Marine Animal Classification With Correntropy-Loss-Based Multiview Learning

Zheng Cao, Shujian Yu, Bing Ouyang, *Member, IEEE*, Fraser Dalgleish, Anni Vuorenkoski, Gabriel Alsenas, and Jose C. Principe, *Fellow, IEEE*

***Abstract*—To analyze marine animals' behavior, seasonal distribution, and abundance, digital imagery can be acquired by a camera or a Lidar. Depending on the quantity and properties of acquired imagery, the animals are characterized as either features (shape, color, texture, etc.) or dissimilarity matrices derived from different shape analysis methods (shape context, internal distance shape context, etc.). For both cases, multiview learning is critical in integrating more than one set of feature/dissimilarity matrix for higher classification accuracy. This paper adopts correntropy loss as the cost function in multiview learning, which has favorable statistical properties for rejecting noise. For the case of features, the correntropy-loss-based multiview learning and its "entrywise" variation are developed based on the multiview intact space learning algorithm. For the case of dissimilarity matrices, the robust Euclidean embedding algorithm is extended to its multiview form with the correntropy loss function. Results from simulated data and real-world marine animal imagery show that the proposed algorithms can effectively enhance classification rate as well as suppress noise under different noise conditions.**

***Index Terms*—Correntropy loss, dissimilarity matrix, marine animal Lidar imagery, multiview learning.**

I. Introduction

THE study of marine animals' behavior, seasonal distribution, and abundance is vital for various environmental agencies, commercial fishermen, and marine research institutes. To this end, extensive amount of digital imagery and video are acquired using imaging sensors mounted on autonomous underwater vehicles, remotely operated vehicles, remotely operated towed vehicles, and fixed installations. Color images are more intuitive to humans compared to sonar signal, yet manual labeling of images is still a daunting task considering the sheer size of the data. An automated solution is thus preferred. While the automated solution consists of two major steps—detection and classification—this paper will focus on the latter problem only. There are a good number of works concentrating on feature extraction for colored marine imagery. Shape (Fourier descriptors), color (normalized color histograms), and texture (Gabor filters and gray-level co-occurrence) [1] are among the most exploited features. Biological characteristics, such as body part ratio [1] and morphological measurements [2], can distinguish different species as well. Today, the rapid development of convolutional neural networks (CNNs) has opened new possibility for accurate image representation, which has since benefited marine animal classification. To acquire CNN features of an image, one can either input the image to a CNN pretrained by a large database (e.g., ImageNet [3]), which consists of images that are visually similar to the target image [4], or train a new CNN with images homogeneous to the target, as in the example of plankton classification [5].

Recently, the Harbor Branch Oceanographic Institute (HBOI) at the Florida Atlantic University, Fort Pierce, FL, USA, has developed a novel system called unobtrusive multistatic serial Lidar imager (UMSLI) to perform marine hydrokinetic site monitoring and marine animal classification [6], [7]. Initial testing of the UMSLI system has been conducted inside a unique optical test facility at HBOI, which is capable of extensive testing of a variety of electrooptical system configurations under a range of environmental conditions. Compared with optical camera imaging, underwater Lidar imaging has several advantages. First, red laser illuminators are beyond the visible wavelength range of marine life; thus, animals being monitored will not be affected [8]. On the other hand, the optical camera requires a significant amount of white light to illuminate low-light areas and is more obtrusive to marine life. Second, unlike conventional camera whose focus is governed by the lens, Lidar imagery will remain in focus throughout the entire range, which gives it superior detection range. Higher signal-to-noise ratio is also achieved with Lidar due to the higher photon efficiency [9]. Third, the transmitter in the UMSLI system can operate in an adaptive mode, opting for either higher resolution or longer range of detection. A Lidar image of a fish captured by UMSLI is typically 2-D grayscale integrated from the 3-D point cloud Lidar return. Fig. 1 shows examples of Lidar images retrieved from the test tank at HBOI. However, using Lidar imagery for marine animal classification is not without its own issues. As UMSLI is the first attempt to identify an individual marine animal using Lidar imagery, there is virtually no existing database online with similar content. Given that the amount of data obtained from initial UMSLI deployment is also small, training a proper CNN becomes

Manuscript received February 3, 2017; revised October 27, 2017 and March 28, 2018; accepted May 19, 2018. Date of publication August 23, 2018; date of current version October 11, 2019. This work was supported in part by the U.S. Department of Energy under Contract DE-EE0006787 and the FAU/HBOI internal fund. *(Corresponding author: Zheng Cao.)*

Associate Editor: J. Cobb.

Z. Cao, S. Yu, and J. C. Principe are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: zcao87@ufl.edu; yusjlcy9011@ufl.edu; principe@cnel.ufl.edu).

B. Ouyang, F. Dalgleish, A. Vuorenkoski, and G. Alsenas are with the Harbor Branch Oceanographic Institute, Florida Atlantic University, Fort Pierce, FL 34946 USA (e-mail: bouyang@fau.edu; fdalglei@fau.edu; adalglei@fau.edu; galsenas@fau.edu).

Digital Object Identifier 10.1109/JOE.2018.2861500

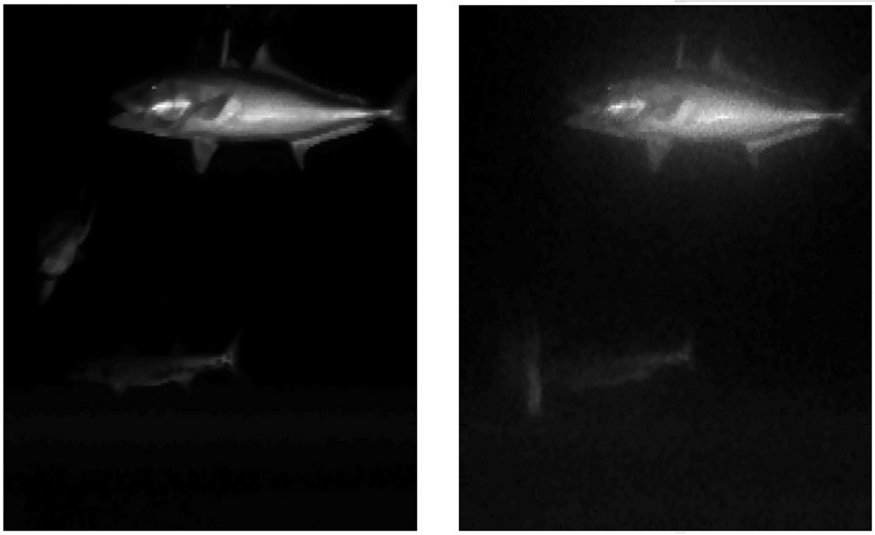


Fig. 1. Lidar image examples. Left image and right image are retrieved under clear water (attenuation coefficient $c = 0.33$) and turbid water ($Ic = 0.73$) conditions, respectively.

problematic with insufficient data. Also, most of the valuable information revealed from the obtained Lidar imagery in the initial experimental data set seems to be the shape of the animal. Using the pixels directly as the feature is not recommended because of orientation variations among shapes, while traditional shape features, such as Zernike [10] or Hu's moments, have relatively weak description ability. There has been an attempt [11] to apply bag of words for quantifying a shape's feature, but the resulting shape vocabulary feature will be very long and highly redundant because the common space in which all shapes reside can have very high dimensions. Instead of feature extraction, most of the existing shape classification or recognition literature adopt a pairwise comparison strategy, creating a matrix directly with each entry representing the similarity/dissimilarity between a pair of shapes. As such, a "descriptor" rather than a "feature" will be enough to represent a shape, which is usually much simpler and intuitive. Commonly used descriptors include shape context (SC) [12], internal distance shape context (IDSC) [13], the triangle descriptor [14], and height functions [15]. With a proper dissimilarity measure (Chi square or earthmover distance [16]) applied on pairs of descriptors, these methods can usually achieve satisfying classification results using a k-nearest neighbor classifier, especially on data with very few training samples. This similarity-matrix-based approach is not limited to shape descriptors only. For instance, Gaussian mixture models [17], [18] have been applied in hand gesture recognition, while the spectral estimation [19] method has been used in an automobile recognition task.

For both feature-based and dissimilarity-matrix-based marine animal classification, utilizing information from multiple sources (different features/descriptors or different view angles) will lead to more robust description of objects, and thus improve classification accuracy [4], [7]. Multiview learning [20] is a group of methods that introduces one function to model one particular view of the data, then jointly optimizes all the functions to improve the learning performance. The goal of this paper is to develop suitable multiview learning algorithms for both data formats, where a "view" can be a feature set or a dissimilarity matrix. There are two major considerations when designing a multiview learning algorithm. First, the algorithm should simultaneously accommodate dimensional reduction because it is a necessary preprocessing step that makes classification faster, and also more accurate when the data are small or have a low-dimensional structure [21], [22]. CNN features usually are high dimensional (4096 for DeCAF [23]) and have considerable redundancy [24], thus it is natural to apply dimensionality reduction. With this principle in mind, one can choose the desired multiview learning framework from various options. For features, the most notable categories of multiview learning algorithms are cotraining [25], multiple kernel learning (MKL) [26], and subspace learning [20]. Only subspace learning involves dimensionality reduction. Examples include multiple spectral embedding [27], multiview nonnegative matrix factorization [28], and multiview intact space learning (MISL) [29]. For dissimilarity matrices, there are very few well-established multiview learning algorithms. It should be noted that MKL cannot be applied to dissimilarity matrices as they are not guaranteed to be positive semidefinite and, hence, are not valid kernel matrices. Cotransduction [30] borrows ideas from cotraining and can be viewed as a multiview approach, yet it lacks a dimensionality reduction mechanism. It may be a good idea to first find a base method that performs dimensionality reduction for a single dissimilarity matrix, then extend it to its multiview version. Potential candidates include principal component analysis (PCA), multidimensional scaling (MDS), manifold learning [31], and autoencoder [32]. Before carrying out any of these algorithms, it is necessary to enforce the dissimilarity matrix to be a proper distance matrix. The robust Euclidean embedding (REE) [33] is an algorithm based on the classical MDS, which also regulates the dissimilarity matrix by enforcing the Euclidean distance. Meanwhile, there is actually one algorithm in the literature with the name multiview MDS (MV-MDS) [34], but it is based on nonclassical MDS and is compatible only with the L_2 cost function, unlike REE.

Another important consideration in designing multiview learning algorithms is the choice of cost functions, which has not been studied in detail in previous works. In a data set, the views given may contain irregularities (noise) of different types and magnitudes. The mean square error is the most widely used cost function, yet its performance is suboptimal for non-Gaussian noise. The correntropy [35] as a nonlinear, local similarity measure that is robust to outliers has attracted researchers in recent years. Notable applications of correntropy include adaptive filtering [36], classification [37], face recognition [38], [39], and robust autoencoder [40]. More recently, the generalized correntropy [41] is proposed and successfully applied to adaptive filtering. It is more versatile than correntropy, as changes in the shape parameter can lead to the suppression of different types of noise.

In this paper, correntropy-loss-based multiview (C-MV) learning algorithms will be developed for both features and dissimilarity matrices. For features, the MISL [29] is employed as the base method for two related C-MV learning algorithms. For dissimilarity matrices, the base method will be REE. As REE is itself single view only, its direct multiview version will be proposed along with a correntropy-based method. For the rest of this paper, the MISL and REE algorithms, as well as the concept of generalized correntropy, will be reviewed first, followed by the derivation of correntropy-loss-based multiview (C-MV) learning algorithms. Experimental results for both simulated data and real-world marine animal data are presented.

II. Background

This section gives a review of two existing algorithms, MISL and REE, as well as the concept of generalized correntropy.

A. MISL Algorithm

The MISL [29] algorithm aims at learning a low-dimensional latent intact subspace from two or more different views. One advantage of MISL is that it does not need the assumption that each view needs to be sufficient; as long as enough views are given, the learned view will be "intact," or fully able to describe the object. MISL uses the Cauchy loss

$$J_{\text{cauchy}}(e) = \log\left(1 + \frac{e^2}{c^2}\right) \tag{1}$$

to minimize the reconstruction error over the latent intact space

$$\min_{\mathbf{x},\mathbf{W}} \frac{1}{MN}\sum_{v=1}^{M}\sum_{i=1}^{N}\log\left(1 + \frac{\|\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)}\mathbf{x}_i\|_2^2}{c^2}\right) + C_1\sum_{v=1}^{m}\|\mathbf{W}_{(v)}\|_F^2 + C_2\sum_{i=1}^{n}\|\mathbf{x}_i\|_2^2. \tag{2}$$

In (1), e refers to an error and c is a user-defined shape parameter. In (2), vectors $\mathbf{z}_i^{(v)}$ and $\mathbf{x}_i$ stand for, respectively, the $d_{(v)} * 1$ feature vector from the vth view and the $d * 1$ common view feature vector to be learned at the ith instance (a single image in the context of image classification), while $\mathbf{W}_{(v)}$ is the vth transformation matrix with dimensions $d_{(v)} * d$. There are M views and N instances in total. The iteratively reweight residuals technique is used to find a solution for (2).

B. REE Algorithm

The classical multidimensional scaling (cMDS) [42] seeks to find the low-dimensional data representation $\mathbf{X}$, whose associated squared Euclidean distance matrix $\mathbf{D}$ approximates the given dissimilarity matrix $\boldsymbol{\Delta}$, whose (i, j)th entry denotes the dissimilarity between the ith and jth instances. Both $\mathbf{D}$ and $\boldsymbol{\Delta}$ are N by N (N is the data size), while $\mathbf{X}$ is N by k ($k < N$) (the ith row of $\mathbf{X}$ represents the ith instance). The cMDS is processed in two stages. First, the dissimilarity matrix is embedded to the Euclidean distance space, which solves the following optimization problem:

$$\min_{\mathbf{D}} ||\mathbf{H}\boldsymbol{\Delta}\mathbf{H} - \mathbf{HDH}||_2^2 \tag{3}$$

where $\mathbf{H} = \mathbf{I} - (1/N)\mathbf{1}\mathbf{1}^T$ is the centering matrix. Second, dimensionality of $\mathbf{X}$ is reduced through PCA.

The REE [33] algorithm states that the robustness of the Euclidean embedding process can be enhanced by two practices: First, replacing the L_2 norm with L_1 norm in the optimization function; and second, rather than projecting the matrix $\mathbf{B} = -(1/2)\mathbf{H}\boldsymbol{\Delta}\mathbf{H}$ to the positive semidefinite cone, REE takes a direct approach by projecting $\boldsymbol{\Delta}$ onto the Euclidean distance matrix. Therefore, REE seeks to solve the optimization problem

$$\min_{\mathbf{D}} W_{ij}|\Delta_{ij} - D_{ij}|. \tag{4}$$

The weighting matrix $\mathbf{W}$ is usually set to all ones. The Gram matrix $\mathbf{B}$ associated with $\mathbf{D}$ should still meet the condition of being positive semidefinite. Since $\mathbf{D}$ and $\mathbf{B}$ are related by

$$D_{ij} = B_{ii} + B_{jj} - B_{ij} - B_{ji} \tag{5}$$

(4) can be optimized by taking subgradient with respect to $\mathbf{B}$, and constrain $\mathbf{B}$ to be positive semidefinite at every iteration.

C. Generalized Correntropy Loss (GC-Loss) Function

The generalized correntropy [41] is a similarity measure between two random variables X and Y

$$V(X,Y) = \mathbf{E}[G_{\alpha,\beta}(X - Y)]. \tag{6}$$

In (6), $G_{\alpha,\beta}(\cdot)$ is called the generalized Gaussian density (GGD) function

$$G_{\alpha,\beta}(e) = \frac{\alpha}{2\beta\Gamma(\frac{1}{\alpha})}\exp\left(-|\frac{e}{\beta}|^{\alpha}\right) = \gamma_{\alpha,\beta}\exp\left(-\lambda|e|^{\alpha}\right) \tag{7}$$

where $\alpha > 0$ is the shape parameter, $\beta > 0$ is the bandwidth parameter, $\lambda = 1/\beta^{\alpha}$ is the kernel parameter, and $\gamma_{\alpha,\beta} = (\alpha/(2\beta\Gamma(1/\alpha)))$ is the normalizing constant [41]. When $\alpha = 2$, the GGD becomes a Gaussian kernel $\kappa(e) = (\sqrt{\lambda}/\sqrt{\pi})\exp(-\lambda e^2) = (1/(\sigma\sqrt{2\pi}))\exp(-(e^2/2\sigma^2))$, which turns $V(X,Y)$ into correntropy—a well-known specific case of generalized correntropy. The generalized correntropy

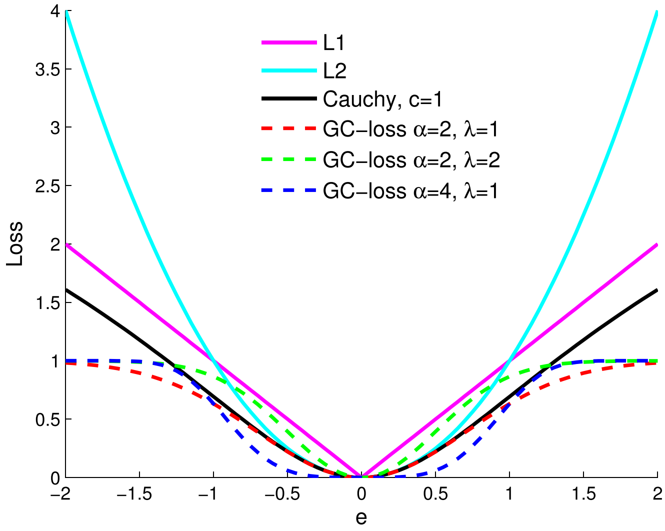


Fig. 2. Comparison of different loss functions. The normalizing constant γ is ignored for GC-loss.

has the advantages of being smooth, positive, and bounded. It also involves higher order absolute moments of the error variable, which makes it more noise resistant compared to the L_2 cost function that uses only the second-order moment [35]

$$V(X,Y) = \gamma_{\alpha,\beta}\sum_{n=0}^{\infty}\frac{(-\lambda)^n}{n!}E\left[|X-Y|^{\alpha n}\right]. \quad (8)$$

In classification tasks, a loss function based on correntropy called correntropy loss is used [43]. The GC-loss as a function of error takes the following form:

$$\begin{aligned} J_{\text{GC}-\text{loss}}(e) &= G_{\alpha,\beta}(0) - G_{\alpha,\beta}(e) \\ &= \gamma_{\alpha,\beta}\left(1-\exp\left(-\lambda|e|^{\alpha}\right)\right). \quad (9)\end{aligned}$$

Fig. 2 shows several loss functions. Clearly, neither L_1 nor Cauchy loss is bounded, unlike GC-loss. In fact, Cauchy loss would resemble GC-loss with $\alpha = 2$ (correntropy loss) when error e is small, according to their Taylor expansion

$$\log\left(1+\frac{e^2}{c^2}\right) = \frac{e^2}{c^2} - \frac{e^4}{2c^4} + \frac{e^6}{3c^6}\cdots$$
$$1-\exp\left(-\lambda|e|^2\right) = \lambda e^2 - \frac{\lambda^2 e^4}{2!} + \frac{\lambda^3 e^6}{3!}\cdots.$$

Given that $\lambda = 1/c^2$, the first two dominant terms are the same for both loss functions. However, Cauchy will suffer from its unboundedness when e is large. GC-loss has the interesting property of behaving like different norms for different values of e [41]. When e is very small, GC-loss acts like $L\alpha$ norm. As e increases, GC-loss moves gradually toward L 0 norm. Therefore, different choices for the shape parameter α are beneficial for different types of noise. Smaller α is better when the distribution of noise is heavy-tailed (e.g., Laplace, α stable), while larger α is better for light-tailed noise (e.g., uniform, binary).

III. C-MV Learning Algorithm

A. *Algorithm for Features*

According to previous analysis, the cost function for C-MV learning can be written as follows:

$$\begin{aligned}\min_{\mathbf{x},\mathbf{W}} R_1(\mathbf{x},\mathbf{W}) &= \frac{1}{MN}\sum_{v=1}^{M}\sum_{i=1}^{N}\gamma_\sigma\left[1-\kappa_\sigma\left(||\mathbf{z}_i^{(v)}-\mathbf{W}_{(v)}\mathbf{x}_i||_2\right)\right] \\ &\quad + C_{10}\sum_{v=1}^{m}||\mathbf{W}_{(v)}||_F^2 + C_{20}\sum_{i=1}^{n}||\mathbf{x}_i||_2^2 \quad (10)\end{aligned}$$

where κ_σ is a Gaussian kernel. Expression (10) can be rewritten in the more compact form as

$$\begin{aligned}\max_{\mathbf{x},\mathbf{W}} R_2(\mathbf{x},\mathbf{W}) &= \sum_{v=1}^{M}\sum_{i=1}^{N}\exp\left(-\frac{||\mathbf{z}_i^{(v)}-\mathbf{W}_{(v)}\mathbf{x}_i||_2^2}{2\sigma^2}\right) \\ &\quad - C_1\sum_{v=1}^{m}||\mathbf{W}_{(v)}||_F^2 - C_2\sum_{i=1}^{n}||\mathbf{x}_i||_2^2. \quad (11)\end{aligned}$$

In (11), variables $\mathbf{z}_i^{(v)}$, $\mathbf{x}_i$, and $\mathbf{W}_{(v)}$ and parameters $d_{(v)}$, C_1, and C_2 have the same connotation as in (2). The main goal is still to solve for the common view $\mathbf{x}_i$. Kernel size σ is usually set equal to or smaller than 1, provided that $z^{(v)}$ is normalized

$$\mathbf{z}^{(v)} := \frac{\mathbf{z}^{(v)}}{\sum_i^N ||\mathbf{z}_i^{(v)}||_2^2/N}.$$

Regularization is used to penalize the terms $\mathbf{x}$ and $\mathbf{W}$. The parameters C_1 and C_2 can be chosen using cross validation.

To solve (11) with half-quadratic (HQ) optimization, a convex function is defined as $g(a) = -a\ln(a) + a$, where $a < 0$. The conjugate function $g^*(b)$ of $g(a)$ is then [44]

$$g^*(b) = \sup_{a<0}\left(ba - g(a)\right) = \exp(-b). \quad (12)$$

In (12), the supremum is achieved when $a = \exp(-b) < 0$. Define

$$b_i^{(v)} = +\frac{||\mathbf{z}_i^v - \mathbf{W}_v\mathbf{x}_i||_2^2}{2\sigma^2}.$$

It follows that

$$\begin{aligned} g^*&\left(\frac{||\mathbf{z}_i^v - \mathbf{W}_v\mathbf{x}_i||_2^2}{2\sigma^2}\right) \\ &= \sup_{a_i^{(v)}<0}\left\{\frac{||\mathbf{z}_i^v - \mathbf{W}_v\mathbf{x}_i||_2^2}{2\sigma^2}a_i^{(v)} - g(a_i^{(v)})\right\} \\ &= \exp\left(-\frac{||\mathbf{z}_i^v - \mathbf{W}_v\mathbf{x}_i||_2^2}{2\sigma^2}\right)\end{aligned}$$

where

$$a_i^{(v)} = -\exp\left(-\frac{||\mathbf{z}_i^v - \mathbf{W}_v \mathbf{x}_i||_2^2}{2\sigma^2}\right) < 0.$$

Therefore,

$$\begin{aligned} R_2(\mathbf{x}, \mathbf{W}) = & \sum_{v=1}^{M}\sum_{i=1}^{N} \sup_{a_i^v < 0} \left\{ \frac{||\mathbf{z}_i^v - \mathbf{W}_v \mathbf{x}_i||_2^2}{2\sigma^2} a_i^v - g(a_i^v) \right\} \\ & - C_1 \sum_{v=1}^{M} ||\mathbf{W}_v||_F^2 - C_2 \sum_{i=1}^{N} ||\mathbf{x}_i||_2^2 \\ = & \sup_{\mathbf{A} \prec 0} \left\{ \sum_{v=1}^{M}\sum_{i=1}^{N} \left[\frac{||\mathbf{z}_i^v - \mathbf{W}_v \mathbf{x}_i||_2^2}{2\sigma^2} a_i^v - g(a_i^v) \right] \right. \\ & \left. - C_1 \sum_{v=1}^{M} ||\mathbf{W}_v||_F^2 - C_2 \sum_{i=1}^{N} ||\mathbf{x}_i||_2^2 \right\}. \end{aligned} \tag{13}$$

With (13), (11) is equivalent to

$$\begin{aligned} & \max_{\mathbf{x}, \mathbf{W}, \mathbf{A} \prec 0} R_3(\mathbf{x}, \mathbf{W}, \mathbf{A}) \\ & = \sum_{v=1}^{M}\sum_{i=1}^{N} \left[\frac{||\mathbf{z}_i^v - \mathbf{W}_v \mathbf{x}_i||_2^2}{2\sigma^2} a_i^v - g(a_i^v) \right] \\ & - C_1 \sum_{v=1}^{M} ||\mathbf{W}_v||_F^2 - C_2 \sum_{i=1}^{N} ||\mathbf{x}_i||_2^2 \end{aligned} \tag{14}$$

where $\mathbf{A}_{(v,i)} = a_i^{(v)} < 0$. A double-loop alternating optimization scheme can be applied to optimize (14). In the outer loop, the alternating optimization is between $\mathbf{A}$ and $\{\mathbf{x}, \mathbf{W}\}$. Given x_i and $\mathbf{W}_{(v)}$ for all i and v, (14) becomes equivalent to

$$\max_{\mathbf{A}} \sum_{v=1}^{M}\sum_{i=1}^{N} \left[\frac{||\mathbf{z}_i^v - \mathbf{W}_v \mathbf{x}_i||_2^2}{2\sigma^2} a_i^v - g(a_i^v) \right] \tag{15}$$

whose analytical solution is

$$a_i^{(v)} = -\exp\left(-\frac{||\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)} \mathbf{x}_i||_2^2}{2\sigma^2}\right). \tag{16}$$

When $a_i^{(v)}$ is given, (14) is equivalent to

$$\begin{aligned} & \max_{\mathbf{x}, \mathbf{W}} \sum_{v=1}^{M}\sum_{i=1}^{N} \frac{a_i^{(v)}}{2\sigma^2} ||\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)} \mathbf{x}_i||_2^2 \\ & \quad - C_1 \sum_{v=1}^{M} ||\mathbf{W}_{(v)}||_F^2 - C_2 \sum_{i=1}^{N} ||\mathbf{x}_i||_2^2 \\ & \Rightarrow \min_{\mathbf{x}, \mathbf{W}} \sum_{v=1}^{M}\sum_{i=1}^{N} (-a_i^{(v)}) ||\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)} \mathbf{x}_i||_2^2 \\ & \quad + C_1 \sum_{v=1}^{M} ||\mathbf{W}_{(v)}||_F^2 + C_2 \sum_{i=1}^{N} ||\mathbf{x}_i||_2^2. \end{aligned} \tag{17}$$

In the latter equation $C_1 := C_1 * 2\sigma^2$, $C_2 := C_2 * 2\sigma^2$. In the inner loop, the alternating optimization is between $\mathbf{W}$ and $\mathbf{x}$. Given $\mathbf{W}_{(v)}$, (17) becomes N independent problems

$$\min_{\mathbf{x}_i} \sum_{v=1}^{M} (-a_i^{(v)}) ||\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)} \mathbf{x}_i||_2^2 + C_2 ||\mathbf{x}_i||_2^2. \tag{18}$$

For (18), taking the derivative with respect to $\mathbf{x}_i$ and equating it to zero, one will get

$$\mathbf{x}_i = \left(\sum_{v=1}^{M} a_i^v \mathbf{W}_v^T \mathbf{W}_{(v)} - C_2 \mathbf{I} \right)^{-1} \left(\sum_{v=1}^{M} a_i^{(v)} \mathbf{W}_{(v)}^T \mathbf{z}_i^{(v)} \right). \tag{19}$$

When $\mathbf{x}_i$ is given, (17) becomes M independent problems

$$\min_{\mathbf{W}^{(v)}} \sum_{i=1}^{N} \left(-a_i^{(v)}\right) ||\mathbf{z}_i^{(v)} - \mathbf{W}_{(v)} \mathbf{x}_i||_2^2 + C_1 ||\mathbf{W}_{(v)}||_F^2. \tag{20}$$

Let the derivative with respect to $\mathbf{W}^{(v)}$ equal zero and solve for $\mathbf{W}^{(v)}$, one will get

$$\mathbf{W}_{(v)} = \left(\sum_{i=1}^{N} a_i^{(v)} \mathbf{z}_i^{(v)} \mathbf{x}_i^T \right) \left(\sum_{i=1}^{N} a_i^{(v)} \mathbf{x}_i \mathbf{x}_i^T - C_1 \mathbf{I} \right)^{-1}. \tag{21}$$

Alternatively to C-MV, this paper will propose another multiview learning algorithm for features called correntropy-loss entrywise multiview (Ce-MV) learning for features to mitigate some of the deficiencies of (11). Its optimization goal is

$$\begin{aligned} & \max_{\mathbf{x}, \mathbf{W}} \sum_{v=1}^{M} \frac{1}{d_{(v)}} \sum_{i=1}^{N} \sum_{j=1}^{d_{(v)}} \exp\left(-\frac{(z_{ij}^{(v)} - \mathbf{W}_j^{(v)} \mathbf{x}_i)^2}{2\sigma^2} \right) \\ & \quad - C_1 \sum_{v=1}^{m} \sum_{j=1}^{d_{(v)}} ||\mathbf{W}_j^{(v)}||_F^2 - C_2 \sum_{i=1}^{n} ||\mathbf{x}_i||_2^2. \end{aligned} \tag{22}$$

Notations of variables and parameters in (22) are the same as in (11). The coefficient $(1/d_{(v)})$ ensures that features with higher dimensions are given the same weight as lower dimensional feature set. The main difference between (22) and (11) is that Ce-MV maximizes the overall correntropy over every individual entry z_{ij} of the input feature, not just every feature vector $\mathbf{z}_i$. The potential advantage is that when certain entries of $\mathbf{z}_i$ have abnormal values caused by noise or low-quality features, the whole $\mathbf{z}_i$ will be affected as a result of the L_2 norm used inside the correntropy function in (11). On the other hand, abnormal values in $\mathbf{z}_i$ will restrict its effect to itself only as in (22), making the good features contribute more effectively. A good choice for kernel size σ for Ce-MV should be much smaller than that for C-MV, preferably rescaled to $1/\sqrt{d_{(v)}}$ of the original.

The optimization for Ce-MV can be done by following the same HQ and the alternating optimization technique. Expression

(22) is equivalent to

$$\Rightarrow \max_{\mathbf{x},\mathbf{W},\mathbf{A}} \sup_{\mathbf{A}\prec 0} \left\{ \sum_{v=1}^{M} \frac{1}{d_v} \sum_{i=1}^{N} \sum_{j=1}^{d_{(v)}} \times \left[\frac{(z_{ij}^{(v)} - \mathbf{W}_j^{(v)}\mathbf{x}_i)^2}{2\sigma_{(v)}^2} a_{ij}^{(v)} - g(a_{ij}^{v}) \right] - C_1 \sum_{v=1}^{m} \sum_{j=1}^{d_{(v)}} ||\mathbf{W}_j^{(v)}||_F^2 - C_2 \sum_{i=1}^{n} ||\mathbf{x}_i||_2^2 \right\}. \tag{23}$$

Solution for $a_{ij}^{(v)}$ in the outer loop of alternating optimization is

$$a_{ij}^{(v)} = -\exp\left(-\frac{(z_{ij}^{(v)} - \mathbf{W}_j^{(v)}\mathbf{x}_i)^2}{2\sigma_{(v)}^2}\right). \tag{24}$$

In the inner loop, solutions for $\mathbf{x}_i$ and $\mathbf{W}_j^{(v)}$ are

$$\mathbf{x}_i = \left(\sum_{v=1}^{M} \frac{1}{d_v} \sum_{j=1}^{d_{(v)}} a_{ij}^{(v)} \left(\mathbf{W}_j^{(v)}\right)^T \mathbf{W}_j^{(v)} - C_2\mathbf{I} \right)^{-1} \times \left(\sum_{v=1}^{M} \frac{1}{d_v} \sum_{j=1}^{d_{(v)}} a_{ij}^{(v)} z_{ij}^{(v)}) \left(\mathbf{W}_j^{(v)}\right)^T \right) \tag{25}$$

$$\mathbf{W}_j^{(v)} = \left(\sum_{i=1}^{N} a_{ij}^{(v)} z_{ij}^{(v)} \mathbf{x}_i^T \right) \left(\sum_{i=1}^{N} a_{ij}^{(v)} \mathbf{x}_i \mathbf{x}_i^T - C_1\mathbf{I} \right)^{-1}. \tag{26}$$

Note that the intermediate variables $a_i^{(v)}$ for C-MV and $a_{ij}^{(v)}$ for Ce-MV are reflective of the importance of an instance $\mathbf{z}_i$ when the algorithm converges. Additionally, $a_{ij}^{(v)}$ can reveal the importance of particular features within the instance, which cannot be said for $a_i^{(v)}$.

Convergence analysis: It can be proved that for C-MV, the sequence $R_3(\mathbf{x}^k, \mathbf{W}^k, \mathbf{A}^k)(k = 1, 2, \ldots$ stands for the number of outer iteration) in (14) converges: From (11), (13), and (14), it is clear that $R_3(\mathbf{x}, \mathbf{W}, \mathbf{A}) \leq R_2(\mathbf{x}, \mathbf{W}) \leq MN$, which means $R_3(\mathbf{x}, \mathbf{W}, \mathbf{A})$ is upper bounded. Then, from (15) and (17), it can be concluded that $R_3(\mathbf{x}^k, \mathbf{W}^k, \mathbf{A}^k) \leq R_3(\mathbf{x}^k, \mathbf{W}^k, \mathbf{A}^{k+1}) \leq R_3(\mathbf{x}^{k+1}, \mathbf{W}^{k+1}, \mathbf{A}^{k+1})$, i.e., $R_3(\mathbf{x}^k, \mathbf{W}^k, \mathbf{A}^k)$ is nondecreasing. Therefore, the sequence $R_3(\mathbf{x}^k, \mathbf{W}^k, \mathbf{A}^k)(k = 1, 2, \ldots)$ converges. By the same token, convergence of Ce-MV can be proven as well.

The C-MV and Ce-MV algorithms are summarized in Algorithm 1. The disadvantage of Ce-MV is that it is much slower, especially when the dimensions of original features $d_{(v)}$ are high, because computing $d_{(v)}$ values of $a_{ij}^{(v)}$ and $\mathbf{W}_j^{(v)}$ are more time consuming than computing a single value of $a_i^{(v)}$ and $\mathbf{W}^{(v)}$.

Algorithm 1: Algorithm for C-MV/Ce-MV.

```
Input: $\mathbf{z}^{(v)}$
Initialization: $\mathbf{W}^{(v)}, \mathbf{x}, \sigma, C_1, C_2$
for $k = 1$ to maximum outer iteration do
    update $a_i^{(v)}$ as in (16)/ $a_{ij}^{(v)}$ as in (24) for all subscripts
    for $k_i = 1$ to maximum inner iteration do
        update $\mathbf{x}_i$ as in (19)/(25) for all $i$
        update $\mathbf{W}^{(v)}$ as in (21)/ $\mathbf{W}_j^{(v)}$ as in (26) for all subscripts
    end for
end for
return $\mathbf{x}$
```

B. Algorithm for Dissimilarity Matrices

First, the REE algorithm can be naturally extended to its own multiview learning version, which will be called multiview REE (MV-REE) in this paper. Suppose the multiple views consist of M $N * N$ real-valued squared dissimilarity matrices $\mathbf{\Delta}^{(1)}, \ldots, \mathbf{\Delta}^{(M)}$, with $(\mathbf{\Delta}^{(v)})^T = \mathbf{\Delta}^{(v)}$ and $\text{diag}(\mathbf{\Delta}^{(v)}) = \mathbf{0}$. Meanwhile, a common view for all M views is the squared Euclidean distance matrix $\mathbf{D}$. A cost function can be written in a similar manner to (4), aiming at the minimization of the overall L_1 distance between all views and the common view

$$\min_{\mathbf{B}} f_0, f_0 = \min \sum_{v=1}^{M} \sum_{ij} W_{ij}^{(v)} |\Delta_{ij}^{(v)} - D_{ij}|. \tag{27}$$

One can optimize (27) with the same subgradient approach. A subgradient for f_0 with respect to the Gram matrix $\mathbf{B}$ is

$$[g_0(\mathbf{B})]_{ij} = \begin{cases} -\sum_{v=1}^{M} W_{ij}^{(v)} \text{sign}(D_{ij} - \Delta_{ij}^{(v)}), & i \neq j \\ \sum_{v=1}^{M} \sum_{k=1}^{N} W_{ik}^{(v)} \text{sign}(D_{ik} - \Delta_{ik}^{(v)}), & i = j \end{cases} \tag{28}$$

where $\mathbf{B}$ is associated with $\mathbf{D}$ as in (5).

The proposed correntropy-loss-based multiview robust Euclidean embedding (C-MV-REE) replaces the L_1 cost function in (4) with the correntropy loss

$$\max_{\mathbf{B}} f, f = \sum_{v=1}^{M} \sum_{ij} W_{ij}^{(v)} \exp\left(-\frac{(\Delta_{ij}^{(v)} - D_{ij})^2}{2\sigma^2}\right). \tag{29}$$

Kernel size σ can be set equal to or smaller than the median values of the dissimilarity matrices. The goal is to optimize (29) with respect to $\mathbf{B}$. Unlike REE, which has a nondifferentiable cost function, expression (29) is differentiable and a gradient ascent optimization approach can be thus adopted. According to matrix calculus, the derivative of f with respect to $\mathbf{B}$ is

$$\frac{\partial f}{\partial B_{ij}} = \text{trace}\left(\frac{\partial f}{\partial \mathbf{D}} \frac{\partial \mathbf{D}}{\partial B_{ij}}\right) \tag{30}$$

where

$$\left[\frac{\partial f}{\partial \mathbf{D}}\right]_{ij} = \sum_{v=1}^{M} W_{ij}^{(v)} \exp\left(-\frac{(\Delta_{ij}^{(v)} - D_{ij})^2}{2\sigma^2}\right) \cdot \frac{\Delta_{ij}^{(v)} - D_{ij}}{\sigma^2} \tag{31}$$

Algorithm 2: Algorithm for C-MV-REE.

```
Input: $\boldsymbol{\Delta}^{(v)}$, $\mathbf{W}$
Initialization: $\mathbf{B}^0$, $\eta$, $\sigma$
1: for k = 1 to maximum iteration do
2:     $\mathbf{B}^k = \mathbf{B}^{k-1} + \eta \frac{\partial f}{\partial \mathbf{B}}$ as in (33)
3:     Decompose $\mathbf{B}$ into $\mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$ (spectral decomposition)
4:     $[\boldsymbol{\Lambda}_+]_{ij} = \max\{\boldsymbol{\Lambda}_{ij}, 0\}$
5:     $\mathbf{B}^k = \mathbf{U}\boldsymbol{\Lambda}_+\mathbf{U}^T$
6: end for
7: return $\mathbf{X} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}$
```

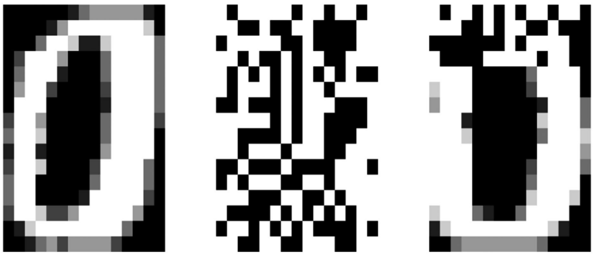


Fig. 3. Examples of the uncorrupted image, complete salt and pepper noise, and partially noisy image.

and

$$\left[\frac{\partial \mathbf{D}}{\partial B_{ij}}\right]_{i'j'} = \begin{cases} -1, & (i'=i, j'=j) \text{ or} \\ & (i'=j, j'=i), \qquad \text{if } i \neq j \\ 0, & \text{others} \end{cases}$$

$$\left[\frac{\partial \mathbf{D}}{\partial B_{ij}}\right]_{i'j'} = \begin{cases} 2, & i'=i \text{ and } j'=i \\ 1, & i'=i \text{ xor } j'=i, \qquad \text{if } i = j \\ 0, & \text{others.} \end{cases} \tag{32}$$

Therefore

$$\frac{\partial f}{\partial B_{ij}} = \sum_{v=1}^{M} W_{ij}^{(v)} \exp\left(-\frac{(\Delta_{ij}^{(v)} - D_{ij})^2}{2\sigma^2}\right) \cdot \frac{D_{ij} - \Delta_{ij}^{(v)}}{\sigma^2}$$
$$\frac{\partial f}{\partial B_{ij}} = \sum_{v=1}^{M}\sum_{k=1}^{N} W_{ik}^{(v)} \exp\left(-\frac{(\Delta_{ik}^{(v)} - D_{ik})^2}{2\sigma^2}\right) \cdot \frac{\Delta_{ik}^{(v)} - D_{ik}}{\sigma^2}. \tag{33}$$

In (33), the upper and lower equations correspond to the cases when $i \neq j$ and $i = j$, respectively.

Convergence analysis: The correntropy loss function f in (29) is an MN^2 dimensional pseudoconvex function. When gradient ascent/descent approach is used for optimization, global convergence is guaranteed provided that the step size is sufficiently small [43]. This will be illustrated by the convergence curves (see Figs. 6 and 8) in Section IV.

The C-MV-REE algorithm is summarized in Algorithm 2. One can obtain the MV-REE algorithm by simply replacing the term $(\partial f/\partial \mathbf{B})$ by $-g_0(\mathbf{B})$. The algorithm returns the $N * N$ configuration matrix $\mathbf{X}$, whose first k columns that corresponds to k dominant eigenvalues of $\mathbf{B}$ comprises a new $N * k$ configuration matrix $\mathbf{X}'$. $\mathbf{X}'$ can be seen as the explicit representation of the original data set from which a new dissimilarity matrix can be calculated. One can also treat $\mathbf{X}'$ as feature vectors, opening up the possibility of using classifiers other than kNN [such as a support vector machine (SVM)].

Computational time of C-MV-REE is at the same order of magnitude as the base method REE. This is because C-MV-REE differs from REE at step 2 only in Algorithm 2. It takes M times as much time for MV-REE to run step 2 compared to REE. Meanwhile, the running time of step 2 for C-MV-REE is slightly less than twice of that of MV-REE, according to MATLAB simulation.

Although all cost functions and derivations in Sections III-A and III-B pertain to correntropy only, the same optimization schemes work for cost functions using generalized correntropy as well. For C-MV and Ce-MV, the HQ optimization is always applicable because the expectation function always exists. For C-MV-REE, it is obvious that the same gradient ascent method will apply for generalized correntropy with any α.

IV. Experiments and Result Analysis

Experiments are performed on both simulated data and real-world marine animal data. The main objectives of experimenting with simulated data set are to study the C-MV learning algorithms' performance in the presence of noise, compared to algorithms with different cost functions. Multiview algorithms' superiority over their single-view counterparts is also studied. The first real-world data are features from color images, on which the feature-based algorithms (C-MV and Ce-MV) will be tested. The second data are dissimilarity matrices that are derived from shape analysis performed on Lidar data, on which the dissimilarity-based algorithm (C-MV-REE) will be tested. C-MV-REE is also suitable for the first data set because features can be turned into dissimilarity matrices.

A. Experiments on Simulated Data Set

1) UCI Handwritten Digit Classification: This experiment is to test different multiview algorithms for the features. The C-MV algorithm and its variant, the Ce-MV algorithm, will be tested on the UCI handwritten digit data set (http://archive.ics.uci.edu/ml/datasets/Multiple+Features). The data set has 2000 instances equally divided into 10 classes (each digit is a class). In the experiment, the data set is evenly divided into two batches, one for training and the other for testing. In each batch, the proportion of any class is still 10%. Two features used for the ith instance are the image itself stretched into a vector $\mathbf{z}_i^{(1)}$ and the Zernike moment $\mathbf{z}_i^{(2)}$, whose dimensions are 240 and 47, respectively. Two separate noise conditions are considered for the first feature. In the first condition (see Fig. 3, left and center plots), a portion of all images (e.g., 250 images out of 2000) are replaced with salt and pepper noise, which is a prominent case of non-Gaussian noise. The mean, maximum, and minimum pixel values of the corrupted images are the same as the rest uncorrupted images. In the second condition (see Fig. 3, right plot), a part (e.g., 60 pixels out of 240) of every

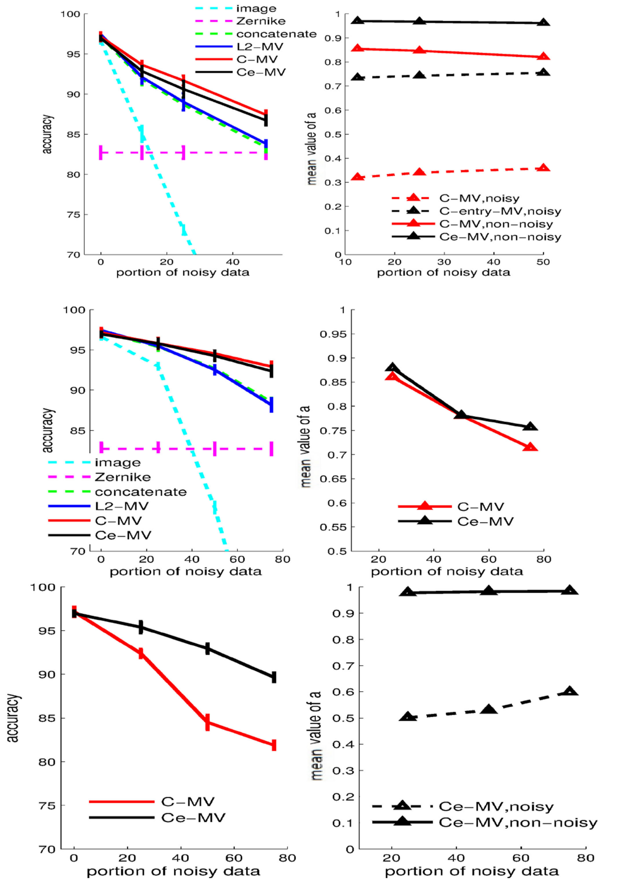


Fig. 4. Left column: Classification accuracy. Right column: Top and center rows—mean value of $a_i^{(1)}$ (mean across instance) for C-MV and Ce-MV; bottom row—mean value of $a_j^{(1)}$ (mean across pixels) for Ce-MV. Note that $a_j^{(1)}$ is unavailable for C-MV. The top row corresponds to the first noise condition, where 0%, 12.5%, 25%, and 50% of images are replaced with salt and pepper noise that has the same mean, maximum, and minimum pixel values as the rest uncorrupted images. The center row corresponds to the second noise condition, where 0%, 25%, 50%, and 75% of pixels in every image are replaced with the same salt and pepper noise. The bottom row also corresponds to the second noise condition, but the magnitude of noise is three times as large as that of the center row. In all cases, mean of $a_i^{(2)}$ or $a_j^{(2)}$ are very close to 1 (around 0.98) and are not shown in plot.

image are replaced with salt and pepper noise. The other feature, Zernike moments, is left unchanged for both conditions.

For C-MV $\sigma = 0.5$, while for Ce-MV, $\sigma_{(v)}$ are rescaled accordingly. Dimension d of the single-view feature to be learned $\mathbf{x}_i$ for both algorithms is set as 60. These parameters are fixed throughout the experiment. The four other methods used for comparison are the following:

1) using the first feature (image) only;
2) using the second feature (Zernike) only;
3) using the concatenated feature;
4) the L_2-MV algorithm, which employs the L_2 cost function but is otherwise the same as C-MV.

The reason that MISL (which uses Cauchy loss function) is not compared is that its performance is very close to that of C-MV under the noise level in this experiment, according to the analysis in Section II-C. To emphasize the importance of the cost function being bounded, the L_2 cost function is used instead. For methods (1)–(3), instead of using the features directly, dimensional reduction is applied by feeding two identical feature sets into the same C-MV algorithm to ensure that dimensionality will not be a factor for any difference in performance. The output feature for all six methods will be consequently used as input to SVM for classification.

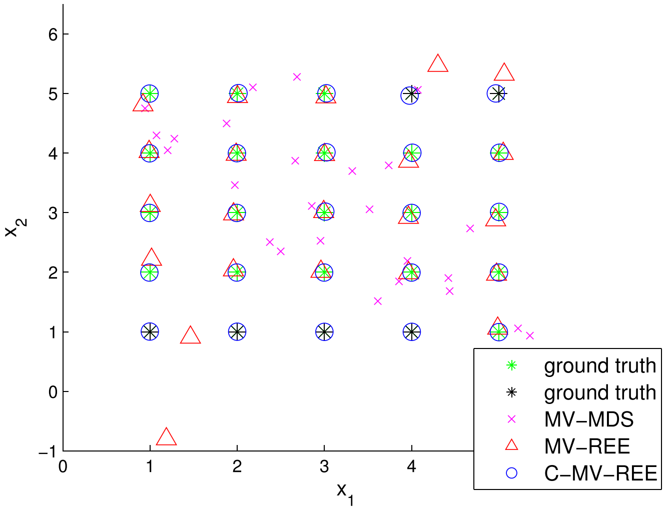


Fig. 5. Comparison of three methods on reconstructing 2-D point set. The original point set is marked with "*," with lighter markers denoting points whose respective distances are uncorrupted by noise, and darker markers denoting otherwise.

Results are shown in Fig. 4. For both noise conditions, applying multiview learning rather than using a single view will significantly boost the classification performance. For the first noise condition (top row), best performances are achieved by C-MV and Ce-MV. This is because C-MV and its entrywise variation are able to distinguish the nonnoisy data instances from the noisy ones, which is reflected by $a_i^{(1)}$. The top right plot shows that the mean value of $a_i^{(1)}$ across nonnoisy instances is consistently higher than those across noisy instances. Meanwhile, L_2-MV assumes the same importance (1) for every instance, whereas doing feature concatenation instead of multiview learning will forcefully give the same importance for different views, leading to suboptimal performance. For the second condition with smaller noise level (center row), C-MV and Ce-MV still have better performance. This is because both algorithms will correctly give smaller $a_i^{(1)}$ than $a_i^{(2)}$ across all instances i. It can be seen that the more noisy instances there are, the smaller $a_i^{(1)}$ is. The advantage of Ce-MV over C-MV begins to show where noise magnitude is higher (bottom row). As explained in Section III-A, C-MV can weight instances only, not individual features (pixels in this setting). Noise on part of the pixels that is big enough will affect the whole instance. On the other hand, Ce-MV can effectively separate the nonnoisy and noisy parts of an instance, giving lesser importance to the noisy pixels and, thus, enhancing the quality of the learned $\mathbf{x}_i$.

2) 2-D Point Set Reconstruction: This experiment is to test different multiview learning algorithms for the dissimilarity matrices. A 2-D point set with $N = 25$ points (see Fig. 5) is considered for this experiment. The goal is to recover the original configuration given two different views of the point set. The Euclidean distance matrix is computed from the original point

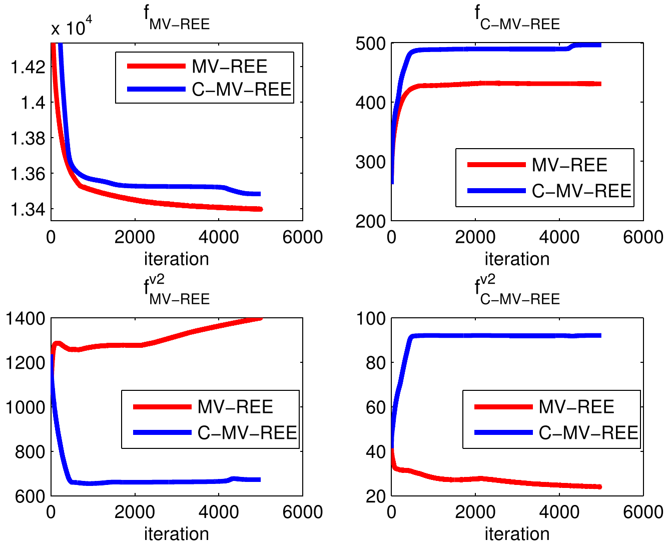


Fig. 6. Values of different cost functions for 2-D point set reconstruction. All red lines represent optimization using MV-REE, and blue lines represent optimization using C-MV-REE. The upper left figure and upper right figure are the values of cost functions (27) and (29), respectively. The lower figures, on the other hand, are values of a fraction of the same cost functions; only the second view ($v = 2$) is considered, and the range of i is from 1 to 4 rather than from 1 to 25 (range of j is still from 1 to 25).

set. Noise is then added: For the first view $\mathbf{\Delta}^{(1)}$, all distances pertaining to the first four points are corrupted with salt and pepper noise $\delta_{i,j} = d_{i,j} + \epsilon_{i,j}$ $(i < j)$, where $\epsilon_{i,j}$ has a magnitude of 10. For the second view $\mathbf{\Delta}^{(2)}$, noise of the same type and magnitude corrupts the 24th and the 25th points. The values of $\delta_{i,j}$ are truncated such that $\delta_{i,j} > 0$. The two distance matrices are then symmetrized.

Three methods are compared: MV-MDS [34], MV-REE, and C-MV-REE, the latter two have been introduced in Section III. For MV-REE, the initial step size η_0 is chosen to be 0.05. The step size at ith iteration will be $\eta = \eta_0/\sqrt{i}$ according to the original REE algorithm. For C-MV-REE, $\eta = 0.1$, which is fixed throughout the iterations. They are chosen such that convergence speed is approximately the same for both algorithms. They are also small enough to ensure good convergence. Since the median values of $\mathbf{\Delta}^{(1)}$ and $\mathbf{\Delta}^{(2)}$ are 9 and 5, respectively, kernel size σ is chosen as 3 for C-MV-REE. From the result (see Fig. 5), it can be seen that MV-MDS performs very poorly as it is based on the nonclassical MDS framework whose mechanism of preserving the Euclidean distances is weak. On the other hand, C-MV-REE achieved a slightly better performance than MV-REE in recovering points that are uncorrupted by noise, and a much better performance in recovering points that have noisy distances in either views. This can be explained intuitively by the fact that C-MV-REE utilizes well the nonnoisy part in one view (e.g., points 1–4 in the second view) that is noisy in the other, which cannot be said for MV-REE. Fig. 6 shows that for both MV-REE and C-MV-REE, the overall cost functions (27) and (29) are correctly optimized over the iterations; however, as the lower figures suggest, C-MV-REE can also correctly optimize the fraction of (29) that corresponds to points 1–4 in the second view, while MV-REE cannot.

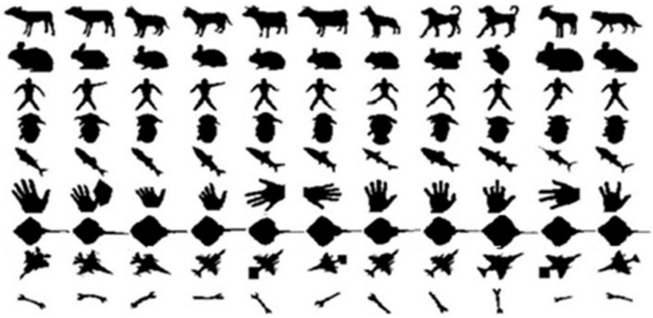


Fig. 7. Kimia-99 consists of nine different classes of objects, with 11 instances in each class.

3) Shape Retrieval: The C-MV-REE algorithm is further tested on the artificial Kimia-99 data set [45], which has been frequently used as a benchmark for shape retrieval tasks. The data set is shown in Fig. 7. For every instance, 10 out of 98 most similar instances are found using two algorithms, namely "SC" [12] and "information point set registration (IPSR)" [46]. Ideally, the ten instances should be from the same class as the query instance. The highest total number of correct findings is, thus, $10 * 99 = 990$. No additional noise is added to the data, but the resulting dissimilarity matrices are expected to be noisy due to variation in shapes and imperfection of SC and IPSR algorithms.

This experiment will first test the performance of the two algorithms individually. Both SC and IPSR will return 99*99 similarity matrices, which are consequently transformed into proper dissimilarity matrices $\mathbf{\Delta}_{\text{SC}}$, $\mathbf{\Delta}_{\text{IPSR}}$ (symmetric, main diagonal are zeros). The ten entries in each row that have the smallest dissimilar values are considered as being from the same class as the query. Next, multiview learning algorithms are applied to combine $\mathbf{\Delta}_{\text{SC}}$ and $\mathbf{\Delta}_{\text{IPSR}}$. The following approaches are also tested for comparison purposes: First, heuristic combination $\mathbf{\Delta}_{\text{heu}}$, which is the elementwise square root of the Hadamard product $\mathbf{\Delta}_{\text{SC}} \circ \mathbf{\Delta}_{\text{IPSR}}$, as in [46]; and second, use (single view) REE only to obtain a new dissimilarity matrix for each of $\mathbf{\Delta}_{\text{SC}}$ and $\mathbf{\Delta}_{\text{IPSR}}$. Parameter settings for SC and IPSR are same as in [46], except that the nonaffine transformation for IPSR is not applied in this experiment for its limited effectiveness. Step sizes for MV-REE and C-MV-REE are 0.02 and 0.01, respectively. Since both $\mathbf{\Delta}_{\text{SC}}$ and $\mathbf{\Delta}_{\text{IPSR}}$ have median values of around 1.9, kernel size σ is selected as 1.5. For all approaches that involve REE, the final configuration matrix $\mathbf{X}'$ is the first eight columns of $\mathbf{X}$. Table I summarizes the results. It first shows that REE can greatly improve shape retrieval accuracy even if only one dissimilarity is considered. This is because REE has the advantages of being a classical MDS algorithm. Meanwhile, the assumption that the dissimilarity among the instances can be quantified by Euclidean distance is also validated. Second, C-MV-REE achieves perfect retrieval result and outperforms MV-REE. Fig. 8 shows that both algorithms converge well, yet MV-REE has heavier oscillation due to its usage of the subgradient method.

TABLE I
KIMIA-99 SHAPE RETRIEVAL RESULTS

Method	Total correct findings
SC only	927
IPSR only	929
SC only, REE	973
IPSR only, REE	980
Heuristic comb.	948
MV-REE	984
C-MV-REE	990

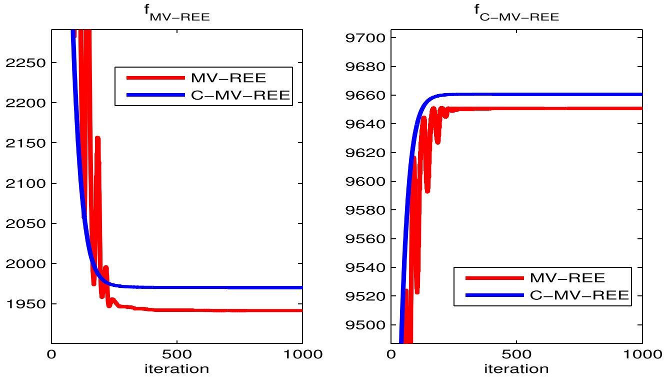


Fig. 8. Values of cost functions (27) and (29) for Kimia-99 data set. Red lines represent optimization using MV-REE, and blue lines represent optimization using C-MV-REE.

B. Marine Animal Classification

1) Classification of Color Images: The first marine animal classification experiment is carried out on color images—the Taiwan sea fish data set (http://sourceforge.net/projects/fish4knowledgesourcecode/). The data can be found inside the "Fish Recognition" folder under the "Files" tab. It has been studied throughout the European Fish4Knowledge project, including works that focus on feature extraction and classification [1], [47]. The original data set has 23 types of fish and over 20 000 fish instances, from which the seven types with the least number of instances (in total 237 instances) are used in the experiment (see Fig. 9). A small data set is used here to be consistent with the classification task for the UMSLI Lidar. In the current stage of the test tank experiment and UMSLI deployment, the retrieved data size is also small. There will not be significant marine animals come into contact with UMSLI deployment, until the monitoring system has been in place for a significant amount of time. The two features used for this data set, namely the CNN feature and the hand-designed feature, have dimensionalities of 4096 and 2626, respectively. Detailed descriptions of the two features can be seen in [4].

This experiment will first test the feature-based multiview algorithms. Kernel size σ is 1 for C-MV and 0.02 for Ce-MV. Dimension d of $\mathbf{x}$ is chosen as 60. Dimensional reduction is also performed when only a single feature (CNN or hand-designed) or the concatenated feature is used. Both SVM and 1-nearest neighbor classifier are tested to evaluate the quality of learned feature more comprehensively. The data set is randomly divided into five sections of equal size. As before, in each section, the proportion of any type of fish equals its proportion in the whole data. Table II compares classification results for six methods with different ratios of training and testing data. Reducing the dimension of data from 4096 + 2626 to a much lower 60 is shown to be beneficial for classification. On the other hand, separately input the CNN and hand features into the multiview learning algorithms (C-MV or Ce-MV) will result in higher classification accuracy than concatenating the features.

Furthermore, in Table III, it is tabulated that the dissimilarity-based multiview algorithm (C-MV-REE) is also a viable option. Any Gram matrix $\mathbf{K}$ computed from features with Gaussian kernel is transformed into dissimilarity matrix $\boldsymbol{\Delta}$ first. The value λ is selected as 50 here

$$\mathbf{K}'_{ij} = \exp\left(-\lambda \mathbf{K}_{ij}\right), \boldsymbol{\Delta}_{\mathbf{ij}} = \frac{2}{\left(\mathbf{K}'_{ij} + \mathbf{K}'_{ji}\right)}. \tag{34}$$

This experiment examines whether the REE scheme itself is effective, and whether multiview learning will outperform heuristic combination of dissimilarity matrix with REE. Kernel size σ for C-MV-REE is set as 1 (median values for both $\boldsymbol{\Delta}_{\text{CNN}}$ and $\boldsymbol{\Delta}_{\text{hand}}$ are around 0.55). It turned out that both REE and the C-MV learning are beneficial for improving classification accuracy.

Observing from both Tables II and III, it is clear that using both CNN and hand-designed features results in better classification performance than using one type of feature only. Another interesting discovery is that CNN features are not as good as hand-designed feature here, possibly because the generalization ability of a pretrained CNN is limited on a small data set, which may differ significantly from the data CNN is trained from.

2) Classification of Lidar Images: Lidar imagery of fiberglass replicas of three different species of marine animals (amberjack, barracuda, and turtle) have been retrieved from the test tank at HBOI. The replicas are mounted on six-degree freedom linear drive such that different poses can be generated. Lidar return of any object first undergoes noise reduction and volume backscattering gating. It is then integrated over time, which generates a 2-D grayscale image. The contour of the object is then obtained through GrabCut segmentation [48]; see Fig. 10. The dissimilarity measure between any pair of objects will be determined by shape analysis in the same manner as in Section IV-A3. Details of Lidar image retrieval, segmentation, and preprocessing are described in [7].

There are 38 testing images in total, 22 of which are obtained under clear water condition. The rest are collected under turbid water environment, so it is expected that these 16 objects are heavily blurred, some not even discernible by human standards. The images used as templates (labels are known) are 2-D projections from different perspectives of 3-D models of the three animal species. These 2-D images are then processed through a radiative transfer model [7]. For each animal species, ten 2-D images of different orientations are generated. The two shape analysis methods used are SC and IDSC. Each method will provide a dissimilarity matrix of size $N{*}N$ $(N = 10 * 3 + 38 = 68)$.

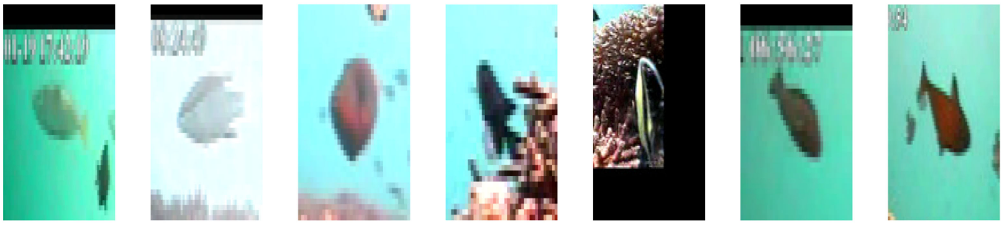


Fig. 9. Examples of color images of fish.

TABLE II
TAIWAN SEA FISH CLASSIFICATION ACCURACY (FEATURE BASED)

train : test	1:4	2:3	3:2	4:1
CNN only	61.9±11.3%	76.6±7.1%	82.5±8.7%	85.8±12.5%
(dimension reduced)	66.6±8.9%	77.8±8.0%	80.9±7.1%	81.5±17.3%
hand only	74.9±8.5%	88.1±6.8%	90.6±6.8%	92.8±7.3%
(dimension reduced)	78.2±10.3%	84.4±8.1%	87.3±9.1%	89.9±11.5%
concatenation	80.7±8.7%	90.8±5.5%	93.6±4.8%	94.2±7.6%
(no dim. reduction)	66.5±7.7%	77.0±8.4%	81.1±8.3%	85.9±9.6%
concatenation	81.6±10.3%	91.4±6.8%	**94.2±5.7%**	**95.5±7.3%**
(dimension reduced)	75.6±12.7%	84.4±9.7%	88.9±7.7%	90.9±9.7%
C-MV	82.1±6.3%	90.9±6.1%	93.7±5.8%	94.8±7.3%
	82.4±9.2%	**90.7±6.5%**	**93.4±7.0%**	**95.4±9.2%**
Ce-MV	**86.9±7.3%**	**91.9±6.6%**	94.0±5.8%	**95.5±7.3%**
	82.6±7.3%	88.5±8.8%	92.5±5.2%	94.9±7.3%

For each method, the first row shows SVM results and the second row shows kNN ($k = 1$) results. C-MV and Ce-MV have the best overall performance.

TABLE III
TAIWAN SEA FISH CLASSIFICATION ACCURACY (DISSIMILARITY MATRIX BASED)

train : test	1:4	2:3	3:2	4:1
CNN only	47.1±15.4%	69.7±8.0%	78.3±6.4%	83.4±9.9%
(with REE)	56.9±4.0%	68.5±6.9%	72.7±8.4%	75.4±13.8%
hand only	66.5±9.8%	83.1±6.8%	86.5±6.8%	90.9±8.1%
(with REE)	83.1±6.8%	87.0±8.1%	89.7±7.4%	92.1±8.7%
Hadamard prod.	60.3±11.6%	81.6±9.1%	87.7±6.4%	90.7±7.5%
(no REE)	82.1±8.0%	89.0±6.8%	92.1±5.9%	94.0±7.4%
Hadamard prod.	66.5±9.5%	84.6±7.8%	90.1±4.9%	93.5±5.0%
(with REE)	**84.5±7.8%**	**90.0±6.2%**	92.5±6.2%	**94.8±7.6%**
MV-REE	67.8±10.1%	84.6±8.5%	88.6±6.5%	90.4±7.7%
	79.6±9.5%	87.8±8.2%	91.4±7.8%	94.2±7.9%
C-MV-REE	**70.9±11.7%**	**86.7±6.0%**	**92.7±4.9%**	**95.8±5.4%**
	84.2±7.2%	89.5±7.1%	**92.6±6.3%**	**94.8±7.6%**

C-MV-REE has the best overall performance.

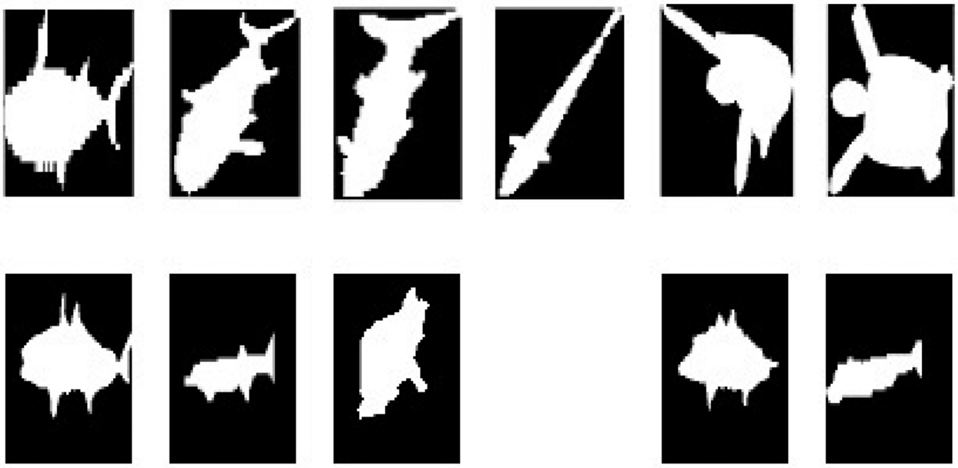


Fig. 10. First row: examples of template images. Second row (left): segmented contours of clear water Lidar images; right: contours of turbid water Lidar images.

TABLE IV
CONFUSION MATRICES FOR HBOI TEST TANK DATA

		I			II		
		A	B	T	A	B	T
SC only (with REE)	A	6.5	8.0	1.5	3	4.5	0.5
	B	1	8	0	1	5	0
	T	2	5.5	5.5	1	3.5	3.5
IDSC only (with REE)	A	12.5	0	3.5	7.5	0	0.5
	B	0	9	0	0	6	0
	T	3	0.5	9.5	2	0.5	5.5
Hadamard prod. no REE	A	12.5	0.5	3	6	0.5	1.5
	B	6	3	0	3.5	2.5	0
	T	7	0	6	4	0	4
Hadamard prod. (with REE)	A	11.5	0	4.5	6.5	0	1.5
	B	1.5	7.5	0	0.5	5.5	0
	T	2	0	11	1	0	7
MV-REE	A	13	0	3	7.5	0	0.5
	B	1	8	0	0.5	5.5	0
	T	2	0	11	1	0	7
C-MV-REE	A	13	0	3	7	0	1
	B	1	8	0	0	6	0
	T	1.5	0	11.5	0.5	0	7.5

All results are averaged from the results of SVM and 1NN classifiers. Column I shows the result for all 38 images, while column II is the result for the 22 clear water images. A, B, and T stand for amberjack, barracuda, and turtle, respectively.

Like in the previous experiment, six dissimilarity matrix based approaches will be tested. The median value for both $\boldsymbol{\Delta}_{\text{SC}}$, $\boldsymbol{\Delta}_{\text{IDSC}}$ are 3.45 so σ is selected as 2 for C-MV-REE. The number of principal components used k is 8. Classification results are average values from 1-NN and SVM classifiers. They are summarized in Table IV. It is evident that the individual method of IDSC performs considerably better than SC in this problem. Using REE on the heuristically combined similarity matrix does not further improve the result. However, the C-MV-REE algorithm still successfully achieves higher classification rate than any of the other methods.

V. CONCLUSION

The correntropy-loss-based multi-view learning algorithms are developed for both cases of features (C-MV and Ce-MV) and dissimilarity matrices (C-MV-REE). In the presence of artificial noise, methods based on correntropy loss are much more robust than their counterparts that apply L_1 or L_2 loss as cost functions. Multiview learning also performs better than using single view only or concatenated feature/heuristically combined dissimilarity matrix.

The developed algorithms are successfully applied to two real-world marine animal classification problems. The significance of the developed multiview learning algorithms for marine animal classification is not only about improving classification accuracy; it provides a versatile framework for combining different marine animal data representations, i.e., feature/feature combination, dissimilarity matrix/dissimilarity matrix combination, and feature/dissimilarity matrix combination. The current Lidar data are described by dissimilarity matrices derived from the 2-D shape analysis, but new properties of Lidar data may be utilized in the forthcoming real-world experiments, such as fish texture that appears in the form of features. With the developed multiview learning algorithms, any new data description can be utilized along with the existing ones.

For future work, since the optimization techniques for both C-MV and C-MV-REE can accommodate GC-loss, it might be interesting to probe into the possibility of employing GC-loss in multiview learning.

REFERENCES

[1] P. X. Huang, B. J. Boom, and R. B. Fisher, "Underwater live fish recognition using a balance-guaranteed optimized tree," in *Proc. Asian Conf. Comput. Vis.*, 2013, pp. 422–433.

[2] Q. Hu and C. Davis, “Accurate automatic quantification of taxa-specific plankton abundance using dual classification with correction,” *Mar. Ecol. Prog. Ser.*, vol. 306, pp. 51–61, 2006.

[3] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, “A large-scale hierarchical image database,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[4] Z. Cao, J. C. Principe, B. Ouyang, F. Dalgleish, and A. Vuorenkoski, “Marine animal classification using combined CNN and hand-designed image features,” in *Proc. MTS/IEEE OCEANS Conf.*, 2015, pp. 1–6.

[5] S. Dieleman *et al.*, “Classifying plankton with deep neural networks,” 2015. [Online]. Available: http://benanne.github.io/2015/03/17/plankton.html

[6] B. Ouyang, F. Dalgleish, A. Vuorenkoski, W. Britton, B. Ramos, and B. Metzger, “Visualization and image enhancement for multistatic underwater laser line scan system using image-based rendering,” *IEEE J. Ocean. Eng.*, vol. 38, no. 3, pp. 566–580, Jul. 2013.

[7] Z. Cao *et al.*, “Marine animal classification using UMSLI in HBOI optical test facility,” *Multimedia Tools Appl.*, vol. 76, no. 21, pp. 23117–23138, Nov. 2017.

[8] K. Matsuda, S. Torisawa, T. Hiraishi, and K. Yamamoto, “Comparison of the color vision and spectral sensitivity of three flatfish species of different ecotypes, and application to selective fishing methods,” *Fisheries Sci.*, vol. 75, no. 1, pp. 35–42, 2009.

[9] F. R. Dalgleish, A. K. Vuorenkoski, and B. Ouyang, “Extended-range undersea laser imaging: Current research status and a glimpse at future technologies,” *Mar. Technol. Soc. J.*, vol. 47, no. 5, pp. 128–147, 2013.

[10] B. Ouyang, “Watermarking based on unified pattern recognition framework,” Ph.D. dissertation, Dept. Electr. Eng., Southern Methodist Univ., Dallas, TX, USA, 2007.

[11] X. Bai, C. Rao, and X. Wang, “Shape vocabulary: A robust and efficient shape representation for shape matching,” *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3935–3949, Sep. 2014.

[12] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 24, pp. 509–522, Apr. 2002.

[13] H. Ling and D. W. Jacobs, “Shape classification using the inner distance,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 2, pp. 286–299, Feb. 2007.

[14] N. Alajlan, M. S. Kamel, and G. H. Freeman, “Geometry-based image retrieval in binary image databases,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 1003–1013, Jun. 2008.

[15] J. Wang, X. Bai, X. You, W. Liu, and L. J. Latecki, “Shape matching and classification using height functions,” *Pattern Recognit. Lett.*, vol. 33, no. 2, pp. 134–143, 2012.

[16] H. Ling and K. Okada, “An efficient earth mover’s distance algorithm for robust histogram comparison,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 840–853, May 2007.

[17] K. Kampa, E. Hasanbelliu, and J. C. Principe, “Closed-form Cauchy–Schwarz pdf divergence for mixture of Gaussians,” in *Proc. Int. Joint Conf. Neural Netw.*, 2011, pp. 2578–2585.

[18] M. Liu, B. C. Vemuri, S.-I. Amari, and F. Nielsen, “Shape retrieval using hierarchical total Bregman soft clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2407–2419, Dec. 2012.

[19] A. Sadeghian, D. Lim, J. Karlsson, and J. Li, “Automatic target recognition using discrimination based on optimal transport,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 2604–2608.

[20] C. Xu, D. Tao, and C. Xu, “A survey on multi-view learning,” 2013, arXiv:1304.5634.

[21] S. Lacoste-Julien, F. Sha, and M. I. Jordan, “DiscLDA: Discriminative learning for dimensionality reduction and classification,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 897–904.

[22] W. Wang and M. A. Carreira-Perpinán, “The role of dimensionality reduction in classification,” in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 2128–2134.

[23] J. Donahue *et al.*, “DeCAF: A deep convolutional activation feature for generic visual recognition,” in *Proc. 31st Int. Conf. Mach. Learn.*, 2014, pp. 647–655.

[24] Z. Cao, J. C. Principe, and B. Ouyang, “Group feature selection in image classification with multiple kernel learning,” in *Proc. Int. Joint Conf. Neural Netw.*, 2015, pp. 1–5.

[25] A. Blum and T. Mitchell, “Combining labeled and unlabeled data with co-training,” in *Proc. 11th Annu. Conf. Comput. Learn. Theory*, 1998, pp. 92–100.

[26] M. Gönen and E. Alpaydın, “Multiple kernel learning algorithms,” *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, 2011.

[27] T. Xia, D. Tao, T. Mei, and Y. Zhang, “Multiview spectral embedding,” *IEEE Trans. Syst., Man, Cybern B*, vol. 40, no. 6, pp. 1438–1446, Dec. 2010.

[28] W. Ou, S. Yu, G. Li, J. Lu, K. Zhang, and G. Xie, “Multi-view nonnegative matrix factorization by patch alignment framework with view consistency,” *Neurocomputing*, vol. 204, pp. 116–124, 2016.

[29] C. Xu, D. Tao, and C. Xu, “Multi-view intact space learning,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 12, pp. 2531–2544, Dec. 2015.

[30] X. Bai, B. Wang, X. Wang, W. Liu, and Z. Tu, “Co-transduction for shape retrieval,” in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 328–341.

[31] Y. Zhao *et al.*, “Multi-view manifold learning with locality alignment,” *Pattern Recognit.*, vol. 78, pp. 154–166, 2018.

[32] Q. Wang and K. L. Boyer, “Feature learning by multidimensional scaling and its applications in object recognition,” in *Proc. 26th IEEE Conf. Graph., Patterns Images*, 2013, pp. 8–15.

[33] L. Cayton and S. Dasgupta, “Robust Euclidean embedding,” in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 169–176.

[34] S. Bai, X. Bai, L. J. Latecki, and Q. Tian, “Multidimensional scaling on multiple input distance matrices,” *Comput. Vis. Pattern Recognit.*, 2016. [Online]. Available: https://arxiv.org/abs/1605.00286

[35] W. Liu, P. P. Pokharel, and J. C. Principe, “Correntropy: Properties and applications in non-Gaussian signal processing,” *IEEE Trans. Signal Process.*, vol. 55, no. 11, pp. 5286–5298, Nov. 2007.

[36] S. Zhao, B. Chen, and J. C. Principe, “Kernel adaptive filtering with maximum correntropy criterion,” in *Proc. Int Joint Conf. Neural Netw.*, 2011, pp. 2012–2017.

[37] R. Pokharel and J. C. Príncipe, “Kernel classifier with correntropy loss,” in *Proc. Int. Joint Conf. Neural Netw.*, 2012, pp. 1–6.

[38] R. He, W. Zheng, and B. Hu, “Maximum correntropy criterion for robust face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1561–1576, Aug. 2011.

[39] W. Ou, G. Li, S. Yu, G. Xie, F. Ren, and Y. Tang, “Robust discriminative nonnegative patch alignment for occluded face recognition,” in *Proc. Int. Conf. Neural Inf. Process.*, 2015, pp. 207–215.

[40] Y. Qi, Y. Wang, X. Zheng, and Z. Wu, “Robust feature learning by stacked autoencoder with maximum correntropy criterion,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process*, 2014, pp. 6716–6720.

[41] B. Chen, L. Xing, H. Zhao, N. Zheng, and J. C. Príncipe, “Generalized correntropy for robust adaptive filtering,” *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3376–3387, 2016.

[42] F. Wickelmaier, “An introduction to MDS,” *Sound Quality Res. Unit, Aalborg Univ., Aalborg, Denmark*, vol. 46, 2003.

[43] M. N. Syed, P. M. Pardalos, and J. C. Principe, “On the optimization properties of the correntropic loss function in data analysis,” *Optim. Lett.*, vol. 8, no. 3, pp. 823–839, 2014.

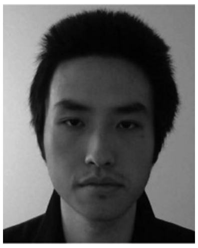
[44] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[45] T. B. Sebastian, P. N. Klein, and B. B. Kimia, “Recognition of shapes by editing their shock graphs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 550–571, May 2004.

[46] Z. Cao, J. Principe, and B. Ouyang, “Information point set registration for shape recognition,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 2603–2607.

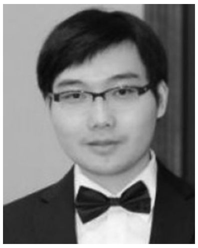
[47] C. Spampinato, D. Giordano, R. D. Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, “Automatic fish classification for underwater species behavior understanding,” in *Proc. 1st ACM Int. Workshop Anal. Retrieval Tracked Events Motion Imagery Streams*, 2010, pp. 45–50.

[48] C. Rother, V. Kolmogorov, and A. Blake, “GrabCut: Interactive foreground extraction using iterated graph cuts,” *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.



Zheng Cao received the B.S. degree in telecommunications engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2010, the M.S. degree in electrical engineering from the University of Wyoming, Laramie, WY, USA, in 2012, and the Ph.D. degree in electrical engineering from the University of Florida, Gainesville, FL, USA, in 2017.

He had previously worked on system identification and kernel adaptive filtering. His current research interests include machine learning and computer vision, with applications in marine animal detection and classification.



Shujian Yu received the B.S. (honors) degree in electrical engineering from the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China, in 2013. He is currently working toward the Ph.D. degree in electrical engineering at the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA.

His research interests include machine learning for signal processing, image analysis, and understanding.

Bing Ouyang (S'02–M'06) received the Ph.D. degree in electrical engineering from Southern Methodist University, Dallas, TX, USA, in 2007.

He was with the Ocean Visibility and Optics Laboratory, Harbor Branch Oceanographic Institute (HBOI), Florida Atlantic University, Fort Pierce, FL, USA, in 2009. Before joining HBOI, he was with Texas Instruments (TI), Inc., Dallas, TX, USA. From 2003 to 2009, he was an Algorithm Engineer with the DLP ASIC algorithm team, with the primary focus in developing front-end algorithms for the video processing ASIC. He holds four U.S. patents in the area of analog video and graphics format detection. His current research interests include underwater computer vision, novel underwater electrooptical system design, underwater LIDAR imaging enhancement, pattern recognition, and analysis for sensor time series data.

Dr. Ouyang is a recipient of the 2013 Young Investigator Research Program Award. He was peer elected to the member of Technical Staff while working at TI.

Fraser Dalgleish received the Ph.D. degree in ocean engineering from Cranfield University, Cranfield, U.K., in 2004.

He directs the Ocean Visibility and Optics Laboratory, Harbor Branch Oceanographic Institute, Florida Atlantic University, Fort Pierce, FL, USA. His research interest focuses on undersea optical sensor development, both for remote and *in situ* environmental measurements and to improve sensing and communications capabilities as an enabling technology for multivehicle imaging and sensing operations. Projects within academia involve collaboration with scientists in sensor package development and ocean observatories for water quality monitoring and imaging missions. His recent development activities in collaboration with government and industry have focused on new laser instrumentation, simulation tools, and detection approaches for ocean sensing and monitoring applications.

Anni Vuorenkoski received the Ph.D. degree in mechanical engineering from Cranfield University, Cranfield, U.K., in 2004.

She is with the Ocean Visibility and Optics Laboratory, Harbor Branch Oceanographic Institute, Florida Atlantic University, Fort Pierce, FL, USA. Her past research activities have included adoption of optical, laser-based techniques to study the properties of turbid media and flows. She has also been involved in the development and application of Monte–Carlo-based computational methods to simulate the effects of multiple scattering in aerosol laser imaging. Her current research interests include the characterization of water column and benthic features by polarimetry, fluorometry, angularly resolved scattering techniques, and time-resolved short-pulse methods, as well as on the experimental validation of computational radiative transfer models.

Gabriel Alsenas received the B.S. and M.S. degrees in ocean and systems engineering from Florida Atlantic University (FAU), Fort Pierce, FL, USA, in 2005 and 2007, respectively.

He has been the Program Manager at FAU Southeast National Marine Renewable Energy Center for seven years.

Jose C. Principe (F'00) is a Distinguished Professor of electrical and computer engineering and biomedical engineering with the University of Florida, Gainesville, FL, USA. He is also the Bell South Professor and the Founding Director of Computational NeuroEngineering Laboratory, University of Florida. He has authored and coauthored more than 600 publications and 30 patents. His primary research interests include advanced signal processing with information theoretic criteria (entropy and mutual information), adaptive models in the reproducing kernel Hilbert spaces, and the application of these advanced algorithms in brain–machine interfaces.

Dr. Principe is a Fellow of the ABME and AIBME. He is the Past Editor-in-chief for the IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, the Past Chair for the Technical Committee on Neural Networks of the IEEE Signal Processing Society, and the Past President for the International Neural Network Society. He was the recipient of the IEEE EMBS Career Award and the IEEE Neural Network Pioneer Award.