SUPERPIXEL BASED ACTIVE LEARNING AND ONLINE FEATURE IMPORTANCE ESTIMATION FOR HYPERSPECTRAL IMAGE ANALYSIS

A Thesis

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in Electrical and Computer Engineering

By

Jielian Guo

December 2015

SUPERPIXEL BASED ACTIVE LEARNING AND ONLINE FEATURE IMPORTANCE ESTIMATION FOR HYPERSPECTRAL IMAGE ANALYSIS

Jielian Guo

Approved:

Chair of the Committee
Dr. Saurabh Prasad, Assistant Professor,
Dept. of Electrical and Computer Engineering

Committee Members:

Dr. Jose L. Contreras-Vidal, Professor,
Dept. of Electrical and Computer Engineering

Dr. Demetrio Labate, Professor,
Dept. of Mathematics

Dr. Suresh K. Khator, Associate Dean,
Cullen College of Engineering

Dr. Badri Roysam, Department Chair,
Dept. of Electrical and Computer Engineering

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor, Dr. Saurabh Prasad, for the continuous support of my master study and related research. His guidance, motivation and immense knowledge helped me in all the time of research and writing of this thesis.

Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Jose Luis Contreras-Vidal and Dr. Demetrio Labate, for their encouragement and insightful comments. I would also like to thank my fellow lab-mates in Hyperspectral Image Analysis Group: Minshan Cui, Wu Hao, Yuhang Zhang, Xiong Zhou, Tanu Priya and Lifeng Yan for their help and support throughout my research.

Last but not the least, I would like to thank my family and friends for supporting me unconditionally throughout writing this thesis and my life in general.

SUPERPIXEL BASED ACTIVE LEARNING AND ONLINE FEATURE IMPORTANCE ESTIMATION FOR HYPERSPECTRAL IMAGE ANALYSIS

An Abstract

of a

Thesis

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in Electrical and Computer Engineering

By

Jielian Guo

December 2015

Abstract

The rapid development of multi-channel optical imaging sensors has led to increased utilization of hyperspectral data for remote sensing. For classification of hyperspectral data, an informative training set is necessary for ensuring robust performance. However, in remote sensing and other image analysis applications, labeled samples are often difficult, expensive and time-consuming to obtain. This makes active learning (AL) an important part of an image analysis framework — AL aims to efficiently build a representative and efficient library of training samples that are most informative for the underling classification task. This thesis proposes an AL framework that leverages from superpixels. First, a semi-supervised AL method is proposed that leverages the label homogeneity of pixels in a superpixel, leading to a faster convergence using few training samples. Secondly, a spatial-spectral AL method is proposed that integrates spatial and spectral features extracted from superpixels in an AL framework. The experiments with an urban land cover classification and a wetland vegetation mapping task show that the proposed methods have faster convergence and superior performance compared to baselines. Importantly, our proposed framework has a key additional benefit in that it is able to identify and quantify feature importance — the resulting insights can be highly valuable to various remote sensing image analysis tasks.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Hyperspectral Imaging

Hyperspectral imaging modalities acquire information from hundreds of bands of the electromagnetic spectrum with fine wavelength resolution (e.g., 0.01μm) and covers a wide range of wavelengths (typically $0.4 - 2.5\mu$m) [1]. For remote sensing, they utilize the fact that different materials have specific reflectance profiles that can be characterized via such imagery. Such imagery provides the unique spectral reflectance signature for objects. For remote sensing applications, hyperspectral sensors are typically deployed on either aircraft or satellites [2]. The data of a hyperspectral sensor is a three-dimensional array with its width and length corresponding to spatial dimensions and the third dimension corresponding to the spectral profile of each point.

1.2 Applications of Hyperpectral Imaging

Hyperspectral imaging is widely used in applications such as precision agriculture, medical diagnosis, food processing and surveillance. For example, hyperspectral images are used to detect grape variety and to obtain an early warning for disease outbreaks [3]. In the food processing industry, hyperspectral images improve the identification of defects and foreign material (FM), hence enhancing the product quality and increase yields. Hyperspectral images can be used to determine the physical, chemical, and biological contamination on

food products [4]. Through detecting a drop in oxygen consumption in the retina, the hyperspectral sensor can indicate potential diseases [5]. In surveillance applications, objects are detected by their unique signature from a large region of the visible and near infrared spectrum [6].

1.3 Active Learning for Hyperspectral Image Classification in Remote Sensing

With the development of remote sensing technology, enormous quantities of data can be obtained from satellite, airborne and ground-based sensors. The basic goal of classification for remote sensing is to assign labels to different objects in an image according to their spectral and spatial information. In practical scenarios, during hyperspectral image collection, only the hyperspectral information for each pixel can be acquired, while the label information is often manually acquired by experts. The process of annotation is time consuming, tedious and often costly. As a result, in recent years, researchers have focused on development of semi-supervised and active learning methods for classification of remote sensing data.

Active learning (AL) is an online learning method that provides a framework for classifier-in-the-loop annotation by an oracle. Based on an appropriate query strategy, AL can induct the most informative data into the training set which labeled by an expert enhances the underlying classification performance. From an image classification point of view, AL query strategies can be divided into three main categories. The first category is designed for a margin-based classifier such as a support vector machine. By this strategy, AL will select the data point which is closest to the separating hyperplane [7][8]. In [9], AL tends to

choose the sample which is not only closest to the separating plane but also far from the existing support vectors. The second category is Uncertainty sampling. Uncertainty sampling tends to query the samples for which the current classifier is least certain to label [10]. In [11], the measurement of the uncertainty is based on the entropy of the predicted labels. Samples with the maximum entropy are considered as the most uncertain ones and are hence queried. Another query strategy based on uncertainty is breaking ties (BT) [12]. In this strategy, the difference between the two highest posterior probabilities is used as the measure of uncertainty. A modified BT algorithm is proposed in [13] to provide unbiased sampling for different classes. The last query strategy is query by committee. The sample which has the greatest disagreement in a committee is taken as the most informative one [14]. It is important to note that the the underlying goal in such work is selecting new training samples based on a strategy using only spectral information — i.e., it is traditionally pixel based AL. However for remote sensing (and other) image analysis problems, samples are intrinsically defined in both the spectral and spatial domain. Hence an active learning strategy which considers both spectral and spatial information can be expected to be more effective for hyperspectral classification.

In this thesis, two superpixel-based active learning techniques are proposed that take advantage of the spatial-spectral information provided by hyperspectral imagery. The first method is referred to as semi-supervised active learning. In this approach, superpixels work as co-trainers. They enable automatic labelling of an unlabeled sample according to a labeled neighbor in the same superpixel. In each active learning query iteration, not only will the human labeled sample be added into the training set, but its neighbors from the same superpixel will also be added into the training set. By collectively inducing all samples

in a superpixel at a query step, local spatial variation in spectral content is accounted for in the training library, hence leading to higher efficiency compared to traditional pixel based active learning. The second proposed method is called spatial-spectral active learning. In this method, a pixel is identified by the spatial-spectral features extracted from superpixels. Hence, when one pixel is added into the training set, the information of its neighbors as encoded in appropriate spatial features (e.g., textural features) will be utilized in the training and classification process, making the system more robust. Finally, we propose an approach to learn feature importance (relevance) within our proposed active learning framework — we note that this can be highly beneficial to gain data specific insights about the underlying image analysis task — such feature importance learning can potentially also be incorporated in an active learning process for visualization of useful features and/or design of tailored query strategies.

Chapter 2

Related Work

Active learning is an online (annotator-in-the-loop) learning method to select the most informative training samples for labeling that when inducted into the training dataset will yield robust classification. In a hyperspectral image, neighboring pixels are highly correlated. The spatial information derived from the neighboring regions have been shown to be crucial for image analysis tasks. Traditional active learning methods are typically based on a pixel-level architecture. This thesis integrates spatial contextual information as provided by superpixels into active learning. We utilize a subspace-based multinomial logistic regression classifier [15] as the backend classifier, since it has the ability to fuse multiple feature types "algorithmically". Moreover, by analyzing the feature weights in the classifier, feature importance can be effectively quantified.

2.1 Superpixel Segmentation

In a real hyperspectral image, contiguous pixels are similar to each other in terms of spectral content, intensity, or texture. Superpixel segmentation is commonly used in color image processing as an approach to oversegment images for efficient downstream processing. With respect to hyperspectral image analysis, superpixel segmentation has the following desired properties: (1) Computationally efficiency: It reduces the complexity of images substantially by grouping similar pixels together, that can then be treated as one unit or as proposed in this work, can be used as an analysis window for feature extraction. (2) Enabling

efficient spatial-spectral classification [16] by incorporating local spatial information.

Mean shift [17], Felzenswalb and Huttenlocher (FH) [18], and watershed [19] are three popular graph based superpixel segmentation methods. Mean shift has tolerance to the local variations; while FH and watershed are very computationally efficient. Generally, superpixels produced by the three methods have large variations in terms of size and shape, some of which cover multiple objects [20][21]. NCuts was proposed by Ren and Malik [22] to address the irregular size and shape problem, but it has a very high computational complexity. Compared to all the above-mentioned segmentation algorithms, entropy-rate superpixel segmentation has the advantage of resulting in superpixels that have compact shape and similar size.

Entropy-Rate superpixel segmentation is a graph based segmentation method, where an image is represented as a graph, $G = (V, E)$, each pixel is represented as the vertex of the graph denoted by set $v_{ij} \in V$ and, pair of pixels are connected by an edge e_{ij} which belongs to the set E. Each edge has been assigned a weight w_{ij} according to the similarity of the pair of pixels connected by it. For an undirected graph, the weights of edge are symmetric, i.e. $w_{ij} = w_{ji}$.

Graph partition : Suppose S is an entire graph, a graph partition divides the vertex set V into disjoint subsets $S = \{S_1, S_2, ...S_K\}$, and K is the number of superpixel. Graph partition selects edges to connect pixels. The pixels which are not connected by selected edges are separated. A selected subset of edges $A \in E$ makes the graph $G = (V, A)$ contain K superpixels.

Entropy: Entropy H is a measurement of the uncertainty of a random variable. Suppose X is a discrete random variable, the entropy $H(X)$ of X is defined as

$$H(X) = -\sum_{x \in \mathfrak{X}} p_X(x) \log p_X(x), \tag{2.1}$$

where$\mathfrak{X}$ is the support of X, and $p_X(x)$ is the probability mass function.

Random walks on graphs: $\boldsymbol{X} = \{X_t | t \in T, X_t \in V\}$ is a random walk on the graph $G = \{V, E\}$. The transition probability is defined as

$$p_{i,j} = p(X_{t+1} = v_j | X_t = v_i), \tag{2.2}$$

where $w_i = \sum_{k:e_{i,k} \in E} w_{i,k}$ is the sum of incident weight of the vertex v_i. The entropy rate of the random walk can be computed by

$$H(\boldsymbol{X}) = H(X_2|X_1) = -\sum_i \sum_j \frac{w_{ij}}{w_T} \log \frac{w_{ij}}{w_T} + \sum_i \frac{w_i}{w_T} \log \frac{w_i}{w_T}. \tag{2.3}$$

Suppose that subset edge set A has been selected to construct a graph $G = (V, A)$. The transition probability can be calculated as

$$p_{ij}(A) = \begin{cases} \frac{w_{ij}}{w_i}, & if \quad i \neq j, e_{ij} \in A, \\ 0, & if \quad i \neq j, e_{ij} \notin A, \\ 1 - \frac{\sum_{j:e_{ij} \in A} w_{ij}}{w_i}, & if \quad i = j. \end{cases} \tag{2.4}$$

Hence, the entropy rate of the random walk on $G = (V, A)$ can be computed as

$$H(A) = -\sum_{i=1} \mu_i \sum_{j=1} p_{ij}(A) log(p_{ij}(A)), \tag{2.5}$$

where $\mu_i = w_i / \Sigma_{i=1}^{|V|} w_i$ is a stationary distribution with respect to a random walk on the graph. Entropy is the criterion used to obtain compact and homogeneous segmentation.

Balancing Function: Balancing function is utilized to encourage clusters to have a similar size. Let A be the selected edge set. Let N_A be the number of connected components

in the graph, and Z_A be the distribution of the cluster membership. Suppose the edge subset A produces the partition $S_A = \{S_1, S_2, S_3, ..., S_{N_A}\}$, then the distribution Z_A is given as

$$p_{Z_A}(j) = \frac{|S_j|}{|V|}, \forall j \in \{1, 2, ..., N_A\}. \tag{2.6}$$

The balancing term is defined as

$$B(A) = H(Z_A) - N_A, \tag{2.7}$$

where $H(Z_A)$ is the entropy of Z_A. It favors clusters with similar sizes, whereas N_A favors fewer number of clusters.

Objective function : The overall objective function is defined as

$$\max_{A \subseteq E} \quad F(A) = H(A) + \lambda B(A), \tag{2.8}$$

where λ is the weight of balance term. The entropy rate $H(A)$ of the random walk on graph tends to produce compact and homogeneous clusters, while the balance term $B(A)$ favors clusters with similar size and fewer clusters. The objective function is submodular and monotonically increasing. It can be optimized by a Greedy algorithm [23]. This algorithm begins with an empty edge set A, and gradually adds edges to set A according to the objective function. At each iteration, this algorithm adds edges that makes the most gain in the objective function of A. The iterations will be terminated once the number of clusters reaches the preset superpixel number K.

2.2 Subspace-based Multinomial Logistic Regression (MLR-sub)

Multinomial logistic regression is widely used in hyperspectral image classification to model the posterior class distributions in a Bayesian framework, based on which a few

state of the art hyperspectral image classification methods are built. The motivation for choosing MLRsub in this work as a backend classifier are as follows — (1) It can integrate multiple features without ad hoc weights (weights are determined via optimization); (2) Feature importance can be efficiently measured by the automatically learned weights in the expression for the posterior probability. Consider an image as a set of pixels. Let $\boldsymbol{x} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ denote the original feature vectors of an image, where N is the number of pixels. $\boldsymbol{Y} = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_N\}$ is the label set of an image. The $\boldsymbol{y}_i$ is the label vector of pixel i. Suppose there are K classes, then $\boldsymbol{y}_i = \{y_i^1, ..., y_i^K\}$. If pixel i belongs to class k, then $y_i^k = 1$, otherwise $y_i^k = 0$, $j \in \{1, 2, ...K\}$. For supervised classification method, a subset of pixels should be labeled by an expert in order to train the classifier. Let $D_L = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_{N_T}, \boldsymbol{y}_{N_T})\}$ be a training set. In hyperspectral image classification, there are two main problems: (1) Limited training samples; (2) Mixed pixels: Due to sensor design considerations, the wealth of spectral information in hyperspectral data is often not complemented by extremely fine spatial resolution. This leads to the problem of mixed pixels, which represent a challenge for accurate hyperspectral image classification [24]. In the MLRsub mode, the resulting model can be expressed as

$$\boldsymbol{x}_i = \sum_{k=1}^{K} r_k \boldsymbol{m}^k + \alpha \prod_{k=1}^{K} \boldsymbol{m}^k, \tag{2.9}$$

where $\boldsymbol{m}^k, k = 1, ...K$ is the spectral of the K end-members. r_k and α are the parameters controlling the linear and nonlinear terms.

The posterior density $p(y_i^k = 1|\boldsymbol{x}_i, \boldsymbol{\omega})$ for a given class is defined as [15]

$$P(y_i^k = 1|\boldsymbol{x}_i, \boldsymbol{\omega}) = \frac{\exp({\boldsymbol{\omega}^{(k)}}^T \boldsymbol{h}(\boldsymbol{x}_i))}{\sum_{j=1}^{K} \exp({\boldsymbol{\omega}^{(j)}}^T \boldsymbol{h}(\boldsymbol{x}_i))}, \tag{2.10}$$

where $\boldsymbol{h}(\boldsymbol{x}_i) = [||\boldsymbol{x}_i||^2, ||{\boldsymbol{x}_i}^T \boldsymbol{U}^{(1)}||^2, ..., ||{\boldsymbol{x}_i}^T \boldsymbol{U}^{(K)}||^2]$, $\boldsymbol{U}^{(k)}$ is a set of $r^{(k)}$ -dimensional

orthonormal-basis vectors for the subspace associated with classes $k = 1, 2, ..., K$. By definition, the final feature vector $\boldsymbol{h}(\boldsymbol{x}_i)$ consists of the energy of the original spectral reflectance vector projected onto all the classes and the energy of the spectral reflectance vector itself. The feature vector is not $\boldsymbol{x}$ itself, but the projections onto the subspaces learned from the training set, which enhances the class separability and reduces the feature dimension.

The training procedure of the subspace-based multinomial logistic regression involves finding an optimal $\boldsymbol{\omega}$ which fits the training set. According to [25] [26], The $\boldsymbol{\omega}$ is estimated by a maximum a posterior (MAP) method. The objective function can be defined as

$$\hat{\boldsymbol{\omega}}_{\text{MAP}} = \arg\max_{\boldsymbol{\omega}} L(\boldsymbol{\omega}) = \arg\max_{\boldsymbol{\omega}} [l(\boldsymbol{\omega}) + \log p(\boldsymbol{\omega})], \tag{2.11}$$

where $p(\boldsymbol{\omega})$ is the prior on the parameters $\boldsymbol{\omega}$, $l(\boldsymbol{\omega})$ is the log-likelihood function,

$$p(\boldsymbol{\omega}) \propto e^{-\beta/2||\boldsymbol{\omega}||^2} \text{ and} \tag{2.12}$$

$$l(\boldsymbol{\omega}) = \sum_{i=1}^{N_T} \log P(\boldsymbol{y}_i | \boldsymbol{x}_i, \boldsymbol{\omega}). \tag{2.13}$$

The optimization function in Eq. 2.11 is concave. Consider the second order Taylor series for the log-likelihood for regressor $\boldsymbol{\omega}$ at t'th iteration

$$l(\boldsymbol{\omega}) \geq l(\boldsymbol{\omega}_t) + (\boldsymbol{\omega} - \boldsymbol{\omega}_t)^{\mathrm{T}} \boldsymbol{g}(\boldsymbol{\omega}_t) + \frac{1}{2}(\boldsymbol{\omega} - \boldsymbol{\omega}_t)^{\mathrm{T}} \boldsymbol{B}(\boldsymbol{\omega} - \boldsymbol{\omega}_t) \text{ and} \tag{2.14}$$

where $\boldsymbol{B}$ is the lower bound of the Hessian of $l(\boldsymbol{\omega})$,

$$\boldsymbol{B} \equiv -\frac{1}{2}\left[\boldsymbol{I} - \mathbf{1}\mathbf{1}^{\mathrm{T}}/(K+1)\right] \bigotimes \sum_{i=1}^{N_T} h(\boldsymbol{x}_i) h(\boldsymbol{x}_i)^{\mathrm{T}}. \tag{2.15}$$

$\boldsymbol{g}(\boldsymbol{\omega}_t)$ is the gradient of $l(\cdot)$ at $\boldsymbol{\omega}_t$. According to the bound optimization algorithm [26], the update equation is

$$\hat{\boldsymbol{\omega}}_{t+1} = (\boldsymbol{B} - \beta \boldsymbol{I})^{-1}(\boldsymbol{B}\hat{\boldsymbol{\omega}}_t - \boldsymbol{g}(\hat{\boldsymbol{\omega}}_t)), \tag{2.16}$$

2.2.1 Data Fusion via MLRsub

Let $p_m(y_i^k = 1|\boldsymbol{x}_i^m, \boldsymbol{\omega}_m)$ be the posterior probability associated with feature m, $m \in \{1, 2, ..., M\}$, there are M features. According to the LOGP rule in [27], for any pixel $i = 1, ..., N$,

$$p_{LOGP}(y_i^k = 1|\boldsymbol{x}_i^1, ..., \boldsymbol{x}_i^M, \boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_M, \alpha_1, ..., \alpha_M) = \frac{\prod_{m=1}^{M} p_m(y_i^k = 1|\boldsymbol{x}_i^m, \boldsymbol{\omega}_m)^{\alpha_m}}{\sum_{j=1}^{K} \prod_{m=1}^{M} p_m(y_i^j = 1|\boldsymbol{x}_i^m, \boldsymbol{\omega}_m)^{\alpha_m}}, \tag{2.17}$$

where $\{\alpha_m|0 \leq \alpha_m \leq 1, \sum_{m=1}^{M} \alpha_m = 1\}$ is a tunable parameter which controls the affection of each feature vector on the final probability. Notice that, for this multiple feature type problem, there are two parameters — (1) the logistic regressors $\boldsymbol{\omega}_m$, which are associated with the classifier and (2) the set of weight parameters α_m associated with the feature m. These two parameters should be optimized in the training process. By using the MLRsub classifier to model the posterior possibility $p_m(y_i^k = 1|\boldsymbol{x}_i^m, \boldsymbol{\omega}_m)$, the regressors $\boldsymbol{\omega}$ and weight parameters α_m can be combined into a new parameter, so only one parameter needs to be optimized. The model of MLRsub is shown in Eq.2.10. Substituting the MLRsub model in Eq.2.10 to the LOGP framework in Eq. 2.17:

$$p_{LOGP}(y_i^k = 1|\boldsymbol{x}_i^1, ..., \boldsymbol{x}_i^M, \boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_M, \alpha_1, ..., \alpha_M) = \frac{\exp\left(\sum_{m=1}^{M} \alpha_m \boldsymbol{\omega}_m^{(k)} \boldsymbol{h}(\boldsymbol{x}_i^m)\right)}{\sum_{j=1}^{K} \exp\left(\sum_{m=1}^{M} \alpha_m \boldsymbol{\omega}_m^{(j)} \boldsymbol{h}(\boldsymbol{x}_i^m)\right)}. \tag{2.18}$$

By letting $\tilde{\boldsymbol{\omega}}_m^{(k)} = \alpha_m \boldsymbol{\omega}_m^{(k)}$,

$$p_{LOGP}(y_i^k = 1|\boldsymbol{x}_i^1, ..., \boldsymbol{x}_i^M, \boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_M, \alpha_1, ..., \alpha_M) = \frac{\exp\left(\sum_{m=1}^{M} \tilde{\boldsymbol{\omega}}_m^{(k)} \boldsymbol{h}(\boldsymbol{x}_i^m)\right)}{\sum_{j=1}^{K} \exp\left(\sum_{m=1}^{M} \tilde{\boldsymbol{\omega}}_m^{(j)} \boldsymbol{h}(\boldsymbol{x}_i^m)\right)}. \tag{2.19}$$

Notice that Eq.2.19 has the same model as the original MLRsub classifier. Hence the combined regressors $\tilde{\boldsymbol{\omega}}$ can be optimized as the aforementioned MAP method. By this fusion framework, the multiple features are integrated without the need for ad hoc weights, hence there is a great flexibility in the fusion of different information. Moreover, from the Eq. 2.19, it can be readily concluded that [28], if $\tilde{\boldsymbol{\omega}}_{m_p}^{(k)}\boldsymbol{h}(\boldsymbol{x}_i{}^{m_p}) > \tilde{\boldsymbol{\omega}}_{m_q}^{(k)}\boldsymbol{h}(\boldsymbol{x}_i{}^{m_q}), m_p, m_q \in \{1, 2, ..., M\}$, then the feature m_p is the dominant feature for classification. This way, the feature importance can be quantified, which is the another motivation for choosing MLRsub as the classifier in this thesis.

2.3 Active Learning for Hyperspectral Image Classification

Active learning aims to select the most informative samples from the unlabeled candidate data. After being labeled by an expert, these selected samples will be added into the training set. The classifier will then be trained with the updated training set, hence enhancing classification performance. In the beginning, the classifier is trained by an initial training set D_L. Then the unlabeled candidate samples will be sent to the classifier to get the predicted label information. According to the predicted label information, the query strategy selects a set of the most informative samples $\boldsymbol{x}^*$. Following this, a human expert manually labels $\boldsymbol{x}^*$ with $\boldsymbol{y}^*$. The pairs $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ will then be inducted into the training set D_L, and the classifier will be retrained by the new training set. These iterations will go on until the classification performance converges to a plateau. The flowchart of general active learning is shown in Figure 2.1. Dashed lines indicate the active learning process.

Generally, active learning strategies can be classified into different types based on the query strategy. Uncertainty sampling is commonly used with multi-class classification.

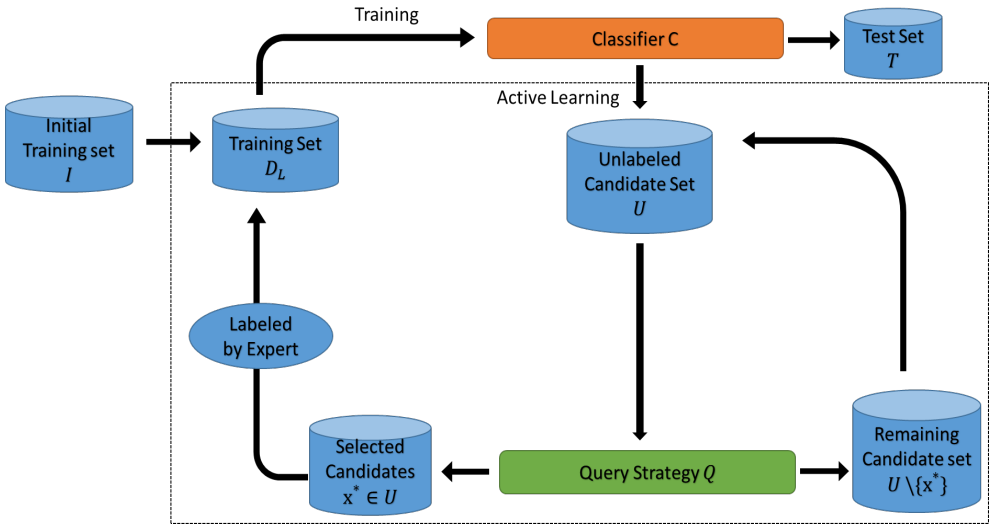


Figure 2.1: Flowchart of general active learning

Among the unlabeled candidate set U, the uncertainty sampling strategy queries the samples for which the current classifier is most uncertain in predicting the class label. For multi-class classification problems, the uncertainty criterion is calculated according to the posterior probability of labels predicted by the current classifier.

There are three uncertainty sampling methods:

1) Least confident (LC) — it can be defined as Eq. 2.20.

$$\boldsymbol{x}^*_{LC} = \arg\max_{\boldsymbol{x}\in U} 1 - p(\hat{\boldsymbol{y}}|\boldsymbol{x}), \tag{2.20}$$

where $\hat{\boldsymbol{y}}$ is the predicted label with highest probability,

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} p(\boldsymbol{y}|\boldsymbol{x}).$$

2) Breaking ties (BT), it can be defined as

$$\boldsymbol{x}^*_{BT} = \arg\min_{\boldsymbol{x}\in U} p(\hat{\boldsymbol{y}}_1|\boldsymbol{x}) - p(\hat{\boldsymbol{y}}_2|\boldsymbol{x}), \tag{2.21}$$

where $\hat{\boldsymbol{y}}_1$ and $\hat{\boldsymbol{y}}_2$ are the first and second most probable labels for $\boldsymbol{x}$ predicted by the current classifier.

3)Entropy (E), it can be defined as

$$\boldsymbol{x}_E^* = \arg\max_{\boldsymbol{x}\in U} -\sum_k p(\boldsymbol{y}^k|\boldsymbol{x}) \log p(\boldsymbol{y}^k|\boldsymbol{x}), \tag{2.22}$$

where $\boldsymbol{y}^k \in \{\boldsymbol{y}^1, ..., \boldsymbol{y}^K\}$. They all have their strengths and weaknesses. An empirical comparison between them is often utilized to identify which of those is suitable for a particular application [29].

2.4 Spatial Feature

Spatial features contain important information about the shape, size and texture, etc. from a fixed or adaptive region of the image in and around objects of interest. In hyperspectral images, the spatial context of a pixel can provide additional information, as spatial context often has information unique to the different objects. By integrating spatial information, better classification can be performed yielding more accurate results. Classical spatial feature extraction methods are based on Gabor filters [30], morphological operators [31], wavelet decomposition [32] and gray-level co-occurrence matrix (GLCM) [33].

GLCM texture measurements have been a popular method for texture extraction in remote sensing images since they were first introduced by Haralick in the 1970s [33]. Given a spatial relationship defined among pixels in a texture, the GLCM represents the joint distribution of grey-level pairs of neighbor pixels. Specifically, for various orientations, GLCM features are estimated as Eq.2.23

$$P_{i,j} = \frac{V_{i,j}}{\sum_{i,j=0}^{N-1} V_{i,j}}, \tag{2.23}$$

where N is the number of grey levels, i is the reference pixel value, j is the neighbor pixel value, $V_{i,j}$ is the number of times the combination (i, j) or (j, i) occurs in a window for the given orientation operator. The GLCM features used in this thesis are summarized in Table 2.1.

Table 2.1: GLCM feature of 7 sources

#	GLCM feature	Formular
1	Contrast	$\sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2$
2	Homogeneity	$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$
3	Energy	$\sqrt{\sum_{i,j=0}^{N-1} P_{i,j}^2}$
4	Entropy	$\sum_{i,j=0}^{N-1} P_{i,j}(-\ln P_{i,j})$
5	Variance	$\sum_{i,j=0}^{N-1} P_{i,j}(i-\mu_i)^2$
6	Mean	$\sum_{i,j=0}^{N-1} i(P_{i,j})$
7	Correlation	$\sum_{i,j=0}^{N-1} P_{i,j}[\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}}]$ [1]

[1] $\mu_i = \mu_j = \sum_{i,j=0}^{N-1} i(P_{i,j})$

$\sigma_i^2 = \sigma_j^2 = \sum_{i,j=0}^{N-1} P_{i,j}(i-\mu_i)^2$

Chapter 3

Superpixel based Active Learning

In this thesis, we propose to utilize spatial information derived from superpixels to enhance the active learning process via two approaches: the first one can be considered to be a semi-supervised variant of active learning, for increasing efficiency, while another is a spectral-spatial superpixel-level active learning with spatial features derived from superpixels. As we will demonstrate, the second method not only reduced the convergence time (in obtaining peak classification performance) for the AL system, but also improved the overall performance itself, compared to baseline methods.

3.1 Data Description

In this thesis, two data sets, University of Houston and a wetland imagery dataset from Galveston are used to validate the proposed methods.

3.1.1 University of Houston

The University of Houston data set is acquired by an ITRES-CASI (Compact Airborne Spectrographic Imager) 1500 hyperspectral imager. It covers the University of Houston campus and its neighboring urban area. The hyperspectral image consists of 144 spectral bands ranging from 380nm - 1050nm with spatial resolution of 2.5m. The classes and the number of labeled superpixels for each class are listed in Table 3.1. There are 15 classes defined in this data. The image was segmented into superpixels with the average size of 81 pixels per superpixel. 8 classes that contain enough superpixels to perform reasonable

validation studies were used for validating the proposed methods, which are shown in bold in Table 3.1. Figure 3.2 is the mean signatures (spectrum) of the defined 15 classes.

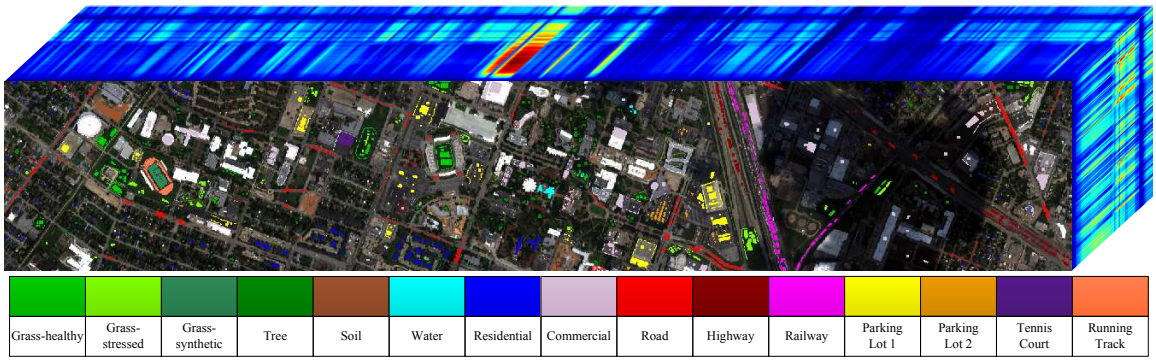


Figure 3.1: True color University of Houston hyperspectral image cube with ground truth

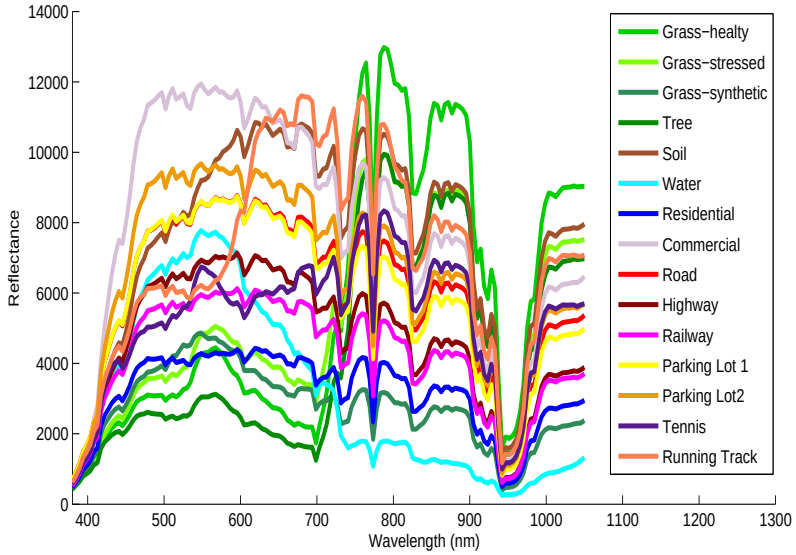


Figure 3.2: Mean signatures of objects in Unversity of Houston hyperspectral image

Table 3.1: Classes and the number of superpixels for each class of UH data

	Name of classes	Number of superpixel
1	**Grass-healthy**	99
2	**Grass-stressed**	97
3	Grass-synthetic	10
4	**Tree**	179
5	Soil	79
6	Water	9
7	**Residential**	127
8	**Commercial**	318
9	**Road**	191
10	**Highway**	94
11	Railway	61
12	**Parking Lot 1**	155
13	Parking Lot 2	60
14	Tennis Court	7
15	Running Track	8

3.1.2 Galveston Wetland

The Galveston wetland data is acquired by the Headwall hyperspectral imager. It covers a small part of wetland in Galveston by the research team in our laboratory. The hyperspectral image consists of 163 spectral bands ranging from 400nm - 1000nm with very high spatial resolution. Analysis of such images is particularly useful in ecological studies over wetlands, and in applications such as monitoring ecological health post natural or anthropogenic disasters. There are 8 classes defined in this data, shown in Table 3.2. The image is segmented into superpixels with an average size of 121 pixels per superpixel. The classes are listed in Table 3.2.

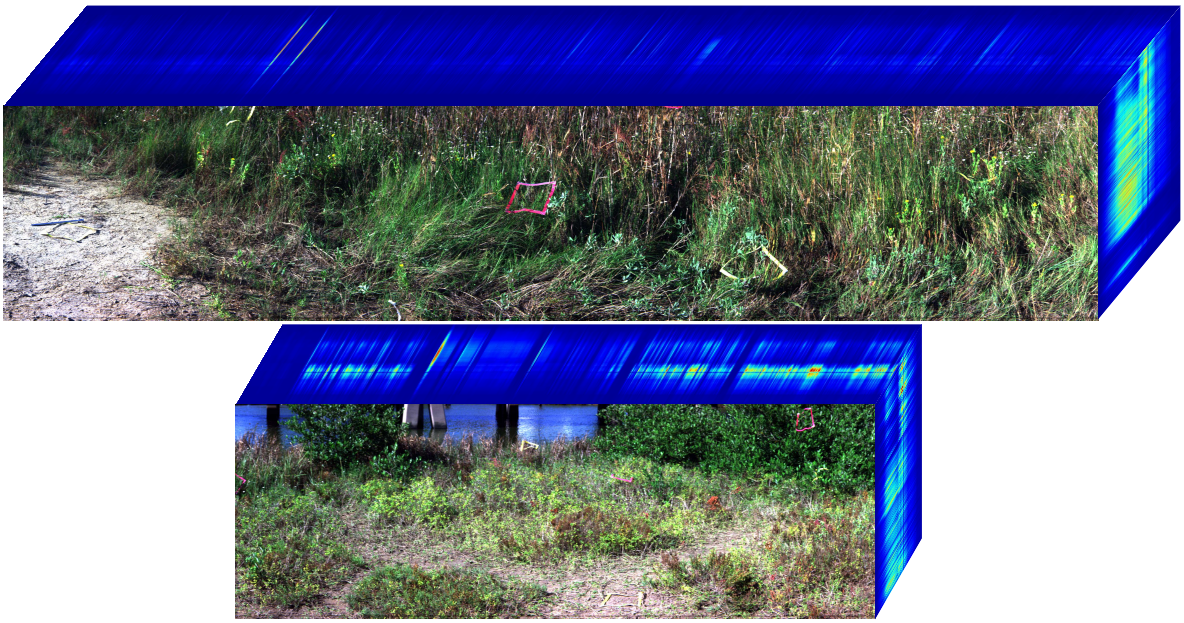


Figure 3.3: True color Galveston wetland hyperspectral image cube

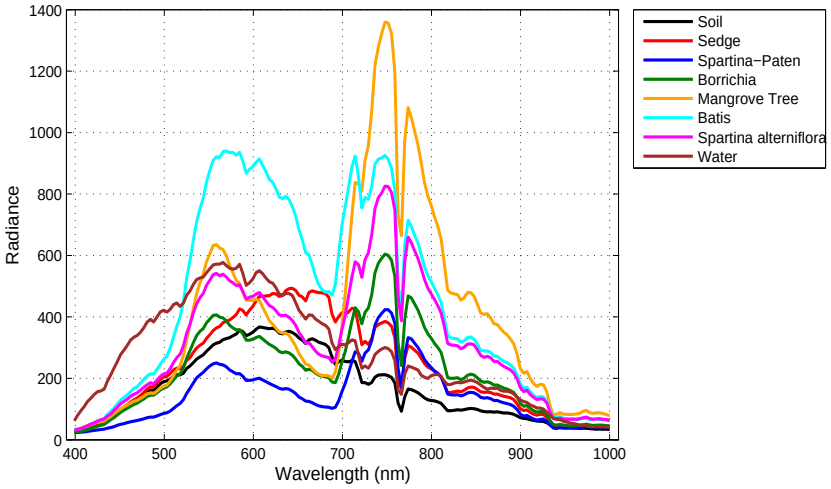


Figure 3.4: Mean signatures of objects in Galveston wetland hyperspectral image

Table 3.2: Classes and the number of superpixels for each class in Galveston wetland data

	Name of classes	Number of superpixel
1	**Soil**	395
2	**Sedge**	247
3	**Borrichia**	214
4	**Spartina paten**	289
5	**Mangrove**	674
6	**Batis**	584
7	**Spartina-alterniflora**	298
8	**Water**	351

3.2 Semi-supervised Active Learning

Semi-supervised classification methods train a classifier with both labeled and unlabeled samples. The motivation is that for real world applications, labeled samples are often difficult, expensive and time-consuming to obtain, while the unlabeled samples are relatively easy to obtain. Semi-supervised methods use labeled and unlabeled data together to train the classifier, aiming to obtain the satisfactory classification result with a small cost of data annotation.

3.2.1 Proposed Method

In general active learning, only the selected informative sample will be labeled and included in the training set. In this thesis, a semi-supervised active learning method is proposed to involve not only the most informative sample but also its similar and contiguous neighbors within the same superpixel into the training set, with the assumption that they belong to the same object and capture local spectral variability. It has been shown that entropy rate based superpixel segmentation have a satisfactory performance in terms of undersegmentation error. Low undersegmentation error implies that high percentage of superpixels contain only one kind of object [16]. It is reasonable to assume that pixels

in the same superpixel should have the same label. Hence, superpixel segmentation in the proposed method works as a co-trainer. It labels the unlabeled data according to its labeled neighbors in the same superpixel. This superpixel-level labeled data will be included in the training set to improve the classifier. The flowchart of the proposed method is shown in Figure 3.5. The dashed box refers to the active learning process.

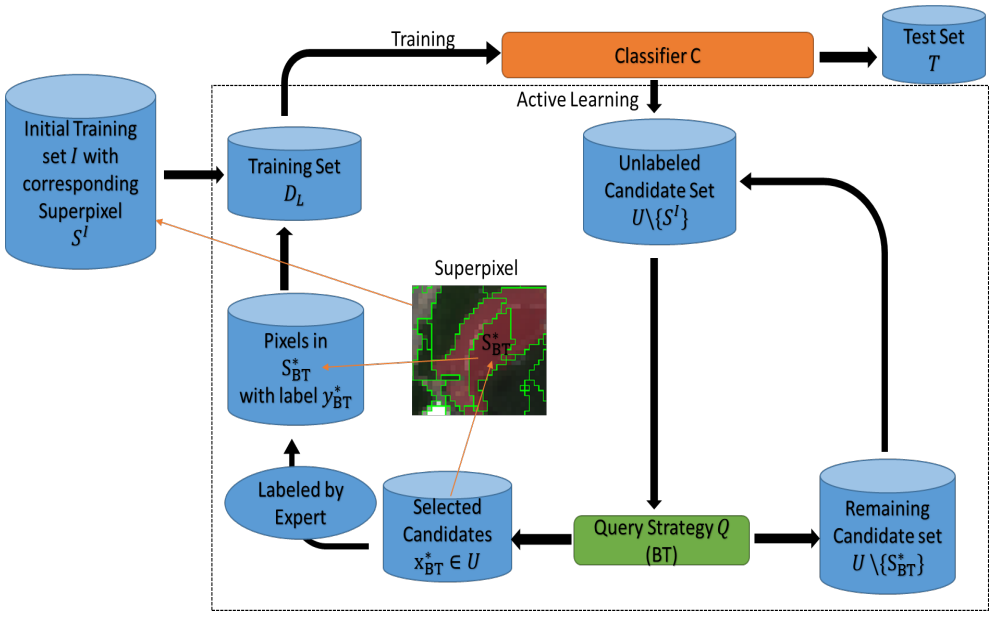


Figure 3.5: Flowchart of the proposed semi-supervised active learning

In the beginning, a small number of samples are randomly selected from an initial training set $I = \{\boldsymbol{x}_1^I, ..., \boldsymbol{x}_{n_1}^I\}$, n_1 is the number of initial training samples. Following this, experts manually assign these initial training samples with labels of $\boldsymbol{y} = \{\boldsymbol{y}_1^I, ..., \boldsymbol{y}_{n_1}^I\}$. Superpixels containing the pixels in initial training set will be selected $S^I = \{S_1^I, ..., S_{n_1}^I\}$. Pixels in a superpixel S_n^I are assumed to be the same class as $\boldsymbol{x}_n^I$, hence they are assigned a label of $\boldsymbol{y}_n^I$. Following this, samples in conjunction with the initial sample will be added to the training set D_L to train a preliminary classifier. The resulting classifier then is used to evaluate the samples in the unlabeled candidate set U. According to Eq. 2.21, the most

informative sample $\boldsymbol{x}^*_{BT}$ is selected. A human expert manually assigns it with a label $\boldsymbol{y}^*_{BT}$. Meanwhile, in the superpixel image (segmented image), the superpixel S^*_{BT} which contains the sample $\boldsymbol{x}^*_{BT}$ will be selected. All the pixels in S^*_{BT} together with the label $\boldsymbol{y}^*_{BT}$ are included into the training set. The classifier will then be trained with the new training set.

3.2.2 Experiments with Superpixel Segmentation

First, entropy rate superpixel generation is used to obtain the superpixels. Figure3.6 shows the superpixels of cropped UH generated by entropy rate segmentation. The number of superpixels for Figure 3.6 is 50000. It can be seen that entropy rate segmentation has a good performance with regards to preserving the edges around objects, in addition to oversegmenting within objects. Further, most superpixels only contain one object type (often, there are multiple superpixels within an object), which is an assumption we make in our proposed semi-supervised active learning method. Moreover, in [16], it is shown that for such hyperspectral imagery, the undersegmentation error decreases as the number of superpixel increases, which implies that with an appropriately chosen number of superpixels, it is possible to make the the resulting superpixels pure.

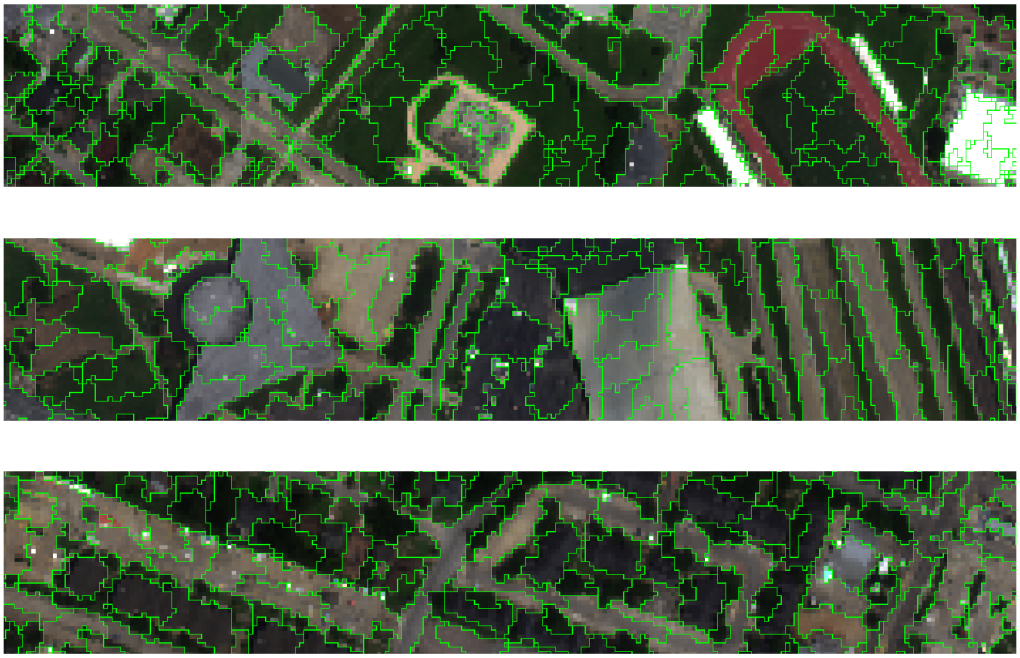


Figure 3.6: Superpixels generated by entropy-rate segmentation from different regions of the University of Houston dataset.

In addition to the number of superpixels, the balancing term λ is an important parameter for superpixel generation via the entropy rate method. With a large value λ, the entropy rate segmentation tends to partition the image into superpixels of balanced size. Figure 3.7 shows the superpixel size distribution with different values of λ. We note that in this work, we found the value of $\lambda = 0.5$ to be optimal empirically.

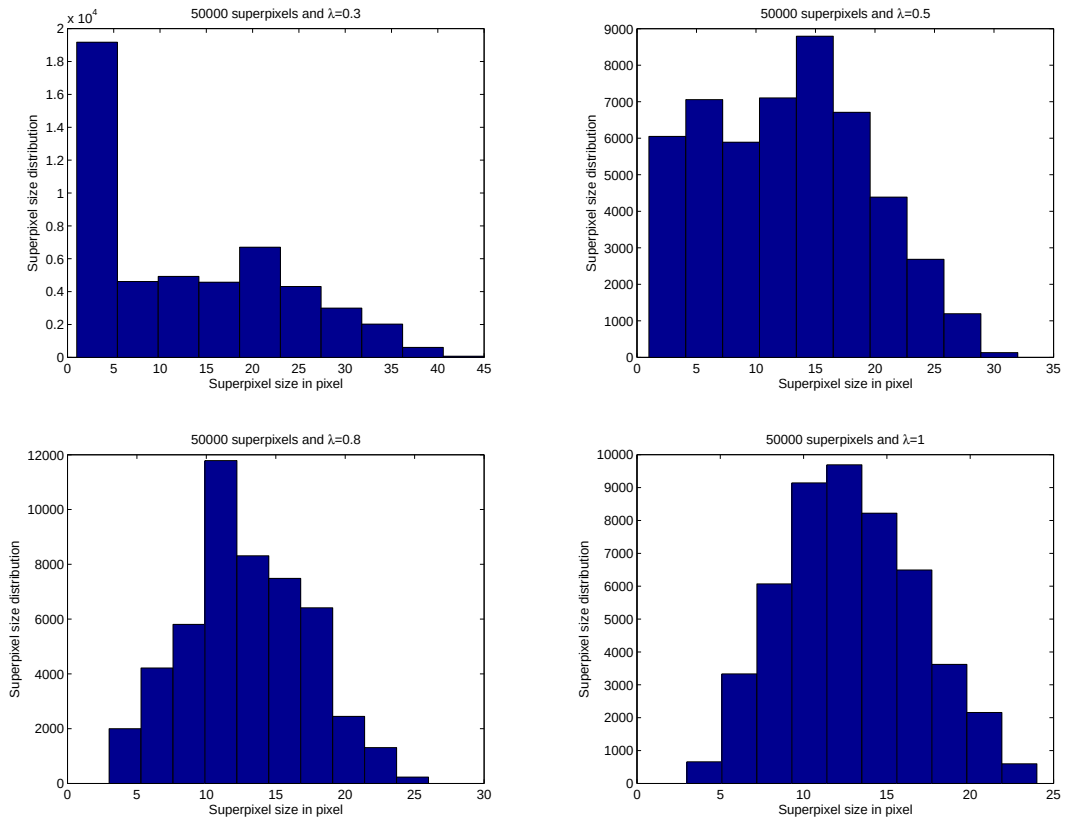


Figure 3.7: Superpixel size distribution

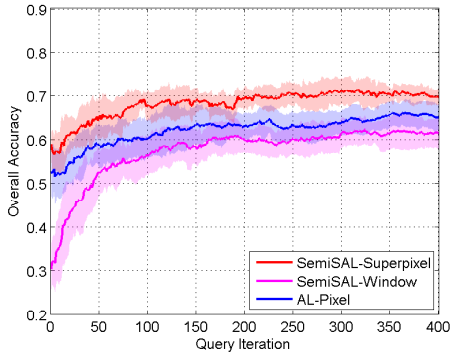
3.2.3 Experimental Setup and Results Semi-supervised Active Learning (SemiSAL)

For traditional breaking ties AL, only the selected most informative pixel will be included in the training set. However, the proposed method will involve all the pixels in a superpixel which contains the most informative pixel to the training set, while assigning them the same label as the most informative pixel. The proposed method were evaluated with two baseline methods. The first one is a window based AL. Similar with superpixel-based AL, the window based AL will add all the pixels covered by the same window frame into the

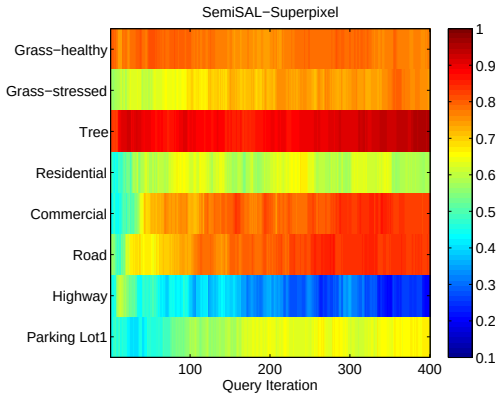
training set. For a fair comparison, the average superpixel size is similar to the window size. For UH data, the window size is 9×9. The second comparative method is pixel level AL, where only the queried pixel is added to the training set.

For the first data set, University of Houston, we note the following settings (1) the initial training set contains 5 samples from different superpixels per class ; (2) Testing set contains 40 samples from different superpixels per class. For a fair comparison, we ensure that the testing samples are not covered by the superpixels which have already covered training samples and candidate samples; (3) The candidate set contains all the left labeled pixels. The batch size for superpixel-based AL is one superpixel. For window-based AL, it is one window. For pixel level AL, it is one pixel.

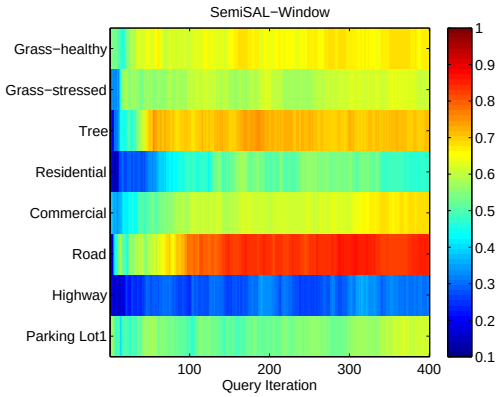
The experiment is repeated 10 times with 10 different randomly selected initial training sets and testing sets. The average results, in terms of over all accuracy (OA) and the class accuracy (CA), are shown in Figure 3.8.From Figure 3.8(a), it can be seen that the proposed method outperforms the two baseline methods for all the query iterations in terms of OA, and it yield a relatively small standard deviation. The window based AL performs even poorer than the pixel level AL, because the window on the boundary of an object will involve the pixels from other objects, resulting in inaccurate classification. The proposed method addressed this problem using the efficacy of superpixels to oversegment objects. Table 3.3 shows that, the accuracy with the proposed method is 8% greater than window based AL and 5% greater than pixel level AL. In terms of class accuracy, the proposed AL outperforms the two baseline methods for Grass-healthy, Grass-stressed, Tree, Commercial, Road and Parking Lot 1. Moreover, Figure 3.8 (b)(c)(d) shows that for the 6 classes the AL result of proposed method converges faster than the baseline methods.



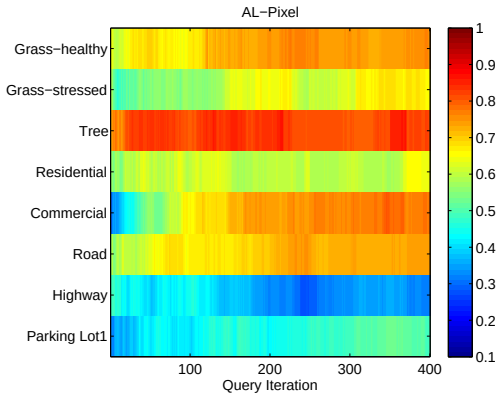
(a) OA along with standard deviation



(b) CA of SemiSAL-SuperPixel



(c) CA of SemiSAL-Window



(d) CA of AL-Pixel

Figure 3.8: OA and CA of Semi-Supervised AL of UH

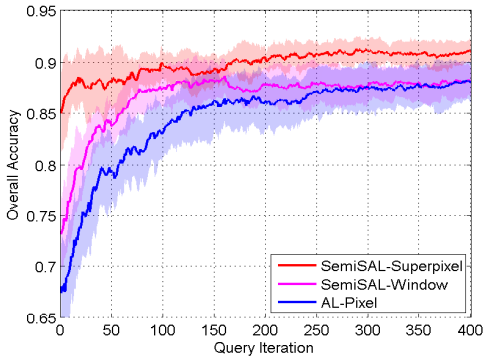
Table 3.3: OA and CA of Semi-Supervised AL of UH

#	Class	SemiSAL Superpixel	SemiSAL Window	AL Pixel
1	Grass-healthy	76.67	66.89	75.65
2	Grass-stressed	74.41	61.60	65.72
3	Tree	94.48	70.08	81.93
4	Residential	58.69	47.56	63.58
5	Commercial	83.82	68.78	78.31
6	Road	84.16	85.09	73.98
7	Highway	19.50	28.50	31.75
8	Parking Lot 1	65.55	61.05	49.62
	OA	69.66	61.19	65.07

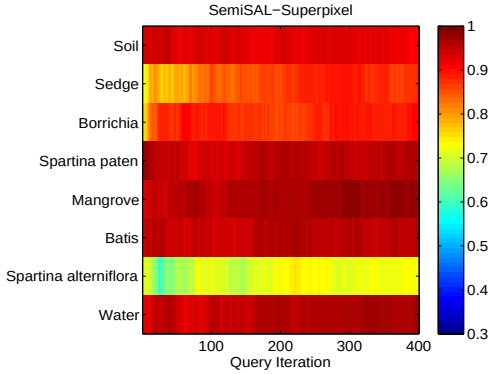
For the second data set, Galveston wetland, we note the following experimental settings — (1) Initial training set contains 5 samples from different superpixels per class. 2) Testing set contains 60 samples from different superpixels per class. For a fair comparison, we ensure that the testing samples is not covered by the superpixels which have already covered training samples and candidate samples. 3) Candidate set contains all the remaining labeled pixels. The batch size for superpixel-based AL, window based AL, and pixel level AL are one superpixel, one window and one pixel, respectively.

Figure 3.9 shows the average of results of 10 random subsamplings of training and test data in terms of overall accuracy and class-specific accuracy. From Figure 3.9(a), it can be seen that the proposed method yields higher accuracy through all the query iterations, especially at the initial iterations. The OA of the proposed method converges much faster than the two baseline methods. Comparing the Figure 3.9(b)(c)(d), it can be seen that the proposed method leads to faster convergence for most of the classes such as Borrichla, Spartina paten, Spartina alterniflora and Water. Table 3.4 shows that the proposed method is 2.84% better than window based AL and 2.74% better than pixel level AL. In terms of class accuracy, the proposed AL outperforms the window based AL for 7 classes and outperforms

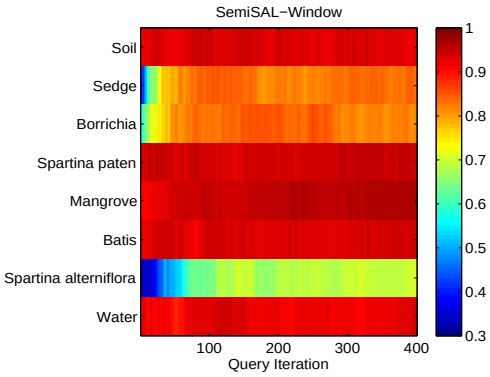
pixel level AL for 6 classes.



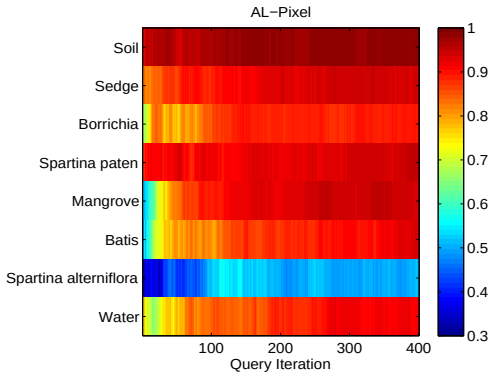
(a) OA along with standard deviation



(b) CA of SemiSAL-SuperPixel



(c) CA of SemiSAL-Window



(d) CA of AL-Pixel

Figure 3.9: OA and CA of Semi-Supervised AL of Galveston wetland

Table 3.4: OA and CA of Semi-Supervised AL of Galveston wetland

#	Class	SemiSAL Superpixel	SemiSAL Window	AL Pixel
1	Soil	90.83	92.33	98.00
2	Sedge	87.67	82.67	93.83
3	Borrichia	89.37	81.26	89.02
4	Spartina paten	95.83	95.00	94.67
5	Mangrove	97.67	96.17	93.67
6	Batis	96.00	93.67	94.17
7	Spartina-alterniflora	72.83	70.33	50.67
8	Water	97.19	93.21	91.38
	OA	90.92	88.08	88.18

3.2.4 Conclusion

Pixels in one superpixel are similar, but they do capture the local spectral variability about the objects in the image. The proposed semi-supervised active learning utilized this spatial variability based on the assumption that pixels in one superpixel have the same label. In each iteration, the proposed method can involve extra "free" (and information bearing) pixels to improve the performance of the classifier. The results of experiments with UH and Galveston hyperspectral images show that the proposed method yields higher classification accuracy compared to related baseline methods with samilar labeling costs.

3.3 Spatial-spectral Active Learning

Pixel-wise spectral features are limited to spectral reflectance information and susceptible to the noise. The hyperspectral images have information in both the spectral and spatial domain. Hence, it is expected (and well understood) that spatial information is important for hyperspectral classification. In the second method we propose, spatial information derived from superpixels is utilized to improve the performance of active learning and classification.

3.3.1 Proposed Method

In this thesis, GLCM features are used as the spatial-spectral features for classification in AL process — we note that this is simply for illustration, and that these features can be replaced by any other spatial features as required by the task at hand. Traditional GLCM is extracted within a fixed size sliding window. In order to capture the texture information for an object, the window must be smaller than the object, but large enough to include the characteristic variability of the object. Hence choosing a proper window size is a challenging task. Moreover, a fixed window size can not fit the objects of different sizes and shapes throughout the image. In order to address these problems, we propose to extract texture from superpixels as adaptive analysis windows. Superpixel boundaries perfectly fits the requirements for an adaptive analysis window, on the one hand, superpixels (when appropriately generated) are generally pure, and can be set to be large enough to capture the underlying texture. On the other hand, the size of superpixels is not fixed, so it can automatically fit objects of different sizes and shapes — infact, typically, they serve as an oversegmentation, and an object is partitioned into several superpixels of varying shapes and sizes, while preserving edges. The 7 GLCM features extracted in this thesis are shown in Table 2.1. The MLRsub classifier is used to automatically fuse multiple feature types because it can efficiently fuse multiple feature types without the need for adhoc parameter estimation. This advantage of MLRsub is detailed in Chapter 2. Moreover, during classification, the importance of each feature can be measured by the weights $\tilde{\boldsymbol{\omega}}_{m_p}^{(k)}\boldsymbol{h}(\boldsymbol{x}_i{}^{m_p}), m_p \in \{1, 2, ..., n\}$ from MLRsub in Eq. 2.19. We propose that these weights can be utilized as a tool to measure online feature importance during active learning and classification. The features in this thesis are defined in two ways: (1) by partitioning the

wavelength spectrum into appropriate subbands. All the bands in an image are divided into 7 groups according to the wavelength range as indicated in Tab 3.5. Each feature in this setup contains the 7 GLCM features extracted from every band in its corresponding wavelength range. (2) by partitioning based on the type of GLCM features, shown in Table 2.1. One GLCM feature derived over of all bands are stacked to form one feature set and this is repeated to generate the multiple feature types.

Table 3.5: Wavelength range

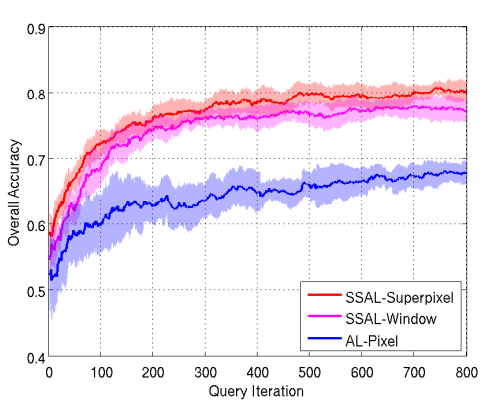
Source	Name	Wavelength (nm)
1	Violet	380-450
2	Blue	450-495
3	Green	495-570
4	Yellow	570-590
5	Orange	590-620
6	Red	620-750
7	Near-infrared	750-2500

3.3.2 Experimental Setup and Results for Spatial-spectral Active Learning (SSAL)

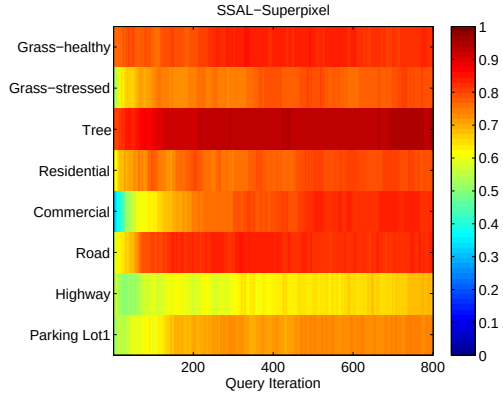
The University of Houston data set and Galveston wetland data are used to test the proposed spatial-spectral AL method. The results of proposed method are compared with the results of a window-based method and a pixel-level (spectral-only) method.

For the first data set, University of Houston, the training set, testing set and candidate set are formed in the same way as in experiment for semi-supervised AL. From Figure 3.10 (wavelengths fusion) and Figure 3.11 (GLCM features fusion), it can be seen that, the spectral-spectral AL outperforms the spectral-only method, pixel level AL. Moreover, the superpixel-based spatial-spectral AL has higher accuracy than window based spatial-spectral AL at all the query iterations. The change of class accuracy during AL process

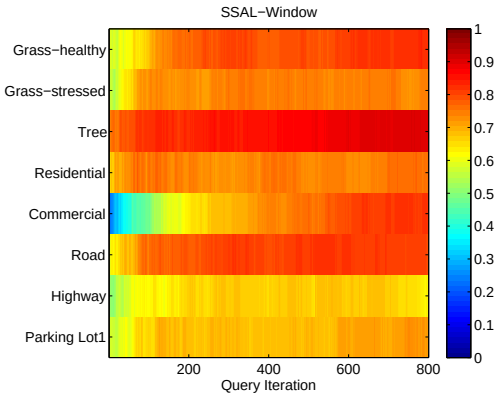
for the three methods are plotted in Figure 3.10 (c) (b) (d) for wavelengths fusion and Fig 3.11 (c) (b) (d) for GLCM features fusion, it can be found that for most of the classes, the class accuracy of superpixel-based spatial-spectral AL is higher and converge faster than the two baseline methods, such as Grass-healthy and Tree. The OA and CA of wavelengths fusion, at the 400th iteration are shown in Table 3.6. The OA of the proposed method is 2.63% and 12.12% higher than that of the window-based method and that the pixel level method, respectively. With regards to class-specific accuracy, the SSAL-superpixel outperforms SSAL-window for 7 classes and outperforms AL-pixel for 8 classes.



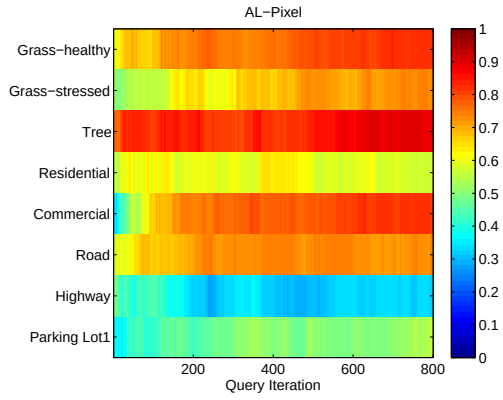
(a) OA along with standard deviation



(b) CA of SSAL-Superpixel



(c) CA of SSAL-Window

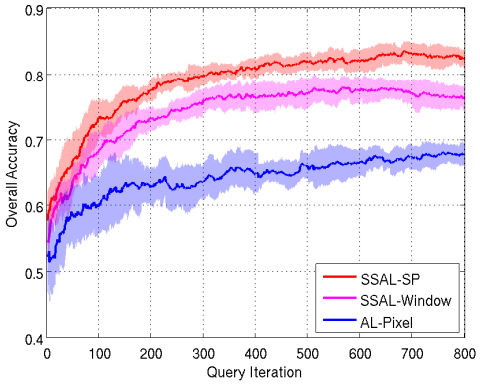


(d) CA of AL-Pixel

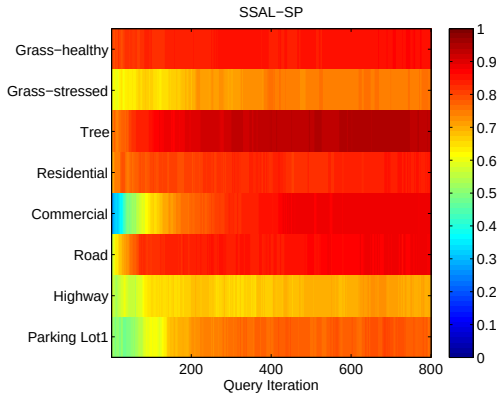
Figure 3.10: OA and CA of UH data with wavelengths fusion

Table 3.6: OA and CA at the final iteration of UH data with wavelengths fusion

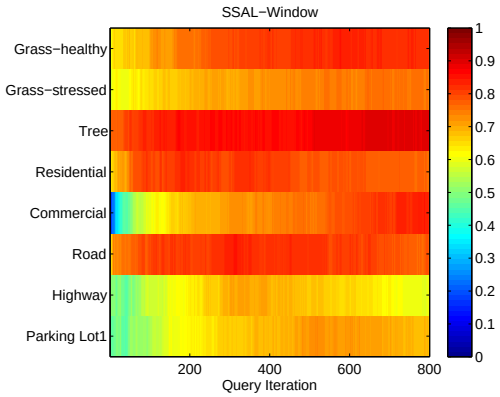
#	Class	SSAL Superpixel	SSAL Window	AL Pixel
1	Grass-healthy	82.20	82.92	81.59
2	Grass-stressed	78.60	74.80	73.59
3	Tree	93.87	89.82	89.07
4	Residential	79.47	75.01	56.91
5	Commercial	82.60	81.66	81.58
6	Road	83.10	79.37	71.98
7	Highway	66.25	64.01	33.50
8	Parking Lot 1	73.64	71.11	54.59
	OA	79.97	77.34	67.85



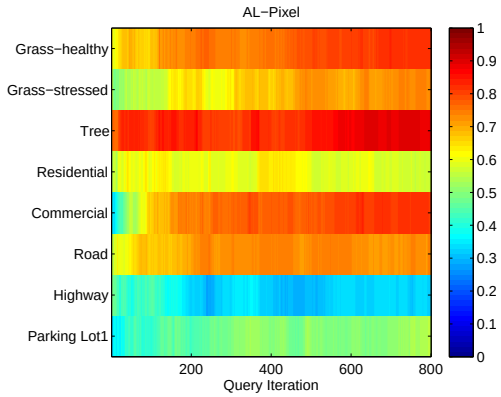
(a) OA along with standard deviation



(b) CA of SSAL-Superpixel



(c) CA of SSAL-Window



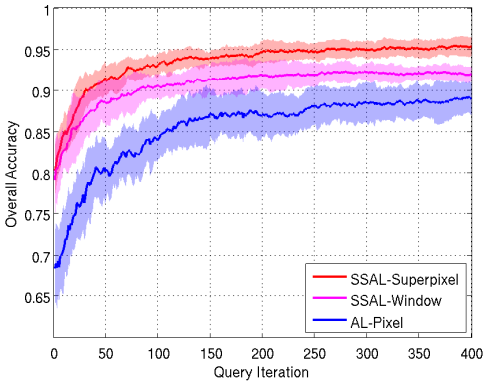
(d) CA of AL-Pixel

Figure 3.11: OA and CA of UH data with GLCM features fusion

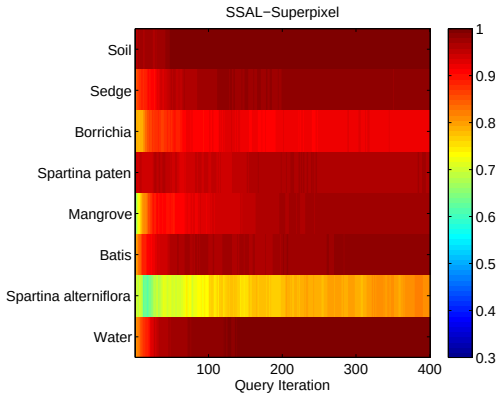
Table 3.7: OA and CA at the final iteration of UH data with GLCM features fusion

#	Class	SSAL Superpixel	SSAL Window	AL Pixel
1	Grass-healthy	84.77	81.99	81.59
2	Grass-stressed	75.23	74.92	73.59
3	Tree	93.62	89.32	89.07
4	Residential	84.74	76.23	56.91
5	Commercial	88.67	83.15	81.58
6	Road	87.56	77.59	71.98
7	Highway	69.00	60.50	33.50
8	Parking Lot 1	75.65	69.11	54.59
	OA	82.40	76.60	67.85

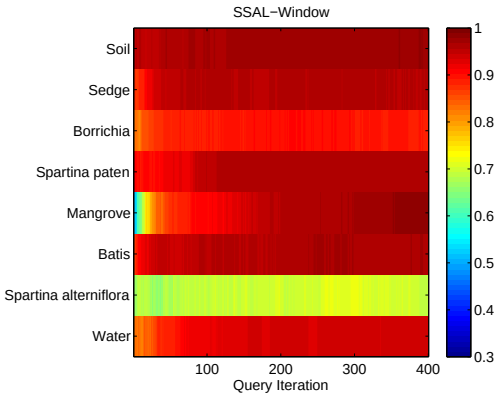
A similar trend was observed for the Galveston wetland dataset. The training set, testing set and candidate set are set in the same way as in experiment for semi-supervised AL. From the Figure 3.12 (a) for wavelength based fusion and Figure 3.13 (a) for GLCM feature based fusion, it can be found that the proposed superpixel-based spatial-spectral AL yields the highest OA through all the query iterations and it has relatively small standard deviation. Figure 3.12 (c) (b) (d) and Figure 3.13 (c) (b) (d) show how the class accuracy for each class changes through all the iterations using different AL method. It can be found the SSAL-superpixel converges faster and has higher class accuracy for most of the classes, such as soil, spartina alterniflora and water. Table 3.8 shows that with the wavelengths fusion, at the converged iteration, the OA of proposed superpixel-based spatial-spectral AL is 3.33% and 7.20% higher than that of the window based spatial-spectral method and that of pixel level AL, respectively. Table 3.9 shows that with the GLCM feature fusion, at converged iterations, the OA of the proposed superpixel-based spatial-spectral AL is 3.00% and 6.10% better than that of the window based spatial-spectral method and that of the pixel level AL respectively.



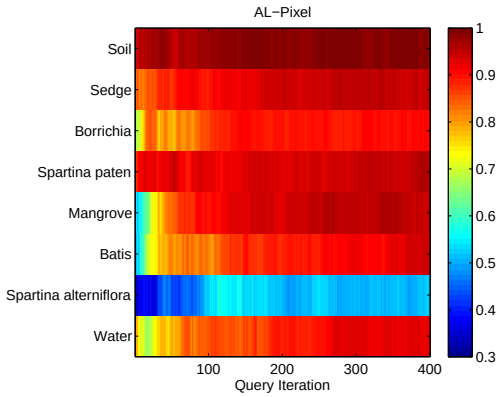
(a) OA along with standard deviation



(b) CA of SSAL-Superpixel



(c) CA of SSAL-Window

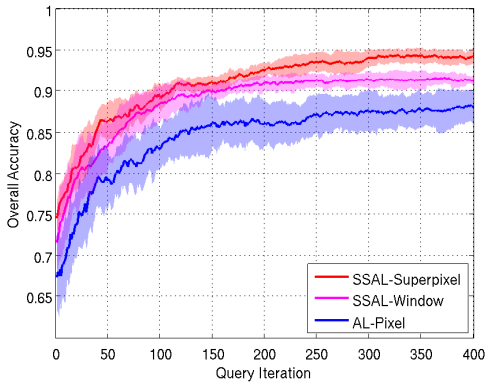


(d) CA of AL-Pixel

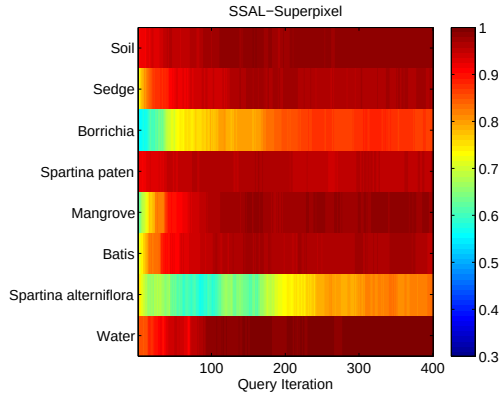
Figure 3.12: OA and CA of Galveston wetland with wavelengths fusion

Table 3.8: OA and CA at the final iteration of Galveston wetland with wavelengths fusion

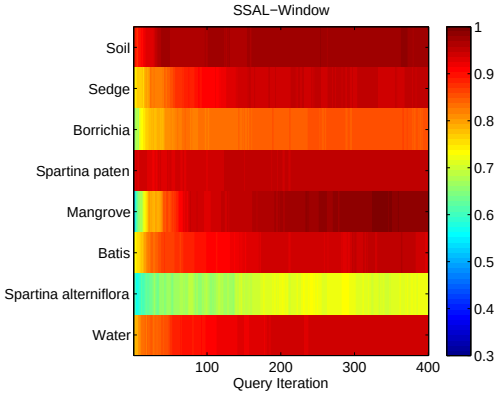
#	Class	SSAL Superpixel	SSAL Window	AL Pixel
1	Soil	99.83	97.90	98.02
2	Sedge	98.00	95.83	93.83
3	Borrichia	91.86	88.69	89.02
4	Spartina paten	96.83	96.50	94.56
5	Mangrove	97.17	98.00	93.76
6	Batis	98.67	96.50	94.17
7	Spartina-alterniflora	81.33	69.17	50.67
8	Water	99.34	93.83	91.38
	OA	95.38	92.05	88.18



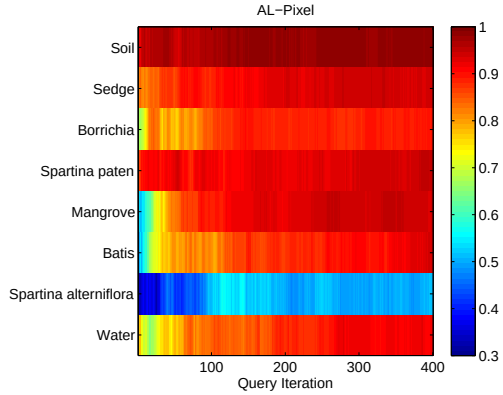
(a) OA along with standard deviation



(b) CA of SSAL-Superpixel



(c) CA of SSAL-Window



(d) CA of AL-Pixel

Figure 3.13: OA and CA of Galveston wetland with GLCM features fusion

Table 3.9: OA and CA at the final iteration of Galveston wetland with GLCM features fusion

#	Class	SSAL Superpixel	SSAL Window	AL Pixel
1	Soil	98.33	97.57	98.02
2	Sedge	97.01	94.83	93.83
3	Borrichia	86.57	85.02	89.02
4	Spartina paten	95.50	95.00	94.56
5	Mangrove	97.67	98.67	93.76
6	Batis	95.75	94.47	94.17
7	Spartina-alterniflora	83.67	70.83	50.67
8	Water	99.71	93.83	91.38
	OA	94.28	91.28	88.18

3.3.3 Conclusion

In this work, the spatial-spectral features are extracted from within the superpixel boundary. The motivation for this idea is that superpixels not only oversegment while effectively protecting the boundary of an object, but can also serve as an adaptive analysis window. Next, the features are utilized in an AL framework. Experimental results show that the proposed superpixel-based spatial-spectral AL drive the OA to converge faster and higher than the window-based method and the pixel level method. Looking at the class accuracy changing process in detail, it can be seen that the class accuracy of the proposed method converges faster and higher for most of the classes.

3.4 Feature Importance

Feature importance can be considered as a quantitative measurement of the contribution of one or more features for the underlying classification task. As discussed in Chapter 2, the importance of feature m_p can be measured by the weight $\tilde{\boldsymbol{\omega}}_{m_p}^{(k)}\boldsymbol{h}(\boldsymbol{x}_i{}^{m_p})$ in Eq. 2.19, which is learned automatically.

3.4.1 Experimental Setup and Results for Feature Importance to Determine Useful Spectral Channels

The wavelength features are defined according to the wavelength range of the visible spectrum and near-infrared spectrum. All the bands are divided into 7 features listed in Table 3.5. The weights are calculated within each class respectively and normalized to [0 1].

The normalized feature weights of UH and Galveston wetland are shown in Table 3.10 and Table 3.11, respectively. Features whose weight is larger than a preset threshold are considered as dominant features for the underlying classification task. The dominant features for each class of UH with the threshold of 0.5 are shown in Table 3.12. The dominant features for each class of Galveston with the threshold of 0.5 are shown in Table 3.13.

Table 3.10: Normalized wavelength feature weights for each class of UH

	$\boldsymbol{\omega}_{s_1}\boldsymbol{h}((\boldsymbol{x}_i)_{s_1})$	$\boldsymbol{\omega}_{s_2}\boldsymbol{h}((\boldsymbol{x}_i)_{s_2})$	$\boldsymbol{\omega}_{s_3}\boldsymbol{h}((\boldsymbol{x}_i)_{s_3})$	$\boldsymbol{\omega}_{s_4}\boldsymbol{h}((\boldsymbol{x}_i)_{s_4})$	$\boldsymbol{\omega}_{s_5}\boldsymbol{h}((\boldsymbol{x}_i)_{s_5})$	$\boldsymbol{\omega}_{s_6}\boldsymbol{h}((\boldsymbol{x}_i)_{s_6})$	$\boldsymbol{\omega}_{s_7}\boldsymbol{h}((\boldsymbol{x}_i)_{s_7})$
Grass-healthy	0.00	0.45	0.44	0.33	0.45	0.13	**1.00**
Grass-stressed	0.16	0.00	**1.00**	**0.74**	0.48	0.07	**0.97**
Tree	0.30	**0.50**	0.15	0.00	**0.62**	0.11	**1.00**
Residential	**0.81**	**0.88**	0.00	0.32	0.40	**1.00**	**0.93**
Commercial	0.40	**1.00**	0.17	0.00	0.19	**0.60**	**0.57**
Road	**0.95**	0.08	**0.93**	**1.00**	0.00	**0.81**	**0.71**
Highway	**0.57**	0.28	0.30	**1.00**	0.00	**0.76**	0.34
Parking Lot 1	**1.00**	0.13	**0.92**	0.00	**0.68**	**0.77**	0.37

Table 3.11: Normalized wavelength feature weights for each class of Galveston wetland

	$\boldsymbol{\omega}_{s_1}\boldsymbol{h}((\boldsymbol{x}_i)_{s_1})$	$\boldsymbol{\omega}_{s_2}\boldsymbol{h}((\boldsymbol{x}_i)_{s_2})$	$\boldsymbol{\omega}_{s_3}\boldsymbol{h}((\boldsymbol{x}_i)_{s_3})$	$\boldsymbol{\omega}_{s_4}\boldsymbol{h}((\boldsymbol{x}_i)_{s_4})$	$\boldsymbol{\omega}_{s_5}\boldsymbol{h}((\boldsymbol{x}_i)_{s_5})$	$\boldsymbol{\omega}_{s_6}\boldsymbol{h}((\boldsymbol{x}_i)_{s_6})$	$\boldsymbol{\omega}_{s_7}\boldsymbol{h}((\boldsymbol{x}_i)_{s_7})$
Soil	**0.53**	0.00	**1.00**	0.32	0.28	0.31	0.04
Sedge	**0.60**	**0.83**	0.00	0.39	**0.60**	**1.00**	**0.81**
Borrichia	0.05	0.00	**1.00**	0.09	0.00	0.05	0.42
Spartina paten	0.34	0.28	0.00	**0.51**	0.47	0.40	**1.00**
Mangrove	0.08	0.30	0.00	0.21	0.21	0.18	**1.00**
Batis	0.25	0.00	**0.62**	**1.00**	**0.61**	0.11	**0.55**
Spartina-alterniflora	0.00	0.19	0.21	0.00	0.08	0.18	**1.00**
Water	**1.00**	0.17	0.41	0.12	0.10	0.09	0.00

Table 3.12: Dominant wavelength features for each class of UH

Class	Dominant Feature
Grass-healthy	s_7
Grass-stressed	s_3 s_4 s_7
Tree	s_2 s_5 s_7
Residential	s_1 s_2 s_6 s_7
Commercial	s_2 s_6 s_7
Road	s_1 s_3 s_4 s_6 s_7
Highway	s_1 s_4 s_6
Parking Lot 1	s_1 s_3 s_5 s_6

Table 3.13: Dominant wavelength features for each class of Galveston wetland

Class	Dominant Feature
Soil	s_1 s_3
Sedge	s_1 s_2 s_5 s_6 s_7
Borrichia	s_3
Spartina paten	s_4 s_7
Mangrove	s_7
Batis	s_3 s_4 s_5 s_7
Spartina-alterniflora	s_7
Water	s_1

From Table 3.12, It can be seen that for UH data, 6 classes are dominated by the feature S_7. 4 classes are dominated by feature S_1, and 5 classes are dominated S_6. It can be concluded that the classification of UH data is dominated by feature S_1, S_6 and S_7. With this result, It can be gathered that the information from visible spectrum and near-infrared spectrum are both important for classification, which is something that we indeed expect from a-prior knowledge in this example. In order to further validate the measurement of feature importance by weights from MLR-sub. The Fisher's ratio of each feature is calculated. The normalized Fisher's ratio (range$-[0\ \ 1]$) is shown in Table 3.14. Note that the three largest Fisher's ratio comes (across all classes) come from S_1, S_6 and S_7, which is consistent with the result derived from the weights of MLRsub. We note that

these weights can hence be used to quantify feature importance in an online setting, as more training data is inducted via active learning. These weights can also be utilized as a query metric to induct samples that have several dominant features.

Table 3.14: Normalized Fisher's ratio for wavelengths of UH

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
Fisher's ratio	1.00	0.14	0.07	0.00	0.09	0.18	0.35

Similarly, for Galveston data, 7 classes are dominated by the feature S_7. 3 classes are dominated by feature S_1, and 3 classes are dominated by feature S_3. Hence S_1, S_3 and S_7 are the dominant feature for the classification of Galveston data. From the Table 3.15, the three most dominant features are S_1, S_5 and S_7. Two of the most important features derived from two methods are the same, which further validate the feature importance measurement by MLRsub weights.

Table 3.15: Normalized Fisher's ratio for wavelengths of Galveston Wetland

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
Fisher's ratio	0.58	0.00	0.15	0.26	0.28	0.22	1.00

3.4.2 Experimental Setup and Results Demonstrating the Use of Feature Importance to Determine Useful GLCM Features

The 7 GLCM features extracted in this thesis are shown in Table 2.1. The normalized feature weights of UH and Galveston wetland are shown in Table 3.16 and Table 3.19, respectively. The feature whose weight is larger than the preset threshold is considered as the dominant feature for classification. The dominant features for each class of UH with the threshold of 0.5 is shown in Table 3.17. The dominant features for each class of Galveston with the threshold of 0.5 is shown in Table 3.20.

From Table 3.17, it can be found that the classification of UH data is dominated by feature S_2, S_4 and S_6, among which the feature S_6 is the average spectral signature, and S_2 and S_4 are two texture features. Hence, both spectral feature and texture feature are important for classification, which would be a good motivation for the proposed spatial-spectral AL. The dominant features calculated by Fisher' ratio are also S_2, S_4 and S_6.

For Galveston data, the dominant features calculated by MLRsub weights are S_1, S_4, S_6, which means the both the spectral feature and the texture feature contribute to the classification. From the Fisher's ratio values, the first four dominant features are S_1, S_2, S_4 and S_6. These results further validates the utility of MLRSub weights as a means to quantify feature importance.

Table 3.16: Normalized GLCM feature weights for each class of UH

	$\boldsymbol{\omega}_{s_1}\boldsymbol{h}((\boldsymbol{x}_i)_{s_1})$	$\boldsymbol{\omega}_{s_2}\boldsymbol{h}((\boldsymbol{x}_i)_{s_2})$	$\boldsymbol{\omega}_{s_3}\boldsymbol{h}((\boldsymbol{x}_i)_{s_3})$	$\boldsymbol{\omega}_{s_4}\boldsymbol{h}((\boldsymbol{x}_i)_{s_4})$	$\boldsymbol{\omega}_{s_5}\boldsymbol{h}((\boldsymbol{x}_i)_{s_5})$	$\boldsymbol{\omega}_{s_6}\boldsymbol{h}((\boldsymbol{x}_i)_{s_6})$	$\boldsymbol{\omega}_{s_7}\boldsymbol{h}((\boldsymbol{x}_i)_{s_7})$
Grass-healthy	0.27	0.00	0.44	0.28	0.28	**1.00**	0.28
Grass-stressed	0.17	0.00	0.32	0.05	0.17	**1.00**	0.26
Tree	0.16	0.00	0.24	**1.00**	0.18	0.48	0.22
Residential	0.27	0.48	0.22	**1.00**	0.24	0.00	0.06
Commercial	0.04	0.00	0.00	0.09	0.03	**1.00**	0.03
Road	0.04	0.00	0.03	0.02	0.03	**1.00**	0.10
Highway	0.41	**1.00**	0.00	**0.52**	0.40	**0.66**	0.38
Parking Lot 1	0.16	**0.64**	0.00	0.21	0.20	**1.00**	0.24

Table 3.17: Dominant GLCM features for each class of UH

Class	Dominant Feature
Soil	s_6
Sedge	s_6
Borrichia	s_4
Spartina paten	s_4
Mangrove	s_6
Batis	s_6
Spartina-alterniflora	s_2 s_4 s_6
Water	s_2 s_6

Table 3.18: Normalized Fisher's ratio for GLCM features of UH

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
Fisher's ratio	0.04	0.47	0.29	0.40	0.00	1.00	0.00

Table 3.19: Normalized source weight for each class of Galveston

	$\boldsymbol{\omega}_{s_1}\boldsymbol{h}((\boldsymbol{x}_i)_{s_1})$	$\boldsymbol{\omega}_{s_2}\boldsymbol{h}((\boldsymbol{x}_i)_{s_2})$	$\boldsymbol{\omega}_{s_3}\boldsymbol{h}((\boldsymbol{x}_i)_{s_3})$	$\boldsymbol{\omega}_{s_4}\boldsymbol{h}((\boldsymbol{x}_i)_{s_4})$	$\boldsymbol{\omega}_{s_5}\boldsymbol{h}((\boldsymbol{x}_i)_{s_5})$	$\boldsymbol{\omega}_{s_6}\boldsymbol{h}((\boldsymbol{x}_i)_{s_6})$	$\boldsymbol{\omega}_{s_7}\boldsymbol{h}((\boldsymbol{x}_i)_{s_7})$
Soil	0.11	0.00	0.18	**1.00**	0.19	**0.53**	0.14
Sedge	0.05	0.19	0.07	**1.00**	0.06	0.00	0.06
Borrichia	0.00	0.34	0.11	**1.00**	0.11	0.07	0.11
Spartina paten	0.07	0.17	0.15	0.00	0.14	**1.00**	0.21
Mangrove	**0.91**	0.11	0.12	0.00	0.19	**1.00**	0.09
Batis	**1.00**	0.24	0.30	0.00	0.30	**0.62**	0.29
Spartina-alterniflora	**1.00**	0.45	0.41	0.00	0.48	**0.55**	0.43
Water	**1.00**	0.28	0.28	0.00	0.38	**0.71**	0.28

Table 3.20: Dominant GLCM features for each class of Galveston wetland

Class	Dominant Feature
Grass-healthy	s_4 s_6
Grass-stressed	s_4
Tree	s_4
Residential	s_6
Commercial	s_1 s_6
Road	s_1 s_6
Highway	s_1 s_6
Parking Lot 1	s_1 s_6

Table 3.21: Normalized Fisher's ratio for GLCM features of Galveston Wetland

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
Fisher's ratio	1.00	0.20	0.04	0.23	0.03	0.16	0.00

3.4.3 Conclusion

In this work, the feature importance is calculated through the feature's corresponding weight in the MLRsub algorithm. The results are validated with Fisher's ratio. It demonstrates that this feature importance method is effective and efficient and can facilitate rapid online importance scoring of disparate features.

Chapter 4

Conclusion

In hyperspectral image analysis applications, the annotation of data is expensive and time consuming. Hence, constructing an efficient training set is an important part of a hyperspectral image analysis workflow. Active learning methods facilitate annotation driven by incorporating the classifier in the loop, and hence result in training libraries that are most informative to the resulting classifier. In the AL process, an expert labels the most uncertain samples for the classifier, after they are inducted in the training library, enhancing classification performance. This thesis proposed two superpixel-based active learning methods, utilizing the spatial-spectral information for efficient and robust classification. The first method can be considered as a semi-supervised variant of active learning based on superpixels. Pixels in the same superpixel are expected to have similar labels, yet contain information about local spectral variability. Compared to the traditional AL method, the proposed method not only involves the query strategy selected pixel but also its neighbors in the same superpixel with the assumption that they all have the same label. As a result, in each query iteration, the proposed method collected more information without any additional annotation cost. The experimental results shows that the proposed method converges much faster than the baseline method. The second method is a spatial-spectral active learning system based on superpixel. It is built on the observation that in addition to spectral information, spatial context is very useful for classification — superpixels are

used to integrate spatial and spectral information simultaneously for active learning and classification. The proposed method derives spatial features (e.g., texture) using the superpixels as an adaptive analysis window. The experimental results show that the classification performance of the proposed method converges faster and higher compared to the baseline methods. Moreover, this thesis proposes a scheme to learn online feature importance based on the feature weights in the MLRsub classifier. These can be useful to visualize and interpret importance of features in online (e.g., active) learning and can also be used in a metric that seeks samples with a large number of activated (dominant) features.

References

[1] D. Manolakis and G. Shaw, "Detection algorithms for hyperspectral imaging applications," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 29–43, 2002.

[2] D. W. Stein, S. G. Beaven, L. E. Hoff, E. M. Winter, A. P. Schaum, and A. D. Stocker, "Anomaly detection from hyperspectral imagery," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 58–69, 2002.

[3] F. Lacar, M. Lewis, and I. Grierson, "Use of hyperspectral imagery for mapping grape varieties in the barossa valley, south australia," in *Geoscience and Remote Sensing Symposium, 2001. IGARSS'01. IEEE 2001 International*, vol. 6. IEEE, 2001, pp. 2875–2877.

[4] Y.-Z. Feng and D.-W. Sun, "Application of hyperspectral imaging in food safety inspection and control: a review," *Critical reviews in food science and nutrition*, vol. 52, no. 11, pp. 1039–1058, 2012.

[5] A. Shahidi, S. Patel, J. Flanagan, and C. Hudson, "Regional variation in human retinal vessel oxygen saturation," *Experimental eye research*, vol. 113, pp. 143–147, 2013.

[6] P. W. Yuen and M. Richardson, "An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition," *The Imaging Science Journal*, vol. 58, no. 5, pp. 241–253, 2010.

[7] P. Mitra, B. U. Shankar, and S. K. Pal, "Segmentation of multispectral remote sensing

images using active support vector machines," *Pattern recognition letters*, vol. 25, no. 9, pp. 1067–1074, 2004.

[8] A. Vlachos, "Active learning with support vector machines," *Master of Science, School of Informatics, University of Edinburgh, UK*, 2004.

[9] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, "Svm active learning approach for image classification using spatial information," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, no. 4, pp. 2217–2233, 2014.

[10] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. Springer-Verlag New York, Inc., 1994, pp. 3–12.

[11] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semi-supervised hyperspectral image segmentation," in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, 2009. WHISPERS'09. First Workshop on*. IEEE, 2009, pp. 1–4.

[12] T. Luo, K. Kramer, S. Samson, A. Remsen, D. Goldgof, L. Hall, and T. Hopkins, "Active learning to recognize multiple types of plankton," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3. IEEE, 2004, pp. 478–481.

[13] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Hyperspectral image segmentation using a new bayesian approach with active learning," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 49, no. 10, pp. 3947–3960, 2011.

[14] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proceedings of the fifth annual workshop on Computational learning theory.* ACM, 1992, pp. 287–294.

[15] M. Khodadadzadeh, J. Li, A. Plaza, and J. M. Bioucas-Dias, "A subspace-based multinomial logistic regression for hyperspectral image classification," 2014.

[16] T. Priya, S. Prasad, and H. Wu, "Superpixels for spatially reinforced bayesian classification of hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 5, pp. 1071–1075, 2015.

[17] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603–619, 2002.

[18] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[19] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE transactions on pattern analysis and machine intelligence*, vol. 13, no. 6, pp. 583–598, 1991.

[20] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: Fast superpixels using geometric flows," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 12, pp. 2290–2297, 2009.

[21] O. Veksler, Y. Boykov, and P. Mehrani, "Superpixels and supervoxels in an energy optimization framework," in *Computer Vision–ECCV 2010.* Springer, 2010, pp. 211–224.

[22] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on.* IEEE, 2003, pp. 10–17.

[23] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functionsi," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[24] A. Plaza, J. A. Benediktsson, J. W. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, and A. Gualtieri, "Recent advances in techniques for hyperspectral image processing," *Remote sensing of environment*, vol. 113, pp. S110–S122, 2009.

[25] D. Böhning, "Multinomial logistic regression algorithm," *Annals of the Institute of Statistical Mathematics*, vol. 44, no. 1, pp. 197–200, 1992.

[26] B. Krishnapuram, L. Carin, M. A. Figueiredo, and A. J. Hartemink, "Sparse multinomial logistic regression: Fast algorithms and generalization bounds," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 6, pp. 957–968, 2005.

[27] J. A. Benediktsson, J. R. Sveinsson, and P. H. Swain, "Hybrid consensus theoretic classification," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 35, no. 4, pp. 833–843, 1997.

[28] J. Li, P. Reddy Marpu, A. Plaza, J. M. Bioucas-Dias, and J. Atli Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 51, no. 9, pp. 4816–4829, 2013.

[29] B. Settles, "Active learning literature survey," *University of Wisconsin, Madison*, vol. 52, no. 55-66, p. 11, 2010.

[30] S. E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on gabor filters," *Image Processing, IEEE Transactions on*, vol. 11, no. 10, pp. 1160–1167, 2002.

[31] P. Soille, *Morphological image analysis: principles and applications*. Springer Science & Business Media, 2013.

[32] P.-H. Hsu, Y.-H. Tseng, and P. Gong, "Dimension reduction of hyperspectral images for classification applications," *Geographic Information Sciences*, vol. 8, no. 1, pp. 1–8, 2002.

[33] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *Systems, Man and Cybernetics, IEEE Transactions on*, no. 6, pp. 610–621, 1973.