Received 16 May 2023, accepted 1 June 2023, date of publication 5 June 2023, date of current version 22 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3282932

RESEARCH ARTICLE

Any-Shot Learning From Multimodal Observations (ALMO)

MEHMET AKTUKMAK[1], **YASIN YILMAZ**[2], **(Senior Member, IEEE), AND ALFRED O. HERO III**[1], **(Life Fellow, IEEE)**

[1]Electrical and Computer Engineering Department, University of Michigan, Ann Arbor, MI 48103, USA
[2]Electrical Engineering Department, University of South Florida, Tampa, FL 33620, USA

Corresponding author: Mehmet Aktukmak (maktukmak@usf.edu)

This work was supported in part by the Army Research Office (ARO) under Grant W911NF-19-1-0269 and Grant W911NF-15-1-0479, and in part by the Department of Energy (DOE) under Grant DE-NA0003921.

ABSTRACT In this paper, we propose a framework (ALMO) for any-shot learning from multi-modal observations. Using training data containing both objects (inputs) and class attributes (side information) from multiple modalities, ALMO embeds the high-dimensional data into a common stochastic latent space using modality-specific encoders. Subsequently, a non-parametric classifier is trained to predict the class labels of the objects. We perform probabilistic data fusion to combine the modalities in the stochastic latent space and learn class conditional distributions for improved generalization and scalability. We formulate ALMO for both few-shot and zero-shot classification tasks, demonstrating significant improvement in recognition performance on the Omniglot and CUB-200 datasets as compared to state-of-the-art baselines.

INDEX TERMS Few-shot learning, zero-shot learning, multi-modal machine learning, probabilistic modeling.

I. INTRODUCTION

Humans can recognize objects that belong to the classes they have heard about but have never seen before. However, AI systems still have difficulty generalizing from training data for which some classes have few or no samples. This has motivated the very active research area called any-shot learning, which aims to imitate this aspect of the human learning process [1], [2], [3]. Most any-shot learning algorithms in the literature rely on a single data modality view of an object (e.g., an image) to learn a predictor. The multi-modality any-shot learning problem, where multiple types of observations of the same object are available (e.g., image and text), is largely unexplored. Diverse modalities can provide additional valuable and complementary information, yielding improved prediction performance. The primary motivation behind this paper is to fill this gap, providing an any-shot learning framework that exploits the dependency between multi-modal observations with improved generalization and scalability.

As described in [2], in the any-shot classification problem, class attribute side information is available to supplement the labeled object samples in the training data. This allows generalizable learning of a classifier from limited training data when some classes have very few samples (few-shot) or no samples at all (zero-shot). There have been many approaches to any-shot classification but the so-called prototype learning methods [4] have been among the most popular. Two disjoint sets called support and query sets are formed from the object samples and class attributes for each episode in advance. Prototype methods use metric learning [5], [6] to embed the support set into a low-dimensional semantic space. The nearest neighbor classifier is then used to classify the samples in the query set. Specifically, a representation for each class is learned given its support samples in the form of a prototype. Then, the class probability of a query sample is approximated by using the distance of the query sample representation from each prototype in the soft-max function. The objective is to maximize the class posterior of the query sample labels given the support samples and the network parameters. However, computing prototypes from few samples is prone to over-fitting [7].

The associate editor coordinating the review of this manuscript and approving it for publication was Ikramullah Lali.

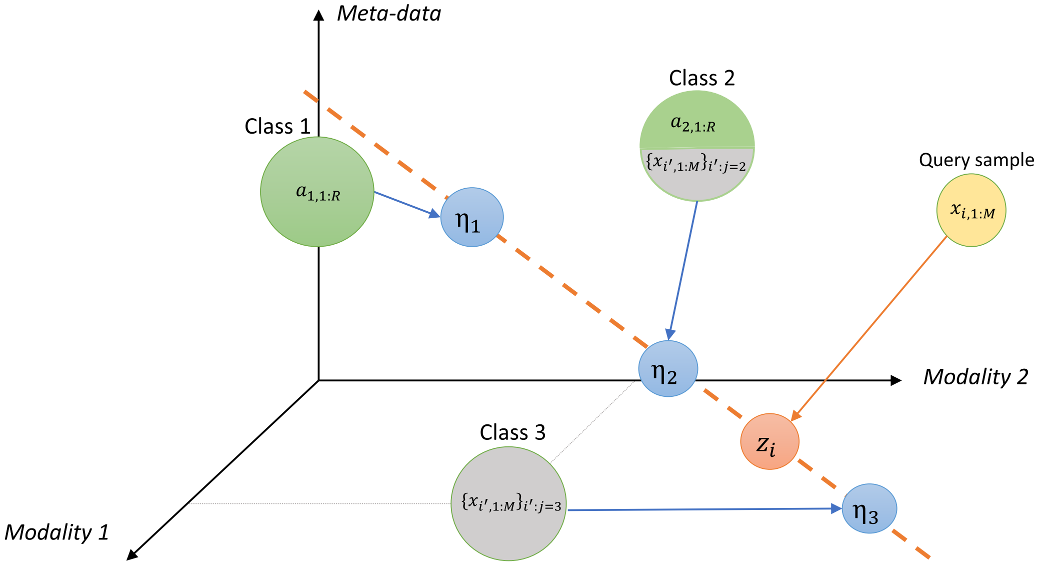


FIGURE 1. Semantic illustration of the proposed ALMO framework when there are $M = 2$ modalities of object observations and class meta-data, $C = 3$ classes, and the dimensionality of the latent space is $L = 1$. Data in the support set for class 2 has multi-modal (object, class attribute) data pairs $\left(\{x_{i',m=1:M}, y_{i'}\}_{i':y_{i'}=2}, a_{2,r=1:R}\right)$, while class 1 data is missing objects and class 3 data is missing class attributes. The support set data is projected to a stochastic latent space (orange dotted line) on a per-class basis to learn a latent representation. Given this representation, a query sample is classified according to a distance computed in the stochastic latent space. In the zero-shot classification setting, the objects are missing in the support set, e.g., the class 1 support set $\{x_{i',m=1:M}, y_{i'}\}_{i:y_{i'}=1}$ is empty. Whereas in the few-shot classification setting, support data have very few samples, and may also be missing class attributes, e.g., the class 3 attribute set $a_{3,r=1:R}$ is empty.

To alleviate over-fitting, we use low dimensional latent variable representations in a Bayesian framework that benefits from uncertainty quantification. Our latent variable models induce a probability distribution over the representation space that allows us to integrate multiple modalities into the any-shot learning framework using probabilistic data fusion. This results in a method we call any-shot learning for multimodality observations (ALMO), which offers a unified stochastic latent representation of prototypes and query samples. Specifically, we first project the information from all the samples in the support set, including multi-modal object and class attribute observations to a lower-dimensional latent space using neural network encoders. Since there is more than one information source for the representations, we adopt a probabilistic data fusion framework to combine the multi-modal projections. Then, the (multi-modal) observations of each query sample are projected to the same latent space using modality-specific encoders. Subsequently, non-parametric classification is performed in this space. We derive analytical expressions for the classification loss and class conditional distributions given the support classes and query objects by integrating out the associated latent variables. An illustration of the proposed model is given in Figure 1.

The proposed ALMO framework has two main innovations relative to previous methods: it enables fusing multi-modal class and object information for few-shot and zero-shot learning with a scalable fusion mechanism; and it does this in a stochastic latent space under a hierarchical Bayesian model, which provides automatic regularization for improving generalization performance. We formulate ALMO for both few-shot and zero-shot classification tasks (Section III) and conduct experiments on the Omniglot and CUB-200 datasets to demonstrate the effectiveness of the proposed model (Section IV).

II. RELATED WORK

Early data fusion algorithms for high-dimensional multi-modal data rely on linear latent variable models. Supervised PCA [8], [9], PLS [10], CCA [11], and MMFA [12] allow fusing multi-modal data by assuming shared low-dimensional latent variables.

Recent multi-modal data fusion algorithms exploit neural networks. In [13] and [14] the authors use a single latent variable per object to explain the object modalities. Optimization of the network weights is performed by maximizing evidence lower bound (ELBO). To push the model to be more discriminative, the likelihood of the response variable is augmented. The authors adopt the product of the experts framework to combine the predictions from each modality. In contrast, in [15] and [16] the authors use a mixture of experts to combine the encoder predictions. However, this method requires the ELBO to be computed through the evaluation of M^2 terms of the conditional likelihood function, which may be computationally prohibitive when the number of modalities M is large. In [17] the latent variables are separated into two categories: joint and individual. Joint latent variables are shared among the input modalities whereas individual latent variables are associated only with their corresponding modalities. Hence, the objective function includes both discriminative terms associated with the response variable through the joint latent variables and the generative likelihoods of the input modalities through individual latent variables. However, the authors adopt late fusion [18] to combine multiple predictions of the encoders, which, unlike early fusion, ignores relevant fine-scale information of the data. Our proposed model, on the other hand, performs probabilistic data fusion, which results in better generalization, as we show in Section IV. Unlike the aforementioned algorithms, the objective of the ALMO is metric learning (Section III) as contrasted to maximizing label or modality likelihoods, which makes it more suitable for few-shot and zero-shot tasks.

Learning class representations in the form of prototypes [4], [19], [20] is a contemporary technique for few-shot and deep metric learning [6], [21]. In [4] it is proposed to learn the class prototypes within the support set, followed by nearest neighbor classification of the query samples using Euclidean distance measured in the embedding space. The objective in [4] is to minimize the classification error of the query samples. The authors of [20] additionally construct a linear mapping for each task to achieve task adaptation from prototypes. In [22] a soft-max attention function is used to perform the query samples classification. In contrast to these methods, our model learns the class and object representations from multiple modalities using latent variables having attributed uncertainty, which allows us to apply probabilistic data fusion, reducing over-fitting. Similarly, the authors of [19] learn the parameters of the Gaussian priors associated with each class from the corresponding support set samples. Their objective is to maximize the likelihood of the support set samples given the class labels. Hence, they propose a generative model that separates modeling class conditional densities and performs the classification after training, which is a nonlinear extension of Discriminant Analysis. In contrast, our model uses a discriminative objective function and episodic training, which is more naturally adapted to few-shot and zero-shot classification tasks. Recently, in [23] a multi-modal few-shot learning algorithm was proposed, in which the prototypes are informed from multiple views of the objects. A fusion method is implemented by the weighted average over a deterministic latent space with fixed weights. In contrast, our method performs adaptive weighting induced by the sample covariance matrices (See Equation 8) in the stochastic latent space. Notably, in [24] a multi-modal approach is proposed where one modality is the image itself and the other modality is the Discrete Cosine Transform of the corresponding image. Their fusion scheme is deterministic, i.e., a simple concatenation in the latent space.

In the context of few-shot learning, the authors of [25] and [26] propose a method based on optimizing transforms. The features obtained from the feature extractor are first processed using a power transform and then the maximum a posteriori (MAP) algorithm is used to estimate class centers by including the labeled and unlabeled data in the query set. Hence, this method is only applicable in a transductive setting where the training and test samples are fitted simultaneously. Another self-optimal transform method is proposed in [27], where the cosine similarity matrix is computed from all training instances. Then, a transport matrix is multiplied with this matrix to transform all instances into another latent space. However, this method may suffer from scalability due to high $O(n^3)$ memory requirements and high computational complexity due to the use of matrix multiply operations. As another transductive method, [28] extends prototypical networks to a generalized FSL setting where novel class training samples are used during training. Data augmentation is adopted for the limited number of samples in novel under-represented classes.

For meta-learning tasks, many algorithms based on class representation learning have been proposed to further improve few-shot/zero-shot learning performance by focusing on better learning of image representations. The authors of [29] incorporated PixelCNN [30] when learning class priors. In [31], [32], and [33] attention mechanisms are used to jointy perform attribute localization and fine-grained feature learning. More complicated feature extraction methods developed for different learning tasks can also be incorporated, e.g., federated learning [34], [35]. Our proposed framework can be used to learn better image representations by modifying the encoders to map the images directly to distribution estimates of the object-level latent variables. In this paper, the focus is on improving data fusion and reducing generalization errors.

In summary, this paper fills a gap by providing a computationally tractable methodology to solve multi-modal data fusion in few-shot and zero-shot learning settings that are capable of significantly reducing generalization error. In Section III, we define the model architecture and formulate its optimization algorithms.

III. PROPOSED MODEL

A. NOTATION

Define the initial object dataset $\{\boldsymbol{x}_{i,m=1:M}, y_i\}_{i=1}^{I}$, where $\boldsymbol{x}_{i,m=1:M} = \{\boldsymbol{x}_{i,m}\}_{m=1}^{M}$is the i-th observed object $\boldsymbol{x}_{i,m} \in \mathbb{R}^{D_m}$ in the m-th modality, $y_i \in \{1, \ldots, C\}$ is the class label of object i. Here C is the number of classes, M is the number of input modalities, I is the number of objects, and D_m is the dimension of the observation space for the mth modality. To form a training/testing episode, N classes are randomly selected, then from each class, K objects are randomly selected without replacement to form the support set, and another K object to form the query set. The indices of the eth support set and the eth query set are stored in S_e and Q_e, respectively. Furthermore, let $S_{e,j} = \{(i' : y_{i'} = j) \in S_e\}$ and $Q_{e,j} = \{(i : y_i = j) \in Q_e\}$ denote the subsets that only contain the sample indexes with label j. The indices of the sampled class labels are stored in Ω_e. Note that each episode contains only N distinct classes and $2 \times N \times K$ objects.

Additionally, for class $y \in \{1, \ldots, C\}$ define the meta-data $\boldsymbol{a}_{y,r} \in \mathcal{R}^{D_r}$ for the r-th modality, $r \in \{1, \ldots, R\}$ and D_r is the dimension of the metadata for modality r. The set of multi-modal meta-data for class y is denoted $\boldsymbol{a}_{y,r=1:R} = \{\boldsymbol{a}_{y,r}\}_{i=1}^{R}$. Likewise the meta-data used in the episode e is denoted as $\{\boldsymbol{a}_{j,r=1:R}\}_{j\in\Omega_e}$.

B. MODEL DEFINITION

We propose a discriminative latent variable model that can perform any-shot learning from multi-modal observations. To this end, specific latent variables are assigned for each prototype and each query sample. The latent variables share the same latent space with dimension L. Let $\boldsymbol{\eta}_j \in \mathbb{R}^L$ denote the prototype latent variable for class j and $z_i \in \mathbb{R}^L$ denote the object latent variable for query sample i. We assume the conditional distribution of z_i given the query sample is a multivariate normal distribution with parameters $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$, i.e., $p(z_i|\boldsymbol{x}_{i,m=1:M}) = \mathcal{N}(z_i|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$. Similarly, the conditional distribution of prototype $\boldsymbol{\eta}_j$ given a support set is $p(\boldsymbol{\eta}_j|\{\boldsymbol{x}_{i',m=1:M}\}_{i'\in S_{e,j}}) = \mathcal{N}(\boldsymbol{\eta}_j|\boldsymbol{m}_j, \boldsymbol{S}_j)$. The free parameters of the distributions $\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \boldsymbol{m}_j, \boldsymbol{S}_j$ are parameterized by neural networks. Since the label is discrete, we use categorical distribution to model the observed labels, i.e., $p(y_i|z_i, \{\boldsymbol{\eta}_j\}_{j\in\Omega_e}) = Cat(y_i|\boldsymbol{\nu}_i)$, where the vector $\boldsymbol{\nu}_i$, that lies in a N dimensional probability simplex, is a non-random non-linear transformation of z_i and $\{\boldsymbol{\eta}_j\}_{j\in\Omega_e}$. The vector $\boldsymbol{\nu}_i$ is given as follows:

$$\boldsymbol{\nu}_i = \Bigg[\frac{\exp(-D(z_i, \boldsymbol{\eta}_{\Omega_{e,1}}))}{\sum_{j'\in\Omega_e}\exp(-D(z_i, \boldsymbol{\eta}_{j'}))}, \ldots, \frac{\exp(-D(z_i, \boldsymbol{\eta}_{\Omega_{e,N}}))}{\sum_{j'\in\Omega_e}\exp(-D(z_i, \boldsymbol{\eta}_{j'}))}\Bigg], \quad (1)$$

where $D(z_i, \boldsymbol{\eta}_j)$ is a distance measure between the latent variables z_i and $\boldsymbol{\eta}_j$. Given the conditional distributions, the marginal likelihood of the class label of a query object is given as

$$\begin{aligned} p(y_i|\boldsymbol{x}_{i,m=1:M}, S_e) &= \int\int\cdots\int p(y_i|z_i, \{\boldsymbol{\eta}_j\}_{j\in\Omega_e}) \\ &\quad \times p(z_i|\boldsymbol{x}_{i,m=1:M})\Big[\prod_{j\in\Omega_e} p(\boldsymbol{\eta}_j|\{\boldsymbol{x}_{i',m=1:M}\}_{i'\in S_{e,j}})\Big] \\ &\quad \times d\boldsymbol{\eta}_{\Omega_{e,1}}\ldots d\boldsymbol{\eta}_{\Omega_{e,N}} dz_i. \end{aligned} \quad (2)$$

The marginal likelihood involves integrating out normally distributed random variables over a categorical likelihood, hence the expression above is intractable to compute. We first replace the integration with the expectation operator, then compute the log marginal likelihood of the labels for the query samples in an episode:

$$\begin{aligned} \ell_e &= \sum_{i\in Q_e} \log p(y_i|\boldsymbol{x}_{i,m=1:M}, S_e) \\ &\geq \sum_{i\in Q_e} \mathbb{E}[\log p(y_i|z_i, \{\boldsymbol{\eta}_j\}_{j\in\Omega_e})], \end{aligned} \quad (3)$$

where the last line follows from Jensen's inequality, and the expectation is taken with respect to the distributions of $\{\boldsymbol{\eta}_j\}_{j\in\Omega_e}$ and z_i. From Equation 1, the explicit expression for the expected log term is given as follows:

$$\begin{aligned} &\mathbb{E}[\log p(y_i = j|z_i, \{\boldsymbol{\eta}_j\}_{j\in\Omega_e})] \\ &\quad = \mathbb{E}[-D(z_i, \boldsymbol{\eta}_j) - \log \sum_{j'\in\Omega_e} \exp(-D(z_i, \boldsymbol{\eta}_{j'}))] \end{aligned} \quad (4)$$

The first term depends linearly on the distance. However, the second term involves non-linear transformation, specifically the log-sum-exp function. The complexity of this expectation depends on the distance metric. Given the distance metric is squared Euclidean, i.e., $D(z_i, \boldsymbol{\eta}_j) = ||z_i - \boldsymbol{\eta}_j||_2^2$, and the co-variance matrices $\boldsymbol{\Sigma}_i = \text{diag}([\sigma_{i1}, \ldots, \sigma_{iL}])$ and $\boldsymbol{S}_j = \text{diag}([s_{i1}, \ldots, s_{iL}])$ are diagonal, the first term of the expected log expression can be analytically computed as follows:

$$\mathbb{E}[||z_i - \boldsymbol{\eta}_j||_2^2] = \sum_{l=1}^{L}(\mu_{il} - m_{jl})^2 + \sigma_{il}^2 + s_{jl}^2, \quad (5)$$

where l indexes the components of the latent variables. However, the second term, which involves the log-sum-exp function, does not have an analytical solution. Hence, we derive an upper bound for the expected log-sum-exp function.

Proposition 1: Given the distance measure between independent latent variables $\boldsymbol{\eta}_j \sim \mathcal{N}(\boldsymbol{m}_j, \boldsymbol{S}_j)$ and $z_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is squared Euclidean, and the co-variance matrices $\boldsymbol{\Sigma}_i$ and $\boldsymbol{S}_j$ are diagonal, then the expected log-sum-exp function can be upper bounded as follows:

$$\begin{aligned} &\mathbb{E}[\log \sum_{j'\in\Omega_e} \exp(-||z_i - \boldsymbol{\eta}_{j'}||_2^2)] \\ &\quad \leq \log \sum_{j'\in\Omega_e} \exp\Big\{\sum_{l=1}^{L} -\frac{(\mu_{il} - m_{j'l})^2}{1 + 2(\sigma_{il}^2 + s_{j'l}^2)} \\ &\qquad - \frac{1}{2}\log(1 + 2(\sigma_{il}^2 + s_{j'l}^2))\Big\} \end{aligned} \quad (6)$$

See Supplementary Section I for the proof.

Combining Equation 5 and 6, we can define the overall objective of a training episode as follows:

$$\begin{aligned} \ell'_e = \sum_{i\in Q_e}\sum_{j\in\Omega_e}\Big[&\sum_{l=1}^{L}\mathbb{I}(y_i = j)[-(\mu_{il} - m_{jl})^2 - \sigma_{il}^2 - s_{jl}^2] \\ &- \log \sum_{j'\in\Omega_e} \exp\Big\{\sum_{l=1}^{L} -\frac{(\mu_{il} - m_{j'l})^2}{1 + 2(\sigma_{il}^2 + s_{j'l}^2)} \\ &- \frac{1}{2}\log(1 + 2(\sigma_{il}^2 + s_{j'l}^2))\Big\}\Big], \end{aligned} \quad (7)$$

where $\mathbb{I}()$ is the binary indicator function. Note that two consecutive bounds are applied for the sum over the log marginal likelihood ℓ_e, which results in a tractable objective ℓ'_e which is a proper lower bound to ℓ_e, i.e., $\ell'_e \leq \ell_e$.

Hence, maximizing ℓ'_e guarantees maximizing ℓ_e. The first bound is a lower bound due to Jensen's inequality which is used to interchange expectation and the log operator, which results in the analytic computation of the first RHS term in Equation 4. The second bound is an upper bound to the log-sum-exp function that results in a lower bound to Equation 4. This enables taking expectation of log-sum-exp function with respect to $\{\boldsymbol{\eta}_j\}_{j\in\Omega_e}$ and z_i, therefore, results in a tractable objective in Equation 7. Note that this objective is parameterized by neural networks through the free parameters of the distributions and the gradients of the objective with respect to the neural network parameters can be computed using backpropagation without Monte Carlo sampling.

C. FEW-SHOT LEARNING OPTIMIZATION

We set up feature extractors based on neural networks that compute low-dimensional representations of the query and support samples in a common stochastic latent space. Let $f_{\theta_m} : \mathcal{R}^{D_m} \rightarrow \mathcal{R}^L$ denote a function approximated by a neural network with parameter set θ_m, which projects mth modality of a query sample $\boldsymbol{x}_{i,m}$ to the low dimensional continuous stochastic latent space, i.e., $\boldsymbol{x}_{i,m} \xrightarrow{f_{\theta_m}} \{\boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m}\}$. Similarly, let $g_{\phi_m} : \mathcal{R}^{D_m} \rightarrow \mathcal{R}^L$ denote a function with parameter set ϕ_m that projects mth modality of a support sample $\boldsymbol{x}_{i',m}$ to the same latent space, i.e., $\boldsymbol{x}_{i',m} \xrightarrow{g_{\phi_m}} \{\boldsymbol{\mu}_{i,m'}, \boldsymbol{\Sigma}_{i,m'}\}$.

Let $p(z_i|\boldsymbol{x}_{i,m}) = \mathcal{N}(z_i|\boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m})$ denote the probability distribution induced over z_i by the modality m of the query sample $\boldsymbol{x}_{i,m}$. Multiple modalities contribute to the probability distribution of z_i since a single latent variable is assumed for all the modalities by model definition (See Equation 2). A natural choice to combine the predictions is to assume factorization and multiply the probability density functions of the modality-specific estimates [13], [36] as $p(z_i|\boldsymbol{x}_{i,m=1:M}) = \prod_{m=1}^{M} p(z_i|\boldsymbol{x}_m)$. When the aggregate probability estimate is denoted as $p(z_i|\boldsymbol{x}_{i,m=1:M}) = \mathcal{N}(z_i|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, the free parameters are given by:

$$\boldsymbol{\Sigma}_i^{-1} = \sum_{m=1}^{M} \boldsymbol{\Sigma}_{i,m}^{-1},$$
$$\boldsymbol{\mu}_i = \boldsymbol{\Sigma}_i \left(\sum_{m=1}^{M} \boldsymbol{\Sigma}_{i,m}^{-1} \boldsymbol{\mu}_{i,m} \right). \tag{8}$$

Next, we compute the distributions of the prototypes assigned for each class. As opposed to the latent variables of the query samples, the prototypes are inferred from multiple samples, and each sample has multiple modalities. Let $p(z_{i'}|\boldsymbol{x}_{i',m}) = \mathcal{N}(z_{i'}|\boldsymbol{\mu}_{i',m}, \boldsymbol{\Sigma}_{i',m})$ denote the induced distribution from mth modality of i'th support sample. If we factorize the distribution of the prototype as $p(\boldsymbol{\eta}_j|\{\boldsymbol{x}_{i',m=1:M}\}_{i'\in S_{e,j}}) = \prod_{i'\in S_{e,j}} \prod_{m=1}^{M} p(z_{i'}|\boldsymbol{x}_{i',m})$, the conditional distribution of prototype $\boldsymbol{\eta}_j$ is obtained as $p(\boldsymbol{\eta}_j|\{\boldsymbol{x}_{i',m=1:M}\}_{i'\in S_{e,j}}) = \mathcal{N}(\boldsymbol{\eta}_j|\boldsymbol{m}_j, \boldsymbol{S}_j)$, where the parameters $\boldsymbol{m}_j, \boldsymbol{S}_j$ are given as:

$$\boldsymbol{S}_j^{-1} = \sum_{i'\in S_{e,j}} \sum_{m=1}^{M} \boldsymbol{\Sigma}_{i',m}^{-1},$$
$$\boldsymbol{m}_j = \boldsymbol{S}_j \left(\sum_{i'\in S_{e,j}} \sum_{m=1}^{M} \boldsymbol{\Sigma}_{i',m}^{-1} \boldsymbol{\mu}_{i',m} \right). \tag{9}$$

Note that the covariance matrices are assumed to be diagonal. Hence, the matrix inverse operations are replaced with reciprocals, and matrix multiplications are replaced with element-wise multiplications. See Supplementary Section II for details on how to perform these computations in a numerically stable way. Afterwards, one can plug the estimates of the distributions $\{\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i\in Q_e}$, $\{\boldsymbol{m}_j, \boldsymbol{S}_j\}_{j\in\Omega_e}$ in Equation 7 to compute the objective. The pseudo-code for few-shot learning is given in Supplementary Section III.

D. ZERO-SHOT LEARNING OPTIMIZATION

In the case of zero-shot learning, we observe class meta-data $\{\boldsymbol{a}_{j,r=1:R}\}_{j\in\Omega_e}$ from R modalities in the support set, instead of the objects $x_{i',m}$ considered in few-shot learning. In the query set, like in few-shot learning, we have only objects $x_{i,m}$. In this case, Equation 2 can be modified as:

$$\begin{aligned} p(y_i|\boldsymbol{x}_{i,m=1:M}, S_e) &= \int\int\cdots\int p(y_i|z_i, \{\boldsymbol{\eta}_j\}_{j\in\Omega_e}) \\ &\quad\times p(z_i|\boldsymbol{x}_{i,m=1:M})\Big[\prod_{j\in\Omega_e} p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r=1:R})\Big] \\ &\quad\times d\boldsymbol{\eta}_{\Omega_{e,1}}\ldots d\boldsymbol{\eta}_{\Omega_{e,N}} dz_i, \end{aligned} \tag{10}$$

where the new term $p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r=1:R})$ denotes the conditional distribution of prototype $\boldsymbol{\eta}_j$, and $\boldsymbol{a}_{j,r} \in \mathcal{R}^{D_r}$ denotes the rth modality of the meta-data of class j. In this setting, the conditional distribution of $\boldsymbol{\eta}_j$ depends on $\boldsymbol{a}_{j,r=1:R}$. To this end, we set up neural networks $g_{\phi_r} : \mathcal{R}^{D_r} \rightarrow \mathcal{R}^L$ to project each modality of class meta-data to the latent space, i.e., $\boldsymbol{a}_{j,r} \xrightarrow{g_{\phi_r}} \{\boldsymbol{\mu}_{j,r}, \boldsymbol{\Sigma}_{j,r}\}$. Let $p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r}) = \mathcal{N}(\boldsymbol{\eta}_j|\boldsymbol{\mu}_{j,r}, \boldsymbol{\Sigma}_{j,r})$ denote the induced distribution of $\boldsymbol{a}_{j,r}$ over $\boldsymbol{\eta}_j$. Using the similar factorization used in few-shot setting for the aggregate distribution $p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r=1:R}) = \prod_{r=1}^{R} p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r})$, we can compute the parameters of the conditional distribution $p(\boldsymbol{\eta}_j|\boldsymbol{a}_{j,r=1:R}) = \mathcal{N}(\boldsymbol{\eta}_j|\boldsymbol{m}_j, \boldsymbol{S}_j)$ as:

$$\boldsymbol{S}_j^{-1} = \sum_{r=1}^{R} \boldsymbol{\Sigma}_{j,r}^{-1},$$
$$\boldsymbol{m}_j = \boldsymbol{S}_j \left(\sum_{r=1}^{R} \boldsymbol{\Sigma}_{j,r}^{-1} \boldsymbol{\mu}_{j,r} \right). \tag{11}$$

Using the objects in the query set, the distribution parameters are computed as in Equation 8, where $\boldsymbol{\mu}_{i,m}$ and $\boldsymbol{\Sigma}_{i,m}$ are calculated through the neural network f_{θ_m}. Given these estimates, one can compute the objective using Equation 7, to evaluate the zero-shot classification performance.

The pseudo-code for zero-shot learning is given in Supplementary Section IV.

IV. EXPERIMENTAL STUDY

We conduct experiments on Omniglot [3] and CUB datasets [37]. Omniglot provides data for 1623 handwritten characters from 50 alphabets. Each character has 20 drawings by different people, and each drawing provides an image and stroke data as two different views of the same object. We conduct an experiment on the Omniglot dataset to demonstrate that our stochastic latent space formulation improves generalization. Then we show that the additional information from the stroke data can be leveraged by the proposed method to improve few-shot learning performance.

The CUB dataset provides 11788 images of 200 bird species. Each species has 40 to 60 images. This dataset also provides class meta-data per species in the form of 312-dimensional real-valued attribute vectors. Additionally, [38] collected another meta-data corresponding to the visual descriptions of the species. On this dataset, we conduct experiments to demonstrate that the proposed method can utilize images, class attributes, and visual descriptions to improve the zero-shot and few-shot classification performances.

A. IMPLEMENTATION DETAILS

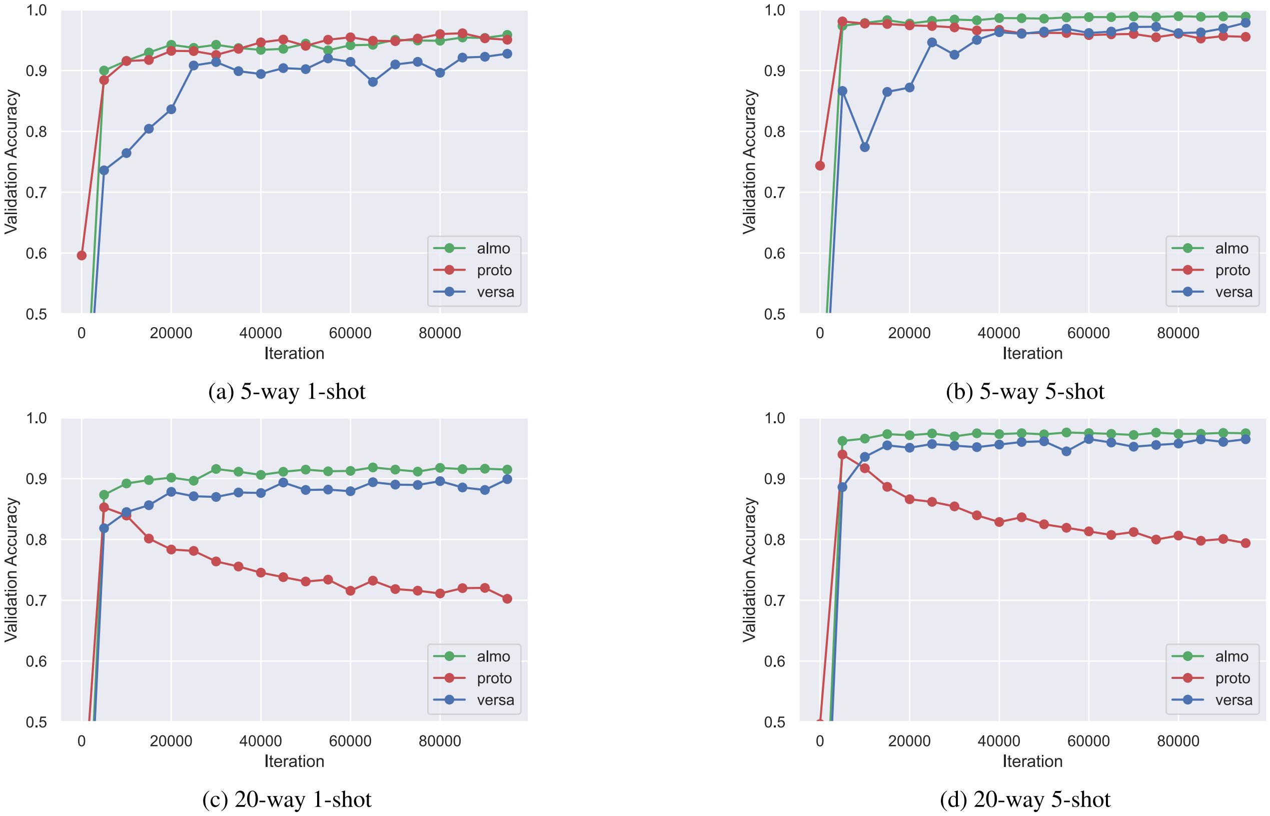
For the Omniglot dataset, we utilize the original splits, respectively called the background and evaluation sets. To generate the training episodes, we use 964 characters residing in the background set. To generate the testing episodes, we use 423 characters residing in the evaluation set. We generate 100,000 training episodes to train the model and 600 testing episodes to test the few-shot classification performance. Each episode contains randomly chosen $2 \times N \times K$ objects, where each object has two modalities, i.e., $M = 2$, as image and stroke data. The details of how each episode is formed are given in Section III-A. To preprocess the images, we follow the procedure of [22]. We first downsize the images to 28×28, then normalize and invert. For the stroke data, we follow [39] and use a 3-dimensional real-valued sequential data format. The first dimension of the data is a binary indicator that corresponds to the first touch of the stylus after a break while drawing a character. Either on the first touch or on the touch after a break, this bit is on, otherwise, it is off. The second dimension corresponds to the coordinates of the stylus which shows the x-axis of the pixel location, and the third dimension corresponds to the y-axis. When a touch occurs, the second and third dimension takes the absolute coordinate, otherwise, it takes the relative coordinates of the stylus. We fix the network structures throughout the experiments. For the image data, we use four stacked 2D Convolution layers each with 64 filters, having a kernel size of 3, and with stride 2. The ReLU activation function is used in all layers. Then, a final dense layer with an identity activation is stacked for mapping the previous layer's hidden activations to the mean parameter, and another dense layer with the same configuration for mapping the previous layer's hidden activations to the diagonal covariance estimates in the stochastic latent space. For the stroke data, we first zero-pad the sequences to equal length, then use a masking layer, followed by an LSTM layer with 128 hidden nodes with RELU activation. Similarly, two dense layers with identity activations are stacked for the mean and covariance mappings. We fix the dimension of the latent space as $L = 256$.

For the CUB-200 dataset, we use the standard train-validation-test split described in [40], in which there are 100 training, 50 validation, and 50 test classes. We create 30,000 training and 500 validation episodes where the validation episodes are used for early stopping. In the zero-shot setting, the support set in each episode contains class attributes and visual descriptions, whereas the query set contains only the images. In the few-shot setting, both support and query sets contain all the modalities. For few-shot learning, choosing 1 and 5 shots for 5-way and 20-way classification is common practice in the literature. To be able to align and compare with the baseline algorithms, we conduct the few-shot experiments using these settings.

We employ 2048-dimensional numerical features as input, which are extracted using ResNet for the images by following the procedure in [40]. Class attributes are 312-dimensional numerical features, which are provided with the dataset. For the visual descriptions, we use the 400-dimensional features extracted by [38]. All these features are normalized in advance by using the statistics computed from the training set. Then, two linear mappings are realized to project the class attributes and description features to the latent space by using dense layers with identity activation functions. Another linear mapping is realized through a dense layer to project the image features to the latent space. For optimization over the training data, we use the Adam optimizer with 10^{-3} learning rate throughout all the experiments.

B. FEW-SHOT LEARNING

We first conduct an experiment on the Omniglot dataset to fairly compare the generalization performance of ALMO with two closely related well-established algorithms by providing the same experimental conditions, i.e., the network architecture, latent space dimension, data pre-processing, and single modality (images). Prototypical networks (PROTO) that [4] uses deterministic latent space for the embeddings, and VERSA [41] which amortizes the conditional distributions of task-specific parameters. Specifically, VERSA induces distributions over the prototypes, however, it assumes deterministic query embeddings. In addition, VERSA uses Monte-Carlo integration, which results in crude marginal likelihood estimation and slow convergence, whereas ALMO uses analytic integration. These differences translate to significant performance deviation. Fig 2 shows the learning curves of the algorithms. PROTO severely over-fits due to point estimation of the embeddings, especially in 20-way settings. VERSA does not over-fit, however, it performs inferior to our algorithm due to the differences mentioned above. ALMO



(a) 5-way 1-shot

(b) 5-way 5-shot

(c) 20-way 1-shot

(d) 20-way 5-shot

FIGURE 2. Learning curves of PROTO, VERSA, and the proposed model ALMO. Overfitting is significant for PROTO when the number of shots/ways is large. Since the proposed ALMO fuses multimodal data in a stochastic latent space it mitigates over-fitting. VERSA suffers from its crude Monte-Carlo approximation to the marginal likelihood.

integrates out the latent variables and exhibits robustness to over-fitting.

We next fairly compare ALMO with several few-shot learning algorithms in terms of few-shot classification accuracy by providing the same experimental conditions. Both Omniglot and CUB-200 datasets are considered. In addition to PROTO and VERSA, we also implement well-established Matching Networks (MN) [22], TapNeT [20], and PTMAP [27]. For each algorithm, there are two variants, -I and -S, which use either images or strokes, respectively, for the Omniglot dataset. Similarly -I and -D, use either images or descriptions/attributes, for the CUB-200 dataset. In addition to ALMO as a multi-modal classifier, we consider MProto [23], which has recently been proposed as the multi-modal version of PROTO, that fuses the modality embeddings by average pooling.

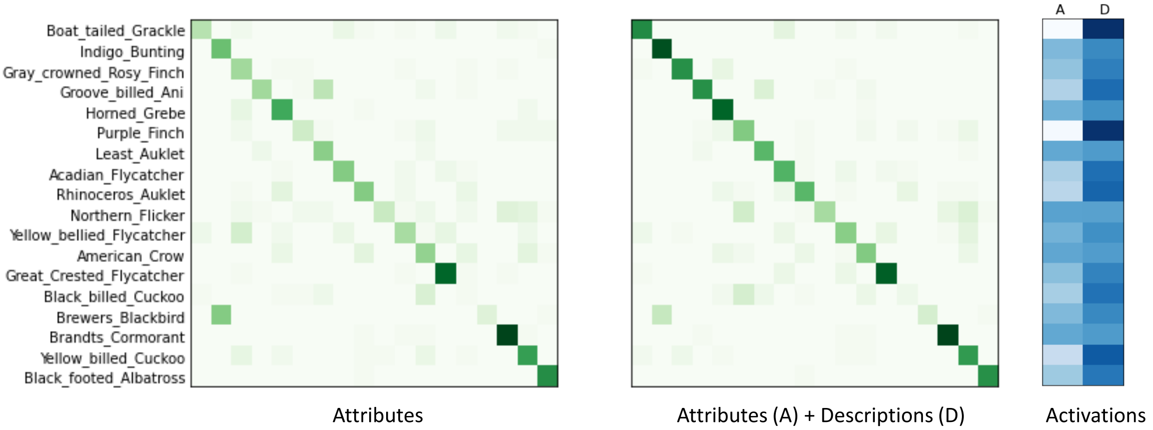
Few-shot classification results are compiled in Table 1 for different N (5-way and 20-way) and K (1-shot and 5-shot) settings. The mean accuracy is computed by averaging the accuracies of all testing episodes. The optimizers are stopped early when the validation accuracies are non-increasing. We observe that ALMO achieves the highest accuracy by leveraging modalities effectively on both datasets. For Omniglot, images provide spatial information, and strokes/descriptions provide sequential information. For CUB-200, multi-modal information on the class attributes/descriptions is even more effective. It may be claimed that multiple modalities provide complementary information for the objects, and ALMO outperforms ALMO-I, ALMO-S, and other baselines by effectively leveraging these different views. [1]

C. ZERO-SHOT LEARNING

Next, we conduct an experiment to assess the zero-shot learning performance of ALMO by using the CUB-200 dataset. We choose $N = 50$ to form episodes containing samples from 50 different classes. For each class, a single image is chosen randomly as the query object. The optimizer is chosen as Adam with 10^{-3} learning rate. After the model is trained, we report average per-class top-1 accuracy on the test set [40]. In Table 2, we compare the performance with the recent zero-shot learning algorithms that also utilize ResNet features for images as a fair comparison. [2]

[1] While the reported results may seem inferior compared to the state-of-the-art results, see [42] and [43] for the related discussion about fair comparison and comparable results for several settings.

[2] Although there are some recently reported better results [33], [44], [45] based on customized feature extraction via attribute attention for zero-shot learning, we believe comparison with them would not be fair since our focus is on fusing different data modalities, not feature extraction. Also, we focus on inductive zero-shot learning settings where test images are not available during the training.



Attributes

Attributes (A) + Descriptions (D)

Activations

FIGURE 3. Confusion matrices of the test classes that benefit most from the visual descriptions. The left confusion matrix is obtained when only attributes are used. The right confusion matrix is obtained when both attributes and descriptions are incorporated. The activation of each modality is given in the rightmost figure.

TABLE 1. Few-shot performance on Omniglot and CUB-200 datasets. Our proposed ALMO performs the best in 7 of the 8 cases studied.

	5-way		20-way	
	1-shot	5-shot	1-shot	5-shot
Omniglot				
Proto-I [4]	96.1	98.0	85.2	93.9
MN-I [22]	90.4	96.0	81.8	90.1
Versa-I [41]	92.7	97.8	89.9	96.4
TapNet-I [20]	95.7	98.2	85.8	92.7
PTMAP-I [27]	94.2	97.1	84.2	91.2
ALMO-I	95.9	98.9	91.8	97.6
Proto-S [4]	80.6	83.1	71.6	66.5
MN-S [22]	68.3	84.2	56.4	59.8
Versa-S [41]	80.1	82.9	74.8	82.5
TapNet-S [20]	80.5	83.2	70.9	66.1
PTMAP-S [27]	79.3	82.1	69.3	68.4
ALMO-S	80.4	85.2	77.5	84.4
MProto [23]	96.4	98.3	88.1	94.5
ALMO	**96.9**	**99.4**	**93.4**	**98.5**
CUB-200				
Proto-I [4]	76.2	86.0	51.5	68.3
MN-I [22]	62.6	82.8	45.0	65.8
Versa-I [4]	75.3	86.1	52.5	69.3
TapNet-I [20]	75.4	85.5	50.9	67.7
PTMAP-I [27]	74.3	84.9	50.3	66.6
ALMO-I	76.0	86.6	54.9	72.5
Proto-D [4]	71.4	83.4	45.4	63.3
MN-D [22]	59.3	80.0	42.4	58.2
Versa-D [4]	70.9	84.7	47.2	65.3
TapNet-D [20]	70.8	83.0	45.1	61.1
PTMAP-D [27]	70.1	82.4	43.9	63.2
ALMO-D	69.2	86.3	49.0	68.5
MProto [23]	**83.6**	91.9	60.2	78.5
ALMO	83.1	**92.3**	**65.9**	**85.4**

TABLE 2. Inductive zero-shot performance on CUB-200 dataset. The proposed ALMO performs the best overall.

CUB-200	Meta-data	Accuracy
DA-SJE [38]	Attributes	50.9
SYNC [46]	Attributes	54.1
GFZSL [47]	Attributes	53.0
Proto [4]	Attributes	54.6
DS-SJE [38]	Descriptions	56.8
ALMO-A	Attributes	54.8
ALMO-D	Descriptions	56.7
ALMO	Attributes + Descriptions	**59.5**

The competing algorithms use only attributes or descriptions as the class meta-data. On the other hand, our proposed algorithm ALMO can exploit both visual descriptions and attributes to achieve higher zero-shot classification accuracy by leveraging the information coming from both modalities of the class meta-data. It is clear that such multimodal data provide complementary information for some classes and our algorithm effectively utilizes it.

D. INTERPRETABILITY

The contribution of each modality can be interpreted by analyzing the data fusion process in the latent space. In the zero-shot setting, to infer the class-level latent variables, the predictions computed from the attributes and the descriptions are combined using the proposed framework, as discussed in Section III-D. After training, one can compute the KL divergences between the combined prediction $\{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\}$ and the individual predictions $\{\boldsymbol{\mu}_{j,r}, \boldsymbol{\Sigma}_{j,r}\}$ as a proxy for the contributions of the modalities. A softmax function subsequently can be used to normalize the weights.

In Figure 3, we show the partial confusion matrices consisting of the classes that benefit most in terms of accuracy increase when the descriptions are included additionally to the attributes. The classes are in descending order in terms of accuracy increase. For instance, the visual descriptions of the *boat-tailed grackle* seem very useful to enhance the classification of the images of this species. We show the corresponding activation weights associated with each class on the rightmost matrix. We observe that, in most of these classes, the activation of the visual descriptions is larger than the activation of the attributes.

V. CONCLUSION

A. SUMMARY

In this paper, we developed a framework, ALMO, for improving any-shot learning for multimodal data. The proposed framework allows diverse information sources to be fused by implementing an encoder network that exploits the multimodality nature of the data. The framework is flexible and can incorporate object-level multimodal information as well as class-level multimodal information. On the Omniglot dataset, we demonstrated that ALMO can incorporate object-level multimodal information (image and stroke) to improve few-shot classification performance. On the CUB-200 dataset, we showed that ALMO can exploit class-level multimodal information (attributes and visual descriptions) as well as object-level information (image) to improve any-shot classification performance.

B. LIMITATIONS

As with any model, our framework has some limitations. First, the main assumption that facilitates analytic computation of the model objective in Eq. 7 is that the estimated co-variances are diagonal. Each feature extractor provides a mean vector of size L and a diagonal covariance vector of size L for each modality, which implies that estimated distributions in the latent space have independent components. This is a common assumption in many probabilistic deep learning models, for instance, in Variational Auto-encoders [48]. However, this assumption may be overly strong and thus may reduce predictive accuracy. A straightforward solution is to compute the full covariance instead. However, this will increase the number of parameters needed to map the covariance from L to L^2. A compromise would be to estimate the upper triangle using a linear layer with $L(L-1)$ parameters. In addition to the diagonal covariance assumption, we have exploited two bounds to derive the analytic objective, Jensen's inequality and Taylor series expansion. Although these types of bounds are common in machine learning, in particular for variational inference, analysis of the errors due to minimizing such a bound remains an open problem. Furthermore, our framework assumes that the distance between the prototypes and query samples in the latent space is Euclidean. if other distance metrics are considered, our analysis would need to be extended, i.e., the analytic objective should be derived from scratch. Finally, the modalities are fused using the product of expert formulation in the proposed framework. Other options, such as mixture of experts, can be considered as future work. However, using a mixture of Gaussian instead of a single Gaussian would also require reformulating the objective function.

APPENDIX A
UPPER BOUND FOR THE LOG-SUM-EXP FUNCTION

Applying first-order order Taylor series expansion to the natural logarithm function, the expected log-sum-exp function can be upper bounded as follows:

$$\mathbb{E}[\log \sum_j \exp(-||\boldsymbol{z}_i - \boldsymbol{\eta}_j||_2^2)] \leq \log \sum_j \mathbb{E}[\exp(-||\boldsymbol{z}_i - \boldsymbol{\eta}_j||_2^2)],$$

where the inner expression corresponds to the expectation of the exponential negative squared Euclidean distance. Let $\boldsymbol{a} = \boldsymbol{z}_i - \boldsymbol{\eta}_j$, then $\boldsymbol{a} \sim N(\boldsymbol{m}, \boldsymbol{\Sigma})$ where $\boldsymbol{m} = \boldsymbol{\mu}_i - \boldsymbol{m}_j$ and $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_i + \boldsymbol{S}_j$. In the diagonal case, the latent components are independent. Let the component l be denoted as $a_l \sim N(m_l, \sigma_l^2)$. Then the expectation can be factorized as:

$$\mathbb{E}[e^{-||\boldsymbol{a}||_2^2}] = \prod_{l=1}^{L} E[e^{-a_l^2}].$$

If we re-parameterize a_l by $a_l = m_l + \sigma_l \epsilon$ where $\epsilon \sim N(0, 1)$, the expectation for the component l is computed analytically by solving the following Gaussian integral:

$$\begin{aligned} E[e^{-a_l^2}] &= \frac{1}{\sqrt{2\pi}} \int e^{-(m_l+\sigma_l\epsilon)^2} e^{-\epsilon^2/2} d\epsilon \\ &= \frac{1}{\sqrt{2\pi}} e^{-m_l^2} \int e^{-\sigma_i^2\epsilon^2 - 2m_l\sigma_l\epsilon - \epsilon_l^2/2} d\epsilon \\ &= \frac{1}{\sqrt{2\pi}} e^{-m_l^2} \int e^{-\frac{1}{2}(\epsilon^2(1+2\sigma_l^2)+4m_l\sigma_l\epsilon)} d\epsilon \\ &= \frac{1}{\sqrt{2\pi}} e^{-m_l^2} \int e^{-\frac{1+2\sigma_l^2}{2}\left[(\epsilon+\frac{2m_l\sigma_l}{1+2\sigma_l^2})^2 - \frac{4m_l^2\sigma_l^2}{(1+2\sigma_l^2)^2}\right]} d\epsilon \\ &= \frac{1}{\sqrt{1+2\sigma_l^2}} e^{-\frac{m_l^2}{1+2\sigma_l^2}}. \end{aligned}$$

Replacing the components into the original vector:

$$\mathbb{E}[e^{-||\boldsymbol{a}||_2^2}] = \exp\Big\{\sum_{l=1}^{L} -\frac{m_l^2}{1+2\sigma_l^2} - \frac{1}{2}\log(1+2\sigma_l^2)\Big\},$$

and subsequently replacing the original variables $\boldsymbol{z}_i$ and $\boldsymbol{\eta}_j$, we get the following log-sum-exp function:

$$\begin{aligned} &\log \sum_j \mathbb{E}[\exp(-||\boldsymbol{z}_i - \boldsymbol{\eta}_j||_2^2) \\ &= \log \sum_j \exp\Big\{\sum_{l=1}^{L} -\frac{(\mu_{il} - m_{jl})^2}{1+2(\sigma_{il}^2 + s_{jl}^2)} \\ &\quad - \frac{1}{2}\log(1+2(\sigma_{il}^2 + s_{jl}^2))\Big\}, \end{aligned}$$

which can be computed analytically in a numerically stable way by using the conventional log-sum-exp trick.

APPENDIX B
NUMERICAL STABILITY AND COMPUTATIONAL COMPLEXITY OF THE DATA FUSION

We focus on the parameters of the conditional distribution for the jth prototype latent variable. The parameters of the query latent variables can be computed by analogy. Recall that the multiplication of the probability density estimates of each modality and each support sample yields:

$$\boldsymbol{S}_j^{-1} = \sum_{i' \in S_{e,j}} \sum_{m=1}^{M} \boldsymbol{\Sigma}_{i',m}^{-1},$$
$$\boldsymbol{m}_j = \boldsymbol{S}_j \left(\sum_{i' \in S_{e,j}} \sum_{m=1}^{M} \boldsymbol{\Sigma}_{i',m}^{-1} \boldsymbol{\mu}_{i',m} \right),$$

where the covariance matrix $\boldsymbol{\Sigma}_{i',m} = \mathrm{diag}([\sigma^2_{i'1m}, \ldots, \sigma^2_{i'Lm}])$ is assumed to be diagonal. The neural network estimates a vector $[\log \sigma^2_{i'1m}, \ldots, \log \sigma^2_{i'Lm}]$ for each modality m with log transformation by using an identity activation function on the last layer. If we denote $\boldsymbol{S}_j = \mathrm{diag}([s^2_{j1}, \ldots, s^2_{jL}])$ as the covariance matrix of the jth the prototype latent variable, then the estimation of each component l reduces to:

$$\log s^2_{jl} = -\log(\sum_{i' \in S_{e,j}} \sum_{m=1}^{M} e^{-\log \sigma^2_{i'lm}}) = -\mathrm{lse}(-\log \boldsymbol{\sigma}^2_{jl}),$$

where $\boldsymbol{\sigma}^2_{jl} = [\sigma^2_{0l0}, \ldots, \sigma^2_{0lM}, \ldots, \sigma^2_{Kl0}, \ldots, \sigma^2_{KlM}]$ is a vector consisting of the lth component of each modality and each support sample variance estimates and $\mathrm{lse}(\boldsymbol{x})$ is log-sum-exp-function which has to be computed in a numerically stable way. Specifically, the conventional lse trick, $\mathrm{lse}(\boldsymbol{x} - b\mathbf{1}) + b$, where $b = \max_l x_l$, is used for the computation of this function.

Similarly, the neural network maps each sample to a mean vector denoted as $\boldsymbol{\mu}_{i,m} = [\mu_{i1m}, \ldots, \mu_{iLm}]$, where the components are obtained with identity activation function on the last layer. Denote $\boldsymbol{m}_j = [m_{j1}, \ldots, m_{jL}]$ as the mean vector of the jth prototype latent variable. Then, the expression for the component l reduces to:

$$m_{jl} = \frac{\sum_{i' \in S_{e,j}} \sum_{m=1}^{M} \mu_{i'lm} e^{-\log \sigma^2_{i'l}}}{\sum_{i' \in S_{e,j}} \sum_{m=1}^{M} e^{-\log \sigma^2_{i'lm}}} = \boldsymbol{\mu}_{jl}^T \mathcal{S}(-\log \boldsymbol{\sigma}^2_{jl})),$$

where $\boldsymbol{\mu}_{jl} = [\mu_{0l0}, \ldots, \mu_{0lM}, \ldots, \mu_{Kl0}, \ldots, \mu_{KlM}]$ is a vector comprising the lth component of each modality and each support sample mean estimates, and $\mathcal{S}$ denotes the soft-max function. Note that the fusion is achieved through the weighted average of the means, where the weights are computed via the soft-max function of each component which takes the corresponding variances as the input. In this expression, the soft-max function has to be computed in a numerically stable way. Since $\log \mathcal{S}(\boldsymbol{x}) = \boldsymbol{x} - \mathrm{lse}(\boldsymbol{x})$, one can use the above lse trick to compute $\log \mathcal{S}$, then take the exponential of it.

The proposed data fusion algorithm introduces a slight computational overhead as compared to the deterministic variants. Particularly, we need to compute component-wise log-sum-exp functions to obtain the covariances $\boldsymbol{S}_j$ and component-wise softmax functions to obtain the means $\boldsymbol{m}_j$ for each prototype. The deterministic variants only compute the means, which require solely add operators. Hence, we additionally execute L softmax and L log-sum-exp functions for each prototype with $L \times N$ operators per function, where L is the latent space dimension and N is the number of classes in the episode, whereas the deterministic variants requiring only $L \times N$ add operators for each prototype. Fortunately, the total number of operators induced is linearly scaled with the latent space dimension and the few-shot class number, hence does not affect the scalability of the proposed algorithm. We also empirically observed that this computational overhead of the operators ALMO additionally carries is statistically insignificant since the dominant part of the computations is coming from the feature extraction. The average running time of ALMO for classifying a single query from the Omniglot testing dataset was found to be 12.169 msec in the 5-shot and 5-way setting, whereas MProto takes 12.073 msec in an NV 1080TI GPU. In terms of parameters, ALMO induces additional parameters due to the extra linear mapping for the variances as compared to the deterministic data fusion. In particular, ALMO requires two linear layers to map the final hidden states to the mean and covariance, whereas deterministic data fusion requires only a single linear layer to map the final hidden states to the mean. This overhead is also insignificant because most of the parameters are devoted to the feature extraction networks (around %99).

APPENDIX C
PSEUDO-CODE FOR FEW-SHOT LEARNING

Algorithm 1 Few-Shot Optimization per Episode

```
Input: ${x_{i,m=1:M}, y_i}_{i \in Q_e}$, ${x_{i',m=1:M}, y_{i'}}_{i' \in S_e}$, $\Omega_e$
Initialize $\theta_{m=1:M}$
for $i'$ in $S_e$ do
    for $m = 1$ to $M$ do
        $f_{\theta_m} : \boldsymbol{x}_{i',m} \rightarrow \{\boldsymbol{\mu}_{i',m}, \boldsymbol{\Sigma}_{i',m}\}$
    end for
end for
for $j$ in $\Omega_e$ do
    Infer $p(\boldsymbol{\eta}_j | \{\boldsymbol{x}_{i',m=1:M}\}_{i' \in S_{e,j}}) = \mathcal{N}(\boldsymbol{m}_j, \boldsymbol{S}_j)$ using Equation 9
end for
for $i$ in $Q_e$ do
    for $m = 1$ to $M$ do
        $f_{\theta_m} : \boldsymbol{x}_{i,m} \rightarrow \{\boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m}\}$
    end for
    Infer $p(z_i | \boldsymbol{x}_{i,m=1:M}) = \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ using Equation 8
    for $j$ in $\Omega_e$ do
        Compute $\mathbb{E}[\log p(y_i = j | z_i, \{\boldsymbol{\eta}_j\}_{j \in \Omega_e})]$ using Equations 4-6
    end for
end for
Compute $\ell'_e$ using Equation 7
for $m = 1$ to $M$ do
    Compute $d\ell'_e / d\theta_m$ and update $\theta_m$
end for
```

APPENDIX D
PSEUDO-CODE FOR ZERO-SHOT LEARNING

Algorithm 2 Zero-Shot Optimization per Episode

```
Input: ${a_{j,r=1:R}}_{j \in \Omega_e}, \{x_{i,m=1:M}, y_i\}_{i \in Q_e}$
Initialize $\phi_{r=1:R}, \theta_{m=1:M}$
for $j$ in $\Omega_e$ do
    for $r = 1$ to $R$ do
        $g_{\phi_r} : a_{j,r} \rightarrow \{\mu_{j,r}, \Sigma_{j,r}\}$
    end for
    Infer $p(\eta_j | a_{j,r=1:R}) = \mathcal{N}(m_j, S_j)$ using 11
end for
for $i$ in $Q_e$ do
    for $m = 1$ to $M$ do
        $f_{\theta_m} : x_{i,m} \rightarrow \{\mu_{i,m}, \Sigma_{i,m}\}$
    end for
    Infer $p(z_i | x_{i,m=1:M}) = \mathcal{N}(\mu_i, \Sigma_i)$ using Equation 8
    for $j$ in $\Omega_e$ do
        Compute $\mathbb{E}[\log p(y_i = j | z_i, \{\eta_j\}_{j \in \Omega_e})]$ using Equations 4-6
    end for
end for
Compute $\ell'_e$ using Equation 7
for $m = 1$ to $M$ do
    Compute $d\ell'_e / d\theta_m$ and update $\theta_m$
end for
for $r = 1$ to $R$ do
    Compute $d\ell'_e / d\phi_r$ and update $\phi_r$
end for
```

REFERENCES

[1] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5542–5551.

[2] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2014.

[3] B. Lake, R. Salakhutdinov, J. Gross, and J. Tenenbaum, "One shot learning of simple visual concepts," in *Proc. Annu. Meeting Cogn. Sci. Soc.*, vol. 33, 2011, pp. 2568–2573.

[4] J. Wang and Y. Zhai, "Prototypical Siamese networks for few-shot learning," in *Proc. IEEE 10th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2020, pp. 178–181.

[5] B. Kulis, "Metric learning: A survey," *Found. Trends Mach. Learn.*, vol. 5, no. 4, pp. 287–364, 2013.

[6] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, Aug. 2019.

[7] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[8] S. Yu, K. Yu, V. Tresp, H.-P. Kriegel, and M. Wu, "Supervised probabilistic principal component analysis," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2006, pp. 464–473.

[9] J. Bernardo, M. Bayarri, J. Berger, A. Dawid, D. Heckerman, A. Smith, and M. West, "Bayesian factor regression models in the 'large *p*, small *n* paradigm,'" *Bayesian Statist.*, vol. 7, pp. 733–742, 2003.

[10] L. Sun, S. Ji, S. Yu, and J. Ye, "On the equivalence between canonical correlation analysis and orthonormalized partial least squares," in *Proc. IJCAI*, vol. 9, 2009, pp. 1230–1235.

[11] A. Klami, S. Virtanen, and S. Kaski, "Bayesian exponential family projections for coupled data sources," in *Proc. 26th Conf. Uncertainty Artif. Intell.* AUAI Press, 2010.

[12] Y. Yilmaz, M. Aktukmak, and A. O. Hero, "Multimodal data fusion in high-dimensional heterogeneous datasets via generative models," *IEEE Trans. Signal Process.*, vol. 69, pp. 5175–5188, 2021.

[13] M. Wu and N. Goodman, "Multimodal generative models for scalable weakly-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 5575–5585.

[14] R. Vedantam, I. Fischer, J. Huang, and K. Murphy, "Generative models of visually grounded imagination," 2017, *arXiv:1705.10762*.

[15] Y. Shi, N. Siddharth, B. Paige, and P. Torr, "Variational mixture-of-experts autoencoders for multi-modal deep generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 15718–15729.

[16] W. Wang, X. Yan, H. Lee, and K. Livescu, "Deep variational canonical correlation analysis," 2016, *arXiv:1610.03454*.

[17] Y.-H. H. Tsai, P. P. Liang, A. Zadeh, L.-P. Morency, and R. Salakhutdinov, "Learning factorized multimodal representations," in *Proc. Int. Conf. Represent. Learn.*, 2019, pp. 1–20.

[18] B. Nojavanasghari, D. Gopinath, J. Koushik, T. Baltrušaitis, and L.-P. Morency, "Deep multimodal fusion for persuasiveness prediction," in *Proc. 18th ACM Int. Conf. Multimodal Interact.* New York, NY, USA: Association for Computing Machinery, Oct. 2016, pp. 284–288.

[19] H. Edwards and A. Storkey, "Towards a neural statistician," 2016, *arXiv:1606.02185*.

[20] S. W. Yoon, J. Seo, and J. Moon, "TapNet: Neural network augmented with task-adaptive projection for few-shot learning," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7115–7123.

[21] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, "No fuss distance metric learning using proxies," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 360–368.

[22] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3630–3638.

[23] F. Pahde, M. Puscas, T. Klein, and M. Nabi, "Multimodal prototypical networks for few-shot learning," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 2643–2652.

[24] X. Chen and G. Wang, "Few-shot learning by integrating spatial and frequency representation," in *Proc. 18th Conf. Robots Vis. (CRV)*, May 2021, pp. 49–56.

[25] Y. Hu, V. Gripon, and S. Pateux, "Leveraging the feature distribution in transfer-based few-shot learning," in *Proc. 30th Artif. Neural Netw. Mach. Learn. (ICANN)*. Bratislava, Slovakia: Springer, Sep. 2021, pp. 487–499.

[26] Y. Hu, S. Pateux, and V. Gripon, "Squeezing backbone feature distributions to the max for efficient few-shot learning," *Algorithms*, vol. 15, no. 5, p. 147, 2022.

[27] D. Shalam and S. Korman, "The self-optimal-transport feature transform," 2022, *arXiv:2204.03065*.

[28] A. Li, T. Luo, T. Xiang, W. Huang, and L. Wang, "Few-shot learning with global class representations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9714–9723.

[29] L. B. Hewitt, M. I. Nye, A. Gane, T. Jaakkola, and J. B. Tenenbaum, "The variational homoencoder: Learning to learn high capacity generative models from few examples," 2018, *arXiv:1807.08919*.

[30] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, "Conditional image generation with PixelCNN decoders," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, Dec. 2016, pp. 4790–4798.

[31] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5219–5227.

[32] Y. Zhu, J. Xie, Z. Tang, X. Peng, and A. Elgammal, "Semantic-guided multi-attention localization for zero-shot learning," in *Proc. 33rd Adv. Neural Inf. Process. Syst.*, 2019, pp. 14943–14953.

[33] W. Xu, Y. Xian, J. Wang, B. Schiele, and Z. Akata, "Attribute prototype network for zero-shot learning," 2020, *arXiv:2008.08290*.

[34] Z. Xiao, X. Xu, H. Xing, F. Song, X. Wang, and B. Zhao, "A federated learning system with enhanced feature extraction for human activity recognition," *Knowl.-Based Syst.*, vol. 229, Oct. 2021, Art. no. 107338.

[35] H. Xing, Z. Xiao, R. Qu, Z. Zhu, and B. Zhao, "An efficient federated distillation learning system for multitask time series classification," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[36] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002.

[37] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD Birds 200," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. 2010.

[38] S. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 49–58.

[39] A. Graves, "Generating sequences with recurrent neural networks," 2013, *arXiv:1308.0850*.

[40] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, Sep. 2019.

[41] J. Gordon, J. Bronskill, M. Bauer, S. Nowozin, and R. E. Turner, "Meta-learning probabilistic inference for prediction," 2018, *arXiv:1805.09921*.

[42] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "The Omniglot challenge: A 3-year progress report," *Current Opinion Behav. Sci.*, vol. 29, p. 97–104, Oct. 2019.

[43] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," 2019, *arXiv:1904.04232*.

[44] Y. Liu, J. Guo, D. Cai, and X. He, "Attribute attention for semantic disambiguation in zero-shot learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6697–6706.

[45] G. Xie, L. Liu, X. Jin, F. Zhu, Z. Zhang, J. Qin, Y. Yao, and L. Shao, "Attentive region embedding network for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9376–9385.

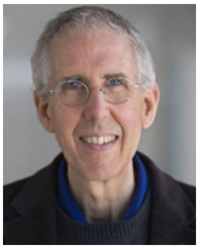
[46] S. Changpinyo, W. Chao, B. Gong, and F. Sha, "Synthesized classifiers for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5327–5336.

[47] V. K. Verma and P. Rai, "A simple exponential family framework for zero-shot learning," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part II 10*. Springer, 2017.

[48] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

MEHMET AKTUKMAK received the B.S. degree in electrical and electronics engineering from Hacettepe University, Ankara, Turkey, in 2009, the M.S. degree in electrical and electronics engineering from Middle East Technical University, Ankara, in 2012, and the Ph.D. degree in electrical engineering from the University of South Florida, Tampa, FL, USA, in 2020. He is currently a Postdoctoral Research Fellow with the Electrical and Computer Engineering Department, University of Michigan, Ann Arbor, MI, USA. His research interests include multimodal-multitask learning, Bayesian modeling, and variational inference, and their applications to image/video processing, matrix completion, meta-learning, and recommender systems.

YASIN YILMAZ (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from Columbia University, New York, NY, USA, in 2014. He is currently an Associate Professor of electrical engineering with the University of South Florida, Tampa, FL, USA. His research interests include machine learning, statistical signal processing, and their applications to computer vision, cybersecurity, biomedical systems, energy systems, transportation systems, and communication systems.



ALFRED O. HERO III (Life Fellow, IEEE) received the B.S. degree (summa cum laude) in electrical engineering from Boston University, in 1980, and the Ph.D. degree in electrical engineering from Princeton University, in 1984.

Since 1984 he has been with the University of Michigan, Ann Arbor, where he is currently the John H. Holland Distinguished University Professor of electrical engineering and computer science and the R. Jamison and Betty Williams Professor of engineering. He was the Founding Co-Director of the University's Michigan Institute for Data Science (MIDAS), from 2015 to 2018. From 2008 to 2013, he held the Digiteo Chaire d'Excellence with Ecole Superieure d'Electricite, Gif-sur-Yvette, France. His primary appointment is with the Department of Electrical Engineering and Computer Science and also has appointments, by courtesy, with the Department of Biomedical Engineering and the Department of Statistics. His current research interests include high-dimensional spatiotemporal data analysis, multimodal data integration, statistical signal processing, machine learning, predictive mathematical models for the biological and physical sciences, social networks, network security and forensics, and personalized health and disease.

Dr. Hero is a fellow of the Society for Industrial and Applied Mathematics (SIAM). From 2011 to 2020, he was a member of the Committee on Applied and Theoretical Statistics (CATS) of the U.S. National Academies of Science, where he was the Chair, from 2017 to 2020. Several of his research articles have received best paper awards. He was awarded the University of Michigan Distinguished Faculty Achievement Award, in 2011, the Stephen S. Attwood Excellence in Engineering Award, in 2017, and the H. Scott Fogler Award for Professional Leadership and Service, in 2018. He received the IEEE Signal Processing Society Meritorious Service Award, in 1998, the IEEE Third Millenium Medal, in 2000, the IEEE Signal Processing Society Technical Achievement Award, in 2014, the Society Award from the IEEE Signal Processing Society, in 2015, and the Fourier Award from the IEEE, in 2020. He was the Co-General Chair of the 2019 IEEE International Symposium on Information Theory (ISIT) and the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing. He was the President of the IEEE Signal Processing Society, from 2006 to 2008. From 2009 to 2011, he was on the IEEE Board of Directors, where he served as the Director of Division IX (Signals and Applications). He is a Section Editor of the *SIAM Journal on Mathematics of Data Science* and a Senior Editor of IEEE JOURNAL ON SELECTED TOPICS IN SIGNAL PROCESSING. He is on the editorial board of the Harvard Data Science Review (HDSR).