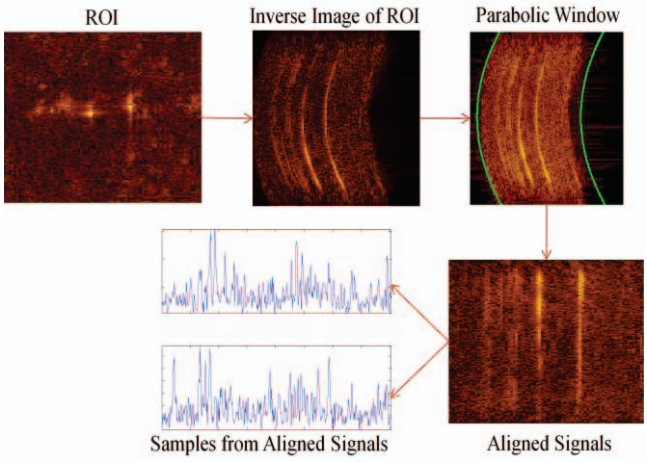

Figure 3. Example extraction of time-signals from the inverse imaging of a SAS image ROI.

2.1. Acoustic Color

Another feature extraction method involving the inverse image data stems from the formation of an acoustic color (AC), or frequency-aspect (FA), matrix. This allows for the possibility of extracting more information from the raw signals of a ROI by analyzing the frequency components of the signals at different angles, or aspects. The acoustic color matrix is a representation of the frequency distribution of the acoustic responses of an object at various sensor viewing angles, as shown above in Figure 4. Therefore, features from the AC should be invariant of the object orientation. Figure 10 shows the generation of the AC plots for both the high and low frequency of an object in an ROI.

Since, both the raw signals and the acoustic color can be high-dimensional the goal here is to reduce the computational burden on the classifier by embedding the data in a lower-dimensional space. The following discussion on Diffusion Maps will demonstrate the process we have used to embed and feature this data.

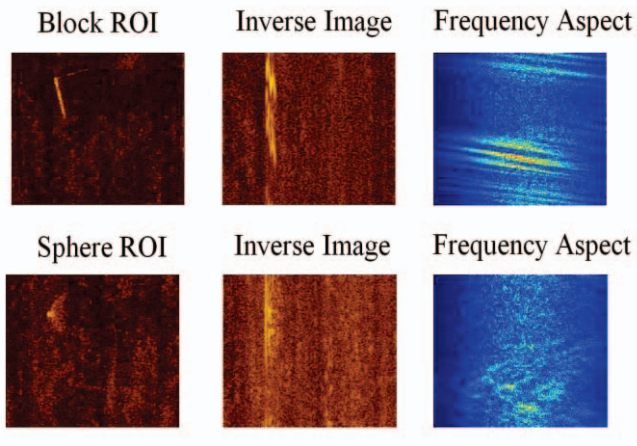

Figure 4. Acoustic color plots from object ROIs for both a block and a sphere.

3. Diffusion Maps

3.1. Overview

Diffusion Maps are defined as the embedding of complex data onto a low dimensional Euclidian space, via the eigenvectors of suitably normalized random walks over the given dataset. It has been shown, both theoretically in [2] and by examples in [3] how this embedding can be used for dimensionality reduction, manifold learning, geometric analysis of complex datasets, and fast simulations of stochastic dynamical systems.

Diffusion Maps are said to preserve the local proximity between data points by first constructing a graph representation for the underlying manifold. The vertices, or nodes of this graph, represent the data points, and the

edges connecting the vertices, represent the similarities between adjacent nodes. If properly normalized, these edge weights can be interpreted as transition probabilities for a random walk on the graph. After representing the graph with a matrix, the spectral properties of this matrix are used to embed the data points into a lower dimensional space, and gain insight into the geometry of the dataset. It has been shown in [2] and [3] that the eigenfunctions of Markov matrices can be used to construct coordinates called Diffusion Maps that generate these efficient representations of the complex geometric structures and the associated family of diffusion distances, obtained by iterating the Markov matrix, defines the multi-scale geometries that prove to be useful in the context of data parameterization and dimensionality reduction. The process of constructing these Diffusion Maps as described in [2] and [3] is discussed in sections 3.1 through 3.5.

3.2. Construction of a random walk on the data

Given a data set Ω with a distribution μ of the points on Ω and a kernel k: $\Omega \times \Omega \rightarrow$R that satisfies the following properties:

- k is symmetric: $k(x, y) = k(y, x)$,
- k is positivity preserving: $k(x, y) \geq 0$.

This kernel represents some notion of affinity or similarity between points of Ω as it describes the relationship between pairs of points in this set and in this sense, one can think of the data points as being the nodes of a symmetric graph whose weight function is specified by k. The kernel constitutes an a priori presumption of the local geometry of Ω, and since a given kernel will capture a specific feature of the data set, its choice should be guided by the application that one has in mind; the Gaussian kernel is used here.

It is known that to any reversible Markov process, one can associate a symmetric graph. In addition, the converse is also true, i.e., from the graph defined by (Ω, k), one can construct a reversible Markov chain on Ω. This technique is known as the normalized graph Laplacian construction. The steps are as follows: define

$$d(x) = \int_{\Omega} k(x, y) d\mu(y) \tag{1}$$

to be a local measure of the degree of node x in this graph and define P^t to be an n × n matrix whose entries are given by

$$p_t(x, y) = \frac{k(x, y)}{d(x)} \tag{2}$$

which is the probability of transition from x to y in one time step. For $t = 1$ this can be interpreted as the first-order neighborhood structure of the graph.

3.3. Powers of P and multi-scale geometric analysis of Ω

The matrix P contains geometric information about the data set Ω. The transitions that it defines directly reflect the local geometry defined by the immediate neighbors of each node in the graph of the data. In other words, $p_1(x, y)$ represents the probability of transition in one time step from node x to node y and it is proportional to the edge-weight $k(x, y)$. For $t \geq 0$, the probability of transition from x to y in t time steps is given by $p_t(x, y)$, the kernel of the t^{th} power P^t of P. Larger powers of P, allows the integration of the local geometry and therefore will reveal relevant geometric structures of Ω at different scales, i.e., larger neighborhoods.

3.4. Spectral Analysis of the Markov Chain

Powers of P constitute an object of interest for the study of the geometric structures of Ω at various scales. A classical way to describe the powers of an operator is to employ the language of spectral theory, namely eigenvectors and eigenvalues. Although for general transition matrices of Markov chains, the existence of a spectral theory is not guaranteed, the random walk constructed here exhibits very particular mathematical properties, i.e., if the graph is connected, which we now assume, then the stationary distribution is unique and we have

$$\lim_{t \to +\infty} p_t(x, y) = \phi_0(y) \tag{3}$$

where the Markov chain has a stationary distribution given by

$$\phi_0(y) = \frac{d(y)}{\sum_{z \in \Omega} d(z)}. \tag{4}$$

The chain is reversible, i.e., it follows the detailed balance condition:

$$\phi_0(x) p_1(x, y) = \phi_0(y) p_1(y, x). \tag{5}$$

The vector ϕ_0 is the top left eigenvector of P. The spectral analysis of the Markov chain is governed by the following eigen-decomposition

$$p_t(x, y) = \sum_{l \geq 0} \lambda_l^t \psi_l(x) \phi_l(y), \tag{6}$$

where $\{\lambda_l\}$ is the sequence of *eigenvalues* of P (with $|\lambda_0| \geq |\lambda_1| \geq |\lambda_2| \geq \cdots$) and $\{\psi_l\}$ and $\{\phi_l\}$ are the corresponding biorthogonal right and left eigenvectors.

3.5. Diffusion Distances and Diffusion Maps

The spectral properties of the Markov chain can now be linked to the geometry of the data set Ω. As previously mentioned, the idea of defining a random walk on the data set relies on the following principle: the kernel k specifies the local geometry of the data and captures some geometric feature of interest. The Markov chain defines fast and slow directions of propagation, based on the values taken by the kernel, and as one runs the walk forward, the local geometry information is being propagated and accumulated the same way local transitions of a system can be integrated in order to obtain a global characterization of this system.

Running the chain forward is equivalent to computing the powers of the operator P. For this computation, we could, in theory, use the eigenvectors and eigenvalues of P. Therefore, we are going to directly employ these objects in order to characterize the geometry of the data set Ω. The family of diffusion distances $\{D_t\}_{t\in N}$ is given by

$$D_t^2(x,z)=\sum_{y\in\Omega}\frac{p_t(x,y)-p_t(z,y)}{\phi_0(y)}^2. \quad (7)$$

In other words, $D_t(x, z)$ is a functional weighted l_2 distance between the two posterior distributions $p_t(x, \cdot)$ and $p_t(z, \cdot)$. For a fixed value of t , D_t defines a distance on the set Ω. By definition, the notion of proximity that it defines reflects the connectivity in the graph of the data. Indeed, $D_t(x, z)$ will be small if there is a large number of short paths connecting x and z, that is, if there is a large probability of transition from x to z and vice versa. The main interesting features of diffusion distance are: 1) the points are closer if they are highly connected, 2) $D_t(x, z)$ involves summing over all paths and is therefore robust to noise perturbations, 3) the distance takes into account all evidence relating x and z. $D_t(x, z)$ does not have to be computed explicitly. It can be computed using the eigenvectors and eigenvalues of P:

$$D_t^2(x,z)=\sum_{l\geq 1}\lambda_l^{2t}(\psi_l(x)-\psi_l(z))^2. \quad (8)$$

As previously mentioned, the eigenvalues $\lambda_1,\lambda_2, \ldots, \lambda_N$ tend to 0 and have a modulus strictly less than 1. As a consequence, the above sum can be computed to a preset accuracy $\delta>0$ with a finite number of terms: if we define as the number of elements retained to meet this accuracy. Then, up to relative precision δ, we have

$$D_t(x,z)=\left(\sum_{l\geq 1}^{s(\delta,t)}\lambda_l^{2t}(\psi_l(x)-\psi_l(z))^2\right)^{\frac{1}{2}}. \quad (9)$$

We can therefore introduce a family of diffusion maps $\{\Psi_t\}_{t\in N}$ given by

$$\psi_t : x\rightarrow\begin{pmatrix}\lambda_1^t\psi_1(x)\\ \lambda_2^t\psi_2(x)\\ \vdots\\ \lambda_{s(\delta,t)}^t\psi_{s(\delta,t)}(x)\end{pmatrix} \quad (10)$$

Each component of Ψ_t (x) is termed diffusion coordinate. The map Ψ_t: $\Omega\rightarrow R^{s(\delta,t)}$ embeds the data set into a Euclidean space of $s(\delta, t)$ dimensions. This method constitutes a universal and data driven way to represent a graph, or any generic data set, as a cloud of points in a Euclidean space. Moreover, $s(\delta, t)$ depends on the properties of the random walk and not on the number of features of the original representation.

4. Experiment and Results

4.1. Experimental Setup

The problem here is to analyze the discriminating feature embeddings of the resultant diffusion features for the classification of specific targets in SAS data. The SAS dataset is divided into ten groups that are as equal as possible for 10-fold cross validation. One group is set aside for the training set and nine groups for the dedicated testing set. This procedure is continued until all groups have represented as a testing set. The average performance overall ten-folds is presented as the probability of classification (P_C), or sensitivity, and the probability of false alarm (P_{FA}), or specificity. This is done to demonstrate the trade-off between correctly classifying true cases versus incorrectly classifying false cases. Each experiment uses the same groups for each dataset so that the possibility of poor individual performance due to the distribution of the draw is reduced as best as possible.

An Adaboost boosted decision tree is used to classify the datasets. Adaboost is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire [6]. It is a meta-algorithm, and can be used in conjunction with many other learning algorithms to improve their performance. AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost is somewhat sensitive to noisy data and outliers. Otherwise, it is less susceptible to the overfitting problem than most learning algorithms. AdaBoost calls a weak classifier repeatedly in a series of rounds. For each call a

distribution of weights is updated that indicates the importance of examples in the dataset for the classification, i.e., the difficulty of each sample. For each round, the weights of each incorrectly classified example are increased (or alternatively, the weights of each correctly classified example are decreased), so that the new classifier focuses more on those difficult examples. Experiments here are performed with 100 iterations of boosting a simple decision tree. The experimental variable values are listed below.

Experimental Variables
δ= 15, α = 1, b = 2, μ = 1, σ^2 =3, d = 3.

Where δ is the diffusion precision, α is the diffusion probability distribution scaling, σ^2 is the variance for the Gaussian kernel.

4.2. Dataset and Results

The experiment discussed above tests the diffusion features for target classification enhancement on the following sonar dataset [13] with the class distribution as shown in Table 1 below.

Table 1. Experimental Dataset

Samples	A	B	C	D	E	F
2100	700	240	230	240	240	450

Table 2. Results for the Acoustic Color and Raw Data

Features	Sensitivity and Specificity (%)for Specific Target Type					
	A	B	C	D	E	F
AC	71.57	68.75	66.09	54.58	68.75	76.67
	94.14	99.35	99.36	98.44	99.09	95.33
Raw	94.71	93.33	86.09	86.67	94.17	91.78
	98.71	99.95	99.52	99.52	99.95	98.61
Fusion	92.43	95.00	90.43	89.17	95.42	90.44
	98.71	99.95	99.73	99.78	99.89	98.85

Table 3. Results for the DF of the Acoustic Color and Raw Data

Features	Sensitivity and Specificity (%)for Specific Target Type					
	A	B	C	D	E	F
DF-AC	81.57	81.25	73.48	56.67	76.25	75.11
	93.64	99.62	99.41	98.66	99.73	97.09
DF-Raw	92.86	95.83	91.30	85.83	90.42	85.78
	97.71	99.89	99.89	99.84	99.84	98.42
DF-Fusion	91.14	94.17	92.61	88.33	90.83	90.89
	98.86	99.99	99.95	99.84	99.89	99.39

The experimental results for the diffusion map time-signal and acoustic color feature extraction is shown above in Tables 2 and 3. The tables are listed per original data and diffusion feature embedded data (DF). The rows correspond to the dataset results, either raw signals or acoustic color. Table 2 shows that for the AC dataset that the classification of specific targets is quite poor. However the raw signal data provides sufficient information to classify the specific targets with nearly 90% probability over all target types. Fusion results show varied improvement for sensitivity and little or no change in specificity. Table 3 results show that the diffusion features for the AC data vastly improves performance, yet the raw signal diffusion features improve with targets B and C and decrease with the remaining targets.

5. Conclusions

Diffusion feature fusion results show excellent false alarm reduction with little or no improvement in specific target discrimination. The diffusion feature acoustic color results are far better than the original acoustic color classifications results, especially noted is the improvement in the sensitivity for targets A through E. There is no apparent direct influence on the raw signals for target specific discrimination for this sensor. However, it had been shown previously in [8] that the diffusion map features allow for a much improved classification of targets vs. non-targets for SAS datasets over using the original inverse image time-signals. The number of diffusion features was not optimized and this could be done in future work. In addition, it may prove that for this particular sensor, the acoustic color data is less discriminating than either the image or raw signal data overall.

References

[1] M. Belkin; P. Niyogi, "Laplacian Eigenmaps for dimensionality reduction and data representation," Neural Computation, v.15 n.6, p.1373-1396, June 2003.

[2] R. Coifman; S. Lafon, "Diffusion Maps," Applied and Computational Harmonic Analysis, special issue on diffusion maps and wavelets, vol. 21, pp. 5-30, July 2006.

[3] R. Coifman; S. Lafon; A. Lee; M. Maggioni; B. Nadler; F. Warner; S. Zucker, "Geometric Diffusions as a Tool for Harmonics Analysis and Structure Definition of Data: Multiscale Methods," Proc. Nat"l Academy of Sciences, vol. 102, no. 21, pp. 7432-7437, May 2005.

[4] G. Dobeck, J. Hyland, and L. Smedley, "Automated detection/classification of sea mines in sonar imagery," Proc. SPIE Vol. 3079, 90-110 (1997).

[5] D. Donoho; C. Grimes, "Hessian Eigenmaps: New Locally Linear Embedding Techniques for High-Dimensional Data," Proc. Nat'l Academy of Sciences, vol. 100, no. 10, pp. 5591-5596, May 2003.

[6] Y. Freund and R.E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting." In *Computational Learning Theory: Eurocolt '95*, pages 23–37. Springer-Verlag, 1995.

[7] J.C. Isaacs; “Laplace-Beltrami Eigenfunctions for 3D Shape Matching,” Proc. of SPIE DSS, April 2011.
[8] J.C. Isaacs, and J.D. Tucker, "Signal diffusion features for automatic target recognition in synthetic aperture sonar," Proc. IEEE Digital Signal Processing Workshop, 461-465 (2011).
[9] J.C. Isaacs, and A. Srivastava, “Geodesic shape distance and integral invariant shape features for automatic target recognition,” Proc. IEEE OCEANS, 1-6 (2010).
[10] J.C. Isaacs, “Diffusion map kernel analysis for target classification,” Proc. IEEE OCEANS, vol., no., 1-7 (2009).
[11] P.T. Gough, “A synthetic aperture sonar system capable of operating at high speed and in turbulent media.” IEEE Jour. Oceanic Eng., 11(2), pp333, 1986.
[12] S. Roweis; L. Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding,” Science, vol. 290, pp. 2323-2326, 2000.
[13] G. Sammelmann; J. Christoff; J. Lathrop; “Synthetic Images of Proud Targets”, Proc. IEEE/MTS OCEANS 2004, pp. 266-271.
[14] Z. Zhang; H. Zha, “Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment,” Technical Report CSE-02-019, Dept. of Computer Science and Eng., Pennsylvania State Univ., 2002.