4-2-2020

Learning Set Representations for LWIR In-Scene Atmospheric Compensation

Nicholas M. Westing [*]
Air Force Institute of Technology

Kevin C. Gross
Resonant Sciences

Brett J. Borghetti
Air Force Institute of Technology

Jacob A. Martin
Air Force Research Laboratory

Joseph Meola
Air Force Research Laboratory

Recommended Citation

N. Westing, K. C. Gross, B. J. Borghetti, J. Martin and J. Meola, "Learning Set Representations for LWIR In-Scene Atmospheric Compensation," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 1438-1449, 2020, doi: 10.1109/JSTARS.2020.2980750.

Learning Set Representations for LWIR In-Scene Atmospheric Compensation

Nicholas Westing, *Student Member, IEEE*, Kevin C. Gross, Brett J. Borghetti, Jacob Martin, and Joseph Meola

***Abstract*—Atmospheric compensation of long-wave infrared (LWIR) hyperspectral imagery is investigated in this article using set representations learned by a neural network. This approach relies on synthetic at-sensor radiance data derived from collected radiosondes and a diverse database of measured emissivity spectra sampled at a range of surface temperatures. The network loss function relies on LWIR radiative transfer equations to update model parameters. Atmospheric predictions are made on a set of diverse pixels extracted from the scene, without knowledge of blackbody pixels or pixel temperatures. The network architecture utilizes permutation-invariant layers to predict a set representation, similar to the work performed in point cloud classification. When applied to collected hyperspectral image data, this method shows comparable performance to Fast Line-of-Sight Atmospheric Analysis of Hypercubes-Infrared (FLAASH-IR), using an automated pixel selection approach. Additionally, inference time is significantly reduced compared to FLAASH-IR with predictions made on average in 0.24 s on a 128 pixel by 5000 pixel data cube using a mobile graphics card. This computational speed-up on a low-power platform results in an autonomous atmospheric compensation method effective for real-time, onboard use, while only requiring a diversity of materials in the scene.**

***Index Terms*—Atmospheric compensation, dimension reduction, hyperspectral imagery, neural networks.**

I. Introduction

HYPERSPECTRAL sensors continue to improve in both spatial and spectral resolution, allowing for a wide range of applications such as land cover mapping, search and rescue operations, and target detection [1]–[3]. Each hyperspectral image (HSI) pixel contains information sampled across hundreds of narrow spectral channels creating a 3-D data cube: Width by height by spectral channel. HSI data collected in the long-wave infrared (LWIR) region of the electromagnetic spectrum (8–14 μm) contains surface emissivity and temperature information. These measurements are important for atmospheric modeling, climate change studies, and urban heat island analysis [4], [5]. Efficiently and accurately extracting emissivity and temperature information remains a challenging problem. The goal of this article is, with limited assumptions on material content in a scene, to develop and evaluate an efficient method for extracting atmospheric information from a LWIR HSI data cube.

Compared to the reflective region of the electromagnetic spectrum (0.4–2.5 μm), the LWIR domain is dominated by emission of surface materials and atmospheric constituents. The at-sensor radiance $L(\lambda)$ for a Lambertian surface can be described by the simplified LWIR radiative transfer model [3], as follows:

$$L(\lambda) = \tau(\lambda)\left[\epsilon(\lambda)B(\lambda,T) + \left[1-\epsilon(\lambda)\right]L_d(\lambda)\right] + L_a(\lambda) \quad (1)$$

where

λ	wavelength;
T	material temperature;
$\tau(\lambda)$	atmospheric transmission;
$\epsilon(\lambda)$	material emissivity;
$B(\lambda, T)$	Planckian distribution;
$L_d(\lambda)$	downwelling atmospheric radiance;
$L_a(\lambda)$	atmospheric path (upwelling) radiance.

The Planckian distribution is as follows:

$$B(\lambda,T) = \frac{2hc^2}{\lambda^5}\frac{1}{e^{hc/\lambda kT}-1} \quad (2)$$

where c is the speed of light, k is Boltzmann's constant, and h is Planck's constant. Transmittance and path radiance are spectrally varying quantities which depend on the spatially varying temperature and constituent concentrations in the atmosphere [6]. Water vapor, ozone, and carbon dioxide are among the most important infrared-active gases affecting the remotely sensed spectrum. Path radiance represents atmospheric emission directly into the sensor line of sight, while downwelling radiance represents atmospheric emission toward the surface. Assuming a Lambertian surface, downwelling radiance is a cosine-weighted average over the entire hemisphere above the surface. Downwelling radiance reflected off the surface enters the sensor line of sight, requiring reflective materials to estimate this term.

Emissivity retrieval from $L(\lambda)$ can be divided into two steps: Atmospheric compensation and temperature/emissivity separation (TES). Atmospheric compensation methods estimate the transmittance, upwelling radiance, and downwelling radiance (TUD) $(\tau(\lambda), L_a(\lambda), L_d(\lambda))$ vector allowing estimation of the

Manuscript received January 13, 2020; revised February 24, 2020; accepted March 7, 2020. Date of publication April 2, 2020; date of current version April 23, 2020. *(Corresponding author: Nicholas Westing.)*

Nicholas Westing and Brett J. Borghetti are with the Department of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, OH 45433 USA (e-mail: nicholas.westing@afit.edu; brett.borghetti@afit.edu).

Kevin C. Gross is with Resonant Sciences, Beavercreek, OH 45430, USA and also with the Adjunct Faculty in the Department of Engineering Physics at the Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, OH 45433 USA (e-mail: kevin.gross@resonantsciences.com).

Jacob Martin and Joseph Meola are with the EO Detection and Surveillance Branch, Sensor Directorate, Air Force Research Laboratory, Wright-Patterson Air Force Base, Dayton, OH 45433 USA (e-mail: jacob.martin.12@us.af.mil; joseph.meola.1@us.af.mil).

Digital Object Identifier 10.1109/JSTARS.2020.2980750

surface-leaving radiance $L_s(\lambda)$

$$L_s(\lambda) = \frac{L(\lambda) - L_a(\lambda)}{\tau(\lambda)} = \epsilon(\lambda)B(\lambda, T) + [1 - \epsilon(\lambda)]\, L_d(\lambda). \tag{3}$$

Next, TES algorithms are applied to simultaneously estimate $\epsilon(\lambda)$ and T. Separating these terms is complicated by their coupling in the emissive portion of the surface-leaving radiance. For a sensor with K spectral bands, estimating $\epsilon(\lambda)$ and T is an underdetermined problem as there are $K+1$ unknowns $(\epsilon(\lambda), T)$ and only K observed radiance values. TES algorithms apply constraints to $\epsilon(\lambda)$ making the estimation problem more tractable. Typically, $\epsilon(\lambda)$ is an assumed smoother function of wavelength than the observed atmospheric features [7]. Additionally, if the downwelling radiance can be estimated, emissivity can be expressed as follows [8]:

$$\epsilon(\lambda) = \frac{L_s(\lambda) - L_d(\lambda)}{B(\lambda, \hat{T}) - L_d(\lambda)} \tag{4}$$

where $\hat{T}$ is the estimated pixel temperature. Pixel temperature is determined by minimizing atmospheric features in the estimated emissivity profile, resulting in a smoother spectral emissivity. More recent methods such as subspace-based TES [9] project the original data to a lower dimensional subspace to determine maximum-likelihood estimates of both temperature and emissivity.

This study presents a new method for in-scene LWIR atmospheric compensation using a neural network approach with minimal assumptions on scene material content, while efficiently producing comparable results to other compensation methods on collected HSI data. The DeepSets network introduced in [10] is the basis of our approach and so our method is named DeepSet Atmospheric Compensation (DAC). The DAC algorithm relies on a nonlinear TUD vector dimension reduction performed using an autoencoder (AE) network to reconstruct spectrally resolved TUD vectors. This low-dimensional TUD representation is utilized with permutation-invariant neural network layers to fully reconstruct the TUD vector for a given set of pixels. No blackbody pixel assumptions are made and pixel temperature estimates are not necessary to predict the underlying TUD vector. In the next section, a review of various atmospheric compensation methods is presented, highlighting differences between previous methods and our new compensation approach.

A. Atmospheric Compensation Methods

Atmospheric compensation algorithms can be divided into two paradigms: Model-based methods and in-scene methods. Radiative transfer models such as MODerate resolution atmospheric TRANsmission (MODTRAN) support model-based compensation methods using the known or estimated atmospheric state information (column water vapor, trace gas content) to calculate the TUD vector in (1) [11], [12]. Model-based methods are computationally more expensive than in-scene methods, but they can be implemented efficiently if lookup tables encompassing expected conditions are computed before collecting data [13].

One model-based approach considered in this study is Fast Line-of-Sight Atmospheric Analysis of Hypercubes-Infrared (FLAASH-IR), which retrieves scene atmospheric parameters based on a lookup table of precomputed TUD vectors from MODTRAN [11]. The lookup table is generated by varying atmospheric surface temperature, water vapor column density, and an ozone scaling factor. Typically, 10–20 pixels must be selected consisting of varying brightness and emissivity profiles. High reflectivity materials are useful for downwelling radiance estimation and should be included in pixel selection. Mean-squared error between the observed radiance and predicted radiance is minimized by varying surface temperature and atmospheric scaling parameters to recover the TUD vector. As will be shown later, our approach also benefits when reflective materials are present in the scene but selects these materials automatically through a spectral angle measurement.

Another LWIR atmospheric compensation approach utilizing a MODTRAN lookup table and a coupled subspace model is presented in [14]. This approach utilizes singular value decomposition (SVD) to form basis matrices of transmittance and upwelling radiance. Blackbody pixels are identified using the basis matrices to retrieve surface leaving radiance. The DAC algorithm utilizes an AE model to perform dimension reduction on the TUD vectors, similar to the SVD approach employed in [14]. Nonlinear dimension reduction using an AE model allows for lower reconstruction error compared to linear approaches but requires additional training data to properly fit the network weights.

In-scene methods typically do not rely on lookup tables to estimate atmospheric parameters, but some material information is required to make the atmospheric compensation problem tractable. One of the most common approaches is the in-scene atmospheric compensation method that first identifies blackbody pixels $(\epsilon(\lambda) \approx 1)$, where at-sensor radiance can be described by [15]

$$L(\lambda) = \tau(\lambda)B(\lambda, T) + L_a(\lambda). \tag{5}$$

A linear fit is performed on each spectral channel to estimate $\tau(\lambda)$ and $L_a(\lambda)$. Each pixel temperature must be determined prior to this fitting procedure. Temperature estimates are made in the most transmissive spectral bands but can be systematically biased. Water absorption features near 11.73 μm are used to reduce biases introduced by inaccurate surface temperature estimates. Treating $\tau(\lambda)$ and $L_a(\lambda)$ as independent fit parameters, when in fact they are strongly correlated, can also exaggerate fit errors [16].

II. Methodology

Training the DAC algorithm requires a library of worldwide atmospheric measurements, forward modeled with MODTRAN, forming a training set of diverse TUD vectors. Additionally, a low-dimensional representation of these TUD vectors is used to reduce model fitting complexity [17]. Next, the TUD vector dimension reduction process is reviewed and how this method fuses with DAC is explained.

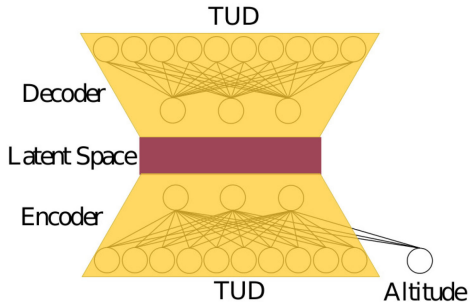

Fig. 1. TUD vectors are compressed by the encoder into the latent space and then reconstructed by the decoder network. Reconstruction error is minimized through weight updates during the training process. Additionally, a scalar altitude input is also presented with the TUD vector allowing the model to scale to multiple altitudes.

A. TUD Vector Dimension Reduction

The Thermodynamic Initial Guess Retrieval (TIGR) database is derived from over 80 000 radiosonde measurements collected worldwide and consists of 2311 atmospheric temperature, water vapor, and ozone measurements on a fixed pressure grid [18], [19]. These measurements are filtered for cloud-free conditions with 96% relative humidity threshold reducing the number of atmospheric measurements to 1755.

To increase the number of training samples used for neural network fitting, a data augmentation strategy is employed on the remaining 1755 atmospheric measurements. First, principal component analysis (PCA) is applied to the measurements using 15 components for each air mass category (Tropical, Polar, etc.). Reconstructing low altitude atmospheric measurements is weighted more heavily in the PCA fitting process since these measurements have a greater impact on the resulting TUD vector. A Gaussian mixture model (GMM) is fit to the 15-dimensional space and sampling this GMM creates new measurement components that are transformed back into pressure level measurements. After filtering these measurements for relative humidity, they are included in the training data.

All atmospheric measurements are forward modeled using MODTRAN 6.0 to create high-resolution TUD vectors based on a NADIR sensor zenith angle. In this study, the Spatially Enhanced Broadband Array Spectrograph System (SEBASS) instrument line shape (ILS) is applied to the high-resolution output, creating sensor-specific TUD vectors [20]. Training, validation, and test data partitioning is based on total optical depth of the original TIGR measurements resulting in 161 validation TUD vectors and 179 test TUD vectors. Combining the remaining TIGR samples with the augmented data results in 8450 training TUD vectors. Each training sample is considered at 17 different altitudes (0.15–3.05 km), leading to a training set of 143 650 TUD vectors. This altitude range was used because previously collected SEBASS data spanned these altitudes. Validation and test samples are generated at altitudes not considered in the training set.

Similar to the work performed in [17], a low-dimensional representation of the generated TUD vector library is created using an AE network (Fig. 1). An AE consists of two networks, an encoder and decoder, to perform nonlinear data compression. The encoder transforms input data $\mathbf{y} \in \mathbb{R}^d$ to the latent space representation $\mathbf{z} \in \mathbb{R}^l$ where $l \ll d$. The decoder reconstructs the input from $\mathbf{z}$ to produce $\hat{\mathbf{y}}$ and weights are updated based on the error between $\mathbf{y}$ and $\hat{\mathbf{y}}$. The entire AE data transformation can be expressed with

$$\begin{aligned} \mathbf{z} &= f(\mathbf{W}_z\mathbf{y} + \mathbf{b}_z) \\ \hat{\mathbf{y}} &= f(\mathbf{W}_y\mathbf{z} + \mathbf{b}_y) \end{aligned} \tag{6}$$

where $\mathbf{W}_z$ and $\mathbf{W}_y$ are the encoder and decoder weight matrices, respectively. Additionally, a bias term is also used at each node, represented by $\mathbf{b}_z$ and $\mathbf{b}_y$. The function f is a nonlinear transform used throughout the network. In this work, we consider two activation functions: The leaky rectified linear unit (RELU) and the exponential linear unit (ELU) described by

$$\text{Leaky RELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha x, & \text{if } x \leq 0 \end{cases}$$

$$\text{ELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(\exp(x) - 1), & \text{if } x \leq 0 \end{cases}$$

where α allows information to flow through the network when the activation function output is negative.

Networks weights are updated using their individual contribution to overall network error, measured by the loss function. The loss function, which was presented previously [17], features both a standard reconstruction error term as well as an at-sensor apparent radiance error

$$\begin{aligned} \mathscr{L}(\hat{\mathbf{y}}, \mathbf{y}) = &\frac{1}{3K}\sum_{i=1}^{3K}(\hat{y}_i - y_i)^2 \\ &+ \frac{\gamma}{MK}\sum_{j=1}^{M}\sum_{i=1}^{K}\left(L_{\hat{\mathbf{y}}}(\lambda_i, \epsilon_j) - L_{\mathbf{y}}(\lambda_i, \epsilon_j)\right)^2. \end{aligned} \tag{7}$$

Here, $\mathbf{y}$ is the truth TUD vector, $\hat{\mathbf{y}}$ is the reconstructed vector, K is the number of spectral channels, and $L_{\hat{\mathbf{y}}}(\lambda_i, \epsilon_j)$ and $L_{\mathbf{y}}(\lambda_i, \epsilon_j)$ are the at-sensor radiance values for the vectors $\hat{\mathbf{y}}$ and $\mathbf{y}$. Additionally, a linear sampling of M gray-body emissivity values between 0 and 1 are used to calculate this loss term, improving reconstruction error for reflective and emissive materials [17].

The hyperparameter γ controls the relative importance between the contribution of the TUD mean-squared error and the at-sensor radiance mean-squared error within the loss function and is set to $\gamma = 1$ in this study. As γ approaches zero, more emphasis is placed on TUD mean-squared error resulting in higher reconstruction error for reflective materials as shown in [17]. Similarly, when $\gamma > 1$, more emphasis is placed on at-sensor radiance error. The TUD mean-squared error term is necessary to stabilize training, and in our experience, increasing γ to a large value can lead to unstable AE training.

The TUD AE presented here differs from the work of [17] because sensor altitude is also included in the model. This scalar input allows the AE to correctly reconstruct TUD vectors at a range of altitudes making the model more applicable to real-world scenarios where sensor altitude varies. For sensors

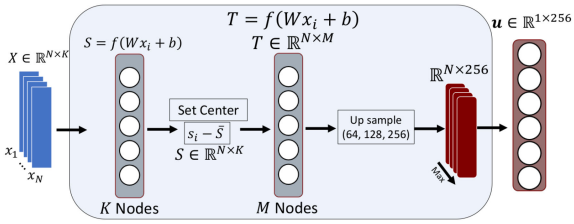

Fig. 2. Permutation-invariant transform $\phi(\mathbf{X})$ used in this study consists of a neural network applied to all pixels in the set X followed by a centering operation of the transformed pixel representations. The set S is transformed by a dense layer and upsampled. The $N \times 256$ set representation is collapsed into a single permutation-invariant set representation through a max operation.

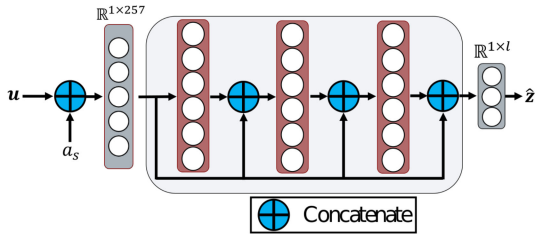

Fig. 3. $\rho(\cdot)$ network is shown highlighting the use of skip connections to propagate the set representation to deeper layers. The input to this network is the result of the permutation-invariant set extraction concatenated with the sensor altitude, a_s. Each layer inside the network block contains 50 nodes and the predicted latent space contains 4 components.

operating at a constant altitude, this model can be retooled to consider a small range of altitudes in the sensor's operating range or a single altitude can be considered as was previously demonstrated in [17]. Validation and test sets consist of hold out samples where neither the atmospheric state nor the sensor altitude were observed in the training set. Performance on validation and test sets explains the network's ability to generalize to new samples or highlights models that are overfit to the training data.

1) Autoencoder Metrics: TUD vector reconstruction error must be placed in context of the overall remote sensing goal to select AE models with the best performance. Predicted at-sensor radiance is calculated using the predicted TUD vector, a range of gray-body emissivity values, and an assumed pixel temperature. Since this study is focused on the LWIR domain, spectral radiance values were transformed to brightness temperature $T_{BB}(\lambda)$ for conveniently representing model errors. $T_{BB}(\lambda)$ is computed by inverting Planck's function, as follows:

$$T_{BB}(\lambda) = \frac{hc}{\lambda k \ln\left(\frac{2hc^2}{\lambda^5 L(\lambda)} + 1\right)}. \tag{8}$$

The root mean square error (RMSE) in Kelvin can be calculated with

$$E_t = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(T_{BB}(\lambda_i) - \hat{T}_{BB}(\lambda_i)\right)^2} \tag{9}$$

where index t corresponds to a test gray body $\epsilon_t(\lambda)$ used in (1) to compute $L(\lambda)$. The test emissivity values range from 0 to 1 producing an RMSE describing overall performance between reflective and emissive materials for the AE model. Next, the entire in-scene atmospheric compensation method is introduced, utilizing the fit AE model.

B. In-Scene Atmospheric Compensation

Numerous methods are available for LWIR in-scene atmospheric compensation, typically relying on the selection of blackbody pixels to determine $\tau(\lambda)$ and $L_a(\lambda)$. Rather than following this paradigm, the DAC algorithm relies on an automated selection of diverse pixels, without the knowledge of $\epsilon(\lambda)$ or pixel temperature to estimate atmospheric effects on LWIR HSI data.

Given N diverse pixels $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, extracted from a data cube collected across K bands, $\mathbf{x}_i \in \mathbb{R}^K$, the DAC network, $D(\mathbf{X})$, must predict the AE low-dimensional representation $\mathbf{z}$. Since this function operates over a set of extracted pixels, $D(\mathbf{X})$ must be permutation-invariant to the pixel selection order. Additionally, this network must also provide similar predictions as the number of pixels varies within the set $\mathbf{X}$. After predicting $\mathbf{z}$, the previously trained decoder network $d(\cdot)$ can be used to reconstruct the full TUD vector. Since the TUD vector is the same for all pixels in the scene, the DAC network will utilize information from multiple pixels at once to estimate the latent space $\mathbf{z}$. This class of problems is referred to as set-input learning where a single target corresponds to a set of input samples [10], [21], [22]. Recently, new network architectures have been investigated for domains such as point cloud classification, anomaly detection, and image tagging to address set-input learning, referred to as DeepSets or PointNet for point cloud classification [10], [23]. These architectures utilize a permutation-invariant function $\phi(\cdot)$ to extract a set feature vector. This operation can be broken down into two steps: Set transformation and set decomposition. Set transformation is performed by $\phi(\cdot)$ with U output nodes to produce the set $\mathbf{V}$ as follows:

$$\mathbf{V} = \phi(\mathbf{X}) \quad \mathbf{V} \in \mathbb{R}^{N\times U}. \tag{10}$$

Next, set decomposition can be performed using any permutation-invariant function. In this study, the maximum value is taken across the N pixels resulting in the set feature vector $\mathbf{u}$ as follows:

$$\mathbf{u}_j = \max_{i\in N} \mathbf{V}_{ij} \quad \mathbf{u} \in \mathbb{R}^{1\times U}. \tag{11}$$

The entire permutation-invariant function can be written as follows:

$$D(\mathbf{X}) = \rho\left(\max_{i\in N} \phi(\mathbf{X})\right) \tag{12}$$

where ρ is another transformation (neural network) applied to the set feature vector $\mathbf{u}$. Additionally, $\phi(\mathbf{X})$ can be applied to any number of N inputs since the max operator pools all N samples into a set feature vector.

The max decomposition of $\phi(\mathbf{X})$ is shown in Fig. 2. First, a K node neural network layer transforms the at-sensor radiance pixels into the set $\mathbf{S}$. Next, $\mathbf{S}$ is centered to encode the overall set information in the learned representation improving

convergence during training. This operation is similar to batch normalization [24]; however, we apply this normalization across the transformed set, rather than the batch. After set centering, a layer containing M nodes is used to transform the centered representation before upsampling. The number of nodes M is a hyperparameter we vary during model evaluation. Upsampling layers are required to provide enough information in the set feature vector for the following $\rho(\cdot)$ network to predict the target value. The upsampling layers create the set $\mathbf{V} \in \mathbb{R}^{N\times 256}$.

The $\rho(\cdot)$ network predicts the low-dimensional TUD representation $\hat{\mathbf{z}}$ using the set information extracted by the max decomposition. The previously fit decoder network $d(\cdot)$ returns the fully spectrally resolved TUD vector such that $(\hat{\tau}(\lambda), \hat{L}_a(\lambda), \hat{L}_d(\lambda)) = d(\hat{\mathbf{z}})$.

Predictions made by $\rho(\cdot)$ are altitude dependent. To include this information in $D(\mathbf{X})$, the sensor altitude a_s is concatenated to the input of $\rho(\cdot)$ (Fig. 3). This allows $\rho(\cdot)$ to modify its low-dimensional prediction $\hat{\mathbf{z}}$ to changes in altitude, ultimately making the model more applicable for real-world conditions. The $\rho(\cdot)$ network shown in Fig. 3 makes extensive use of skip connections, allowing the extracted 1×257 vector to propagate to deeper network layers. Finally, combining the max decomposition and $\rho(\cdot)$ networks, the DAC algorithm can be expressed as follows:

$$D(\mathbf{X}, a_s) = \rho\left(\max_{i\in N}\left[\phi(\mathbf{X})\right], a_s\right). \tag{13}$$

To identify the best network architecture, hyperparameter sweeps were performed on the number of nodes M in $\phi(\mathbf{X})$ and the number of nodes per layer in $\rho(\cdot)$. Additionally, batch size, learning rates, and activation functions were also varied. The network architecture used in these study sets $M = 90$ in the $\phi(\mathbf{X})$ network utilizing the ELU activation function. Additionally, the $\rho(\cdot)$ network contains three layers containing 50 nodes, all using the ELU activation function. The network was trained with $N = 50$; however, this can be varied during model evaluation since the max decomposition is along the pixel axis. A learning rate of 0.001 and a batch size of 64 was used to fit network weights. The Adam optimization algorithm was used for calculating weight updates [25]. Networks were constructed using Python 3.6.8, Keras version 2.2.4, Tensorflow 1.15, and hyperparameter sweeps were conducted across 20 graphic processing units (GPUs) using Ray Tune version 0.7.6 [26], [27]. Since the AE and DAC models only use 132 MB of memory, multiple models can be trained in parallel on a single GPU. The model contains 109 026 weights that are fit through the training process discussed next.

C. Algorithm Training

Each training example consists of a set of at-sensor radiance spectra $\mathbf{X}$ and the low-dimensional TUD representation $\mathbf{z}$ generated by the encoder network. Creating the at-sensor radiance spectra requires a library of emissivity spectra, TUD vectors, and a method for assigning pixel temperatures.

Pixel emissivity spectra are selected from the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) database and downsampled using the SEBASS ILS. To verify model performance on new data, 200 emissivity spectra are held out leaving 978 profiles for training. The held out emissivity spectra contain a range of reflective and emissive materials to evaluate model performance.

Algorithm 1: Set Generation.

Input: ϵ, $\boldsymbol{\tau}$, $\boldsymbol{L_a}$, $\boldsymbol{L_d}$, t_o, N
Output: L
Emissivity Selection :
$\epsilon_t = \mathrm{U}(0.75, 1.0)$
$\boldsymbol{\epsilon}_f = \boldsymbol{\epsilon}$ s.t. $\bar{\boldsymbol{\epsilon}} < \epsilon_t$
$\boldsymbol{\epsilon}_R = \boldsymbol{\epsilon}_f$ s.t. $\bar{\boldsymbol{\epsilon}}_f < \epsilon_t - 0.10$
$\boldsymbol{\epsilon}_E = \boldsymbol{\epsilon}_f$ s.t. $\bar{\boldsymbol{\epsilon}}_f \geq \epsilon_t - 0.10$
$P_E = \mathrm{U}(0.5, 0.95)$
$N_E = \mathrm{int}(P_E \cdot N)$
$N_R = N - N_E$
$\boldsymbol{\epsilon}_S = [N_R$ samples from $\boldsymbol{\epsilon}_R$, N_E samples from $\boldsymbol{\epsilon}_E]$
At-Sensor Radiance Generator
$w = \mathrm{U}(2, 20)$
for $i = 0$ to N **do**
  $\mathrm{T} = \mathrm{U}(t_o - w, t_o + w)$
  $\boldsymbol{L_s} = \boldsymbol{\epsilon}_S[i]B(T) + (1 - \boldsymbol{\epsilon}_S[i])\boldsymbol{L_d}$
  $\mathrm{L[i]} = \boldsymbol{\tau}\boldsymbol{L_s} + \boldsymbol{L_a}$
end for
return $\boldsymbol{L}$

Selecting the N emissivity spectra for a particular training set begins by dividing the ASTER database into emissive and reflective samples. To model a wide range of scenes, an initial emissivity threshold is calculated, $\epsilon_t \sim \mathrm{U}(0.75, 1.0)$, where U is a uniform distribution. Emissivity spectra with means exceeding this threshold are removed resulting in a filtered emissivity database $\boldsymbol{\epsilon}_f$. The filtered database is divided into emissive samples $\boldsymbol{\epsilon}_E$ and reflective samples $\boldsymbol{\epsilon}_R$ based on a threshold of $\epsilon_t - 0.10$.

Next, the percent of emissive samples P_E in the set N is sampled according to $P_E \sim \mathrm{U}(0.5, 0.95)$. The number of reflective, N_R, and emissive, N_E, materials in the scene are determined by P_E. Emissivity spectra are sampled from the emissive and reflective portions of the filtered ASTER database forming the set emissivity spectra $\boldsymbol{\epsilon}_S$. This process is outlined in Algorithm 1 under emissivity selection, where $\boldsymbol{\epsilon}$ corresponds to the training or validation ASTER database samples. By varying P_E and ϵ_t, training sets dominated by blackbody pixels with little diversity can be created, and highly diverse scenes can be created containing both reflective and emissive materials.

After selecting a set of N emissivity spectra, N pixel temperatures must be assigned to calculate at-sensor radiance. The surface-level temperature measurement t_o from the TIGR data is used to assign pixel temperatures T_i such that $T_i \sim \mathrm{U}(t_o - w, t_o + w)$, where $w \sim \mathrm{U}(2, 20)$. By allowing the T_i uniform distribution width to vary, scenes containing little temperature variation and high temperature variation can be generated. After initializing the emissivity spectra and pixel temperatures, a set of at-sensor radiance values $\mathbf{L}$ are calculated for a single TUD vector as shown by the entire set generation process in Algorithm

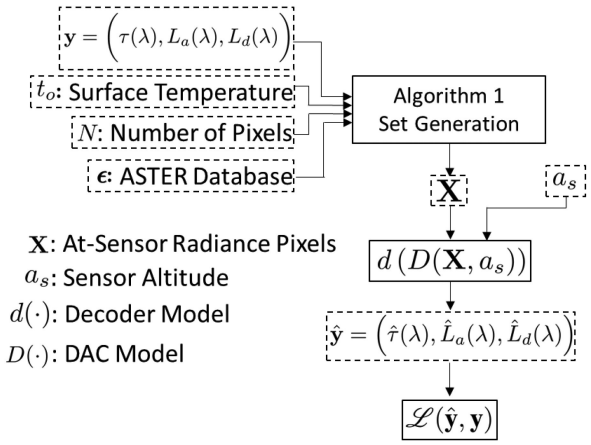

Fig. 4. Entire DAC network training pipeline is shown highlighting the inputs to the set generation algorithm resulting in the at-sensor radiance pixels $\mathbf{X}$. Using the known sensor altitude a_s the DAC model $D(\cdot)$ predicts the low-dimensional TUD representation such that the decoder model $d(\cdot)$ can reconstruct the TUD vector. The loss function $\mathscr{L}(\hat{\mathbf{y}}, \mathbf{y})$ (7) directs weight updates within the DAC network.

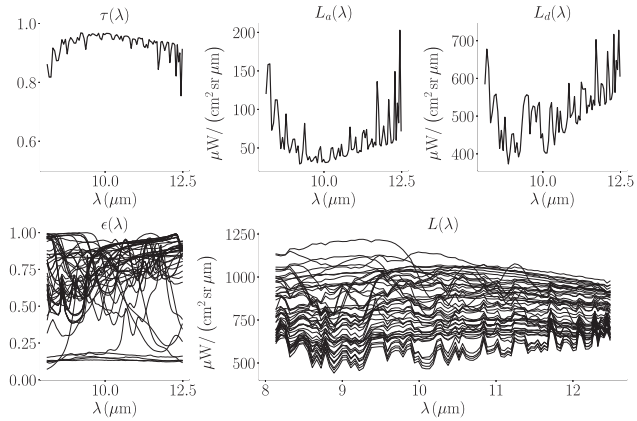

Fig. 5. Example of set $\mathbf{X}$ is shown in the bottom right plot where $N = 50$. The lowest atmospheric temperature measurement was 296 K for the given TUD vector. The emissivity threshold ϵ_t was 0.85, sampled P_E was 0.75, resulting in a mean emissivity less than 0.75 for 25% of the materials and a mean emissivity between 0.75 and 0.85 for 75% of the materials.

1. Fig. 4 shows the entire training process to include the role of the DAC model, decoder model, and overall network loss calculation. Fig. 5 shows the result of this process for 50 sampled emissivity spectra, where $\epsilon_t = 0.85$, $P_E = 0.75$, and the surface temperature was 296 K.

Considering the 978 emissivity training spectra and the set size $N = 50$, the number of possible emissivity training sets is $\binom{978}{50} = 3 \times 10^{84}$. Additionally, each pixel emissivity temperature is randomly sampled following the strategy outlined in Algorithm 1 further increasing the number of training samples for a single TUD vector. Using 8450 TUD vectors sampled at 17 altitudes results in a large training data set to fit the DAC network. Based on the large number of training samples possible, spectral noise was not added to the ASTER emissivity spectra. Similarly for the validation set, the number of emissivity spectra combinations based on 200 hold out emissivity spectra is $\binom{200}{50} = 4.5 \times 10^{47}$, and 161 TUD vectors are considered across two altitudes, none of which were a part of the training data.

While the number of possible training and validation samples is very large, we find that training only requires 150 iterations for network performance to converge. During each training iteration, 50 batches are randomly generated from the training TUD database. Specifically, using a batch size of 64, the set generation algorithm shown in Algorithm 1 is executed 64 times to generate a single training batch. During each training iteration, weight updates are made based on 3200 TUD vectors.

The encoder network of the previously trained AE model is used to map the underlying TUD vector to the low-dimensional representation $\mathbf{z}$. The DAC algorithm predicts $\hat{\mathbf{z}}$ such that the decoder network $d(\cdot)$ can fully reconstruct the TUD vector. The same loss function used in the AE model training (7) is used for the DAC network, $\mathscr{L}(d(\hat{\mathbf{z}}), d(\mathbf{z}))$, where the inputs are the decoder transformed TUD vectors. Again, *M* gray-body emissivity values e_j between 0 and 1 are used to calculate DAC loss, allowing network weight updates to minimize at-sensor radiance error for reflective and emissive materials.

D. Pixel Selection

Applying the trained model to real data requires the selection of N pixels to predict $\hat{\mathbf{z}}$ and ultimately the cube TUD vector. This selection process should be automated, increasing data throughput, while providing reliable results. As shown in Fig. 2, the set $\mathbf{X}$ must contain some pixel diversity to extract a set representation. Specifically, if all N pixels are identical, $\phi(\mathbf{X})$ will converge to zero after centering the set $\mathbf{S}$.

To extract N pixels from a real data cube, the mean at-sensor radiance spectrum $\bar{L}(\lambda)$ is calculated. Next, the spectral angle θ_i between pixel i and $\bar{L}(\lambda)$ is calculated as follows:

$$\theta_i = \cos^{-1}\left(\frac{L_i(\lambda) \cdot \bar{L}(\lambda)}{\|L_i(\lambda)\|\|\bar{L}(\lambda)\|}\right) \tag{14}$$

where $\|\cdot\|$ denotes the l_2 norm and $L_i(\lambda)$ is the at-sensor radiance for pixel i. After sorting all pixels by spectral angle from the mean radiance spectrum, the 10% largest spectral angles are used for pixel selection. First, the lowest spectral angle pixel (90th percentile) is selected and a one-pixel guard band is applied spatially. Any pixels within this guard band are removed from the sorted array and no longer considered for selection. This process is repeated by uniformly sampling N pixels across the sorted spectral angle array. Endmember extraction techniques were investigated for pixel selection but added significant computational overhead without noticeable improvement in algorithm performance. Anomaly detection approaches such as Mahalanobis distance were also considered but did not yield noticeable improvements while also requiring data covariance calculation.

III. Results

This section first presents the AE model results applied to the TIGR data across a range of altitudes. The trained AE model is then used to create the training data samples for fitting the DAC model. After reporting training results for both methods, several

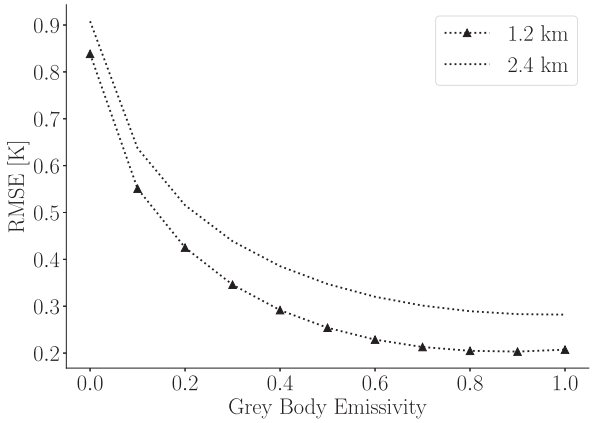

Fig. 6. Brightness temperature RMSE is reported for TUD samples never used in training the AE. The sensor altitudes also were never observed in the training data demonstrating the model's ability to generalize across new altitudes.

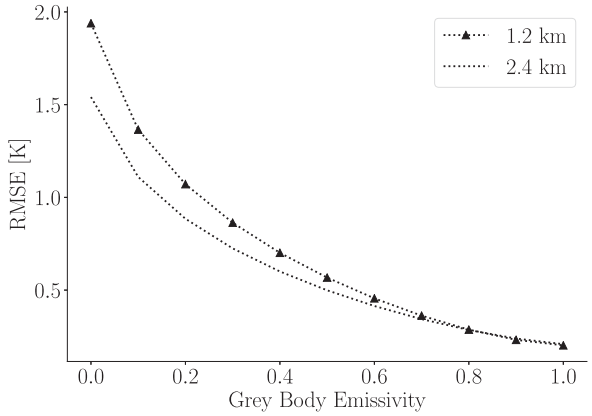

Fig. 7. DAC brightness temperature RMSE for the hold out synthetic data is reported as a function of gray-body emissivity. The hold out data consisted of TUD vectors never observed in the training data. Additionally, these hold out samples were tested at new altitudes not included in the training set showing the model interpolates to new TUD vectors and altitudes.

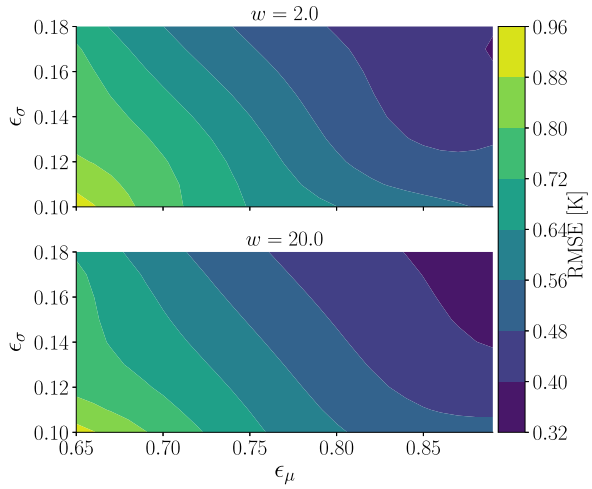

Fig. 8. At-sensor brightness temperature error contours are shown for the DAC model for randomly sampled sets of pixel emissivity spectra with overall mean and standard deviations shown. Here, w is the uniform distribution width for sampling pixel temperatures. Errors decrease with increasingly diverse sets of pixels and increased mean emissivity as expected from the LWIR radiative transfer equation.

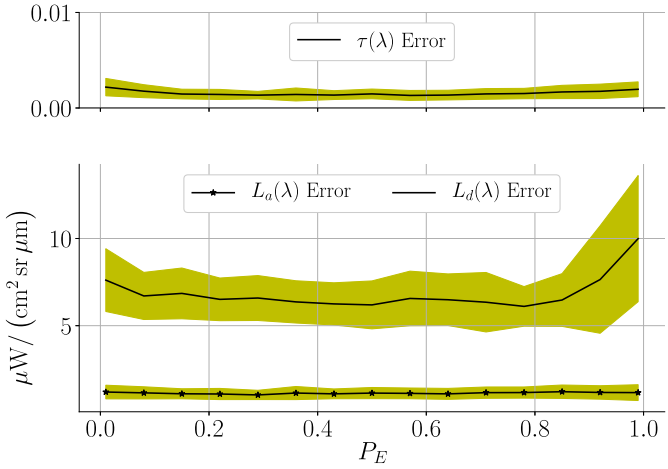

Fig. 9. Increasing the percentage of emissive samples P_E leads to larger error in the model's downwelling prediction. This error is expected from (1) since blackbody materials provide little information to resolve $L_d(\lambda)$. The sets used to generate this plot held $\epsilon_t = 1.0$ in Algorithm 1.

different measured hyperspectral data sets are used to verify whether DAC performance is comparable to FLAASH-IR.

A. Autoencoder Results

The relative humidity filtered TIGR data and augmented samples were used to fit the AE model. A hyperparameter sweep was performed across the number of nodes per layer, number of layers, batch size, learning rate, activation functions, latent components, and loss functions. The network with minimum brightness temperature RMSE on the validation TUD vectors was selected. Additionally, each model was trained ten times starting from a random weight initialization, and the model with the best mean performance was selected as the best overall architecture. The selected model consisted of a two-layer encoder with four latent components: 276-48-16-4 where 276 is the TUD vector dimension and 48-16 are the encoder layer dimensions. The decoder is the reverse order of the encoder: 4-16-48-276. The leaky RELU activation function was used with the at-sensor radiance loss described in (7). Model training executed for 300 iterations, using a batch size of 64 and a learning rate of 1×10^{-4}.

The RMSE in brightness temperature of the AE model is shown in Fig. 6. These results are based on TUD vectors and altitudes not included in the training data, representing model performance when new data is presented. Brightness temperature RMSE increases for lower emissivity (higher reflectivity) materials where errors in transmission and downwelling radiance are multiplied in the simplified LWIR radiative transfer equation. The errors reported in Fig. 6 represent the lowest achievable error of the DAC method since all DAC low-dimensional predictions are transformed through the decoder network.

B. Synthetic Data Results

The DAC network results are shown in Fig. 7 for the validation TUD samples, emissivity profiles, and altitudes. Fig. 7 was formed by transforming validation TUD error to at-sensor

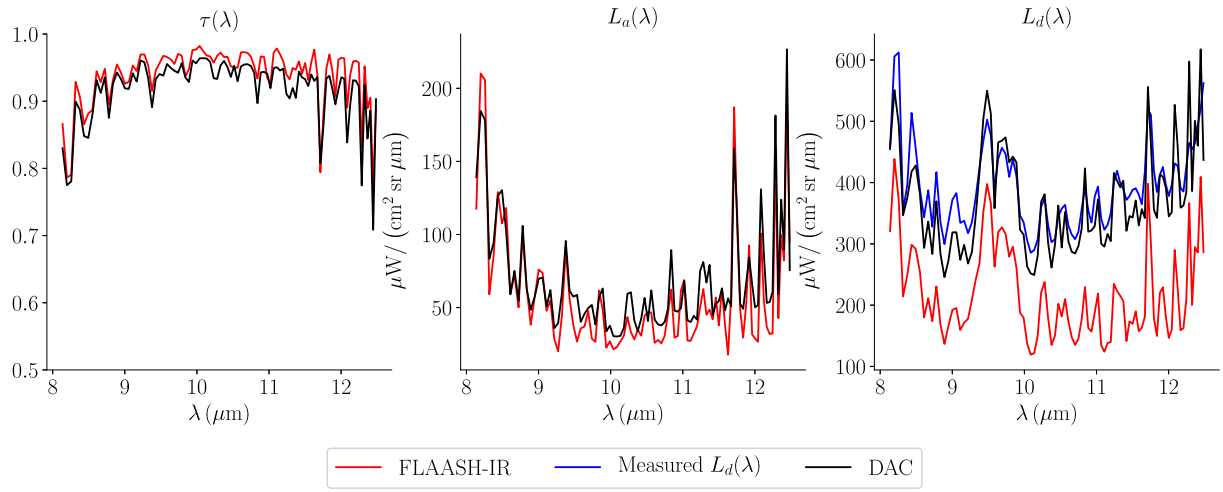

Fig. 10. Real data TUD predictions where close agreement is observed for the atmospheric terms, $\tau(\lambda$ and $L_a(\lambda)$, while larger deviations are seen for $L_d(\lambda)$. This cube was collected at 1856 L from an altitude of 0.45 km under clear sky conditions.

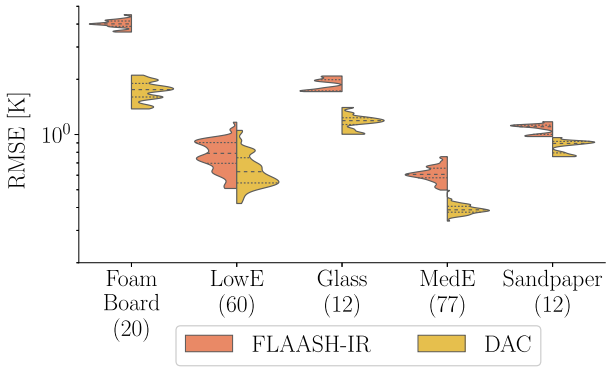

Fig. 11. Brightness temperature error is shown when using the measured pixel emissivity and pixel labels to estimate individual pixel temperatures. The materials are organized from left to right in increasing emissivity. The dashes within each plot are the inner quartile range where the thick dashes are the median error. The number of pixels per material are shown in parentheses.

TABLE I
PREDICTED MATERIAL TEMPERATURES [K] USING PIXEL LABELS TO MINIMIZE EMISSIVITY ERROR

	Foam Board	LowE	Glass	MedE	Sandpaper
Measured	318.27	330.32	321.85	341.79	338.27
FLAASH-IR	335.34	335.81	330.44	344.95	341.15
DAC	330.29	334.17	329.90	344.85	342.12

brightness temperature using the gray-body emissivity shown and the surface temperature measurement t_o from the TIGR data. The performance shown in Fig. 7 demonstrates the DAC the model's ability to generalize while only increasing brightness temperature RMSE by approximately 1 K for reflective and emissive materials. Next, we consider scenes with varying emissivity and temperature statistics to determine how these parameters impact TUD estimation error.

To explore the DAC algorithm's dependency on a diversity of pixels, sets of N pixel sets were randomly sampled from the ASTER database with varying scene statistics. The spectrally averaged emissivity $\bar{\epsilon}_i$ for a selected emissivity ϵ_i measured across K bands is $\bar{\epsilon}_i = \frac{1}{K}\sum_{j=1}^{K}\epsilon_i(\lambda_j)$ and the set mean emissivity of N selected emissivity spectra is $\epsilon_\mu = \frac{1}{N}\sum_{i=1}^{N}\bar{\epsilon}_i$. Additionally, the set standard deviation ϵ_σ is calculated according to the following:

$$\epsilon_\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\epsilon_\mu - \bar{\epsilon}_i}. \tag{15}$$

The set mean and standard deviation were calculated for each randomly sampled set. Sampling continued until a range of set means and standard deviations were recorded. The standard deviation represents the diversity of pixels within the scene, while the set mean corresponds to reflective versus emissive scenes.

These N pixel sets were used with varying temperature distributions to determine DAC error. Fig. 8 shows the at-sensor error in brightness temperature as a function of set mean and standard deviation for the TUD validation set. The errors were calculated using the sampled emissivity spectra and sampled pixel temperatures where $T_i = \mathrm{U}(t_o - w, t_o + w)$. The two plots in Fig. 8 correspond to different temperature uniform distribution widths w.

Errors decrease with increased mean emissivity because errors in downwelling radiance play a less significant role as emissivity approaches 1.0. Additionally, errors are also reduced as the pixel diversity increases within a scene, supporting that DAC relies on diverse pixels to estimate the TUD vector. These trends are consistent for low temperature variance ($w = 2$) and high temperature variance ($w = 20$), and the overall error is lower as temperature variance increases. Additionally, a portion of the error shown in Fig. 8 is derived from the AE errors shown in Fig. 6. This is because all DAC predictions are transformed through the AE decoder network. The errors shown in Fig. 8 represent at-sensor error but do not fully explain individually how $\tau(\lambda)$, $L_a(\lambda)$ and $L_d(\lambda)$ errors vary. Next, additional N pixel sets are created to further identify trends in these errors.

From the simplified LWIR radiative transfer equation, it is expected that downwelling radiance prediction error will

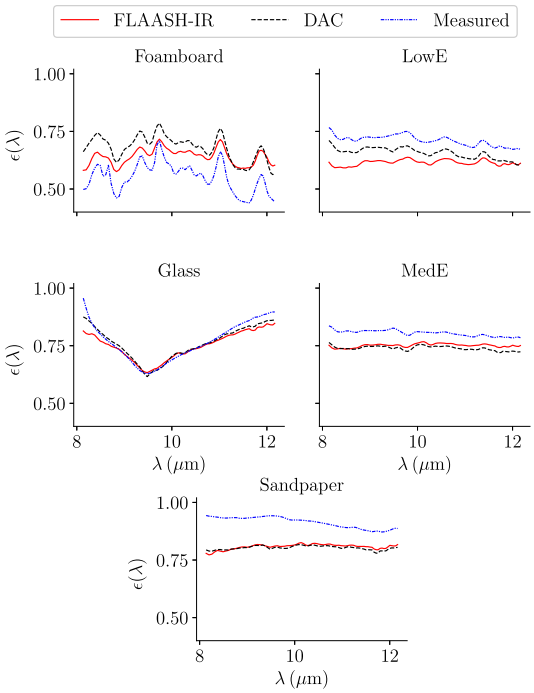

Fig. 12. Predicted emissivity curves are shown for both FLAASH-IR and DAC. Emissivity estimates were made using the maximum-smoothness TES technique [7].

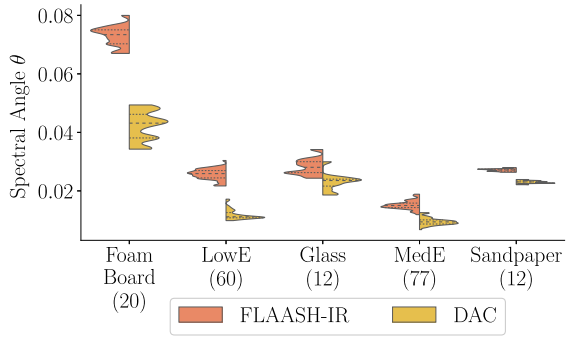

Fig. 13. Brightness temperature errors are shown between the two methods where TES was used with each TUD prediction to determine $\epsilon(\lambda)$ and T. These estimates were forward modeled to determine the at-sensor radiance. Comparable performance is observed, but the DAC method operates under 1 s including automated pixel selection.

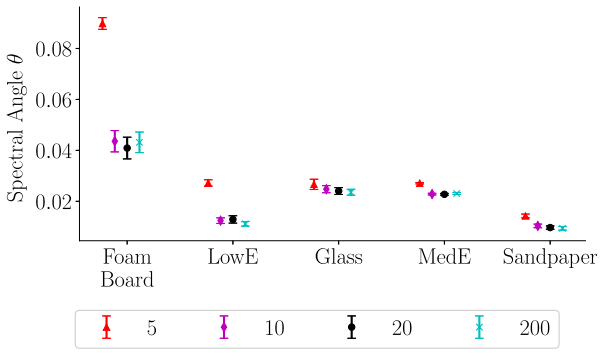

Fig. 14. Varying the input set size N between 5 and 200 pixels and performing TES using the predicted TUD vectors demonstrates how set size impacts the overall spectral angle error. For this cube, 5 pixels is not adequate to correctly predict the TUD vector, but increasing to 10 pixels captures the necessary diversity in the data.

increase when reflective materials are not present in the scene. To verify this observation with the DAC model, synthetic data sets were created containing an increasingly higher percentage of emissive materials P_E. Additionally, the emissivity threshold in Algorithm 1 was set to 1.0 for set generation. As shown in Fig. 9, the downwelling radiance error increases significantly when scenes consist of materials with a mean emissivity greater than 0.9. Also, transmittance and upwelling radiance errors are unaffected by scenes consisting of nearly all blackbody pixels as expected from (1). Next, the trained model is applied to collected data cubes to evaluate atmospheric compensation performance in a real-world scenario.

C. Real HSI Data Results

This study uses the same data cubes reported in [28] and [29], collected at altitudes ranging from 0.45 to 2.7 km with the SEBASS LWIR imager. First, we consider a 128 by 5000 pixel cube collected at 0.45 km under clear sky conditions. The collected data contains varying size material panels at different tilt angles and surface roughness. Only flat panels within the scene are considered to evaluate downwelling radiance prediction accuracy. The labeled materials are: Foam Board, Low Emissivity Panel (LowE), Glass, Medium Emissivity Panel (MedE), and Sandpaper. The ground truth emissivity for each material was measured with a D&P spectrometer. Measured emissivity spectra are shown in Fig. 12 with emissivity predictions using TES to be discussed later. Additionally, downwelling radiance was also measured with a D&P spectrometer by measuring radiance from an infragold sample.

The first hyperspectral data cube considered was collected at 1856 L from an altitude of 0.45 km under clear sky conditions. Predictions from DAC and FLAASH-IR are shown in Fig. 10, where the largest difference is in the downwelling radiance component. The DAC $L_d(\lambda)$ prediction closely aligns with the D&P spectrometer measurement demonstrating the ability of this method to extract information from reflective pixels in the scene. Truth data for $\tau(\lambda)$ and $L_a(\lambda)$ are not available; however, these predictions are considered in the total at-sensor radiance error discussed next.

Using the pixel labels to assume a known emissivity, at-sensor radiance error can be calculated if pixel temperatures can be estimated. Pixel temperature T_i is determined by minimizing the mean-squared error between the measured and predicted emissivity

$$\min_{T_i} \frac{1}{K} \sum_{j=1}^{K} \left(\epsilon(\lambda_j) - \hat{\epsilon_i}(\lambda_j)\right)^2 \tag{16}$$

where the predicted emissivity is also a function of pixel temperature

$$\hat{\epsilon}_i(\lambda)(T_i) = \frac{L_i(\lambda) - \hat{L_a}(\lambda) - \hat{\tau}(\lambda)\hat{L_d}(\lambda)}{\hat{\tau}(\lambda)\left[B(\lambda, T_i) - \hat{L_d}(\lambda)\right]}. \tag{17}$$

Radiance predictions are made using the estimated TUD, estimated pixel temperature, and measured emissivity. Fig. 11 shows the resulting errors in brightness temperature where materials

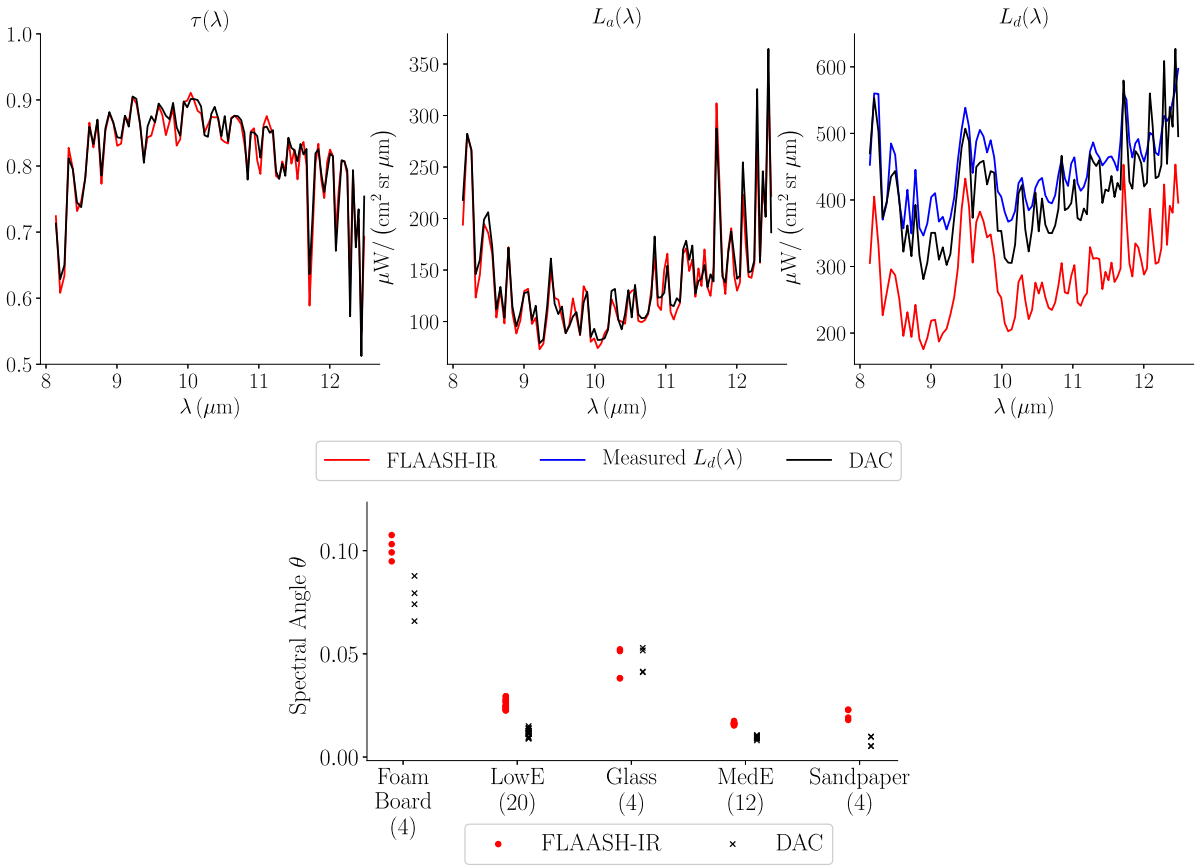

Fig. 15. Applying DAC and FLAASH-IR to a data cube collected at 1.2 Km shows good agreement between the two approaches. This data cube was collected at a different time of day from the cube results reported in Figs. 10 and 13. The spectral angle errors are based on applying maximum-smoothness TES [7] using the predicted TUD vectors. This cube was collected at 1638 L under clear sky conditions 5 d after the previous cube was collected. Violin plots are not shown because the number of pixels per material is significantly smaller at this altitude.

are organized in increasing emissivity from left to right. The number of pixels per material is shown in parentheses and violin plots are used to display the distribution of errors. For this data cube, the DAC predictions result in lower error across a range of material emissivity spectra, with only small improvements for the highest emissivity material, sandpaper. A log-scale is used in Fig. 11 to highlight differences in LowE and MedE errors that approach 0.4 K. Additionally, the estimated temperatures from each compensation method are shown in Table I with close agreement observed between both methods.

The previous results used the known pixel emissivity to estimate pixel temperature; however, this is unrealistic in real-world conditions since pixel emissivity is not known beforehand. Next, TES is applied to the HSI data to compare compensation performance. A total of 2048 temperatures between 280 and 350 K are considered to maximize the smoothness of the estimated emissivity spectra with a seven-point local averaging filter based on the method presented in [7]. The mean TES estimated emissivity spectra are shown in Fig. 12, where both DAC and FLAASH-IR provide similar estimates. The FLAASH-IR estimates are derived from the TES method described in [7] to compare TUD predictions, rather than using the reported emissivity within the FLAASH-IR software.

It is important to note the estimated ILS agreed closely with the observed measurements. Errors in ILS estimation will impact the DAC algorithm just as they would any other atmospheric compensation approach. An area of future work to allow DAC to efficiently operate over a time-varying ILS is to train the entire model on higher resolution TUD vectors and downsample predictions to the current ILS estimate.

Numerous target detection algorithms exist differing in background clutter modeling, subpixel replacement strategies, and detection statistic calculation. For many algorithms, detection statistics are based on a spectral angle measurement between a known emissivity measurement and the extracted emissivity from TES. Using the TES emissivity estimates, spectral angles are calculated using the measured emissivity spectra with spectral angle error θ shown in Fig. 13. Lower spectral angle errors for the DAC algorithm are observed across all materials supporting the utility of this approach for target detection scenarios.

The $\phi(\mathbf{X})$ network can use any number of pixels since the max decomposition is performed along the pixel axis. Varying the set size from 5 to 200 pixels and calculating the spectral angle error after conducting TES is shown in Fig. 14. Using only 5 pixels does not contain enough information to accurately predict the scene TUD vector leading to higher spectral angle error. The model was trained using sets of 50 pixels; however, from Fig. 14, only 10 pixels are needed for this cube. For more diverse scenes such as urban areas, additional pixels are expected to further improve TUD prediction accuracy.

Next, a data cube collected at an altitude of 1.2 km is considered to demonstrate DAC performance at a new altitude. This cube was collected in the afternoon 5 d after the previous cube collection. Weather was noted as clear sky conditions during this collection. Fig. 15 shows the predicted TUD vector and the resulting spectral angle error after applying TES. While radiosonde data is not available to compare atmospheric state vectors between the data cubes, a significant change in $\hat{\tau}(\lambda)$ and $\hat{L_a}(\lambda)$ is noted between the collects.

Finally, inference time is another important factor to consider when deploying these methods in real-world scenarios. The DAC algorithm benefits from accelerated computation using a graphics card; however, to compare inference time between FLAASH-IR and DAC, both methods were tested on an Intel i7-4710MQ processor. Inference time for DAC was on average 0.35 s, while FLAASH-IR took approximately 67 s not including lookup table generation. Running DAC on an Nvidia RTX 2060 mobile graphics card reduced inference time to 0.24 s. The DAC inference times include automatic pixel selection using the spectral angle method detailed in (14).

IV. Conclusion

The use of in-scene atmospheric compensation algorithms allows for efficient estimation of key components in the LWIR radiative transfer equation but typically with higher error versus their model-based counterparts. This study has presented a hybrid approach, dependent on previously generated MODTRAN data but applicable to a wide range of conditions and altitudes. The inference step only requires in-scene data, without the need for lookup table generation, making this method applicable for real-time predictions. We demonstrated comparable performance to FLAASH-IR with an inference time of 0.24 s using a mobile graphics card. This computational speedup is important for efficiently dealing with the large volumes of data generated by modern LWIR sensors.

A key enabler of the DAC algorithm presented here was the use of permutation-invariant neural network layers. This approach allowed the model to estimate the underlying TUD vector from in-scene data without generating spatially resolved hyperspectral data cubes. Additionally, permutation-invariant layers were necessary to handle the diversity of possible at-sensor radiance pixel sets, derived from varying materials, material temperatures, and atmospheric conditions.

The results and analysis presented included both synthetic data and collected HSI, confirming that this method generalizes to real-world conditions. The entire training pipeline can be retooled for a particular sensor, only requiring a modified ILS for training data generation. There is a wide range of future work in this area, including testing against additional measured HSI data sets, varying types of AE models, pixel selection strategies, and modifications to the DAC network. Future work will also consider off-NADIR sensor zenith angles and modifications to the neural network architecture to support this additional information in the data compression and TUD estimation steps.

Acknowledgment

The authors would like to thank Nathan Wurst and Seung Hwan An from the Air Force Research Laboratory Sensors Directorate for providing the SEBASS HSI data used in this analysis. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the US Government.

References

[1] D. G. Manolakis, R. B. Lockwood, and T. W. Cooley, *Hyperspectral Imaging Remote Sensing: Physics, Sensors, and Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2016.

[2] M. T. Eismann, A. D. Stocker, and N. M. Nasrabadi, "Automated hyperspectral cueing for civilian search and rescue," *Proc. IEEE*, vol. 97, no. 6, pp. 1031–1055, Jun. 2009.

[3] M. T. Eismann, *Hyperspectral Remote Sensing*. Bellingham, WA, USA: SPIE, 2012.

[4] J. A. Sobrino *et al.*, "Emissivity mapping over urban areas using a classification-based approach: Application to the dual-use european security IR experiment (DESIREX)," *Int. J. Appl. Earth Observ. Geoinform.*, vol. 18, pp. 141–147, 2012.

[5] D. Zhou *et al.*, "Satellite remote sensing of surface urban heat islands: Progress, challenges, and perspectives," *Remote Sens.*, vol. 11, no. 1, 2019, Art. no. 48.

[6] B.-C. Gao, M. J. Montes, C. O. Davis, and A. F. Goetz, "Atmospheric correction algorithms for hyperspectral remote sensing data of land and ocean," *Remote Sens. Environ.*, vol. 113, pp. S17–S24, 2009.

[7] C. Borel, "Error analysis for a temperature and emissivity retrieval algorithm for hyperspectral imaging data," in *Proc. Int. Soc. Optics Photonics*, in Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XIII, vol. 6565, pp. 612–623, May 2007.

[8] D. Manolakis *et al.*, "Longwave infrared hyperspectral imaging: Principles, progress, and challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 72–100, Jun. 2019.

[9] N. Acito, M. Diani, and G. Corsini, "Subspace-based temperature and emissivity separation algorithms in LWIR hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1523–1537, Mar. 2019.

[10] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, "Deep sets," in *Advances in Neural Information Processing Systems*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Red Hook, NY: Curran Associates, Inc., 2017, pp. 3391–3401. [Online]. Available: http://papers.nips.cc/paper/6931-deep-sets.pdf

[11] S. Adler-Golden, P. Conforti, M. Gagnon, P. Tremblay, and M. Chamberland, "Long-wave infrared surface reflectance spectra retrieved from Telops hyper-cam imagery," in *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XX*, vol. 9088. International Society for Optics and Photonics, 2014, p. 90880 U.

[12] A. Berk, P. Conforti, R. Kennett, T. Perkins, F. Hawes, and J. van den Bosch, "MODTRAN6: A major upgrade of the MODTRAN radiative transfer code," in *Proc. Int. Soc. Optics Photonics*, in Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XX, vol. 9088, pp. 113–119, Jun. 2014.

[13] B. D. Bue *et al.*, "Real-time atmospheric correction of AVIRIS-NG imagery." *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6419–6428, Dec. 2015.

[14] N. Acito, M. Diani, and G. Corsini, "Coupled subspace-based atmospheric compensation of LWIR hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5224–5238, Aug. 2019.

[15] S. J. Young, B. R. Johnson, and J. A. Hackwell, "An in-scene method for atmospheric compensation of thermal hyperspectral data," *J. Geophys. Res.: Atmospheres*, vol. 107, no. D24, 2002.

[16] C. T. Lane, "In-scene atmospheric compensation of thermal hyperspectral imaging with applications to simultaneous shortwave data collection," Air Force Institute of Technology, WPAFB, Dayton, OH, USA, Tech. Rep. AFIT-ENP-DS-17-D-008, Dec. 2017.

[17] N. Westing, B. Borghetti, and K. C. Gross, "Fast and effective techniques for LWIR radiative transfer modeling: A dimension-reduction approach," *Remote Sens.*, vol. 11, no. 16, Aug. 2019, Art. no. 1866. [Online]. Available: http://dx.doi.org/10.3390/rs11161866

[18] A. Chedin, N. Scott, C. Wahiche, and P. Moulinier, "The improved initialization inversion method: A high resolution physical method for temperature retrievals from satellites of the TIROS-N series," *J. Climate Appl. Meteorol.*, vol. 24, no. 2, pp. 128–143, 1985.
[19] F. Chevallier, F. Chéruy, N. Scott, and A. Chédin, "A neural network approach for a fast and accurate computation of a longwave radiative budget," *J. Appl. Meteorol.*, vol. 37, no. 11, pp. 1385–1397, 1998.
[20] J. A. Hackwell *et al.*, "LWIR/MWIR Imaging Hyperspectral Sensor for Airborne and Ground-Based Remote Sensing," in *Proc. Int. Soc. Optics Photonics*, vol. 2819, Nov. 1996, pp. 102–107. [Online]. Available: https://doi.org/10.1117/12.258057
[21] H. Edwards and A. Storkey, "Towards a neural statistician," 2016, *arXiv:1606.02185*.
[22] J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer," 2018, *arXiv:1810.00825*.
[23] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2017, pp. 652–660.
[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[26] F. Chollet *et al.*, Keras.2015. [Online]. Available: https://keras.io
[27] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica, "Tune: A research platform for distributed model selection and training," 2018, *arXiv:1807.05118*.
[28] N. P. Wurst, S. H. An, and J. Meola, "Comparison of longwave infrared hyperspectral target detection methods," in Proc. *Int. Soc. Optics Photonics*, in Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery XXV, vol. 10986, pp. 343–354, May 2019.
[29] B. M. Rankin, J. Meola, and M. T. Eismann, "Spectral radiance modeling and Bayesian model averaging for longwave infrared hyperspectral imagery and subpixel target identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 6726–6735, Dec. 2017.

Nicholas Westing (Student Member, IEEE) received the B.S. degree in electrical and computer engineering from the University of Minnesota-Duluth, Duluth, MN, USA, in 2010, and the M.S. degree in electrical engineering from the Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, OH, USA, in 2014. He is currently working toward the Ph.D. degree at the Air Force Institute of Technology, investigating methods for hyperspectral data analysis and exploitation.

Kevin C. Gross received the B.S. degree in chemistry and mathematics and the M.S. degree in chemistry from Wright State University, Dayton, OH, USA, in 1998 and 2001, respectively, the Ph.D. degree in engineering physics from the Air Force Institute of Technology (AFIT), Wright-Patterson Air Force Base, Dayton, OH, USA, in 2007.

He is currently the Director of EO/IR Technologies with Resonant Sciences, Dayton, OH, USA, and the Adjunct Associate Professor with AFIT. His academic interests are remote sensing, spectroscopy, radiative transfer, and the development of physics-based algorithms to extract information content from electro-optical data.

Brett J. Borghetti received the Ph.D. degree in computer science from the University of Minnesota, Twin Cities, Minneapolis, MN, USA, in 2008, the M.S. degree in computer systems from the Air Force Institute of Technology (AFIT), Dayton, OH, USA, in 1996, and the B.S. degree in electrical engineering from Worcester Polytechnic Institute (WPI), Worcester, MA, USA, in 1992.

He is an Associate Professor with the Department of Electrical and Computer Engineering, Graduate School of Engineering Management, Air Force Institute of Technology. He has research experience in estimating human cognitive performance, statistical machine learning, genetic algorithms, self-organizing systems, neural networks, game theory, information theory, and cognitive science. His research interests focus on improving human–machine team performance in complex environments using artificial intelligence and machine learning.

Jacob Martin received the B.S. degree in physics from Michigan State University, East Lansing, MI, USA, in 2010, and the M.S. and Ph.D. degrees in applied physics from the Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, OH, USA, in 2012 and 2016, respectively.

He currently works in the hyperspectral tech area with the Electo-Optic Target Detection and Surveillance Branch, Sensors Directorate, Air Force Research Laboratory, Wright-Patterson Air Force Base, Dayton, OH, USA. His research interests include hyperspectral target detection, machine learning, sensor characterization, and performance modeling.

Joseph Meola received the B.S. and M.S. degrees in electrical engineering from the University of Dayton, Dayton, OH, USA, in 2004 and 2006, respectively, and the Ph.D. degree in electrical and computer engineering from The Ohio State University, Columbus, OH, USA, in 2011.

He is currently the Hyperspectral Technical Area Lead with the Electro-Optic Target Detection and Surveillance Branch, Sensors Directorate, Air Force Research Laboratory, Wright-Patterson Air Force Base, Dayton, OH, USA, where he is also an Adjunct Faculty Member with the Air Force Institute of Technology. His research interests include hyperspectral data modeling, sensor calibration and characterization, data exploitation, target detection, and change detection.