UNCERTAINTY-AWARE GRAPH-BASED HYPERSPECTRAL IMAGE CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Hyperspectral imaging (HSI) technology captures spectral information across a broad wavelength range, providing richer pixel features compared to traditional color images with only three channels. Although pixel classification in HSI has been extensively studied, especially using graph convolution neural networks (GCNs), quantifying epistemic and aleatoric uncertainties associated with the HSI classification (HSIC) results remains an unexplored area. These two uncertainties are effective for out-of-distribution (OOD) and misclassification detection, respectively. In this paper, we adapt two advanced uncertainty quantification models, evidential GCNs (EGCN) and graph posterior networks (GPN), designed for node classifications in graphs, into the realm of HSIC. We first analyze theoretically the limitations of a popular uncertainty cross-entropy (UCE) loss function when learning EGCNs for epistemic uncertainty estimation. To mitigate the limitations, we propose two regularization terms. One leverages the inherent property of HSI data where pixel features can be decomposed into weighted sums of various material features, and the other is the total variation (TV) regularization to enforce the spatial smoothness of the evidence with edge-preserving. We demonstrate the effectiveness of the proposed regularization terms on both EGCN and GPN on three real-world HSIC datasets for OOD and misclassification detection tasks. The code is available at `https://anonymous.4open.science/r/HSI_torch-1586/`

1 INTRODUCTION

Hyperspectral (HS) data is widely used in various real-world applications including atmospheric science (Saleem et al., 2020), food processing (Ayaz et al., 2020), and forestry (Khan et al., 2020), benefiting from rich spectral information measured at individual pixels. Unlike human eyes which possess only three color receptors sensitive to blue, green, and red channels, HS data provides a wide spectrum of light (visible and near-infrared range) for every pixel in the scene, which enables more faithful classification results compared to traditional classification using color images. As a result, *hyperspectral image classification* (HSIC) attracts considerable research interests (Chen et al., 2014; Ahmad et al., 2017; Hong et al., 2018; Ahmad et al., 2021). Specifically, graph convolution neural network (GCN) (Kipf & Welling, 2016) has found extensive use in HSIC (Shahraki & Prasad, 2018; Qin et al., 2018; Wan et al., 2020; Hong et al., 2020) due to its ability to effectively model the interdependency among pixels (especially when they are far away).

However, there is limited work related to predictive uncertainty quantification for HSIC. For example, it is not practical to assume that all categories (materials) in the scene are known and have available samples for model training. In such scenarios, the model is expected to have the capability to *know what they do not know*, which can be measured by *epistemic uncertainty* from a probabilistic view (uncertainty of model parameters due to limited training data). On the other hand, pixels may be misclassified due to various factors, such as environmental noise, material similarity, and atmospheric effects. Thus, it is desirable for a training model to identify the *unknown what they do not know*, which can be measured by *aleatoric uncertainty* (uncertainty due to randomness). Overall, it is necessary to quantify these two uncertainties to ensure the reliability of HSIC models.

The epistemic and aleatoric uncertainties at the pixel level can be used to detect out-of-distribution (OOD) pixels that belong to unknown materials and detect pixels that are misclassified to the wrong categories, respectively. OOD detection in HSIC performs ID classification and OOD detection simultaneously, which is different from *HSI anomaly detection*, as the latter only involves detect-

ing pixels whose spectral characteristics deviate significantly from surrounding or background pixels. Literature has found that epistemic uncertainty is the most effective for OOD detection, while aleatoric uncertainty is most effective for misclassification detection (Zhao et al., 2020; Stadler et al., 2021).

Graph-based models for HSIC construct a graph by regarding each pixel as a node and the interdependency among nodes is defined by an adjacency matrix. As a result, the nodes on the graph are dependent on each other. In contrast to extensive literature for independent inputs (Lakshminarayanan et al., 2017; Gal & Ghahramani, 2016; Charpentier et al., 2022), uncertainty estimation for semi-supervised node classification on a graph with dependent inputs is more complex and thus less explored (Abdar et al., 2021). Notably, two primary investigations have been conducted employing deterministic methodologies. One is the evidential graph neural network (EGCN) (Zhao et al., 2020), and it extends the evidential neural network (ENN) (Sensoy et al., 2018) on images (independent inputs) to graph data (dependent inputs) with GCNs and graph-based kernel Dirichlet estimation. Throughout the paper, we refer to this model as GKDE or EGCN for brevity. The other is graph posterior network (GPN) (Stadler et al., 2021) that adapted the posterior network (PN) (Charpentier et al., 2020) together with an evidence propagation through the graph nodes. Both GKDE and GPN predict the conjugate prior distribution of categorical distribution, i.e. Dirichlet distribution at each node, and incorporate the uncertainty-cross-entropy (UCE) loss (Biloš et al., 2019) in the overall optimization problem to train the model parameters.

However, the UCE loss function has limitations in effectively learning uncertainty quantification models. First, the models learned based on UCE tend to peak the Dirichlet distribution and become overly concentrated on the predictive classes in the simplex (feasible) space composed of the class-probability vectors (Bengs et al., 2022), which can be alleviated by introducing entropy-based regularization to encourage the predicted Dirichlet distributions to be uniform (Charpentier et al., 2020; Stadler et al., 2021). Second, it is known empirically that learning models based on UCE alone do not produce accurate epistemic uncertainty for OOD detection, which can be aided by additional regularization terms, e.g., the aforementioned GKDE (Zhao et al., 2020). However, we show in our experiments both GKDE and GPN do not have satisfactory results in HSIC.

In this work, we consider the uncertainty quantification task for graph-based hyperspectral image classification. **Our Contributions** are summarized as follows. **First**, we provide a theoretical analysis of the limitations of UCE for learning EGCNs to enable accurate epistemic uncertainty estimation. In particular, minimizing the UCE loss does not help an EGCN to learn embeddings that are capable of mapping OOD nodes into the detectable region near the decision boundary. **Second**, we propose a multidimensional uncertainty estimation framework for HSIC. To the best of our knowledge, this is a pioneer work in discussing the uncertainty estimation on the graph-based HSIC models. **Third**, we introduce a physics-guided unmixing-based regularization (UR) to address the shortcomings of the UCE loss when quantifying epistemic uncertainty. Here, we assume that OOD pixels are mostly composed of an unknown material and the UR term is the reconstruction squared loss for decomposing into the in-distribution (or known) materials and the OOD material. **Fourth**, we adopt the total variation regularization to propagate predicted evidence along the decision boundary (not across), thus preserving spatial edges between ID and OOD nodes. **Finally**, we present extensive empirical experiments to demonstrate the effectiveness of the proposed regularization terms on both EGCN and GPN using three real-world HSIC datasets for OOD and misclassification detection tasks in comparison with 5 competitive baselines.

2 PRELIMINARY

In this section, we review graph-based hyperspectral image classification in Section 2.1, EGCNs in Section 2.2, and relevant concepts of uncertainty quantification in Section 2.3.

2.1 GRAPH-BASED HYPERSPECTRAL IMAGE CLASSIFICATION (HSIC)

HSIC aims to assign a unique label to each pixel based on its spectral and spatial properties. Mathematically, the input HS data can be represented as $\boldsymbol{X} = [\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^{(HW)}] \in \mathbb{R}^{(H\times W)\times B}$, where B is the number of spectral bands (feature dimension) and $H \times W$ is the spatial dimension. Letting $N = HW$, we stack the 2D spatial domain to a vector, and hence each pixel i is associated with a feature vector $\boldsymbol{x}^i \in \mathbb{R}^B$, $\forall i \in [N]$. For classification purposes, each pixel i has a class label $y^i \in [C]$ associated with a specific constituent material, where C is the number of classes known as *a priori*.

The graph-based HSIC technique (Qin et al., 2020) builds a graph, in which each vertex corresponds to a pixel in the 2D spatial domain and the weighted adjacency matrix $\mathbb{A}$ is calculated based on similarities between node-level features: i.e.,

$$\mathbb{A}_{ij} = \exp(-d(\mathbf{x}^i, \mathbf{x}^j)/\sigma), \quad \forall i, j \in [N], \tag{1}$$

where $d(\mathbf{x}^i, \mathbf{x}^j)$ is the Euclidean distance (or minus cosine similarity) between vertices i and j, and σ is tuned to optimize how similar two nodes are. Suppose the resulting graph is defined as $\mathcal{G}(\mathbb{V}, \mathbb{E}, \boldsymbol{X}, \boldsymbol{Y}_{\mathbb{L}})$, where $\mathbb{V} = \{1, \cdots, N\}$ is a ground set of nodes, $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ is a ground set of edges, $\boldsymbol{X} = [\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^N] \in \mathbb{R}^{N \times B}$ is the node-level feature matrix, $\boldsymbol{x}^i \in \mathbb{R}^B$ is the feature vector of node i, $\boldsymbol{y}_{\mathbb{L}} = \{y^i | i \in \mathbb{L}\} \in \mathbb{R}^{|\mathbb{L}|}$ is the label for the training node set $\mathbb{L} \subset \mathbb{V}$, and $y^i \in [C]$ is the label for node i. The GCN-based HSIC method (Hong et al., 2020) can be formulated as $[\mathbf{p}^i]_{i \in \mathbb{V}} = f(\mathbb{A}, \mathbf{X}; \boldsymbol{\theta})$, where $\mathbf{p}^i$ is the probability vector of node i and $f(\cdot)$ is a standard GNN function that depends on the adjacency matrix $\mathbb{A}$, the data matrix $\mathbf{X}$, and a set of network parameters, denoted by $\boldsymbol{\theta}$.

2.2 Evidential Graph Convolutional networks for node classification

An evidential GCN (EGCN) (Zhao et al., 2020) takes graph $\mathcal{G}$ as INPUT and predicts an evidence vector $\mathbf{e}^i = [e_1^i, \cdots, e_C^i]$ for each node i as OUTPUT: $[\mathbf{e}^i]_{i \in \mathbb{V}} = f(\mathbb{A}, \mathbf{X}; \boldsymbol{\theta})$, where e_c^i is a measure of the amount of support collected form the training labels $y_{\mathbb{L}}$ in favor of node i to be classified to the class c. EGCN is the same as a classical GCN, except that the activation function (e.g., exponential or ReLU) of the output layer is unbounded, outputting an evidence vector, instead of the softmax function outputting class probabilities. The evidence vector can quantify predictive uncertainty through a well-defined theoretical framework called subjective logic (SL) (Jsang, 2018). More specifically, a multinomial opinion $\omega = (\mathbf{b}, u)$ in SL can be defined as:

$$b_c = \frac{e_c}{S} \text{ and } u = \frac{C}{S}, \text{ for } c = 1, \cdots, C, \tag{2}$$

where $\mathbf{b} = [b_1, \cdots, b_C]^T$ represents the beliefs of the C classes, u is the uncertainty mass representing *the vacuity of evidence*, and $S = \sum_{c=1}^{C}(e_c + 1)$. It is straightforward that $b_c \geq 0, u \geq 0$, and $\sum_{c=1}^{C} b_c + u = 1$. A multinomial opinion ω can be equivalently represented by a Dirichlet distribution: $P(\mathbf{p}) \sim \text{Dir}(\boldsymbol{\alpha})$, where $\mathbf{p} = [p_1, \cdots, p_C]$ is a probability vector of C classes and $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_C]$ are called *concentration parameters* with $\alpha_c = e_c + 1$. The class label $\mathbf{y}^i$, probability vector $\mathbf{p}^i$, and the evidence vector $\mathbf{e}^i$ for node i have the following probabilistic relations:

$$\mathbf{y}^i \sim \text{Cat}(\mathbf{p}^i), \;\; \mathbf{p}^i \sim \text{Dir}(\mathbf{p}^i | \boldsymbol{\alpha}^i), \;\; \boldsymbol{\alpha}^i = \mathbf{e}^i + \mathbf{1}, \;\; [\mathbf{e}^i]_{i \in \mathbb{V}} = f(\mathbb{A}, \mathbf{X}; \boldsymbol{\theta}). \tag{3}$$

The expected class probability is equal to the mean of the Dirichlet distribution, i.e. $\bar{\boldsymbol{p}} = \frac{\boldsymbol{\alpha}}{S}$ in the sense that S can also be defined by $S = \sum_{c=1}^{C} \alpha_c$. Based on the principles of evidential theory, a lack of evidence, i.e., "I don't know," can be expressed through a close-to-zero vacuity u (or equivalently a uniform Dirichlet).

An EGCN is trained based on the uncertainty cross-entropy (UCE) loss function, defined by

$$\text{UCE}(\boldsymbol{\alpha}^i, \mathbf{y}^i; \boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{p}^i \sim \text{Dir}(\boldsymbol{p}^i | \boldsymbol{\alpha}^i)} \left[-\log \mathbb{P}(\boldsymbol{y}^i | \boldsymbol{p}^i) \right], \tag{4}$$

which can be interpreted as the expectation of the standard cross-entropy loss with respect to the distribution of class probabilities: $\boldsymbol{p}^i \sim \text{Dir}(\boldsymbol{p}^i | \boldsymbol{\alpha}^i)$. Alternatively, Stadler et al. (2021) proposed a new network architecture (as opposed to a classical GNN architecture), namely graph posterior networks (GPN), to predict node-level Dirichlet distributions. Specifically, GPN consists of three modules: multilayer perceptron (MLP) layers for node-level feature embedding, a normalizing flow module to estimate node-level densities in the embedded space, and a personalized page rank propagation layer (Gasteiger et al., 2018) to smooth the concentration parameters among neighboring nodes.

2.3 Uncertainty Quantification

Aleatoric uncertainty is the uncertainty in the class prediction that is measured by the entropy of categorical distribution(Malinin et al., 2017), i.e. $u^{\text{alea}} = \mathbb{H}(\text{Cat}(\bar{\boldsymbol{p}}))$ or *confidence* (Charpentier et al., 2020), i.e., $u^{\text{alea}} = -\max_c \bar{\boldsymbol{p}}_c$. It exhibits higher values when the categorical distribution is flat. In contrast, epistemic uncertainty is the uncertainty on categorical distribution and can be measured

by the total evidence count, i.e. $u^{\text{epis}} = C/S$, which is referred to as *vacuity* from the viewpoint of evidential uncertainty (Josang et al., 2018). When the distribution of categorical distribution, which is the Dirichlet distribution in the evidential-based models, is spread out, the epistemic uncertainty is high. Aleatoric uncertainty is proven to be more effective for detecting misclassifications while epistemic is better for identifying OOD samples (Zhao et al., 2020).

3 UNCERTAINTY-AWARE REGULARIZED LEARNING

This section first discusses the limitations of the UCE loss function and the GKDE-based regularization term on epistemic uncertainty quantification in Section 3.1. We then propose two regularization terms specifically designed for HSIC: unmixing-based regularization (UR) and evidence-based total variation (TV) regularization in Section 3.2.

3.1 LIMITATIONS OF UCE AND EXISTING REGULARIZATION TECHNIQUES

Without the loss of generality, we focus on binary classification task throughout this section; the generalization to multiple classes can be analyzed similarly to Collins et al. (2023) and Kristiadi et al. (2020). A typical EGCN architecture has several graph convolutional (GC) layers followed by one MLP layer (Zhao et al., 2020). The GC layers produce node-level embeddings to capture the graph dependency among the nodes in the sense that nodes that are neighbors in the graph are more likely to be spatial neighbors in the embedded space, denoted by $\mathcal{D} \subset \mathbb{R}^D$. Specifically for homophily graphs (Ma et al., 2021), GC layers can generate embeddings that can separate different classes. Note that the three HSIC datasets used in our experiments are indeed homophily graphs, as discussed in Appendix B. The MLP layer in EGCN helps to reduce the dimensions of the embedded space while producing node-level evidence. We demonstrate that the MLP layer learned based on UCE fails to produce accurate evidence predictions, even in the ideal case where the GC layers can produce perfectly separable node embeddings. Let $\mathbf{z}^i \in \mathbb{R}^D$ denote the embedded vector of node $i \in \mathbb{V}$. The MLP layer for node-level evidence prediction can be formulated as:

$$\mathbf{e}(\mathbf{z};\boldsymbol{\theta}) := [e_+(\mathbf{z};\boldsymbol{\theta}), e_-(\mathbf{z};\boldsymbol{\theta})] = [\sigma(\mathbf{w}^T\mathbf{z}+b), \sigma(-\mathbf{w}^T\mathbf{z}-b)], \tag{5}$$

where $\boldsymbol{\theta} = \{\mathbf{w}, b\}$, $\mathbf{w} \in \mathbb{R}^D$, $b \in \mathbb{R}$, and $\sigma(\cdot)$ is the activation function (e.g. ReLU and exponential) that outputs evidence values. We start with the lower and upper bounds for MLP-based ENNs.

Proposition 1. *Suppose $\mathbf{z} \in \mathcal{D} \subset \mathbb{R}^D$ is a data point in the embedded space and $y \in \{-1, +1\}$ is its binary class label. An MLP-based ENN has the lower and upper bounds for the UCE loss:*

$$\frac{1}{e_y(\mathbf{z};\boldsymbol{\theta})+1} \leq \mathit{UCE}(\boldsymbol{\alpha}(\mathbf{z};\boldsymbol{\theta}), y;\boldsymbol{\theta}) \leq \frac{\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil + 1}{e_y(\mathbf{z};\boldsymbol{\theta})}, \tag{6}$$

where $e_y(\mathbf{z};\boldsymbol{\theta})$ is the evidence of classs y, $\boldsymbol{\alpha}(\mathbf{z};\boldsymbol{\theta}) = \mathbf{e}(\mathbf{z};\boldsymbol{\theta}) + 1$, and $\lceil\cdot\rceil$ is the ceiling operator. If the ENN can predict y correctly: $y(e_+(\mathbf{z};\boldsymbol{\theta}) - e_-(\mathbf{z};\boldsymbol{\theta})) > 0$, we have a tighter upper bound:

$$\mathit{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y;\boldsymbol{\theta}) \leq \overline{\mathit{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), y;\boldsymbol{\theta}) := \frac{r+1}{e_y(\mathbf{z};\boldsymbol{\theta})}, \tag{7}$$

where $r = 0$ for the ReLU activation function and $r = 1$ for the exponential activation function.

Please refer to Appendix A.1 for the proof of Proposition 1. Note that the upper bound in (7) is tight as the error bound: $|\frac{r+1}{e_y(\mathbf{z};\boldsymbol{\theta})} - \text{UCE}(\boldsymbol{\alpha}, y;\boldsymbol{\theta})| \leq \frac{r+1}{e_y(\mathbf{z};\boldsymbol{\theta})} - \frac{1}{e_y(\mathbf{z};\boldsymbol{\theta})+1} = \frac{r\cdot e_y(\mathbf{z};\boldsymbol{\theta})+r+1}{(e_y(\mathbf{z};\boldsymbol{\theta})+1)e_y(\mathbf{z};\boldsymbol{\theta})} \to 0$ as $e_y(\mathbf{z};\boldsymbol{\theta}) \to \infty$. Under the assumption of the universal approximation theorem (Pinkus, 1999) for an MLP network, the optimal parameter $\boldsymbol{\theta}^\star$ that minimizes the UCE on a training set has the property: $e_y(\mathbf{z};\boldsymbol{\theta}^*) \to \infty$ and $e_{-y}(\mathbf{z};\boldsymbol{\theta}^*) \to 0$, as demonstrated in Lemma 2 in Appendix A.1.

We establish in Theorem 1 that the optimal solution when minimizing the upper bound $\overline{\text{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y};\boldsymbol{\theta})$ defined in Equation (7) with the exponential activation function $\sigma(\cdot)$ has a closed-form expression that is equivalent to the optimal solution of linear discriminative analysis (LDA) under certain assumptions.

Theorem 1. *We assume that (i) feature vectors belonging to classes $\{\pm 1\}$ follow Gaussian distributions with the same covariance matrix and the means $\pm\boldsymbol{\mu}$, respectively, i.e., $\mathbb{P}(\mathbf{z}, y) = \mathbb{P}(y = +1)\mathcal{N}(\mathbf{z};\boldsymbol{\mu},\boldsymbol{\Sigma}) + \mathbb{P}(y = -1)\mathcal{N}(\mathbf{z};-\boldsymbol{\mu},\boldsymbol{\Sigma})$, with $\mathbb{P}(y = +1) = \mathbb{P}(y = -1) = 0.5$; (ii) the optimal solutions $\boldsymbol{\theta}^\star$ that minimize $\mathbb{E}_{(\mathbf{z},y)\sim\mathbb{P}(\mathbf{z},y)}\overline{\mathit{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y};\boldsymbol{\theta})$ can linearly separate both*

classes: $e_y(\mathbf{z};\boldsymbol{\theta}) > e_{-y}(\mathbf{z};\boldsymbol{\theta}), \forall(\mathbf{z}, y)$. *Let* $\sigma(\cdot)$ *be the exponential function. The optimal solution* $\boldsymbol{\theta}^\star = (\mathbf{w}^\star, b^\star)$ *is the same as the optimal solution of LDA, i.e.,*

$$\mathbf{w}^\star = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} \text{ and } b^\star = 0. \tag{8}$$

Theorem 1 has several important implications when the classes are separable. **First**, the MLP layer in EGCN learned based on $\overline{\text{UCE}}$ has the same objective as in LDA: Finding a projection that maximizes the separation between the projected class means with a small variance within each class. Unfortunately, as illustrated in Fig. 1, this objective does not help learn a projection that maps OOD data points to the **grey region of low evidence** near the decision boundary, where GCNs can effectively detect OODs: $e_y(\mathbf{z};\boldsymbol{\theta}^\star) = 1$ (or equivalently $\mathbf{w}^{\star T}\mathbf{z} + b^\star = 0$), to have the required small evidence values (or large epistemic uncertainty values due to Eq. (2)). For any far-away OOD data point $\tilde{\mathbf{z}} = \delta \cdot \mathbf{z}$, where $\delta \to \infty$ and $(\mathbf{z}, y) \in \mathbb{P}(\mathbf{z}, y)$ is an ID data point, the predicted evidence approaches $+\infty$ evidence (the epistemic uncertainty approaches 0, equivalently): $e_y(\tilde{\mathbf{z}};\boldsymbol{\theta}^\star) = \exp(y(\mathbf{w}^{\star T}\tilde{\mathbf{z}} + b^\star)) = \exp(y\mathbf{w}^{\star T}\delta\mathbf{z}) = (\exp(y\mathbf{w}^{\star T}\mathbf{z}))^\delta \to \infty$, when $\mathbf{z}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} \neq 0$, given that $b^\star = 0$ and $\exp(y(\mathbf{w}^{\star T}\tilde{\mathbf{z}} + b^\star)) = \exp(y\mathbf{w}^{\star T}\tilde{\mathbf{z}}) > 1$.

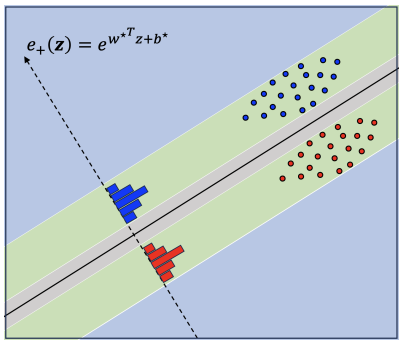

Figure 1: Two-separable-class case: The grey region in the feature space is a detectable OOD region by one-layer MLP-based ENN learned using UCE. The light-blue and light-green OOD regions are not detectable by ENN.

Second, the evidence predictions for the testing data points are not influenced by the distance between the two class means, as it is not a factor in $\mathbf{w}^\star$ and $b^\star$. **Third**, Fig. 1 shows that we can identify the light-blue and light-grey OOD regions of different characteristics in the feature space of $\mathbf{z}$ based on the projection $\mathbf{w}^\star = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$. In particular, the learned MLP layer predicts higher evidence for OOD nodes in the light-blue region than the one of ID nodes; and predicts evidence similar to those of ID data points for OOD data points in the light-grey region. The learned MLP can only predict small evidence for OOD points in the small light-grey region near the decision boundary: $\mathbf{w}^{\star T}\mathbf{z} + b^\star = 0$.

We remark on several assumptions in Theorem 1. First, the assumption on the Gaussian means $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$ can always be true by translating the origin of the feature space to the middle point of two centers as a preprocessing step. Second, we assumed the same covariance matrix $\boldsymbol{\Sigma}$ for the two Gaussians to obtain an analytical solution in Eq. (8). For different covariance matrices, the optimal solution is non-identical to that of LDA. Third, we assume that the classes are linearly separable so that the MLP layer can be defined in Eq. (5). The linear separability has been assumed in OOD-related theoretical analysis such as Ahuja et al. (2021). For the non-separable case, the MLP layer is defined as $\mathbf{e}(\mathbf{z};\boldsymbol{\theta}) := [e_+(\mathbf{z};\boldsymbol{\theta}), e_-(\mathbf{z};\boldsymbol{\theta})] = [\sigma(\mathbf{w}_1^T\mathbf{z} + b_1), \sigma(\mathbf{w}_2^T\mathbf{z} + b_2)]$, where the weight and bias parameters for predicting the evidence values of the two classes are different. As demonstrated in Fig. 2, the grey, light-green, and light-blue regions have more complex shapes compared to the separable case in Fig. 1. The OOD points that can be detected by the MLP layer are within the grey region: $\{\mathbf{z}|e_y(\mathbf{z};\boldsymbol{\theta}) < 1, e_{-y}(\mathbf{z};\boldsymbol{\theta}) < 1\}$. Further, there is an orange region for the non-separable case: $\{\mathbf{z}|e_y(\mathbf{z};\boldsymbol{\theta}) > 1, e_{-y}(\mathbf{z};\boldsymbol{\theta}) > 1\}$, in which the evidence values for both classes are larger than 1. Our theoretical results on EGCN may not be generalizable to GPN. GPN predicts evidence values based on density estimation in the embedded space instead of MLP layers as used in EGCN.

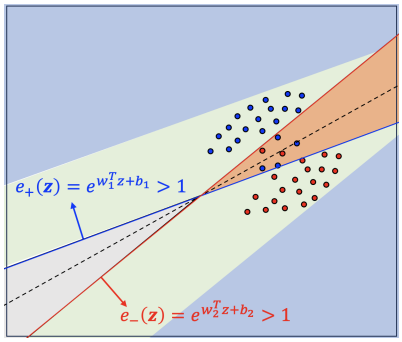

Figure 2: Two-non-separable-class case. The grey, light-green, and light-blue regions are the same as those in Fig.1. In the orange region, the predicted evidence values of both classes are larger than 1.

We demonstrate that minimizing the UCE loss does not help to learn the MLP layer to map the OOD regions (e.g., light-green and light-green OOD regions in Figs. 1 and 2) into the detectable OOD region of EGCN near the decision boundary. Zhao et al. (2020) proposed to use the KL

divergence-based regularization term: $\sum_{i\in\mathbb{L}} \mathrm{KL}(\mathrm{Dir}(\hat{\boldsymbol{p}}^i|\hat{\boldsymbol{\alpha}}^i), \mathrm{Dir}(\boldsymbol{p}^i|\boldsymbol{\alpha}^i))$, to enforce the closeness between $\hat{\boldsymbol{\alpha}}^i$ and $\boldsymbol{\alpha}^i$, $\forall i \in \mathbb{V}$, where $\hat{\boldsymbol{\alpha}}^i$ is a pre-computed teacher based on graph-kernel distance. However, this term assumes that OOD test nodes are far away in terms of graph-based distance from the training (ID) nodes compared to ID test nodes. This assumption is not always valid based on our empirical performance of the GKDE teacher on three HSIC datasets in Appendix E.2.

3.2 Unmixing and Evidence-based Uncertainty Regularizations

Due to the limited spatial resolution of HSI sensors, it is conceivable that each pixel in HSI data may contain a combination of materials, and hence it is desirable to decompose a single pixel into the proportions of constituent materials (a.k.a. abundance). We assume that there exist C pure materials (a.k.a. endmembers) in the scene, each with the corresponding signature $\boldsymbol{m}_c \in \mathbb{R}^B, c \in \{1, \cdots, C\}$. A matrix formed by all of these signatures is called a mixing matrix and denoted by $\boldsymbol{M} = [\boldsymbol{m}_1, \ldots, \boldsymbol{m}_C] \in \mathbb{R}^{B\times C}$. The abundance map obtained by the abundance coefficients of all the pixels can be represented by a matrix $\boldsymbol{V} \in \mathbb{R}^{C\times N}$. We adopt a simple linear mixing model in which the spectral measurement at each pixel is a linear combination of the endmembers, i.e.,

$$\boldsymbol{x}^i = \sum\nolimits_{c=1}^{C} v_c^i \boldsymbol{m}_c + \boldsymbol{\eta}^i, \tag{9}$$

where v_c^i is the abundance coefficient for the c-th material at the ith pixel and $\boldsymbol{\eta}^i$ denotes a noise term. Denote $\boldsymbol{v}^i = [v_1^i, \ldots, v_C^i]$. It is typical to assume $\sum_{c=1}^{C} v_c^i = 1$, as each abundance vector resides within the probability simplex. We adopt the linear mixing model (9) for its simplicity. There are more complicated nonlinear models by taking into account endmember-wise scaling factors (Drumetz et al., 2016), spectral variability (Hong et al., 2018), and illumination-induced variability (Drumetz et al., 2019). Decomposing the HS data $\mathbf{X}$ into a collection of reference spectral signatures $\mathbf{M}$ with associated abundance matrix $\mathbf{V}$ is referred to as *hyperspectral unmixing*.

Unmixing-based Regularization (UR). HSIC is related to hyperspectral unmixing in that the in-distribution (ID) classes are associated with the C known endmembers. Under the OOD detection setting where OOD classes are associated with unknown materials, we can consider a linear mixing model, where the signatures of the ID materials, $\{\mathbf{m}_1, \cdots, \mathbf{m}_C\}$ are given. We assume that OOD nodes are associated with the same unknown material that is denoted by $\mathbf{m}_o$. The hyperspectral unmixing problem can be formulated as

$$\min_{\boldsymbol{m}_o, \boldsymbol{v}^i, v_o^i} \sum\nolimits_{i\in\mathbb{V}} \|\boldsymbol{x}^i - \boldsymbol{M}\boldsymbol{v}^i - v_o^i \boldsymbol{m}_o\|_2^2. \tag{10}$$

We propose to use the beliefs $\mathbf{b}^i(\boldsymbol{\theta})$ and the vacuity $u^i(\boldsymbol{\theta})$ (the epistemic uncertainty measure) to approximate the abundance coefficients $\boldsymbol{v}^i$ of ID materials and the abundance coefficient v_o^i of the OOD material, respectively. The rationale of such approximations, $\mathbf{b}^i(\boldsymbol{\theta}) \approx \boldsymbol{v}^i(\boldsymbol{\theta})$ and $u^i(\boldsymbol{\theta}) \approx v_o^i(\boldsymbol{\theta})$, is threefold. First, the sum-to-one property on beliefs and vacuity: $\sum_{c=1}^{C} b_c^i + u^i = 1$, is aligned with the one on abundance coefficients: $\sum_{c=1}^{C} v_c^i + v_o^i = 1$. Second, the vacuity u^i for ID node i should be close to zero, and hence the beliefs are analogous to class probabilities (Jsang, 2018), which can be used to approximate the abundance coefficients (Chen et al., 2023). Third, the vacuity for an OOD node is close to one and its belief is close to zero, implying that the abundance coefficient $v_o^i(\boldsymbol{\theta})$ should be close to one and $\boldsymbol{v}^i(\boldsymbol{\theta})$ be close to $\mathbf{0}$. Using the approximations, we turn the unmixing problem (10) into an unmixing regularization (UR) term,

$$\min_{\boldsymbol{m}_o, \boldsymbol{\theta}} \mathrm{UR}(\boldsymbol{m}_o, \boldsymbol{\theta}) := \sum\nolimits_{i\in\mathbb{V}} \|\boldsymbol{x}^i - \boldsymbol{M}\boldsymbol{b}^i(\boldsymbol{\theta}) - u_o^i(\boldsymbol{\theta})\boldsymbol{m}_o\|_2^2, \tag{11}$$

where $\boldsymbol{b}^i, u_o^i (\forall i \in \mathbb{V})$ can be derived by the evidence $\boldsymbol{e}^i(\boldsymbol{\theta}) = f_i(\mathbb{A}, \mathbf{X}; \boldsymbol{\theta})$, or $\boldsymbol{e}^i$ for brevity. Minimizing the UR term encourage high vacuity for OOD nodes and low vacuity for ID ones. Given $\boldsymbol{\theta}$, there is a closed-form solution for the optimal $\boldsymbol{m}_o$, i.e.,

$$\boldsymbol{m}_o^* = \frac{\sum_{i\in\mathbb{V}} u_o^i(\boldsymbol{\theta})(\boldsymbol{x}^i - \boldsymbol{M}\boldsymbol{b}^i(\boldsymbol{\theta}))}{\sum_{i\in\mathbb{V}} (u_o^i(\boldsymbol{\theta}))^2}. \tag{12}$$

Please refer to Appendix A.3 for more details. Using the definitions of $\boldsymbol{b} = \frac{\boldsymbol{e}}{C+\sum_{c=1}^{C} e_c}$, $u = \frac{C}{C+\sum_{c=1}^{C} e_c}$, we rewrite $\mathrm{UR}(\boldsymbol{m}_o^*, \boldsymbol{\theta})$ with respect to evidence $\boldsymbol{e}$, i.e.,

$$\mathrm{UR}(\boldsymbol{e}) = \sum\nolimits_{i\in\mathbb{V}} \|\boldsymbol{x}^i - \frac{\sum_{c=1}^{C} e_c^i \boldsymbol{m}_c}{C + \sum_{c=1}^{C} e_c^i} - \frac{C\boldsymbol{m}_o^*}{C + \sum_{c=1}^{C} e_c^i}\|_2^2. \tag{13}$$

Proposition 2. *Assume the linear model (9) holds without noise, the gradient descent for minimizing the UR($\boldsymbol{e}$) regularization increases the predicted evidence of ground-truth class for ID instances and decrease the total evidence for OOD instances with the corresponding pure material contained in the pixel. Formally, we have*

(a) For an instance $(\boldsymbol{x}^i, y^i)$ with feature matrix $\boldsymbol{x}^i = \boldsymbol{m}_{y^i}$ and $y^i \in \{1, \ldots, C\}$ is the ground truth label, one has

$$\frac{\partial UR(\boldsymbol{e})}{\partial e^i_{y^i}} \leq 0. \tag{14}$$

(b) For an OOD instance instance $(\boldsymbol{x}^i, y^i)$ with $\boldsymbol{x}^i = \boldsymbol{m}^*_o$ and $y^i = o \notin \{1, \ldots, C\}$

$$\sum_{c=1}^{C} \frac{\partial UR(\boldsymbol{e})}{\partial e^i_c} \geq 0. \tag{15}$$

We present two desired properties of the UR term in Proposition 2. Part (a) is consistent with minimizing the UCE loss for ID nodes, aiming to predict high class-wise evidence for ground truth class. This often results in an increased total evidence for ID nodes during training iterations. Part (b) implies a decrease in total evidence for OOD samples when minimizing the UR term, resulting in a higher vacuity score, which provides additional information beyond the UCE loss. It is the inherent physical characteristics of hyperspectral data that implicitly help distinguish OOD and ID. Specifically, each pixel in a hyperspectral image contains a spectrum, which is a mixture of the spectra of all materials present in that pixel. ID and OOD pixels naturally contain different materials. The results above are agnostic to model architectures, i.e. the unmixing-based regularization term can be applied for arbitrary uncertainty quantification architectures and is guaranteed to have the above properties with reasonable assumptions.

Evidence-based Total Variation Regularization (TV). The graph $\mathcal{G}$ does not incorporate spatial information. For hyperspectral unmixing, the total variation (TV) regularization (Iordache et al., 2012) was applied to the abundance coefficients to enforce the spatial smoothness, while preserving edges. We propose the use of TV on the node-level vacuity value, which is inversely proportional to the Dirichlet level strengths (or equivalently total evidence). To define the discrete TV regularization, we represent a 2D image of size $H \times W$ as a vector via a linear indexing, i.e., $((h-1)H + m)$-th component denotes the location at (h, m). Define two matrices D_x, D_y to be the finite forward difference operators with periodic boundary conditions in the horizontal and vertical directions, respectively. Then the discrete form of the (anisotropic) TV norm is defined by

$$\text{TV}(\boldsymbol{u}) = \|D_x \boldsymbol{u}\|_1 + \|D_y \boldsymbol{u}\|_1. \tag{16}$$

Regularized Learning. The regularized learning objective has the following form:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{m}_o) = \sum\nolimits_{i \in \mathbb{L}} \Big(\text{UCE}(\boldsymbol{\alpha}^i, \mathbf{y}^i; \boldsymbol{\theta})) + \lambda_1 R(\boldsymbol{\theta}) \Big) + \lambda_2 \text{UR}(\boldsymbol{\theta}, \boldsymbol{m}_o) + \lambda_3 \text{TV}(\boldsymbol{u}(\boldsymbol{\theta})), \tag{17}$$

where $R(\boldsymbol{\theta})$ refers to the model (GKDE or GPN)-specific regularization term and $\lambda_1, \lambda_2, \lambda_3$ are hyperparameters. For GPN, $R(\boldsymbol{\theta}) = \sum_{i \in \mathbb{L}} \text{ENT}(\text{Dir}(\boldsymbol{p}^i | \boldsymbol{\alpha}^i))$. The GKDE regularization term can be found in the last paragraph of Section 3.1. The TV term is applied on the vacuity score $\boldsymbol{u}(\boldsymbol{\theta})$. The last two terms only require node features and are applied to the whole graph $\mathbb{V}$. The model parameter $\boldsymbol{\theta}$ and $\boldsymbol{m}_o$ in UR term can be optimized alternatively: closed-form solution for $\boldsymbol{m}_o$ in (12) and gradient descent to update the model parameters $\boldsymbol{\theta}$.

4 Experiment

4.1 Experiment Setup

Datasets We use three HSIC datasets for evaluation: the University of Pavia (UP), the University of Houston (UH), and the Kennedy Space Center (KSC) dataset. For train/(validation + test) split, we use the public challenge split for UH (Debes et al., 2014), the same split as (Hong et al., 2020) for UP, and a random split for KSC with 20 nodes for training. For validation/test split, we use 0.2/0.8. The number of disjoint train/validation/test samples selected from each class used for all the experimental results is presented in Appendix B.

Competing Schemes We consider two state-of-the-art uncertainty quantification backbones for graph data: EGCN (Zhao et al., 2020) and GPN (Stadler et al., 2021). For EGCN, we include GKDE regularization by default. We apply our two regularization terms to the original model loss and analyze the effect of our proposed UR term and TV term. We then compare our proposed models with three baselines. Softmax-GCN (Kipf & Welling, 2017) is a classic GCN for semi-supervised node classification and uses the softmax as the last activation layer. We use the entropy as the uncertainty score as (Hendrycks & Gimpel, 2016). Though this paper focuses on OOD detection, we also include two anomaly detection models TLRSR (Wang et al., 2022) and RGAE (Fan et al., 2021) in our baselines as anomaly detection is widely researched in HSI. We use the features of all pixels to do the anomaly detection and take the OOD class as the anomalies. The detailed settings and parameter tuning are presented in Appendix D. We evaluate the performance of all above models using the area under the ROC (AUROC) curve and the area under the Precision-Recall (AUPR) curve in both experiments. Specifically, the detection rankings are based on epistemic uncertainty score in OOD detection tasks and aleatoric uncertainty in Misclassification detection tasks.

Table 1: AUROC and AUPR for the Misclassification Detection.

dataset	UP		UH		KSC	
	Mis ROC	Mis PR	Mis ROC	Mis PR	Mis ROC	Mis PR
softmax-GCN	**78.95**±1.18	47.58±0.78	**89.22**±0.25	52.85±1.62	89.22±0.25	52.85±1.62
EGCN	77.37±0.61	47.78±0.44	87.98±0.50	76.18±0.82	90.18±0.32	54.38±1.59
EGCN - UR	77.89±1.37	47.16±0.52	88.47±0.56	**76.77**±1.15	90.21±0.38	53.51±1.79
EGCN - UR - TV(**Ours**)	78.83±0.85	**48.73**±0.49	87.96±0.69	75.76±1.51	**90.37**±0.54	55.09±1.71
GPN	73.39±0.70	44.38±0.60	80.66±0.95	67.24±1.92	78.13±13.82	54.68±7.69
GPN - UR	73.58±0.36	45.02±0.42	81.08±1.05	67.25±1.24	82.38±6.78	**55.28**±6.17
GPN - UR - TV(**Ours**)	73.35±0.33	47.99±2.37	83.36±0.68	67.57±1.83	78.23±6.58	53.08±8.09

4.2 RESULTS

Misclassification detection. The misclassification detection is to detect whether a given prediction is incorrect with an uncertainty score. It is evaluated on the clean graph and the positive cases correspond to wrongly classified nodes and the negative cases represent correctly classified nodes. Table 1 shows the misclassification detection result where the bold numbers are the best results over all models. We can observe that softmax-GCN is not bad on misclassification detection, which indicates that misclassified nodes tend to have predicted class probabilities spread out across ID categories and entropy can capture reasonable aleatoric uncertainty for deterministic softmax models. Besides, our proposed uncertainty quantification frameworks show comparable results with softmax-GCN on the misclassification detection task for UP and UH, and have slightly better ROC and PR on KSC.

OOD detection. OOD detection involves whether a given example is out-of-distribution based on an estimate of uncertainty compared to the training set. The positive class corresponds to OOD nodes and the negative class pertains to ID nodes. The OOD detection is on the Left-out classes setting consistent with Zhao et al. (2020) and Stadler et al. (2021). Note that we remove the Left-Out classes from the training set but keep them in the graph. Within each dataset, we create four random configurations, and in each one, one class is picked as OOD. we display the weighted average, factoring in the count of test OOD nodes for every dataset in Table 2. We report the mean and stand deviation of 5 random runs. Due to the space constraint, the detailed setting and result for each OOD setting can be found in Appendix E.2. The bold numbers are the best results over all models. The underlined numbers are the best results within the same model type if it is not the best of all.

We observe that our proposed uncertainty quantification framework with both two implementations outperforms the softmax-GCN and anomaly detection baselines. This result indicates that softmax entropy can not capture the epistemic uncertainty well, which is the key for OOD detection, leading to worse performance on OOD detection. Anomaly detection techniques are designed to identify pixels with abnormal features across an entire image, without relying on supervised information about what's considered "normal". Compared to models specifically designed for OOD detection, these models are unable to perform ID classification and struggle to recognize OOD. One possible scenario is the most distinct elements might actually belong to an easily classified ID category. In addition, GPN backbone performs best on UP while EGCN performs best on UH and KSC. This may be because the GKDE teacher in the EGCN model (full name EGCN-EGCN) has a better instruction effect on UH and KSC and the performance of EGCN model is highly dependent on the

Table 2: AUROC and AUPR for the OOD Detection.

dataset	UP		UH		KSC	
	OOD ROC	OOD PR	OOD ROC	OOD PR	OOD ROC	OOD PR
softmax-GCN	57.04±5.80	16.34±3.29	56.78±2.63	19.18±0.57	77.12±0.65	54.18±1.29
RGAE	77.22±n.a.	24.81±n.a.	52.51±n.a.	10.59±n.a.	69.62±n.a.	34.10±n.a.
TLRSR	74.03±n.a.	20.11±n.a.	48.95±n.a.	6.24±n.a.	58.14±n.a.	9.84±n.a.
EGCN	87.21±0.67	45.50±0.65	88.64±0.33	39.68±1.77	89.29±0.13	70.17±0.55
EGCN - UR	90.43±0.18	46.06±0.31	89.81±0.56	43.25±2.75	89.45±0.32	70.81±1.11
EGCN - UR -TV(**Ours**)	91.57±0.12	46.44±0.18	**90.69**±0.46	46.77±2.90	**92.21**±0.42	**72.13**±1.63
GPN	82.82±3.21	40.96±2.50	82.16±1.25	46.30±3.07	79.66±3.82	59.30±0.59
GPN - UR	93.63±0.62	48.71±1.87	84.75±0.76	49.57±1.07	88.40±0.80	62.01±0.81
GPN -UR -TV (**Ours**)	**94.55**±0.23	**51.84**±0.72	87.29±1.04	**52.02**±2.36	88.78±1.71	63.11±1.11

teacher. As experimental evidence, we directly use the alpha teacher to do the OOD detection and get 87.24% and 87.2% ROC values on UH and KSC respectively, and 69.88% on UP.

It is also worth noting that ROC and PR values are not always consistent. For example on UH, the GPN-UR-TV has higher PR but shows lower ROC than EGCN-UR-TV. ROC and PR offer different perspectives to measure the quality of a ranking on data points for separating positives and negatives. Davis & Goadrich (2006) pointed out that algorithms that optimize AUROC are not guaranteed to optimize AUPR and vice versa. Yuan et al. (2023) also reported similar empirical observations for classification tasks. A low AUROC but a high AUPR for GPN-UR-TV indicates that the GPN-based produces more true positives among the top-ranked nodes than EGCN-based, while EGCN-based can separate true positives and negatives better than GPN-based among the lower-ranked nodes.

To further illustrate the above observation, Figure 3 displays example curves of ROC and PR on UP dataset with "shadows" selected as the OOD class. Although TLRSR and RGAE exhibit impressive ROC performance (over 93%), their PR outcomes are notably poor (below 12%), in contrast to the PR of our proposed framework (over 93%). A high ROC with an extremely low PR for a balanced dataset means that the model tends to produce tremendous false positive errors. For example, most nodes have very comparably high predicted uncertainty scores, which are not distinguishable for ID and OOD.

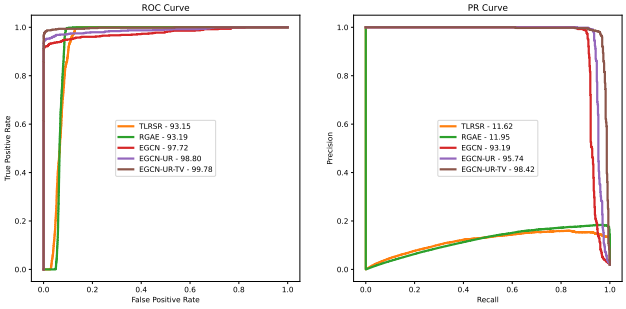

Figure 3: OOD detection on UP with "shadows" as OOD class

Ablation Study. We highlight the contributions of the proposed two terms by comparing the results of EGCN and GPN variants in Table 1 and Table 2. The key findings are: (1) For misclassification detection, the UR and TV can improve the performance of EGCN and GPN. Specifically, UR promotes 5 out of 6 cases, TV promotes 3 out of 6 cases. (2) For OOD detection, when we apply the proposed UR term to EGCN and GPN, they both show significantly promoted performance on all datasets of up to 10% in ROC and 7% in PR on UP dataset with GPN-UR. We then apply TV regularization on EGCN-UR and GPN-UR and the performance is further improved on all datasets. For instance, GPN-UR-TV increased 1% in ROC and 3.1% in PR on UP dataset compared to GPN-UR.

5 Conclusion

We proposed a graph-based uncertainty quantification framework for HSIC. We analyzed the limitations of ENN models based on the UCE loss. To mitigate the limitations, we leveraged inherent physical characteristics of HS data and edge-preserving regularization to propagate evidence in the spatial domain, leading to unmixing regularization (UR) and evidence-based total variation (TV), respectively, both are novel in GNN and hyperspectral literature. We conducted experiments on three datasets to demonstrate the effectiveness of the proposed regularizations. As the effectiveness of the UR term largely relies on the performance of hyperspectral unmixing, we will develop a more stable HSCI model subject to errors introduced by inaccurate mixing model and mixing matrix (please refer to Appendix G for limitations of the proposed approach). Another future direction lies in the scenario with multiple OOD material categories (as opposed to only one in this work).

REFERENCES

Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021.

Muhammad Ahmad, Adil Mehmood Khan, and Rasheed Hussain. Graph-based spatial–spectral feature learning for hyperspectral image classification. *IET image processing*, 11(12):1310–1316, 2017.

Muhammad Ahmad, Sidrah Shabbir, Swalpa Kumar Roy, Danfeng Hong, Xin Wu, Jing Yao, Adil Mehmood Khan, Manuel Mazzara, Salvatore Distefano, and Jocelyn Chanussot. Hyperspectral image classification—traditional to deep models: A survey for future prospects. *IEEE journal of selected topics in applied earth observations and remote sensing*, 15:968–999, 2021.

Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Jean-Christophe Gagnon-Audet, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 34:3438–3450, 2021.

Hamail Ayaz, Muhammad Ahmad, Ahmed Sohaib, Muhammad Naveed Yasir, Martha A Zaidan, Mohsin Ali, Muhammad Hussain Khan, and Zainab Saleem. Myoglobin-based classification of minced meat using hyperspectral imaging. *Applied sciences*, 10(19):6862, 2020.

Emrecan Bati, Akın Çalışkan, Alper Koz, and A Aydin Alatan. Hyperspectral anomaly detection method based on auto-encoder. In *Image and Signal Processing for Remote Sensing XXI*, volume 9643, pp. 220–226. Spie, 2015.

Viktor Bengs, Eyke Hüllermeier, and Willem Waegeman. Pitfalls of epistemic uncertainty quantification through loss minimisation. In *Advances in Neural Information Processing Systems*, 2022.

Jignesh S Bhatt and Manjunath V Joshi. Deep learning in hyperspectral unmixing: A review. In *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 2189–2192. IEEE, 2020.

Marin Biloš, Bertrand Charpentier, and Stephan Günnemann. Uncertainty on asynchronous time event prediction. *Advances in Neural Information Processing Systems*, 32, 2019.

Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. *Advances in Neural Information Processing Systems*, 33:1356–1367, 2020.

Bertrand Charpentier, Oliver Borchert, Daniel Zügner, Simon Geisler, and Stephan Günnemann. Natural posterior network: Deep bayesian predictive uncertainty for exponential family distributions. In *International Conference on Learning Representations*, 2022.

Bohan Chen, Yifei Lou, Andrea L. Bertozzi, and Jocelyn Chanussot. Graph-based active learning for nearly blind hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–16, 2023. doi: 10.1109/TGRS.2023.3313933.

Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected topics in applied earth observations and remote sensing*, 7(6):2094–2107, 2014.

Liam Collins, Hamed Hassani, Mahdi Soltanolkotabi, Aryan Mokhtari, and Sanjay Shakkottai. Provable multi-task representation learning by two-layer relu neural networks. *arXiv preprint arXiv:2307.06887*, 2023.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

Christian Debes, Andreas Merentitis, Roel Heremans, Jürgen Hahn, Nikolaos Frangiadakis, Tim van Kasteren, Wenzhi Liao, Rik Bellens, Aleksandra Pižurica, Sidharta Gautama, et al. Hyperspectral and lidar data fusion: Outcome of the 2013 grss data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):2405–2418, 2014.

Lucas Drumetz, Miguel-Angel Veganzones, Simon Henrot, Ronald Phlypo, Jocelyn Chanussot, and Christian Jutten. Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability. *IEEE Trans. Image Process.*, 25(8):3890–3905, 2016. doi: 10.1109/TIP.2016.2579259.

Lucas Drumetz, Jocelyn Chanussot, and Christian Jutten. Spectral unmixing: A derivation of the extended linear mixing model from the hapke model. *IEEE Geoscience and Remote Sensing Letters*, 17(11):1866–1870, 2019.

Ganghui Fan, Yong Ma, Xiaoguang Mei, Fan Fan, Jun Huang, and Jiayi Ma. Hyperspectral anomaly detection with robust graph autoencoders. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*, 2018.

Renlong Hang, Qingshan Liu, Danfeng Hong, and Pedram Ghamisi. Cascaded recurrent neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5384–5394, 2019.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Danfeng Hong, Naoto Yokoya, Jocelyn Chanussot, and Xiao Xiang Zhu. An augmented linear mixing model to address spectral variability for hyperspectral unmixing. *IEEE Transactions on Image Processing*, 28(4):1923–1938, 2018.

Danfeng Hong, Lianru Gao, Jing Yao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Graph convolutional networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):5966–5978, 2020.

Xing Hu, Chun Xie, Zhe Fan, Qianqian Duan, Dawei Zhang, Linhua Jiang, Xian Wei, Danfeng Hong, Guoqiang Li, Xinhua Zeng, et al. Hyperspectral anomaly detection using deep learning: A review. *Remote Sensing*, 14(9):1973, 2022.

M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza. Total variation spatial regularization for sparse hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens.*, 50(11):4484–4502, 2012.

Tao Jiang, Yunsong Li, Weiying Xie, and Qian Du. Discriminative reconstruction constrained generative adversarial network for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 58(7):4666–4679, 2020.

Audun Josang, Jin-Hee Cho, and Feng Chen. Uncertainty characteristics of subjective opinions. In *2018 21st International Conference on Information Fusion (FUSION)*, pp. 1998–2005. IEEE, 2018.

Audun Jsang. *Subjective Logic: A formalism for reasoning under uncertainty*. Springer Publishing Company, Incorporated, 2018.

Muhammad Hussain Khan, Zainab Saleem, Muhammad Ahmad, Ahmed Sohaib, Hamail Ayaz, and Manuel Mazzara. Hyperspectral imaging for color adulteration detection in red chili. *Applied Sciences*, 10(17):5955, 2020.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *International conference on machine learning*, pp. 5436–5446. PMLR, 2020.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Wei Li, Guodong Wu, and Qian Du. Transferred deep learning for anomaly detection in hyperspectral imagery. *IEEE Geoscience and Remote Sensing Letters*, 14(5):597–601, 2017.

Qingshan Liu, Feng Zhou, Renlong Hang, and Xiaotong Yuan. Bidirectional-convolutional lstm based spectral-spatial feature learning for hyperspectral image classification. *Remote Sensing*, 9 (12):1330, 2017.

Xiaoqiang Lu, Wuxia Zhang, and Ju Huang. Exploiting embedding manifold of autoencoders for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 58(3): 1527–1537, 2019.

Haobo Lyu and Hui Lu. Learning a transferable change detection method by recurrent neural network. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 5157–5160. IEEE, 2016.

Yao Ma, Xiaorui Liu, Neil Shah, and Jiliang Tang. Is homophily a necessity for graph neural networks? *arXiv preprint arXiv:2106.06134*, 2021.

Andrey Malinin, Anton Ragni, Kate Knill, and Mark Gales. Incorporating uncertainty into deep learning for spoken language assessment. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 45–50, 2017.

Ekaterina Merkurjev, Justin Sunu, and Andrea L Bertozzi. Graph mbo method for multiclass segmentation of hyperspectral stand-off detection video. In *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 689–693. IEEE, 2014.

Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta numerica*, 8:143–195, 1999.

Anyong Qin, Zhaowei Shang, Jinyu Tian, Yulong Wang, Taiping Zhang, and Yuan Yan Tang. Spectral–spatial graph convolutional networks for semisupervised hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 16(2):241–245, 2018.

Jing Qin, Harlin Lee, Jocelyn T Chi, Lucas Drumetz, Jocelyn Chanussot, Yifei Lou, and Andrea L Bertozzi. Blind hyperspectral unmixing based on graph total variation regularization. *IEEE Transactions on Geoscience and Remote Sensing*, 59(4):3338–3351, 2020.

Zainab Saleem, Muhammad Hussain Khan, Muhammad Ahmad, Ahmed Sohaib, Hamail Ayaz, and Manuel Mazzara. Prediction of microbial spoilage and shelf-life of bakery products through hyperspectral imaging. *IEEE Access*, 8:176986–176996, 2020.

Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.

Farideh Foroozandeh Shahraki and Saurabh Prasad. Graph convolutional neural networks for hyperspectral data classification. In *2018 IEEE global conference on signal and information processing (GlobalSIP)*, pp. 968–972. IEEE, 2018.

Maximilian Stadler, Bertrand Charpentier, Simon Geisler, Daniel Zügner, and Stephan Günnemann. Graph posterior network: Bayesian predictive uncertainty for node classification. *Advances in Neural Information Processing Systems*, 34:18033–18048, 2021.

Sheng Wan, Chen Gong, Ping Zhong, Shirui Pan, Guangyu Li, and Jian Yang. Hyperspectral image classification with context-aware dynamic graph convolutional network. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1):597–612, 2020.

Minghua Wang, Qiang Wang, Danfeng Hong, Swalpa Kumar Roy, and Jocelyn Chanussot. Learning tensor low-rank representation for hyperspectral anomaly detection. *IEEE Transactions on Cybernetics*, 53(1):679–691, 2022.

Qitian Wu, Yiting Chen, Chenxiao Yang, and Junchi Yan. Energy-based out-of-distribution detection for graph neural networks. *arXiv preprint arXiv:2302.02914*, 2023.

Weiying Xie, Xin Zhang, Yunsong Li, Jie Lei, Jiaojiao Li, and Qian Du. Weakly supervised low-rank representation for hyperspectral anomaly detection. *IEEE Transactions on Cybernetics*, 51 (8):3889–3900, 2021.

Yichu Xu, Lefei Zhang, Bo Du, and Liangpei Zhang. Hyperspectral anomaly detection based on machine learning: An overview. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3351–3364, 2022.

Zhuoning Yuan, Dixian Zhu, Zi-Hao Qiu, Gang Li, Xuanhui Wang, and Tianbao Yang. Libauc: A deep learning library for x-risk optimization. *arXiv preprint arXiv:2306.03065*, 2023.

Xujiang Zhao, Feng Chen, Shu Hu, and Jin-Hee Cho. Uncertainty aware semi-supervised learning on graph data. *Advances in Neural Information Processing Systems*, 33:12827–12836, 2020.

A PROOFS FOR THEORETICAL RESULTS

A.1 PROOFS FOR LIMITATIONS OF UCE AND EXISTING REGULARIZATION TECHNIQUES

Proposition 1. *Suppose $\mathbf{z} \in \mathcal{D} \subset \mathbb{R}^D$ is a data point in the embedded space and $y \in \{-1, +1\}$ is its binary class label. An MLP-based ENN has the lower and upper bounds for the UCE loss:*

$$\frac{1}{e_y(\mathbf{z};\boldsymbol{\theta}) + 1} \leq \mathit{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) \leq \frac{\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil + 1}{e_y(\mathbf{z};\boldsymbol{\theta})}, \tag{18}$$

where $e_y(\mathbf{z};\boldsymbol{\theta})$ refers to the output evidence for the classs y and $\boldsymbol{\alpha}(\mathbf{z}) = \mathbf{e}(\mathbf{z};\boldsymbol{\theta}) + 1$. If $\boldsymbol{\theta}$ can predict y correctly: $y(e_+(\mathbf{z};\boldsymbol{\theta}) - e_-(\mathbf{z};\boldsymbol{\theta})) > 0$, we have the following tighter upper bound:

$$\mathit{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) \leq \overline{\mathit{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) := \frac{r+1}{e_y(\mathbf{z};\boldsymbol{\theta})}, \tag{19}$$

where $r = 0$ if the output activation function is ReLU and $r = 1$ if it is the exponential function.

Proof. The UCE loss function has the analytical form:

$$\begin{aligned} \mathrm{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) &= \mathbb{E}_{\boldsymbol{p}\sim\mathrm{Dir}(\boldsymbol{p}|\boldsymbol{\alpha}(\mathbf{z}))}\left[-\log \mathbb{P}(\boldsymbol{y}|\boldsymbol{p})\right] && (20)\\ &= \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + e_{-y}(\mathbf{z};\boldsymbol{\theta}) + 2) - \Psi(e_y + 1), && (21)\end{aligned}$$

where $\Psi(\cdot)$ is the digamma function and $-y$ refers to a different class label other than y. For example, if $y = +1$, then $-y = -1$ refers to the negative class. As $\Psi(\cdot)$ is a monotonic increasing function, we have the lower bound:

$$\begin{aligned} \mathrm{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) &= \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + e_{-y}(\mathbf{z};\boldsymbol{\theta}) + 2) - \Psi(e_y + 1) && (22)\\ &\geq \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + \lfloor e_{-y}(\mathbf{z};\boldsymbol{\theta})\rfloor + 2) - \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 1). && (23)\end{aligned}$$

It follows from the recurrence relation of the digamma function, i.e., $\Psi(x+1) = \Psi(x) + 1/x, \forall x > 0$ that a lower bound of the UCE loss function:

$$\begin{aligned} \mathrm{UCE}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y}; \boldsymbol{\theta}) &\geq \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + \lfloor e_{-y}(\mathbf{z};\boldsymbol{\theta})\rfloor + 2) - \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 1) && (24)\\ &\geq \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 2) - \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 1) && (25)\\ &= \frac{1}{e_y(\mathbf{z};\boldsymbol{\theta}) + 1}. && (26)\end{aligned}$$

Similarly, we can achieve an upper bound:

$$\begin{aligned} \mathrm{UCE}(\boldsymbol{\alpha}(\mathbf{z}), y; \boldsymbol{\theta}) &= \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + e_{-y}(\mathbf{z};\boldsymbol{\theta}) + 2) - \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 1) && (27)\\ &\leq \Psi(\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil + e_y(\mathbf{z};\boldsymbol{\theta}) + 2) - \Psi(e_y(\mathbf{z};\boldsymbol{\theta}) + 1) && (28)\\ &= \sum_{i=1}^{\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil+1} \frac{1}{e_y(\mathbf{z};\boldsymbol{\theta}) + i} \leq \frac{\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil + 1}{e_y(\mathbf{z};\boldsymbol{\theta})}. && (29)\end{aligned}$$

Since $\lceil e_{-y}(\mathbf{z};\boldsymbol{\theta})\rceil \leq e_{-y}(\mathbf{z};\boldsymbol{\theta}) + 1$, we get a desired upper bound as in (18).

In the separable case, the last MLP layer of the ENN can be defined as:

$$\mathbf{e}(\mathbf{z};\boldsymbol{\theta}) := [e_+(\hat{\mathbf{z}};\boldsymbol{\theta}), e_-(\hat{\mathbf{z}};\boldsymbol{\theta})] = [\sigma(\mathbf{w}^T\hat{\mathbf{z}} + b), \sigma(-\mathbf{w}^T\hat{\mathbf{z}} - b)], \tag{30}$$

where $\hat{\mathbf{z}}$ is the input to the last MLP layer $\boldsymbol{\theta} = \{\mathbf{w}, b\}$, $\mathbf{w} \in \mathbb{R}^D$, $b \in \mathbb{R}$, and $\sigma(\cdot)$ is the activation function (e.g. ReLU and exponential) that outputs evidence values. If the configuration $\boldsymbol{\theta}$ can separate the example $(\mathbf{x}, y)$ correctly: $y(e_+(\hat{\mathbf{z}};\boldsymbol{\theta}) - e_-(\hat{\mathbf{z}};\boldsymbol{\theta})) > 0$, we have that $e_+(\hat{\mathbf{z}};\boldsymbol{\theta}) > e_-(\hat{\mathbf{z}};\boldsymbol{\theta})$ for $y = +1$ and $e_+(\hat{\mathbf{z}};\boldsymbol{\theta}) < e_-(\hat{\mathbf{z}};\boldsymbol{\theta})$ for $y = -1$. We discuss two types of activation functions separately.

a) **Exponential function**: $\exp(\mathbf{w}^T\hat{\mathbf{z}} + b) > \exp(-\mathbf{w}^T\hat{\mathbf{z}} - b)$ for $y = +1$ and $\exp(\mathbf{w}^T\hat{\mathbf{z}} + b) < \exp(-\mathbf{w}^T\hat{\mathbf{z}} - b)$ for $y = -1$, which implies that $0 \leq \exp(-\mathbf{w}^T\hat{\mathbf{z}} - b) \leq 1$ for $y = +1$ and $0 \leq \exp(\mathbf{w}^T\hat{\mathbf{z}} + b) \leq 1$ for $y = -1$. It follows that: $\lceil e_{-y}(\hat{\mathbf{z}}, \boldsymbol{\theta})\rceil \leq 1$. Therefore, we obtain a tighter upper bound in Equation (19) with $r = 1$.

b) **ReLU function**: $\text{ReLU}(\mathbf{w}^T\hat{\mathbf{z}} + b) = 1$ and $\text{ReLU}(-\mathbf{w}^T\hat{\mathbf{z}} - b) = 0$ when $y = +1$. When $y = -1$, we $\text{ReLU}(\mathbf{w}^T\hat{\mathbf{z}} + b) = 0$ and $\text{ReLU}(-\mathbf{w}^T\hat{\mathbf{z}} - b) = 1$. we then have: $\lceil e_{-y}(\hat{\mathbf{z}}, \boldsymbol{\theta})\rceil = 0$. We obtain a tighter upper bound in Equation (19) with $r = 0$.

□

We note that the condition of separability: $y(e_+(\hat{\mathbf{z}};\boldsymbol{\theta}) - e_-(\hat{\mathbf{z}};\boldsymbol{\theta})) > 0$ implies that the training examples in the training set $\mathcal{D}$, i.e., $(\hat{\mathbf{z}}, y)$ can be correctly classified based on the projected class probabilities: $y(p_+(\hat{\mathbf{z}}) - p_-(\hat{\mathbf{z}})) > 0$, where $[p_+(\mathbf{z}), p_-(\hat{\mathbf{z}})] = [(e_+(\hat{\mathbf{z}};\boldsymbol{\theta})+1)/S, (e_-(\hat{\mathbf{z}};\boldsymbol{\theta})+1)/S]$ and $S = e_+(\hat{\mathbf{z}};\boldsymbol{\theta}) + e_-(\hat{\mathbf{z}};\boldsymbol{\theta}) + 2$.

Lemma 2. *Assume that the universal approximation property holds for an MLP-based ENN, i.e., the ENN can learn an arbitrary mapping function from the feature vector $\mathbf{z}$ to the evidence values of binary classes: $\mathbf{e}(\mathbf{z};\boldsymbol{\theta}) = [e_+(\mathbf{z};\boldsymbol{\theta}), e_-(\mathbf{z};\boldsymbol{\theta})]^T \in [0,\infty)^2$. Then, the UCE loss defined based on the train set $\mathcal{D} = \{(\mathbf{z}^i, y^i)\}_{i=1}^N$ approaches the infimum value 0, if the solution $\boldsymbol{\theta}^\star$ has the property: $e_y(\mathbf{z};\boldsymbol{\theta}^\star) \to +\infty$ and $e_{-y}(\mathbf{z};\boldsymbol{\theta}^\star) \to 0, \forall(\mathbf{z}, y) \in \mathcal{D}$.*

Proof. Thanks to the universal approximation property, the optimal solution $\boldsymbol{\theta}^\star$ can predict the Dirichlet distribution $Dir(\boldsymbol{\alpha}^\star)$ that has the minimal UCE loss for each training example $(\mathbf{x}, y) \in \mathcal{D}$, where $\boldsymbol{\alpha}^\star = \mathbf{e}(\mathbf{z};\boldsymbol{\theta}^\star) + \mathbf{1}$. Minimizing over θ is equivalent to minimizing $\boldsymbol{\alpha}$, i.e., for each example $(\mathbf{z}, y) \in \mathcal{D}$:

$$\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha}} UCE(\boldsymbol{\alpha}, y) = \mathbb{E}_{\mathbf{p}\in Dir(\boldsymbol{\alpha})}[\ell_{CE}(\mathbf{p}, y)], \tag{31}$$

where $\ell_{CE}(\mathbf{p}, y)$ is the standard cross entropy function. It holds that:

$$\mathbb{E}_{\mathbf{p}\in Dir(\boldsymbol{\alpha})}[\ell_{CE}(\mathbf{p}, y)] \geq \ell_{CE}(\mathbb{E}_{\mathbf{p}\in Dir(\boldsymbol{\alpha})}[\mathbf{p}], y). \tag{32}$$

Let $\bar{\boldsymbol{\alpha}}$ be the minimizer of the above lower bound:

$$\bar{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \ell_{CE}(\mathbb{E}_{\mathbf{p}\in Dir(\boldsymbol{\alpha})}[\mathbf{p}], y). \tag{33}$$

Let $\bar{\mathbf{p}} = \mathbb{E}_{\mathbf{p}\in Dir(\bar{\boldsymbol{\alpha}})}[\mathbf{p}]$. We can derive that $\bar{\mathbf{p}} = [\bar{p}_+, \bar{p}_-]$, where $\bar{p}_y = 1$ and $\bar{p}_{-y} = 0$. Let $Dir(\hat{\boldsymbol{\alpha}}) = \delta_{\bar{\mathbf{p}}}$, where $\delta(\cdot)$ is the delta function and $\mathbb{E}_{\mathbf{p}\in Dir(\hat{\boldsymbol{\alpha}})}[\mathbf{p}] = \delta_{\bar{\mathbf{p}}}$. Then,

$$\begin{aligned} UCE(\hat{\boldsymbol{\alpha}}, y) &= \mathbb{E}_{\mathbf{p}\in Dir(\hat{\boldsymbol{\alpha}})}[\ell_{CE}(\mathbf{p}, y)] &(34)\\ &= \ell_{CE}(\bar{\mathbf{p}}, y) &(35)\\ &= \ell_{CE}(\mathbb{E}_{\mathbf{p}\in Dir(\bar{\boldsymbol{\alpha}})}[\mathbf{p}], y). &(36)\end{aligned}$$

It follows that $UCE(\boldsymbol{\alpha}, y) \geq UCE(\hat{\boldsymbol{\alpha}}, y), \forall\boldsymbol{\alpha} \in [1,+\infty)^2$, and hence $\hat{\boldsymbol{\alpha}}$ is an optimal solution. As $Dir(\hat{\boldsymbol{\alpha}}) = \delta_{\bar{\mathbf{p}}}$, we conclude that $e_y(\mathbf{z};\boldsymbol{\theta}^\star) \to +\infty$ and $e_{-y}(\mathbf{z};\boldsymbol{\theta}^\star) \to 0$. □

Theorem 1. *We assume that (i) feature vectors belonging to classes $\{\pm 1\}$ follow Gaussian distributions with the same covariance matrix and the means $\pm\boldsymbol{\mu}$, respectively, i.e., $\mathbb{P}(\mathbf{z}, y) = \mathbb{P}(y = +1)\mathcal{N}(\mathbf{z};\boldsymbol{\mu},\boldsymbol{\Sigma}) + \mathbb{P}(y = -1)\mathcal{N}(\mathbf{z};-\boldsymbol{\mu},\boldsymbol{\Sigma})$, with $\mathbb{P}(y = +1) = \mathbb{P}(y = -1) = 0.5$; (ii) the optimal solutions $\boldsymbol{\theta}^\star$ that minimize $\mathbb{E}_{(\mathbf{z},y)\sim\mathbb{P}(\mathbf{z},y)}\overline{UCE}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y};\boldsymbol{\theta})$ can separate both classes: $e_y(\mathbf{z};\boldsymbol{\theta}) > e_{-y}(\mathbf{z};\boldsymbol{\theta}), \forall(\mathbf{z}, y)$. Let $\sigma(\cdot)$ be the exponential function. The optimal solution $\boldsymbol{\theta}^\star = (\mathbf{w}^\star, b^\star)$ is the same as the optimal solution of LDA, i.e.,*

$$\mathbf{w}^\star = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} \;\; \textit{and} \;\; b^\star = 0. \tag{37}$$

Proof. According to the assumption on Gaussian distributions for the two classes, the generalization $\overline{\text{UCE}}$ loss has the following relations:

$$\begin{aligned} &\mathbb{E}_{(\mathbf{z},y)\sim P(\mathbf{z},y)}\overline{\text{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y};\boldsymbol{\theta}) &(38)\\ =\;& \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})}\overline{\text{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), +\mathbf{1};\boldsymbol{\theta}) + \frac{\mathbf{1}}{\mathbf{2}}\mathbb{E}_{\mathbf{z}\sim\mathcal{N}(-\boldsymbol{\mu},\boldsymbol{\Sigma})}\overline{\text{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), -\mathbf{1};\boldsymbol{\theta}) &(39)\\ \propto\;& \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})}\left[\frac{2}{e_+(\mathbf{z};\boldsymbol{\theta})}\right] + \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(-\boldsymbol{\mu},\boldsymbol{\Sigma})}\left[\frac{2}{e_-(\mathbf{z};\boldsymbol{\theta})}\right] &(40)\\ =\;& \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})}2\exp(-\mathbf{w}^T\mathbf{z} - b) + \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(-\boldsymbol{\mu},\boldsymbol{\Sigma})}2\exp(\mathbf{w}^T\mathbf{z} + b) &(41)\\ \propto\;& \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})}\exp(-\mathbf{w}^T\mathbf{z} - b) + \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(-\boldsymbol{\mu},\boldsymbol{\Sigma})}\exp(\mathbf{w}^T\mathbf{z} + b). &(42)\end{aligned}$$

Given that $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we have $\mathbf{w}^T\mathbf{z} + b \sim \mathcal{N}(\mathbf{w}^T\boldsymbol{\mu} + b, \mathbf{w}^T\Sigma\mathbf{w})$. Based on the property: $\mathbb{E}[\exp(r)] = \exp(\mu + \sigma^2/2)$ for $r \sim \mathcal{N}(\mu, \sigma^2)$, we have that

$$\mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})}\left[\exp(-\mathbf{w}^T\mathbf{z} - b)\right] = \exp(-\mathbf{w}^T\boldsymbol{\mu} - b + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2) \tag{43}$$

$$\mathbb{E}_{\mathbf{z}\sim\mathcal{N}(-\boldsymbol{\mu},\boldsymbol{\Sigma})}\left[\exp(\mathbf{w}^T\mathbf{z} + b)\right] = \exp(-\mathbf{w}^T\boldsymbol{\mu} + b + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2). \tag{44}$$

Plugging Equations (43) and (44) into Equation (42), we have the following relations:

$$\begin{aligned}
& \mathbb{E}_{(\mathbf{z},y)\sim P(\mathbf{z},y)}\overline{\mathrm{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y}; \boldsymbol{\theta}) && (45)\\
\propto\ & \exp(-\mathbf{w}^T\boldsymbol{\mu} - b + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2) + \exp(-\mathbf{w}^T\boldsymbol{\mu} + b + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2) && (46)\\
=\ & \exp(-\mathbf{w}^T\boldsymbol{\mu} + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2)(\exp(-b) + \exp(b)). && (47)
\end{aligned}$$

The minimization problem of the generalization $\overline{\mathrm{UCE}}$ loss has the relations:

$$\begin{aligned}
& \min_{\boldsymbol{\theta}} \mathbb{E}_{(\mathbf{z},y)\sim P(\mathbf{z},y)}\overline{\mathrm{UCE}}(\boldsymbol{\alpha}(\mathbf{z}), \mathbf{y}; \boldsymbol{\theta}) && (48)\\
=\ & \min_{\mathbf{w},b} \exp(-\mathbf{w}^T\boldsymbol{\mu} + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2)(\exp(-b) + \exp(b)) && (49)\\
=\ & \min_{\mathbf{w}} \exp(-\mathbf{w}^T\boldsymbol{\mu} + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2) \min_{b}(\exp(-b) + \exp(b)) && (50)
\end{aligned}$$

The following problem has the optimal solution for b:

$$b^\star = 0 = \arg\min_{b}(\exp(-b) + \exp(b)) \tag{51}$$

Due to the monotonicity of the exponential function, we obtain the equivalent optimization problems,

$$\min_{\mathbf{w}} \exp(-\mathbf{w}^T\boldsymbol{\mu} + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2) = \min_{\mathbf{w}} -\mathbf{w}^T\boldsymbol{\mu} + \mathbf{w}^T\boldsymbol{\Sigma}\mathbf{w}/2. \tag{52}$$

By taking the gradient and setting it to zero, i.e., $-\boldsymbol{\mu} + \boldsymbol{\Sigma}\mathbf{w} = 0$, there is a closed-form solution for the optimal solution, given by

$$\mathbf{w}^\star = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}. \tag{53}$$

□

A.2 Gradient analysis of proposed Unmixing-based Regularization(UR term)

Proposition 2. *Assume the linear model equation 9 holds without noise, the gradient descent for minimizing the UR$(\boldsymbol{e})$ regularization increases the predicted evidence of ground-truth class for ID instances and decrease the total evidence for OOD instances with the corresponding pure material contained in the pixel; formally,*

a) For an instance $(\boldsymbol{x}^i, y^i)$ with feature matrix $\boldsymbol{x}^i = \boldsymbol{m}_{y^i}$ and $y^i \in \{1, \ldots, C\}$ is the ground truth ID class label, one has

$$\frac{\partial UR(\boldsymbol{e})}{\partial e^i_{y^i}} \leq 0. \tag{54}$$

b) For an OOD instance instance $(\boldsymbol{x}^i, y^i)$ with $\boldsymbol{x}^i = \boldsymbol{m}^*_o$ and $y^i = o \notin \{1, \ldots, C\}$

$$\sum_{c=1}^{C} \frac{\partial UR(\boldsymbol{e})}{\partial e^i_c} \geq 0. \tag{55}$$

Proof. Given an instance $(\boldsymbol{x}^i, y^i)$ with $\boldsymbol{x} \in \mathbb{R}^B$, $y^i \in \{1, \ldots, C, o\}$. The ID material signature $\boldsymbol{m}_c \in \mathbb{R}^B, c = \{1, \ldots, C\}$, The optimal OOD material signature $\boldsymbol{m}^*_o \in \mathbb{R}^B$; The subjective logic opinion $\omega^i = (\boldsymbol{b}^i, u^i)$ is based on model prediction $\boldsymbol{e}^i(\boldsymbol{\theta})$ (we omitted $\boldsymbol{\theta}$ in the following proof for brevity). $\boldsymbol{b}^i = \frac{\boldsymbol{e}^i}{C+\sum_{c=1}^{C} e^i_c}$, $u = \frac{C}{C+\sum_{c=1}^{C} e^i_c}$. The UR term can be formulated as

$$\mathrm{UR}(\boldsymbol{e}) = \sum\nolimits_{i\in\mathbb{V}} \|\boldsymbol{x}^i - \frac{\sum_{c=1}^{C} e^i_c \boldsymbol{m}_c}{C + \sum_{c=1}^{C} e^i_c} - \frac{C\boldsymbol{m}^*_o}{C + \sum_{c=1}^{C} e^i_c}\|^2_2. \tag{56}$$

Taking the partial derivative to e_k^i, which is the evidence scalar of class k for instance i, we obtain

$$\frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i} = -2\left(\boldsymbol{x}^i - \frac{\sum_{c=1}^C e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_o^*}{C + \sum_{c=1}^C e_c^i}\right)^T \frac{(\sum_{c\neq k} e_c^i + C)\boldsymbol{m}_k - \sum_{c\neq k} e_c^i \boldsymbol{m}_c - C\boldsymbol{m}_o^*}{(C + \sum_{c=1}^C e_c^i)^2}. \tag{57}$$

With the condition that $y^i \in \{1, \ldots, C\}$ and $\boldsymbol{x}^i = \boldsymbol{m}_{y^i}$, it indicates the instance has a pure ID material, then we have the partial gradient with respect to the ground truth class y^i as follows:

$$\begin{aligned}\frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_{y^i}^i} &= -2\frac{\left((C + \sum_{c\neq y_i} e_c^i)\boldsymbol{m}_{y^i} - (\sum_{c\neq y^i} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_0^*)\right)^T \left((\sum_{c\neq y^i} e_c^i + C)\boldsymbol{m}_{y^i} - (\sum_{c\neq y^i} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_0^*)\right)}{(C + \sum_{c=1}^C e_c^i)^3} \\ &= -2\frac{\|(C + \sum_{c\neq i} e_c^i)\boldsymbol{m}_i - (\sum_{c\neq y_i} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_0^*)\|_2^2}{(C + \sum_{c=1}^C e_c^i)^3} \leq 0.\end{aligned} \tag{58}$$

With the condition that $y^i = o$ and $\boldsymbol{x}^i = \boldsymbol{m}_o^*$, it indicates the instance has a pure OOD material. Then it follows from (57) that

$$\begin{aligned}\frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i} &= -2\frac{\left((C + \sum_{c=1}^C e_c^i)\boldsymbol{m}_o^* - (\sum_{c=1}^C e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_o^*)\right)^T \left((\sum_{c\neq k} e_c^i + C)\boldsymbol{m}_k - (\sum_{c\neq k} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_o^*)\right)}{(C + \sum_{c=1}^C e_c^i)^3} \\ &= \frac{-2\left(\sum_{c=1}^C e_c^i \boldsymbol{m}_o^* - \sum_{c=1}^C e_c^i \boldsymbol{m}_c\right)^T}{(C + \sum_{c=1}^C e_c)^3} \cdot \left((\sum_{c\neq k} e_c^i + C)\boldsymbol{m}_k - (\sum_{c\neq k} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_o^*)\right).\end{aligned} \tag{59}$$

The gradient descent to update the total evidence can be expressed as

$$\begin{aligned} e_k^i &:= e_k^i - \delta \frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i} \\ \sum_{k=1}^C e_k^i &:= \sum_{k=1}^C e_k^i - \delta \sum_{k=1}^C \frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i}, \end{aligned} \tag{60}$$

where δ is the learning rate and the summation of class-wise gradient is calculated by

$$\begin{aligned}\sum_{k=1}^C \frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i} &= \frac{-2\left(\sum_{c=1}^C e_c^i \boldsymbol{m}_o^* - \sum_{c=1}^C e_c^i \boldsymbol{m}_c\right)^T}{(C + \sum_{c=1}^C e_c)^3} \cdot \left(\sum_{k=1}^C (\sum_{c\neq k} e_c^i + C)\boldsymbol{m}_k - \sum_{k=1}^C (\sum_{c\neq k} e_c^i \boldsymbol{m}_c + C\boldsymbol{m}_o^*)\right) \\ &= \frac{-2\left(\sum_{c=1}^C e_c^i \boldsymbol{m}_o^* - \sum_{c=1}^C e_c^i \boldsymbol{m}_c\right)^T}{(C + \sum_{c=1}^C e_c)^3} \cdot \left(\sum_{k=1}^C (\sum_{c\neq k} e_c^i)\boldsymbol{m}_k + C\sum_{c=1}^C \boldsymbol{m}_c - \sum_{k=1}^C (C-1)e_k^i \boldsymbol{m}_k - C^2 \boldsymbol{m}_o^*\right) \\ &= \frac{-2\left(\sum_{c=1}^C e_c^i \boldsymbol{m}_o^* - \sum_{c=1}^C e_c^i \boldsymbol{m}_c\right)^T}{(C + \sum_{c=1}^C e_c)^3} \cdot \left(C\sum_{c=1}^C (\boldsymbol{m}_c - \boldsymbol{m}_o^*) + \sum_{k=1}^C (\sum_{c\neq k} e_c^i - (C-1)e_k^i)\boldsymbol{m}_k\right).\end{aligned} \tag{61}$$

Without loss of generality, we assume that there is only one ID class, i.e. $C = 1$, then we have

$$\sum_{k=1}^C \frac{\partial \text{UR}(\boldsymbol{e})}{\partial e_k^i} = \frac{2C^2(\sum_{c=1}^C e_c)}{(C + \sum_{c=1}^C e_c)^3} \cdot \|\boldsymbol{m}_o - \boldsymbol{m}_{ID}\|_2^2 \geq 0. \tag{62}$$

□

A.3 Analytical solution for signature of OOD material

Given the feature set of the whole graph $\{\boldsymbol{x}^i, i \in \mathbb{V}\}$ and each instance has an associated evidence vector $\boldsymbol{e}^i(\boldsymbol{\theta})$ with fixed $\boldsymbol{\theta}$. Then the optimal OOD material's signature with minimizing $\text{UR}(\boldsymbol{m}_o)$ over the graph has analytical form as

$$\boldsymbol{m}_o = \frac{\sum_{i\in\mathbb{V}} (\boldsymbol{x}^i - \frac{\sum_c e_c^i \boldsymbol{m}_c}{S^i}) \frac{1}{S^i}}{\sum_{i\in\mathbb{V}} \frac{C}{S^{i2}}} \tag{63}$$

Proof. The summation of UR term over graph is as follows:

$$\mathrm{UR}(\boldsymbol{m}_o) = \sum_{i \in \mathbb{V}} \|\boldsymbol{x}^i - \frac{\sum_c e_c^i \boldsymbol{m}_c}{S^i} - \frac{C}{S^i}\boldsymbol{m}_o\|_2^2 \tag{64}$$

Take the derivative of Equation 64 with respect to $\boldsymbol{m}_o$, we have

$$\begin{aligned} \frac{\partial \mathrm{UR}(\boldsymbol{m}_o)}{\partial \boldsymbol{m}_o} &= 2\sum_{i \in \mathbb{V}} (\boldsymbol{x}^i - \frac{\sum_c e_c^i \boldsymbol{m}_c}{S^i} - \frac{C}{S^i}\boldsymbol{m}_o)(-\frac{C}{S^i}) \\ &= -2C\left[\sum_{i \in \mathbb{V}} (\boldsymbol{x}^i - \frac{\sum_c e_c^i \boldsymbol{m}_c}{S^i})\frac{1}{S^i} - \sum_{i \in \mathbb{V}} \frac{C}{S^{i^2}}\boldsymbol{m}_o\right] \end{aligned} \tag{65}$$

When $\frac{\partial \mathrm{UR}(\boldsymbol{m}_o)}{\partial \boldsymbol{m}_o} = \mathbf{0}$, we have

$$\boldsymbol{m}_o = \frac{\sum_{i \in \mathbb{V}} (\boldsymbol{x}^i - \frac{\sum_c e_c^i \boldsymbol{m}_c}{S^i})\frac{1}{S^i}}{\sum_{i \in \mathbb{V}} \frac{C}{S^{i^2}}}. \tag{66}$$

For a dataset, in which the feature vectors for ID nodes coincide with the material signature and predicted evidence is sparse for ID pixels and is all zero for OOD pixels, then we have:

$$\boldsymbol{m}_o = \frac{\sum_{i \in \mathbb{V}^{OOD}} \boldsymbol{x}^i}{|\mathbb{V}^{OOD}|}. \tag{67}$$

□

B Dataset Details

HSI captures numerous images at various wavelengths for a given spatial region. Unlike the human eye, which possesses only three color receptors sensitive to blue, green, and red light, HSI precisely measures the complete spectrum of light for every pixel in the scene acquired by sensors such as Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor. It offers detailed wavelength resolution not just within the visible range but also in the near-infrared range.

We use three HSI datasets that are widely used in the HSIC task and details are shown in Table 3. "Spatial" is the 2D dimension in terms of width and height, "Spectral" is the spectral bands, i.e. number of features for each pixel within the "Wavelength" range (nm). "Labeled pixels" counts the labeled pixels in the ground truth datasets and we do not care about the unlabeled pixels. "Training ratio" calculates the proportion of training pixels across all labeled pixels. The "homophily score" measures how likely nodes with the same label are near each other in a graph. The three datasets we used all show high homophily properties (exceeding 79% homophily score), and it indicates that pixels with similar spectral features tend to belong to the same category.

Table 3: Summary of the HSI Datasets used for experimental evaluation

	UP	UH	KSC
Spatial	610 x 610	340 x 1905	512 x 614
Spectral	103	144	176
Wavelength	430-860	0.35-1.05	400-2500
Labeled pixels	42,776	17,270	4,364
Categories	9	15	13
Training ratio	8.67%	27%	7.72%
Homophily score	0.7913	0.7911	0.8109

The Pavia University dataset was acquired by a Reflective Optics System Imaging Spectrometer (ROSIS) sensor during a flight campaign over the university campus at Pavia, Northern Italy. The detailed class description and training/validation/test ratio are presented in Table 4 following (Hong et al., 2020) [1].

[1] https://github.com/danfenghong/IEEE_TGRS_GCN

Table 4: Land-cover classes of the Pavia University (UP) dataset

Class No.	Class Name	Training	Validation	Test
0	Asphalt	327	1260	5044
1	Meadows	503	3629	14517
2	Gravel	284	363	1452
3	Trees	152	582	2330
4	Painted metal sheets	232	222	891
5	Bare Soil	457	914	3658
6	Bitumen	349	196	785
7	Self-Blocking Bricks	318	672	2692
8	Shadows	152	159	636
	Total	2774	7997	32005

The University of Houston dataset is collected by the Compact Airborne Spectrographic Imager (CASI) and released as a data fusion contest by The IEEE Geoscience and Remote Sensing Society. Table 5 presents the classes and dataset split following the contest [2].

Table 5: Land-cover classes of the HOUSTON2013 (UH) dataset

Class No.	Class Name	Training	Validation	Test
0	Healthy grass	198	235	941
1	Stressed grass	190	252	1012
2	Artificial turf	227	113	455
3	Evergreen trees	188	215	861
4	Deciduous trees	186	222	890
5	Bare earth	196	28	115
6	Water	196	256	1024
7	Residential buildings	191	232	931
8	Non-residential buildings	193	272	1089
9	Roads	191	246	987
10	Sidewalks	234	266	1066
11	Crosswalks	192	247	990
12	Major thoroughfares	246	77	309
13	Highways	213	60	240
14	Railways	227	114	457
	Total	3068	2835	11367

The Kennedy Space Center (KSC) dataset was gathered by Airborne Visible/Infrared Imaging Spectrometer (AVIRIS). There is no widely-used public split for KSC and we randomly pick 20 nodes from each class for training following (Kipf & Welling, 2016). The detailed class description and training/validation/test ratio are presented in Table 6.

Table 6: Land-cover classes of the Kennedy Space Center(KSC) dataset

Class No.	Class Name	Training	Validation	Test
0	Scrub	20	111	504
1	Willow swamp	20	33	152
2	Cabbage palm hammock	20	35	160
3	Cabbage palm/oak hammock	20	34	158
4	Slash pine	20	21	96
5	Oak/broadleaf hammock	20	31	142
6	Hardwood swamp	20	12	58
7	Graminoid marsh	20	61	280
8	Spartina marsh	20	75	340
9	Cattail marsh	20	57	261
10	Salt marsh	20	59	272
11	Mud flats	20	72	328
12	Wate	20	136	616
	Total	260	737	3367

C GRAPH CONSTRUCTION

We construct an undirected graph based on the relations between spectral features of pixels. Specifically, pixels that have similar features are more likely to connect with each other. Considering most

[2] http://www.grss-ieee.org/community/technical-committees/data-fusion/2013-ieee-grss-data-fusion-contest/

HSI datasets only have part of the pixels labeled, we also only build the graph based on labeled nodes following Hong et al. (2020).

We model all pixels in one HSI scene as a graph $\mathcal{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} \in \{1, 2, \ldots, N\}$ denotes the vertex set and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ is the edge set. Edges can be represented as a weighted adjacency matrix $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ and the weight is calculated with the radial basis function of the similarity between two node features Qin et al. (2020),

$$\boldsymbol{W}_{ij} = \exp^{-d(\boldsymbol{x}^i, \boldsymbol{x}^j)/\sigma}$$

where $d(\boldsymbol{x}^i, \boldsymbol{x}^j)$ is the distance between two vertices i and j, such as the Euclidean distance or cosine similarity, and $\sigma > 0$ is a control parameter for the similarity. We use the cosine similarity for scale invariance based on the observation that illumination alters the scaling of spectra while preserving their overall shape in the spectral domain (Merkurjev et al., 2014) and avoids the curse of dimensionality.

$$d(\boldsymbol{x}^i, \boldsymbol{x}^j) = 1 - \frac{< \boldsymbol{x}^i, \boldsymbol{x}^j >}{\|\boldsymbol{x}^i\| \|\boldsymbol{x}^j\|}$$

To improve the computation efficiency with better scalability, we only keep the first K nearest neighbors for each node to build a sparse graph. K and σ are hyperparameters. $K = 50$ and $\sigma = 0.1$ in our case.

The built graph exhibits a strong homophily characteristic where nodes typically associate with others that are "similar" or "comparable" from the perspective of their respective categories. It's been shown that GCN manages such highly homophilic graphs effectively (Ma et al., 2021).

D Model Details

For our comparative analysis, we employ five baseline methods. Initially, we contrast our approach with a classification model that utilizes entropy as its uncertainty metric. Subsequently, we select two representative anomaly detection techniques, as highlighted in a recent review (Xu et al., 2022). Additionally, we incorporate two state-of-the-art uncertainty quantification models designed for semi-supervised node classification.

For GCN-based models, we use two graph convolution layers and 0.5 dropout probability. Following the graph size, KSC, UP, and UH have hidden dimensions of 64, 128, 256, respectively We use early stopping with the patience of 30, a maximum of 5,000 epochs, and validation cross-entropy as a stop metric. For all models, we use the Adam optimizer, and the learning rate and weight decay are carefully tuned for each dataset.

Softmax-GCN We use classic two-layer GCNs optimized with Cross-Entropy loss following (Hendrycks & Gimpel, 2016) based on the assumption that correctly classified examples tend to have greater maximum softmax probabilities than erroneously classified and out-of-distribution examples.

Table 7: Hyperparamters for softmax-GCN Model

dataset	lr	wd
UP	1.00E-02	1.00E-05
UH	1.00E-02	1.00E-05
KSC	1.00E-03	1.00E-05

TLRSR Tensor Low-Rank and Sparse Representation model (Wang et al., 2022) first use a principal component analysis (PCA) method as one preprocessing step to exact a subset of HSI bands and then apply a TLR framework to preserve the inherent HSI 3-D structure, and finally extract the LR background part as the dictionary of TLRSR. The problem can be solved by the well-designed alternating direction method of multipliers (ADMMs). Following their parameter analysis, we also search λ and $\lambda^{'}$ from the set $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3\}$ as the original paper. The hyperparameters used in our experiments are in Table 8.

RGAE Robust Graph AutoEncoder (Fan et al., 2021) is a modified autoencoder framework combined with gradient normalization of each sample to make it more robust to noise and anomalies. Besides, it has a graph regularization term for preserving the local geometric structure of the given high-dimensional data. It is mentioned in the original paper that there are three hyperparameters that need to be tuned carefully. (1): Trade-off parameter λ that balances the regularization term and the range is set to $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$ (2) Number of superpixels S and the rage is set to $\{50, 100, 150, 300, 500\}$ (3) Dimension of hidden layers n_{hid} and the range is $\{20, 40, 60, 80, 100, 120, 140, 160\}$. We use the validation OOD ROC to pick the top performance models. The hyperparameters used in our experiments are in Table 8.

Table 8: Hyperparamters for anomaly detection baselines

Model	hyper parameters	UP				UH				KSC			
		4	6	7	8	0	1	2	10	5	6	7	12
RGAE	λ	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.01	0.1	0.1	0.1	0.1
	S	300	100	100	100	150	150	150	150	500	500	150	150
	n_{hid}	160	120	120	120	80	80	20	40	160	160	40	40
TLRSR	λ_1	0.3	0.001	0.001	0.2	0.2	0.2	0.01	0.3	0.2	0.01	0.001	0.3
	λ_2	0.01	0.01	0.01	0.01	0.001	0.001	0.005	0.3	0.05	0.001	0.01	0.3

EGCN-GKDE Similar to ENN, EGCN uses an activation layer instead of the softmax layer to output non-negative values as the parameters for the predicted Dirichlet distribution. The representation learning step uses GCN layers for graph learning. Graph-based Kernel Dirichlet distribution Estimation(GKDE) associated with EGCN is designed to estimate prior Dirichlet distribution parameters for each node, which is calculated based on the shortest path between test nodes and training nodes belonging to different classes on the graph. The necessary condition is that nodes with a high epistemic uncertainty are far away from training nodes and nodes with a high aleatoric uncertainty are near the boundary of classes. It may not show good performance when the condition is not satisfied, that is OOD is close to ID training nodes. We present the detailed form of GKDE for analysis in Section E.1.

In our experiments, we also integrate the GKDE teacher by default. we use $\beta = 0.5$ for KSC and UP, $\beta = 0.2$ for UH. For hyperparameter tuning, we first tune the learning rate (lr) and weight decay (wd), as well as the trade-off parameter for GKDE teacher in λ_2 based on the average result of ID accuracy and OOD/Misclassification ROC. We suppose there are OOD classes involved in the validation set for hyperparameter selection for all models. The hyperparameters used in our experiments are presented in Table9 and Table 10 for misclassification detection and OOD detection, respectively.

Table 9: Hyperparamters for Misclassification Models

dataset	GKDE					GPN				
	lr	wd	λ_1	λ_2	λ_3	lr	wd	λ_1	λ_2	λ_3
UP	1.00E-02	1.00E-04	1.00E-03	1.00E-04	1.00E-04	1.00E-03	5.00E-03	1.00E-03	1.00E-02	1.00E-02
UH	1.00E-02	1.00E-04	1.00E-02	1.00E-04	1.00E-05	1.00E-04	1.00E-04	1.00E-04	1.00E-01	1.00E-04
KSC	1.00E-02	1.00E-04	1.00E-03	1.00E-02	1.00E-04	1.00E-02	1.00E-03	1.00E-03	1.00E-04	1.00E-02

GPN GPN applies multi-layer perceptions for representation learning and then use normalizing flow for density estimation in the latent space. The graph structure is leverages for evidence propagation at last. In detail, there are three components. (1) A feature encoder $g_{\boldsymbol{\phi}}$ maps the original node feature $\boldsymbol{x}^i \in \mathbb{R}^B, i \in \mathbb{V}$ onto a low-dimensional latent space $\boldsymbol{z}^i \in \mathbb{R}^H$ with a simple two-layer multilayer perception (MLP) encoder, H is the latent dimension. i.e. $\boldsymbol{z}^i = g_{\boldsymbol{\phi}}(\boldsymbol{x}^i)$ and $\boldsymbol{\phi}$ is the encoder parameters. (2) A Radial normalizing flow $h_{\boldsymbol{\varphi}}$ estimates the density of the latent space per class, which is used to compute the pseudo evidence (class counts) $\beta_c^i := h_{\boldsymbol{\varphi}}(\boldsymbol{z}^i) = N_c \cdot \mathbb{P}(\boldsymbol{z}^i | c; \boldsymbol{\varphi})$ (3) A personalized page rank message passing scheme diffuses the pseudo counts (density multiplied by the number of training nodes) by taking the graph structures into account, i.e. $\alpha_c^i = \sum_{v \in \mathbb{V}} \prod_{i,v} \beta_c^v$ with $\prod_{i,v}$ is the dense PPR score reflecting the importance of node v on i. Similar to the GPN paper, we use 10 dimensions of latent space and 10 radial layers for normalizing flow and use CE loss to pretrain the flow layer, teleport is equal to 0.2 and propagation is iterated with 10 steps. For the parameters in the optimizer and tradeoffs for the loss function, the tuning process is the same as EGCN.

Those parameters used in our experiments are presented in Table9 and Table 10 for misclassification detection and OOD detection, respectively.

Table 10: Hyperparamters for OOD detedction Models

Dataset	hyper parameters	GPN lr	GPN wd	GPN λ_1	GPN λ_2	GPN λ_3	GKDE lr	GKDE wd	GKDE λ_1	GKDE λ_2	GKDE λ_3
UP	4	1.00E-04	5.00E-03	1.00E-04	1.00E-03	1.00E-03	1.00E-02	1.00E-05	1.00E-01	1.00E-04	1.00E-05
	6	1.00E-03	5.00E-04	1.00E-05	1.00E+00	1.00E-04	1.00E-02	1.00E-05	0.00E+00	1.00E-01	1.00E-05
	7	1.00E-03	1.00E-03	1.00E-03	1.00E-01	1.00E-05	1.00E-03	1.00E-05	0.00E+00	1.00E-02	1.00E-05
	8	1.00E-04	1.00E-04	1.00E-05	1.00E-01	1.00E-05	1.00E-03	1.00E-05	1.00E-02	1.00E-04	1.00E-05
UH	0	1.00E-03	5.00E-03	1.00E-03	1.00E-05	1.00E-05	1.00E-02	1.00E-05	1.00E-02	1.00E-03	1.00E-04
	1	1.00E-03	1.00E-03	1.00E-05	1.00E-05	1.00E-04	1.00E-02	1.00E-05	1.00E-02	1.00E-04	1.00E-05
	2	1.00E-04	1.00E-04	1.00E-04	1.00E+00	1.00E-03	1.00E-02	1.00E-05	1.00E-02	1.00E-04	1.00E-04
	10	1.00E-03	5.00E-04	1.00E-03	1.00E-01	1.00E-02	1.00E-02	1.00E-05	1.00E-04	1.00E-01	1.00E-05
KSC	5	1.00E-03	5.00E-03	1.00E-04	1.00E-04	1.00E-03	1.00E-03	1.00E-05	1.00E-02	1.00E-04	1.00E-03
	6	1.00E-03	5.00E-03	1.00E-04	1.00E+00	1.00E-03	1.00E-03	1.00E-05	1.00E-02	1.00E-01	1.00E-02
	7	1.00E-03	5.00E-03	1.00E-04	1.00E+00	1.00E-05	1.00E-03	1.00E-05	1.00E-02	1.00E-02	1.00E-05
	12	1.00E-03	5.00E-03	1.00E-04	1.00E-05	1.00E-01	1.00E-03	1.00E-05	1.00E-02	1.00E-04	1.00E-05

E Additional Results

E.1 Analysis on GKDE teacher

As our discussion in Section 3.1, GKDE assumes that OOD test nodes are far away in terms of graph-based distance from the training (ID) nodes compared to ID test nodes. This assumption is not always true in practice. We provide the GKDE form as follows:

$$h_c(y_i, d_{ij}) = \begin{cases} 0 & y^i \neq c \\ g(d_{ij}) & y^i = c, \end{cases}$$

with $g(d_{ij}) = \exp(-\frac{d_{ij}^2}{2\beta^2})$ and d_{ij} denoting the graph distance. The evidence prior $\hat{\boldsymbol{e}}_j = \sum_{i \in \mathbb{L}} \mathbf{h}(y^i, d_{ij})$ with $\mathbf{h}(y^i, d_{ij}) = [h_1, \ldots, h_c, \ldots, h_C]$. β is the bandwidth in Gaussian kernel function.

We first investigate the parameter β. On a specific OOD setting, pixels belonging to "shadows" in UP dataset are considered as OOD. Figure 4 shows the OOD detection performance for GKDE prior model with different β. With $\beta = 1$, GKDE can achieve 96.40% ROC and 70.13% PR value. $\beta = 0.1$ is random ranking result while $\beta = 10$ is a much worse result.

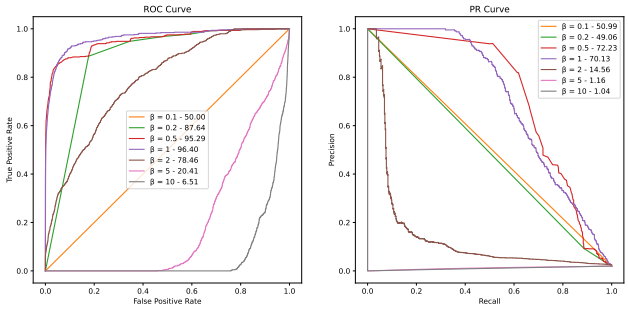

Figure 4: ROC and PR curves for different β of GKDE teacher in OOD detection setting

Figure 5 presents the predicted vacuity map across various values of β, compared to the ground truth map where OOD is labeled as 1, and ID is 0. Note that unlabeled data in the hyperspectral image is marked as 0 for all sub-figures. The choice of β can significantly influence OOD detection outcomes. Within a single sub-figure, a comparative analysis reveals a clearer distinction between OOD and ID regions when $\beta = 1$. For $\beta = 0.1$, nearly all pixels exhibit high vacuity, whereas for $\beta = 10$, they display low vacuity uniformly. Hence, adjusting β is crucial for accessing a good the GKDE prior.

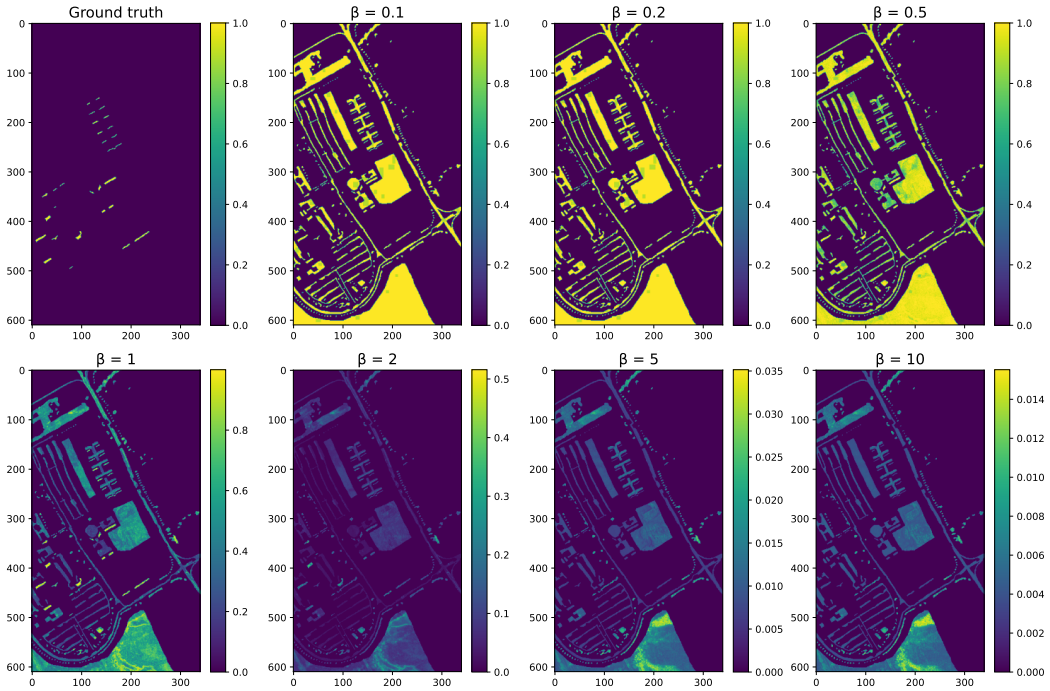

Figure 5: Predicted vacuity map for different β of GKDE teacher in OOD detection setting

In comparison to the GKDE model, which achieves 99.1% on ROC and 96.9% on PR, the GKDE prior doesn't perform as effectively as EGCN. As seen in Table 11 through 16, the performance disparity between GKDE and EGCN can reach up to 27% on ROC and 19% on PR. This suggests that the GKDE teachers may not yield optimal results. Conversely, in certain situations, GKDE can indeed enhance the learning process of EGCN.

E.2 Detailed Result

For OOD detection experiments, we use the mentioned 3 datasets. Within each dataset, we create four random configurations, and in each one, one class is picked as OOD. In the main paper, we display the weighted average, factoring in the count of test OOD nodes for every dataset. In this section, we show the detailed results for each of the twelve configurations.

Table 11 - 12 display the results for UP. It is worth noting that some classes are easier to classify while some are challenging. For example, class-4 is easily discernible by both anomaly detection and uncertainty quantification methods. However, the softmax-GCN struggles in this regard, whereas our introduced framework significantly outperforms the anomaly detection benchmark. This may indicate that class-4 class-4 possesses distinct features compared to other pixels in the image. On the other hand, for class-6, the PR values are suboptimal across all models, indicating that a considerable number of ID pixels have elevated predicted vacuities. When comparing class-7 and class-8 as the OOD class, models based on GPN fare better with class-7, while those based on EGCN excel with class-8. A similar trend is observed with the GKDE teacher, whucg performs commendably with class-8 but falls short with class-7.

Figure 6 presents the predicted vacuity map for different models when "shadows" (class-8) as the OOD class in UP. TLRSR and RGAE can not identify the OOD at all. EGCN tends to predict higher vacuity scores for OOD nodes (exceeding 0.8) while GKDE has a much lower vacuity score for all nodes (below 0.001). After applying UR term, there seem to be fewer false positives and lower vacuity scores for ID nodes after applying TV term.

We have a similar conclusion for the other two datasets. Table 13 - 14 present the result for UH dataset and Table 15 - 16 show the result for KSC.

Table 11: OOD Detection Result for UP

dataset	UP - 4			UP - 6		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	74.1±1.8	62.9±14.6	6.5±6.1	77.4±2.0	29.8±2.7	1.9±0.1
GKDE	68.5±n.a.	99.6±n.a.	96.8±n.a.	71.0±n.a.	59.8±n.a.	3.1±n.a.
RGAE	n.a.±n.a.	99.3±n.a.	86.2±n.a.	n.a.±n.a.	60.6±n.a.	2.9±n.a.
TLRSR	n.a.±n.a.	98.5±n.a.	71.4±n.a.	n.a.±n.a.	70.9±n.a.	3.8±n.a.
EGCN	75.5±1.9	99.9±0.0	97.8±0.3	**77.0**±2.4	72.3±0.9	4.1±0.1
EGCN - UR	74.6±1.2	99.9±0.0	97.1±0.8	76.3±0.7	90.3±0.2	10.4±0.1
EGCN - UR -TV	**76.6**±3.1	**100.0**±0.0	**99.8**±0.0	75.8±2.2	90.1±0.2	10.3±0.2
GPN	71.5±0.0	99.6±0.0	99.1±0.0	69.5±2.7	40.4±15.4	2.1±0.8
GPN - UR	71.5±0.1	99.6±0.0	99.1±0.0	50.9±1.7	89.9±1.4	10.5±1.4
GPN -UR -TV	63.1±0.6	99.6±0.0	98.9±0.1	51.7±2.0	**90.7**±0.6	**11.3**±0.8

Table 12: OOD Detection Result for UP (cont.)

dataset	UP - 7			UP - 8		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	**76.6**±2.6	73.2±3.9	27.4±4.0	75.3±1.5	14.2±5.5	1.3±0.5
GKDE	70.7±n.a.	57.0±n.a.	9.8±n.a.	68.7±n.a.	95.3±n.a.	72.2±n.a.
RGAE	n.a.±n.a.	70.3±n.a.	12.3±n.a.	n.a.±n.a.	96.1±n.a.	18.9±n.a.
TLRSR	n.a.±n.a.	62.3±n.a.	9.9±n.a.	n.a.±n.a.	93.1±n.a.	11.6±n.a.
EGCN	75.6±0.8	84.5±0.9	28.1±1.0	74.3±1.1	99.1±0.2	96.9±0.4
EGCN - UR	75.7±0.8	85.2±0.2	27.5±0.2	73.9±1.2	99.3±0.3	97.3±0.4
EGCN - UR -TV	73.0±0.5	87.3±0.1	26.9±0.2	**75.0**±0.3	**99.8**±0.0	**99.1**±0.1
GPN	65.2±2.6	86.5±1.3	26.3±1.9	68.1±0.3	96.0±0.6	69.6±10.8
GPN - UR	62.8±3.3	92.2±0.6	36.9±2.1	67.9±0.5	96.0±0.7	75.1±4.2
GPN -UR -TV	62.6±1.9	**93.2**±0.2	**40.9**±0.8	60.6±2.5	97.8±0.1	82.5±1.1

Table 13: OOD Detection Result for UH

dataset	Houston - 0			Hosuton - 1		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	66.8±2.9	81.9±2.0	46.4±0.9	67.7±2.2	13.8±3.5	5.0±0.2
GKDE	68.9±n.a.	93.6±n.a.	68.7±n.a.	68.6±n.a.	92.3±n.a.	66.5±n.a.
RGAE	n.a.±n.a.	85.6±n.a.	21.6±n.a.	n.a.±n.a.	46.0±n.a.	7.7±n.a.
TLRSR	n.a.±n.a.	48.9±n.a.	6.3±n.a.	n.a.±n.a.	49.6±n.a.	8.0±n.a.
EGCN	69.5±0.5	93.5±0.4	43.8±1.5	70.2±2.3	96.6±0.2	70.1±4.3
EGCN - UR	**71.7**±1.0	94.4±0.1	47.5±0.8	70.3±1.9	97.1±0.3	73.1±5.0
EGCN - UR -TV	71.3±1.2	96.0±0.7	54.2±6.2	**70.5**±1.0	97.0±0.4	70.9±1.9
GPN	66.7±0.4	99.1±0.3	87.8±7.2	64.5±1.4	96.5±1.8	63.4±3.4
GPN - UR	66.1±1.4	99.1±0.0	90.5±0.3	64.9±0.5	**98.1**±0.6	70.5±2.7
GPN -UR -TV	66.3±0.8	**99.2**±0.1	**90.6**±0.8	63.6±1.2	98.0±0.7	**75.6**±6.1

Table 14: OOD Detection Result for UH (cont.)

dataset	Houston - 2			Houston -10		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	65.9±1.7	62.0±4.9	4.9±0.5	73.4±0.4	73.2±1.4	14.7±0.6
GKDE	68.5±n.a.	92.3±n.a.	**56.9**±n.a.	71.4±n.a.	74.7±n.a.	**32.0**±n.a.
RGAE	n.a.±n.a.	63.8±n.a.	5.1±n.a.	n.a.±n.a.	24.6±n.a.	5.9±n.a.
TLRSR	n.a.±n.a.	49.4±n.a.	3.4±n.a.	n.a.±n.a.	48.2±n.a.	5.7±n.a.
EGCN	70.0±0.8	90.6±0.1	16.7±0.1	73.0±0.3	76.0±0.5	17.0±0.3
EGCN - UR	69.7±1.6	91.0±2.8	25.4±7.7	73.1±0.1	**78.3**±0.2	18.8±0.2
EGCN - UR -TV	**70.4**±0.3	**97.3**±0.4	44.9±4.2	**73.1**±0.6	77.1±0.4	18.1±0.3
GPN	64.1±0.3	70.1±0.6	6.1±0.1	70.9±0.7	58.7±1.9	10.5±0.4
GPN - UR	64.1±0.2	70.8±0.4	6.3±0.1	68.0±1.1	65.4±1.7	12.0±0.6
GPN -UR -TV	54.8±0.3	85.6±0.8	11.8±0.7	70.8±1.0	67.3±2.3	12.8±0.8

Table 15: OOD Detection Result for KSC

dataset	KSC - 5			KSC - 6		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	89.9±0.2	71.6±1.6	7.0±0.4	87.7±0.2	45.5±1.7	2.3±0.0
GKDE	87.5±n.a.	47.0±n.a.	5.1±n.a.	85.0±n.a.	43.2±n.a.	2.2±n.a.
RGAE	n.a.±n.a.	68.6±n.a.	6.9±n.a.	n.a.±n.a.	71.1±n.a.	4.1±n.a.
TLRSR	n.a.±n.a.	67.7±n.a.	6.0±n.a.	n.a.±n.a.	75.6±n.a.	3.1±n.a.
EGCN	89.7±0.0	64.6±0.0	7.1±0.0	87.6±0.1	17.3±0.8	1.9±0.0
EGCN - UR	89.6±0.0	64.4±0.1	7.0±0.0	87.6±0.0	19.5±0.9	1.9±0.0
EGCN - UR -TV	**90.0**±0.1	**76.7**±0.5	**12.0**±0.3	**87.7**±0.1	41.9±2.8	2.2±0.1
GPN	86.8±1.6	54.3±7.4	4.7±0.9	82.6±1.9	26.6±16.3	2.0±0.3
GPN - UR	86.3±2.2	58.7±2.6	5.0±0.3	80.7±0.9	86.7±1.9	7.8±1.4
GPN -UR -TV	88.5±0.6	58.3±9.0	5.2±0.9	82.1±0.6	90.4±2.5	**20.0**±9.0

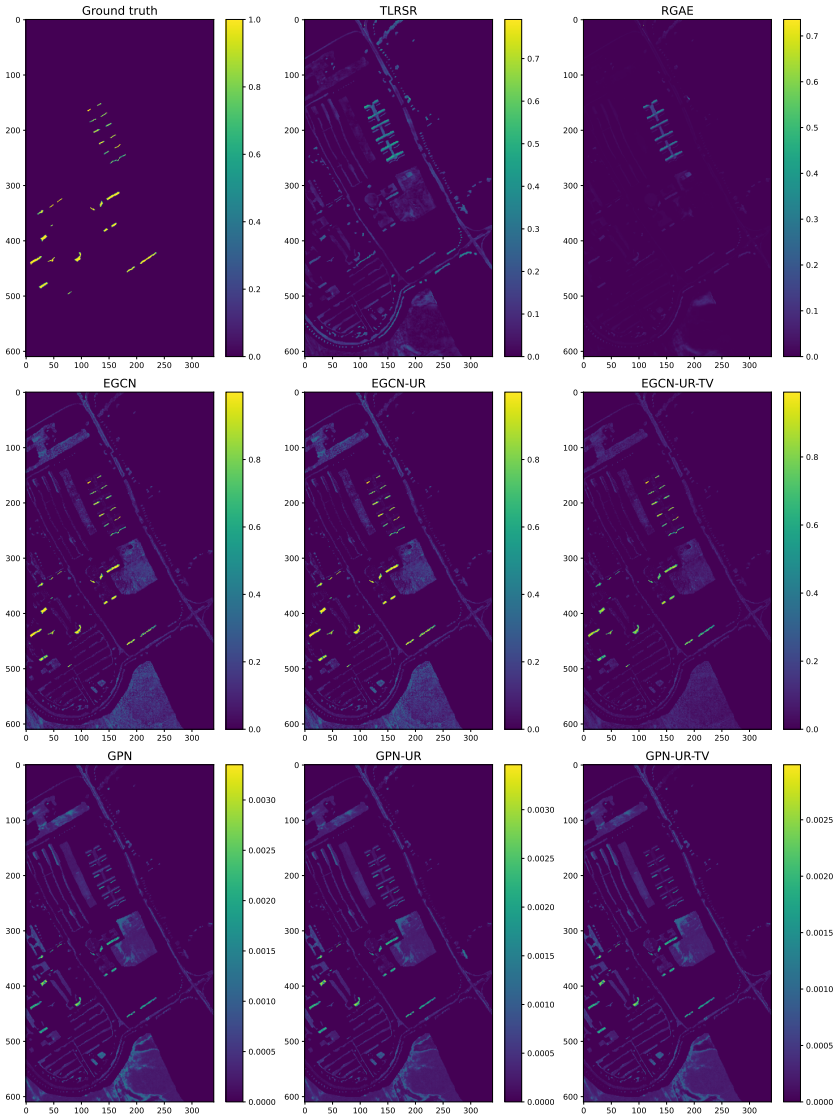

Figure 6: Predicted vacuity map for different models

Table 16: OOD Detection Result for KSC (cont.)

dataset	KSC - 7			KSC - 12		
	ID OA	OOD ROC	OOD PR	ID OA	OOD ROC	OOD PR
softmax-GCN	**88.0**±0.2	38.9±0.8	6.4±0.1	83.9±0.3	98.9±0.2	91.7±2.2
GKDE	85.8±n.a.	88.9±n.a.	**58.7**±n.a.	80.9±n.a.	100.0±n.a.	59.1±n.a.
RGAE	n.a.±n.a.	18.6±n.a.	4.8±n.a.	n.a.±n.a.	93.0±n.a.	56.6±n.a.
TLRSR	n.a.±n.a.	67.9±n.a.	14.7±n.a.	n.a.±n.a.	49.9±n.a.	9.1±n.a.
EGCN	87.8±0.3	93.5±0.3	51.1±2.2	84.3±0.1	100.0±0.0	99.9±0.0
EGCN - UR	87.7±0.2	**93.8**±1.0	53.6±4.3	**84.3**±0.1	100.0±0.0	99.9±0.0
EGCN - UR -TV	87.9±0.1	93.7±0.8	56.2±6.2	84.2±0.1	**100.0**±0.0	**99.9**±0.0
GPN	86.1±1.8	59.0±7.8	9.7±1.8	75.8±2.6	100.0±0.0	99.9±0.0
GPN - UR	82.1±1.1	78.6±1.4	18.9±2.7	77.4±0.9	100.0±0.0	99.9±0.0
GPN -UR -TV	81.6±1.6	79.5±1.6	20.6±2.0	81.0±0.7	**100.0**±0.0	**99.9**±0.0

F Related Work

Hyperspectral Imaging Analysis Due to the wealth of detailed spectral information available in each pixel, hyperspectral imaging (HSI) has found widespread application in various real-world sce-

narios. *HSI classification* (HSIC) aims to assign a distinct class label to each pixel. In their work, (Chen et al., 2014) utilized stacked auto-encoder networks for HS image classification by leveraging dimensionally-reduced HS images obtained through principal component analysis (PCA). Another study (Liu et al., 2017) introduced convolutional neural networks (CNNs) to effectively extract spatial-spectral features from HS images, resulting in improved classification performance. In separate work, a cascaded RNN was proposed (Hang et al., 2019) to utilize spectral information comprehensively for achieving high-accuracy HS image classification. Furthermore, (Hong et al., 2020) developed fusion modules that seamlessly integrate CNNs and miniGCNs in an end-to-end manner. We refer (Ahmad et al., 2021) for a complete review of HSI classification task. *HSI spectral unmixing* decompose the image into a collection of reference spectral signatures with associated proportions, which is a non-negative Matrix Factorization problem (Lee & Seung, 1999), which can be formed to blind and unblind problem with Linear mixing model (Qin et al., 2020) or Extended linear mixing model (Drumetz et al., 2019). We refer (Bhatt & Joshi, 2020) for a systemic introduction. *HSI anomaly detection* involves detecting pixels in an image whose spectral characteristics deviate significantly from the surrounding or overall background pixels and attracts a lot of interest. Deep learning-based methods can be divided into CNN-based (Li et al., 2017), autoencoder-based (Bati et al., 2015), GAN based (Jiang et al., 2020) and RNN based (Lyu & Lu, 2016). There are also some models that leverage other techniques, such as manifold learning (Lu et al., 2019), and low-rank representation (Xie et al., 2021). A comprehensive review is conducted by Hu et al. (2022).

Anomaly detection on HSI involves detecting pixels in an image whose spectral characteristics deviate significantly from the surrounding or overall background pixels and attracts a lot of interest Deep learning-based methods can be divided into CNN-based (Li et al., 2017), autoencoder-based (Bati et al., 2015), GAN based (Jiang et al., 2020) and RNN based (Lyu & Lu, 2016). There are also some models that leverage other techniques, such as manifold learning (Lu et al., 2019), and low-rank representation (Xie et al., 2021). A comprehensive review is conducted by Hu et al. (2022).

Uncertainty quantification models on i.i.d inputs Numerous studies have focused on developing uncertainty quantification models for data that is independent of inputs, such as images. These efforts encompass various approaches, including multi-forward pass models such as ensembles (Lakshminarayanan et al., 2017), dropout-based models (Gal & Ghahramani, 2016), and deterministic models like Bayesian-based methods (Charpentier et al., 2022).

Uncertainty quantification on graph data As pointed out in the survey (Abdar et al., 2021), uncertainty quantification on GNN and semi-supervised learning is under-explored. Most existing models for uncertainty quantification on graphs are either dropout-based or BNN-based methods that typically drop or assign probabilities to edges. Two approaches quantified uncertainty using deterministic single-pass GNNs. One is called graph-based kernel Dirichlet distribution estimation (GKDE) (Zhao et al., 2020), which consists of evidential GCN, graph-based kernel, teacher network, dropout, and loss regularization. Another method is the GPN (Stadler et al., 2021) model that combines PN (Charpentier et al., 2020) and personalized page rank (PPR) message passing to disentangle uncertainty with and without network effects. In addition, a recent method (Wu et al., 2023) used standard classification loss for OOD detection on graphs together with an energy function that is directly extracted from GNN. This method is limited to OOD detection, not generally on the topic of uncertainty quantification. However, these models have feature collapsing issues and the physical mixing properties of HSI are not yet plugged into the models.

G LIMITATION

The UR term we introduce exhibits a strong correlation with the performance of unmixing. Specifically, its effectiveness hinges on the accuracy of the reference endmember matrix, which we assume is available in our problem formulation. Currently, we adopt the approach outlined in (Qin et al., 2020) to obtain an optimized endmember matrix, which serves as our reference. Nevertheless, a universally accepted criterion for assessing the quality of endmember matrices is lacking, and a possible inaccurate one could potentially undermine the positive impact of our UR term. Besides, certain categories pose challenges for unmixing models due to unpredictable noise and intricate

mixtures. In such instances, our proposed UR term might not offer substantial assistance. However, we first introduce the uncertainty quantification problem in the HSI domain, which is less explored but necessary in real applications. Then we propose a promising direction in that we can improve deterministic uncertainty quantification models with domain knowledge and the observed insight can be extended to other domains.