

Hyperspectral tree crown classification using the multiple instance adaptive cosine estimator (#29716)

1

First submission

Editor guidance

Please submit by **6 Sep 2018** for the benefit of the authors (and your \$200 publishing discount).



Structure and Criteria

Please read the 'Structure and Criteria' page for general guidance.



Raw data check

Review the raw data. Download from the location [described by the author](#).



Image check

Check that figures and images have not been inappropriately manipulated.

Privacy reminder: If uploading an annotated PDF, remove identifiable information to remain anonymous.

Files

Download and review all files from the [materials page](#).

18 Figure file(s)

2 Latex file(s)

1 Table file(s)



Structure your review

The review form is divided into 5 sections. Please consider these when composing your review:

1. BASIC REPORTING
2. EXPERIMENTAL DESIGN
3. VALIDITY OF THE FINDINGS
4. General comments
5. Confidential notes to the editor






 You can also annotate this PDF and upload it as part of your review

When ready [submit online](#).





Editorial Criteria

Use these criteria points to structure your review. The full detailed editorial criteria is on your [guidance page](#).





BASIC REPORTING

-  Clear, unambiguous, professional English language used throughout.
-  Intro & background to show context. Literature well referenced & relevant.
-  Structure conforms to [PeerJ standards](#), discipline norm, or improved for clarity.
-  Figures are relevant, high quality, well labelled & described.
-  Raw data supplied (see [PeerJ policy](#)).

EXPERIMENTAL DESIGN

-  Original primary research within [Scope of the journal](#).
-  Research question well defined, relevant & meaningful. It is stated how the research fills an identified knowledge gap.
-  Rigorous investigation performed to a high technical & ethical standard.
-  Methods described with sufficient detail & information to replicate.

VALIDITY OF THE FINDINGS

-  Impact and novelty not assessed. Negative/inconclusive results accepted. *Meaningful* replication encouraged where rationale & benefit to literature is clearly stated.
-  Speculation is welcome, but should be identified as such.
-  Conclusions are well stated, linked to original research question & limited to supporting results.
-  Data is robust, statistically sound, & controlled.



The best reviewers use these techniques

Tip

Support criticisms with evidence from the text or from other sources

Example

Smith et al (J of Methodology, 2005, V3, pp 123) have shown that the analysis you use in Lines 241-250 is not the most appropriate for this situation. Please explain why you used this method.

Give specific suggestions on how to improve the manuscript

Your introduction needs more detail. I suggest that you improve the description at lines 57- 86 to provide more justification for your study (specifically, you should expand upon the knowledge gap being filled).

Comment on language and grammar issues

The English language should be improved to ensure that an international audience can clearly understand your text. Some examples where the language could be improved include lines 23, 77, 121, 128 - the current phrasing makes comprehension difficult.

Organize by importance of the issues, and number your points

1. Your most important issue
2. The next most important item
3. ...
4. The least important points

Please provide constructive criticism, and avoid personal opinions

I thank you for providing the raw data, however your supplemental files need more descriptive metadata identifiers to be useful to future readers. Although your results are compelling, the data analysis should be improved in the following ways: AA, BB, CC

Comment on strengths (as well as weaknesses) of the manuscript

I commend the authors for their extensive data set, compiled over many years of detailed fieldwork. In addition, the manuscript is clearly written in professional, unambiguous language. If there is a weakness, it is in the statistical analysis (as I have noted above) which should be improved upon before Acceptance.

Hyperspectral tree crown classification using the multiple instance adaptive cosine estimator

Sheng Zou ¹, Paul Gader ², Alina Zare ^{Corresp. 1}

¹ Department of Electrical and Computer Engineering, University of Florida, Gainesville, Florida, United States

² Department of Computer & Information Science & Engineering, University of Florida, Gainesville, Florida, United States

Corresponding Author: Alina Zare
Email address: azare@ufl.edu

Tree species classification using hyperspectral imagery is a challenging task due to the high spectral similarity between species and large intra-species variability. This paper proposes a solution using the Multiple Instance Adaptive Cosine Estimator (MI-ACE) algorithm. MI-ACE estimates a discriminative target signature to differentiate between a pair of tree species while accounting for label uncertainty. Additionally, the performance of MI-ACE does not rely on parameter settings that require tuning resulting in a method that is easy to use in application. Results presented are using training and testing data provided by a data analysis competition aimed at encouraging the development of methods for extracting ecological information through remote sensing obtained through participation in the competition.

Hyperspectral tree crown classification using the multiple instance adaptive cosine estimator

Sheng Zou¹, Paul Gader², Alina Zare¹

¹ Department of Electrical and Computer Engineering, University of Florida, Gainesville, Florida, USA

² Department of Computer & Information Science & Engineering, University of Florida, Gainesville, Florida, USA

Corresponding Author:

Alina Zare¹

PO Box 6130, Gainesville, FL, 32611-6130

Email address: azare@ufl.edu

Hyperspectral tree crown classification using the multiple instance adaptive cosine estimator

Sheng Zou¹, Paul Gader², and Alina Zare¹

¹Department of Electrical and Computer Engineering, University of Florida

²Department of Computer & Information Science & Engineering, University of Florida

ABSTRACT

Tree species classification using hyperspectral imagery is a challenging task due to the high spectral similarity between species and large intra-species variability. This paper proposes a solution using the Multiple Instance Adaptive Cosine Estimator (MI-ACE) algorithm. MI-ACE estimates a discriminative target signature to differentiate between a pair of tree species while accounting for label uncertainty. Additionally, the performance of MI-ACE does not rely on parameter settings that require tuning resulting in a method that is easy to use in application. Results presented are using training and testing data provided by a data analysis competition aimed at encouraging the development of methods for extracting ecological information through remote sensing obtained through participation in the competition.

Keywords: tree crown, classification, hyperspectral, multiple instance

INTRODUCTION

Spectral signatures of tree crowns across species often have high spectral similarity as well as significant intra-species variability (Cochrane, 2000), making tree crown classification from hyperspectral imagery a challenging task. In this work, a discriminative multiple instance hyperspectral target characterization method, the Multiple Instance Adaptive Cosine Estimator (MI-ACE) algorithm (Zare et al., 2018b), is proposed for this problem.

In many remote sensing applications, precise pixel level training labels are expensive or infeasible to obtain (Blum and Mitchell, 1998). In the case of tree crown classification, pixel level ground truth labeling for tree crowns can be extremely challenging to collect. When looking at aerial hyperspectral imagery, given overlapping tree crowns and *mixed pixels* in which individual pixels contain responses from multiple neighboring tree species due to the image spatial resolution, manually labeling individual tree crowns is generally infeasible as the precise outline of each tree cannot be easily identified. Marconi et al. (2018) organized data science challenges for airborne remote sensing data. One of these challenges was to perform species classification of individual trees given airborne hyperspectral data. The challenge provided competitors (Anderson, 2018; Sumsion et al., 2018; Dalponte et al., 2018) with training and testing hyperspectral signatures extracted from individual tree crowns in the National Ecological Observatory Network (NEON) hyperspectral data collected at the Ordway-Swisher Biological Station in north-central Florida. These signatures were extracted from the imagery and labeled by the competition organizers by generating individual tree crown polygons using a tablet computer, GIS software, and an external GPS device in the field as described by Marconi et al. (2018). The team loaded the aerial hyperspectral imagery onto tablet computers in the field and simultaneously visually assessed the scene in person and the aerial imagery to mark and digitize the outlines of individual tree crowns. This was a time consuming process that required some subjectivity in assessing the field and the overhead view. The difficulty of this process and the subjectivity needed may result in some individual pixels in tree crown polygons being mislabeled. The MI-ACE algorithm presented in this paper is designed to be robust to this sort of imprecise labels without the need for any parameter tuning or any additional steps for outlier removal.

MI-ACE is a multiple instance learning (MIL) algorithm (Maron and Lozano-Pérez, 1998) where precise instance level labels are not necessary. Instead only a *bag* level label indicating the existence or absence of a target in a bag (or set) of instances is needed. In MIL, a bag is labeled as a *positive bag*

39 containing a target if at least one data point in the bag corresponds to target and a bag is labeled as a
 40 *negative bag* if none of the data in the bag correspond to the target. The MI-ACE algorithm estimates
 41 a discriminative target signature from data with this sort of bag-level labels. This target signature can
 42 then be used within the ACE detector to perform pixel-level target detection and classification (Kraut
 43 et al., 2001). Since MI-ACE needs only bag-level labels, the algorithm naturally addresses the tree crown
 44 classification problem outlined above. Each tree crown (and the associated set of hyperspectral signatures)
 45 are considered a bag and that bag is labeled as the corresponding target tree species. Since MI-ACE
 46 assumes multiple instance style labels, the algorithm does not assume that each pixel in every bag is
 47 representative of the associated tree species (but only assumes that there exists at least one representative
 48 signature in the tree crown) and, thus, addresses imprecision in labeling.

49 PROPOSED APPROACH

50 In this section, a brief review of MI-ACE is presented, then the proposed one-vs-one MI-ACE tree species
 51 classification approach is outlined.

52 MI-ACE target characterization

53 MI-ACE (Zare et al., 2018b) is a discriminative target characterization method that based on ACE detector
 54 and multiple instance concept learning. In multiple instance learning, sets of data points (termed bags) are
 55 labeled as either positive or negative. Specifically, let $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ be training data where
 56 d is the dimensionality of an instance, \mathbf{x}_i , and N is the total number of training instances. The data is
 57 grouped into K bags, $\mathbf{B} = \{\mathbf{B}_1, \dots, \mathbf{B}_K\}$, with associated binary bag-level labels, $L = \{L_1, \dots, L_K\}$ where
 58 $L_j \in \{0, 1\}$ and $\mathbf{x}_{ji} \in \mathbf{B}_j$ denotes the i^{th} instance in bag \mathbf{B}_j . Positive bags (i.e., \mathbf{B}_j with $L_j = 1$, denoted as
 59 \mathbf{B}_j^+) contain at least one instance composed of some target:

$$\text{if } L_j = 1, \exists \mathbf{x}_{ji} \in \mathbf{B}_j^+ \text{ s.t. } \mathbf{x}_{ji} \sim \mathcal{N}(\alpha_i \mathbf{s} + \mu_b, \sigma_1^2 \Sigma_b), \alpha_i \neq 0 \quad (1)$$

where Σ_b is the background covariance, μ_b is the mean of the background, s is the known target signature
 which is scaled by a target abundance, a , and $\sigma_1^2 = \frac{1}{a} (\mathbf{x} - a\mathbf{s})^T \Sigma_b^{-1} (\mathbf{x} - a\mathbf{s})$. However, the number of
 instances in a positive bag with a target component is unknown. If \mathbf{B}_j is a negative bag (i.e., $L_j = 0$,
 denoted as \mathbf{B}_j^-), then this indicates that \mathbf{B}_j^- does not contain any target:

$$\text{if } L_j = 0, \mathbf{x}_{ji} \sim \mathcal{N}(\mu_b, \sigma_1^2 \Sigma_b) \forall \mathbf{x}_{ji} \in \mathbf{B}_j^- \quad (2)$$

Given this problem formulation, the goal of MI-ACE is to estimate the target signature, \mathbf{s} , that
 maximizes the corresponding adaptive cosine estimator (ACE) detection statistic for the target instances
 in each positive bag and minimize the detection statistic over all negative instances. This is accomplished
 by maximizing the following objective:

$$\arg \max_{\mathbf{s}} \frac{1}{N^+} \sum_{j:L_j=1} D_{ACE}(\mathbf{x}_j^*, \mathbf{s}) - \frac{1}{N^-} \sum_{j:L_j=0} \frac{1}{N_j^-} \sum_{\mathbf{x}_i \in \mathbf{B}_j^-} D_{ACE}(\mathbf{x}_i, \mathbf{s}) \quad (3)$$

where N^+ and N^- are the number of positive and negative bags, respectively, N_j^- is the number of
 instances in the j^{th} negative bag, and \mathbf{x}_j^* is the selected instance from the positive bag \mathbf{B}_j^+ that is mostly
 likely a target instance in the bag. The selected instance is identified as the point with the maximum
 detection statistic given a target signature, \mathbf{s} :

$$\mathbf{x}_j^* = \arg \max_{\mathbf{x}_i \in \mathbf{B}_j^+} D_{ACE}(\mathbf{x}_i, \mathbf{s}) \quad (4)$$

Since the first term of the objective function relies only on the selected instance from each positive bag,
 the method is robust to outliers and incorrectly labeled samples. The D_{ACE} is the ACE detection statistic,

$$D_{ACE}(\mathbf{x}, \mathbf{s}) = \left(\frac{\hat{\mathbf{s}}}{\|\hat{\mathbf{s}}\|} \right)^T \left(\frac{\hat{\mathbf{x}}}{\|\hat{\mathbf{x}}\|} \right) = \hat{\mathbf{s}}^T \hat{\mathbf{x}} \quad (5)$$

60 where $\hat{\mathbf{x}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{U}^T (\mathbf{x} - \mu_b)$ and $\hat{\mathbf{s}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{U}^T \mathbf{s}$, \mathbf{U} and \mathbf{D} are the eigenvectors and eigenvalues of the
 61 background covariance matrix, respectively.

As outlined by Zare et al. (2018b), the MI-ACE algorithm optimizes (3) using an alternating optimization strategy. After optimization, an estimate for a discriminative target signature (that is used to distinguish between two classes and perform pixel level classification using the D_{ACE} detector) is obtained. The MI-ACE code is available and published on our GitHub site (Zare et al., 2018a).

Proposed one-vs-one MI-ACE

The original MI-ACE algorithm was designed for target detection. Target detection can also be viewed as a two-class classification problem with one class being target and the other class being non-target or background (often with heavily imbalanced class sizes). In this work we extend MI-ACE using a one-vs-one scheme for application to multi-class classification problems. The basic idea for the proposed approach is to train a set of MI-ACE classifiers. Two MI-ACE classifiers are trained for every pair of two classes in the multi-class classification problem. Two classifiers are trained so that each class can be considered as the target class once in this pair. An MI-ACE classifier consists of a trained discriminative target signature (estimated using the MI-ACE approach outlined in the previous section), a background mean and covariance computed using the training samples from the non-target class, and a threshold value used to assign a target or non-target label to individual data points given their ACE detection confidence computed using the estimated target signature, background mean and background covariance values.

During testing, each trained MI-ACE classifier is applied to an input test point. Since each classifier yields a classification result, the final classification for a testing bag is obtained by aggregating all of the individual results. Specifically, a test bag is assigned the class label associated with the class that had the largest number of votes associated with each class. The votes are tallied by, first, averaging the confidence values estimated from each of the individual classifiers applied to each test point in the bag and, then, thresholding these average confidence values to obtain a binary target vs. non-target label the test bag. Then, the class label with the largest number of votes from the binary classification results is assigned as the final class label. Pseudocode for the proposed method is shown below. In the pseudocode, let \mathbf{X} and \mathbf{Y} be the set of all training and testing bags, respectively, with \mathbf{X}_L being the set of all bags assigned label L , \mathbf{Y}_m being the m^{th} testing bag, and $\mathbf{Y}_{m,n}$ being the n^{th} data point in the m^{th} testing bag. Let \mathbf{L} and \mathbf{R} be the corresponding bag level labels for the training and testing bags, respectively. C denotes the number of classes. M denotes the number of testing bags. The variables $s_{i,j}$ and $\tau_{i,j}$ represent the estimated target signature and classification threshold for target class i and background class j , respectively, and $z_{m,i,j}$ denotes the confidence value estimated by ACE detector given the estimated $s_{i,j}$ and $\tau_{i,j}$ values for the m^{th} test bag. The threshold value is set by determining the threshold that minimizes classification error on the training data. This approach does not have any parameters to tune as all parameters are estimated from the training data.

EXPERIMENTAL RESULTS

The proposed method was applied to and entered into the tree crown classification challenge organized and described by Marconi et al. (2018).

Data description

The training data released by Marconi et al. (2018) contains the hyperspectral signatures from 305 tree crowns collected over the Ordway-Swisher Biological Station (OSBS) by the National Ecological Observatory Network (NEON). The data from NEON included the following data products: 1) woody plant vegetation structure (NEON.DP1.10008); 2) spectrometer orthorectified surface directional reflectance - flightline (NEON.DP1.30008); 3) ecosystem structure (NEON.DP3.30015); and High-resolution orthorectified camera imagery (NEON.DP1.30010). Figure 1 shows a portion of the OSBS containing various tree species. However, for this tree crown classification challenge only individual spectral signatures for each tree crown are stored and provided. Thus, no spatial information is given nor can any image processing approach be applied. The signatures provided contain 426 spectral bands ranging from 383 nm to 2512 nm. Water absorption wavelengths, of which reflectance are set to be 1.5 as shown in Figure 2, correspond to 1345 nm to 1430 nm, 1800 nm to 1956 nm and 2482 nm to 2512 nm. Each training tree crown is paired with a genus class label and a species class label. The genus consisted of 5 classes which are *Acer* (AC), *Liquidambar* (LI), *Pinus* (PI), *Quercus* (QU) and OTHERS (OT). OTHERS represent the tree crowns that cannot be classified to any one of the four known genera. Each genus has a different number of associated species. AC and LI contains only one species, which are *Acer rubrum* (ACRU) and



Procedure 1 One-vs-one MI-ACE classification

1: Train two MI-ACE classifiers for each pair of class labels:

Input: $\mathbf{X}, \mathbf{Y}, \mathbf{L}$

2: **for** Every pair of classes $c_1 = 1 : C$ and $c_2 = 1 : C$ where $c_2 \neq c_1$ **do**

3: Train MI-ACE: $(\mathbf{s}_{c_1,c_2}, \tau_{c_1,c_2}, \mu_{c_2}, \Sigma_{c_2}) = \text{MI-ACE}(X_{L=c_1}, X_{L=c_2})$

4: Train MI-ACE: $(\mathbf{s}_{c_2,c_1}, \tau_{c_2,c_1}, \mu_{c_1}, \Sigma_{c_1}) = \text{MI-ACE}(X_{L=c_2}, X_{L=c_1})$

5: **end for**

6: Test using a one-vs-one voting scheme:

7: **for** Every test bag $m = 1 : M$ **do**

8: **for** Every pair of classes $c_1 = 1 : C$ and $c_2 = 1 : C$ where $c_2 \neq c_1$ **do**

9: **for** Every data point, $n = 1 : N_m$ in bag m **do**

10: Apply the c_1 vs. c_2 classifier: $z_{m,n,c_1,c_2} = \text{ACE}(\mathbf{Y}_{m,n}, \mathbf{s}_{c_1,c_2})$

11: **end for**

12: Average the confidence scores over all points in the bag: $z_{m,c_1,c_2} = \frac{1}{N} \sum_{n=1}^{N_m} z_{m,n,c_1,c_2}$

13: **if** $z_{m,c_1,c_2} > \tau_{c_1,c_2}$ **then**

14: R_{m,c_1,c_2} gets one vote for class c_1

15: **else**

16: R_{m,c_1,c_2} gets one vote for class c_2

17: **end if**

18: **end for**

19: R_m is assigned to the class with the largest number of votes.

20: **end for**

Output: \mathbf{R}

114 *Liquidambar styraciflua* (LIST), respectively. PI and QU contains more than 3 species individually, which
 115 are *Pinus elliottii* (PIEL), *Pinus palustris* (PIPA), *Pinus taeda* (PITA) and OTHERS (for PI), *Quercus*
 116 *geminata* (QUGE), *Quercus laevis* (QULA), *Quercus nigra* (QUNI) and OTHERS (for QU). The number
 117 of tree polygons for each species are shown in Table 1. In the current implementation of this work,
 118 OTHERS in both the genus and species level are not used for training as the proposed approach did not
 119 have a mechanism to identify points that did not belong to any of the labeled training classes.

Species	ACRU	LIST	PIEL	PIPA	PITA	QUGE	QULA	QUNI	OTHERS
Number	6	4	5	197	14	12	54	5	8

Table 1. The number of training tree crowns for each species

120 The testing data was also NEON tree crown hyperspectral data in the same format. There were 126
 121 testing tree crowns. The test labels were not provided by the competition organizers.

122 Data preprocessing and MI-ACE training

123 Prior to application of the MI-ACE algorithm, the water bands of the spectral signatures are removed.
 124 Then, since in our current implementation data points labeled as OTHERS genus or species are not
 125 addressed, the signatures that were labeled as OTHERS are removed from the training set.

126 After removal of the water bands and the OTHERS data points, the target signatures and classification
 127 threshold values are trained using the proposed one-vs-one MI-ACE approach. Training was conducted
 128 at two levels, the genus and the species levels. During the training phase, each training tree crown was
 129 considered a bag for MI-ACE, thus the training label (genus or species level) was the bag label. A
 130 one-vs-one MI-ACE was used in which a set of MI-ACE target signatures representing the difference
 131 between every two genera or species were estimated. For instance, a target signature was trained to
 132 distinguish between the genus PI and the genus QU where tree crown labeled as PI was labeled target
 133 (or '1') and QU was labeled as non-target (or '0'). For this competition only one MI-ACE classifier was
 134 trained for each pair of classes (as opposed to two for each pair) because results were similar between the
 135 two approaches. Similarly, a set of target signatures were estimated between every two species (if there
 136 were at least two species) that belonged to the same genus. For example, a target signature was estimated

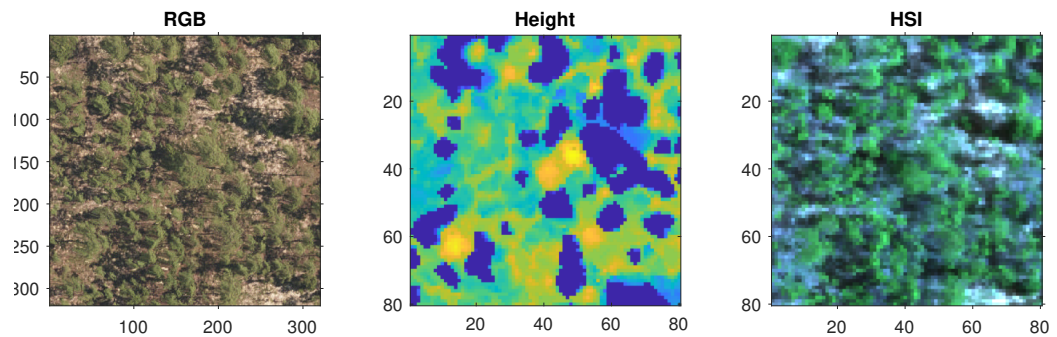


Figure 1. An example RGB, LiDAR, and (the RGB image generated from the corresponding) Hyperspectral image of a region in OSBS 

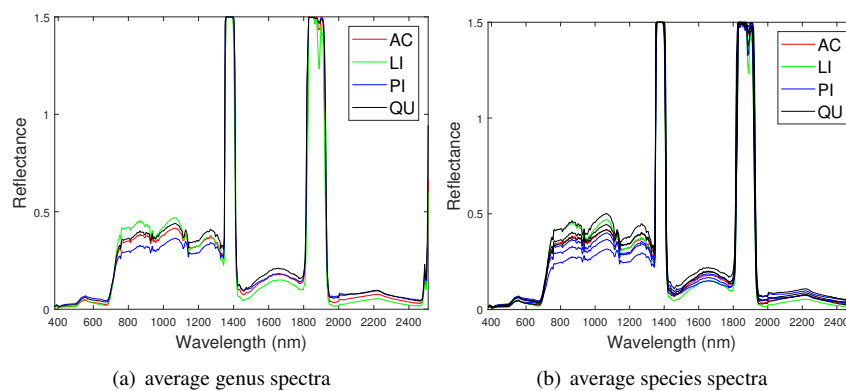



Figure 2. Average spectral signature of (a) all genera and (b) all species, colored by genera. 

137 using training data to distinguish between species PIEL and PITA where the tree crowns labeled as PIEL
 138 were labeled as target (or '1') and PITA was labeled as non-target (or '0').

139 **Testing using ACE detector and voting**

140 Testing was also conducted in two stages where a test tree crown was first classified at the genus level and,
 141 then, further classified at the species level. An ACE detector was used to estimate the confidence value
 142 indicating how similar a test signature is to a trained target signature. Classification of test tree crowns
 143 consisted of the following steps. First, the confidence values of each instance signature inside the a test tree
 144 crown were computed using each of the six genus-level trained MI-ACE classifiers. Second, the confidence
 145 value for each testing crown was estimated by taking the average value over all of the instance-level
 146 confidence values. These average confidence values were then thresholded using the trained threshold
 147 values to obtain a binary classification result. The classification thresholds were determined during
 148 training to minimize the number of misclassified tree crowns in the training data. The final classification
 149 of a tree crown is the class label with the highest number of corresponding binary classifications. After
 150 the genus level classification, a test tree crown can be classified at species level using the same approach
 151 among the species associated with the genus to which the tree crown assigned.

152 **Results**

153 **Genus level classification**

154 Classification result when testing on training samples are shown via confusion matrix in Table 2. The
 155 overall classification accuracy on the training dataset is 97.31%. The pixel confidence distributions
 156 and the aggregated crown confidence distributions for each classifier are shown in Figure 3 (a) and (b),
 157 respectively. In Figure 3, each row represents one of the six classifiers and each column denotes each
 158 ground truth genus type. The associated threshold value for each classifier is **plot** as a vertical blue line in

159 each subfigure. For instance, the top left subfigure in Figure 3 (a) shows the averaged ACE confidence
 160 value distribution of AC tree crowns detected using AC-vs-LI classifier and the same subfigure in Figure 3
 161 (b) shows the corresponding pixel confidence value distributions. A good result for the AC-vs-LI classifier
 162 will have all AC confidence values to the right of the threshold value and all LI confidence values (shown in
 163 the second plot in the first row) to the left of the threshold value. As can be seen, the AC-vs-LI classifier
 164 accurately distinguishes between these two classes on the aggregated crown-level scale. However, when
 165 considering the same plots in Figure 3 (b) for the pixel level confidences, we can see that there are many
 166 AC pixels to the left of the threshold causing significant overlap with the LI pixels confidences. This
 167 indicates that the aggregation procedure helps to improve results.

True \ Predict	AC	LI	PI	QU
AC	6	0	0	0
LI	0	4	0	0
PI	0	0	212	4
QU	0	0	4	67

Table 2. The classification confusion matrix on all training data (except for OHTERS) in genus level

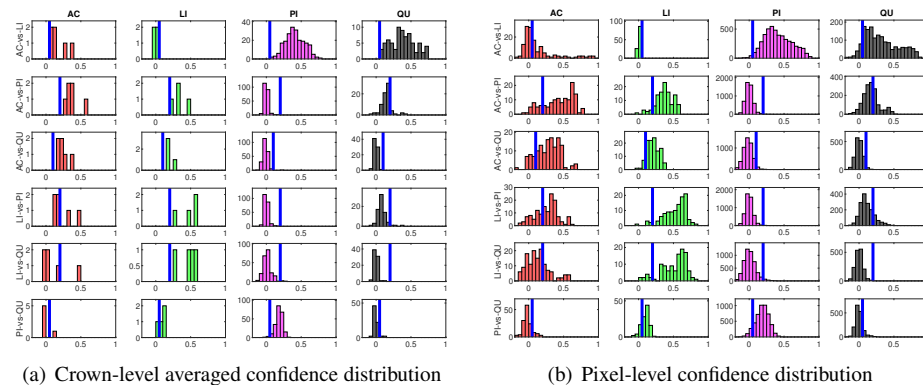


Figure 3. Confidence distributions of crown (left figure) and pixel (right figure) levels in training set. Rows from top to down are AC-vs-LI, AC-vs-PI, AC-vs-QU, LI-vs-PI, LI-vs-QU and PI-vs-QU classifiers, respectively. Columns from left to right are tree crowns genus types of AC, LI, PI, QU, respectively.

168 Since there were six classifiers trained and four classes in this data set, there are only a small set of
 169 possible voting cases. These cases are: a. (3,1,1,1) votes for each class; b. (3,2,1,0) votes for each class; c.
 170 (2,2,1,1) votes for each class and d. (2,2,2,0) votes for each class. For cases a. and b., there is a single
 171 class with the largest amount of votes, thus, labeling of the crown is straightforward. However, for voting
 172 cases c. and d., there are ties among 2 or 3 candidate classes. In our current implementation, we randomly
 173 assign the label of one of the tied classes. We found that cases c. and d. are rare in our training and testing
 174 results. In the testing on training data results, votes for all of the tree crowns fell into either case a. or b.
 175 When applying the trained approach to the testing dataset provided by the competition, there is only one
 176 tree crown in which there was a tie (and resulted in voting case is d). For this tree crown, we found that it
 177 has 2 votes for AC, 2 votes for LI and 2 votes for PI. **Our implementation in test randomly assigned** this
 178 tree crown to the AC class. Since the ground truth labels of testing data are not released by organizer, the
 179 true genus class of this testing tree crown is unknown. However, we found that in our classification results
 180 on testing data, there are three tree crowns that were predicted to be in class AC by our method. Two
 181 of these crowns were correctly classified into class AC and the other false positive tree crown actually
 182 belonged to the OTHER class (see Table 7). Thus, it is likely that this tree crown may be the tree crown
 183 with the tied result.



184 Cross validation studies on the training data were also conducted. There are a limited number of
 185 training tree crowns for the AC and LI classes (only 6 AC and 4 LI tree crowns). Due to this reason, cross
 186 validation experiments were not conducted for AC or LI. In the training phase, the PI training (pixel-level)
 187 samples and QU (pixel-level) training samples are considered target and background, respectively. The
 188 learned target signature is shown in Figure 4 (a), which characterizes the spectral difference between PI
 189 and QU shown in Figure 4 (b). In the testing phase, each pixel signature of PI and QU are compared
 190 with estimated target signature using the ACE detector resulting in a confidence value shown in Figure 5.
 191 Since most confidence values of PI pixels are larger than QU pixels, a threshold value (of 0.05) can be
 192 selected such that the misclassified error is minimized.

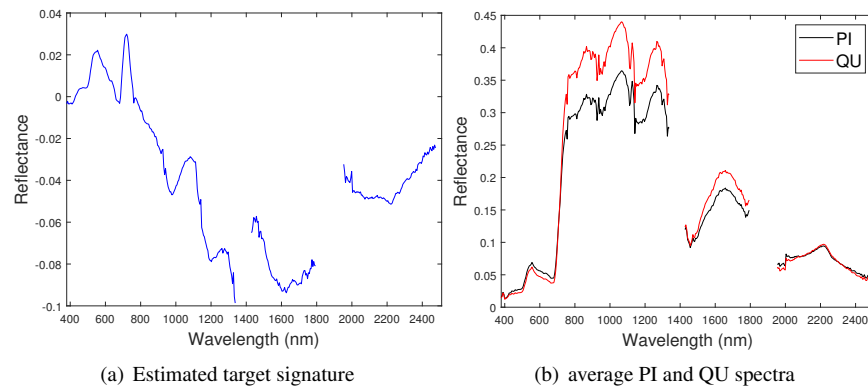



Figure 4. Comparison between estimated target signature and average class signatures. As can be seen, the target signature tends to be positive in the wavelengths where the target class has a larger response than the background class and negative where the target class has a smaller response. Thus, the target signature gives insight into discriminative features for the detection problem. 

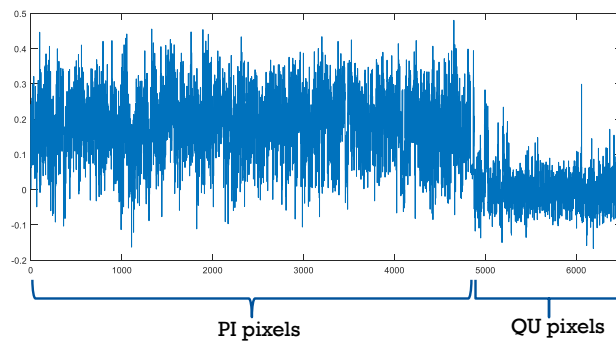




Figure 5. ACE detection statistic on PI & QU pixels 

193 Two-fold cross validation was applied to the PI and QU samples. The PI and QU classes were
 194 randomly split into two datasets (50% of PI and QU tree crowns are selected as d_1 and the rest as d_2)
 195 We train on d_1 and validate on d_2 , followed by training on d_2 and validating on d_1 . After training the
 196 PI-vs-QU target signature and corresponding threshold, the histogram of the average confidence values
 197 on the validation set is shown in Figure 6. In this figure, the PI and QU tree crowns are colored by their
 198 ground truth classes, i.e., red for PI and blue for QU. The threshold value estimated from training can be
 199 directly applied to the validation set for classification of the validation training crowns.

200 The cross validation experiment was repeated ten times and the mean confusion matrix is shown in
 201 Table 3. The average classification accuracy on the PI and QU given two-fold cross validation dataset was
 202 95.8%, which is similar to the test-on-train accuracy indicating robust results. 

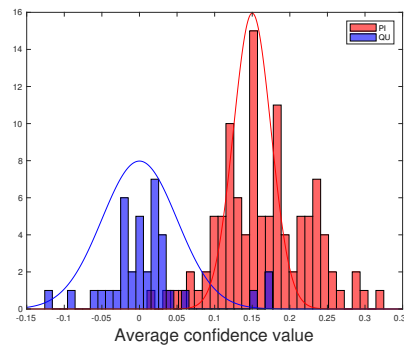


Figure 6. Histogram of average confidence values on validation set

		Predict	
		PI	QU
True	PI	105.8	2.2
	QU	3.8	31.2

Table 3. The mean classification confusion matrix on all PI and QU training data via cross validation

203 **Species level classification**

204 After the genus level classification, the tree crowns were further classified into species. If a tree is
 205 classified as AC or LI, it is classified also as ACRU or LIST automatically. If a tree is classified as PI
 206 or QU, the one-vs-one MI-ACE method is used to classify it into one of the corresponding species. The
 207 confusion matrices for species level classification (testing on training data) are shown in Table. 4. The
 208 classification (rank-1) accuracy is 95.62% on the training dataset in species level with a cross entropy
 209 value of 0.2649.

Predict True	ACRU	LIST	PIEL	PIPA	PITA	QUGE	QULA	QUNI
ACRU	6	0	0	0	0	0	0	0
LIST	0	4	0	0	0	0	0	0
PIEL	0	0	5	0	0	0	0	0
PIPA	0	0	3	188	2	2	2	0
PITA	0	0	0	0	14	0	0	0
QUGE	0	0	0	2	0	10	0	0
QULA	0	0	0	2	0	0	53	0
QUNI	0	0	1	0	0	0	0	4

Table 4. The classification confusion matrix on all training data (except for OHTERS) in species level

210 The confusion matrices for species level classification for testing data are shown in Figure 7 as
 211 provided by the competition organizers. The classification (rank-1) accuracy is 86.40% and cross entropy
 212 is 0.9395 on the testing dataset. However, this accuracy includes data points labeled as OTHERS in the
 213 testing dataset which, using our approach, were all misclassified to one of the four genus types since we
 214 did not implement a mechanism to distinguish outliers in this approach. If the OTHERS tree crowns (3
 215 tree crowns) are excluded, the classification accuracy would come to 88.52% and cross entropy would be
 216 0.7918 on the testing dataset.

217 The species level classification results are further evaluated using several metrics on the testing data
 218 by the organizer, including per-class accuracy, specificity, precision, recall and F1 score. For comparison,
 219 we also evaluated the classification performance using the same metrics on the training data. The accuracy
 220 and specificity score, F1 score, precision, recall for both training and testing dataset are shown in Figure
 221 8, 9, 10 and 11, respectively. As can be seen, accuracy and specificity results between training and testing

Species ID	ACRU	LIST	OTHER	PIEL	PIPA	PITA	QUGE	QULA	QUNI
ACRU	2.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
LIST	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
OTHER	1.00	1.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
PIEL	0.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
PIPA	0.00	0.00	0.00	1.00	81.00	0.00	1.00	0.00	0.00
PITA	0.00	0.00	0.00	1.00	2.00	2.00	0.00	1.00	0.00
QUGE	0.00	1.00	0.00	0.00	0.00	0.00	3.00	0.00	0.00
QULA	0.00	0.00	0.00	0.00	1.00	0.00	4.00	17.00	1.00
QUNI	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

Figure 7. The classification confusion matrix on all testing data (except for OHTERS) in species level (provided by competition)

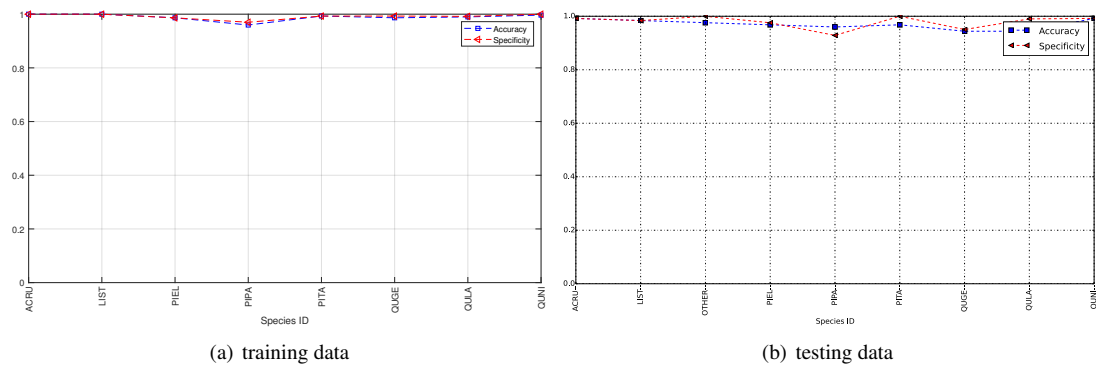


Figure 8. Accuracy and Specificity Scores (Per-Class) for training data (a) and testing data (b - provided by competition)

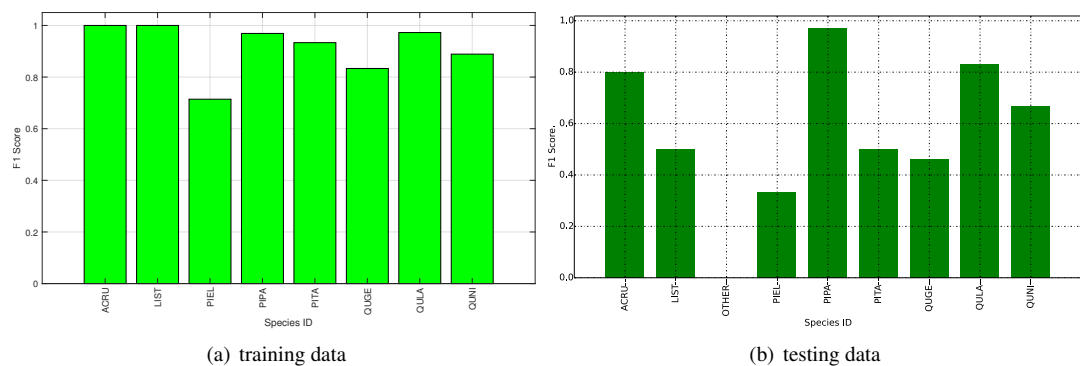


Figure 9. F1 Scores (Per-Class) for training data (a) and testing data (b - provided by competition)

222 data are similar whereas the F1, precision and recall curves highlight that challenging classes in the testing
 223 data were PIEL, PITA and QUGE.

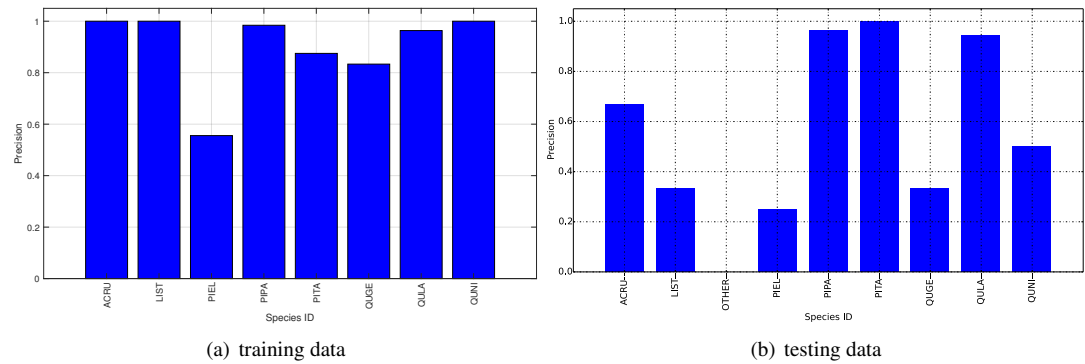


Figure 10. Precision (Per-Class) for training data (a) and testing data (b - provided by competition)

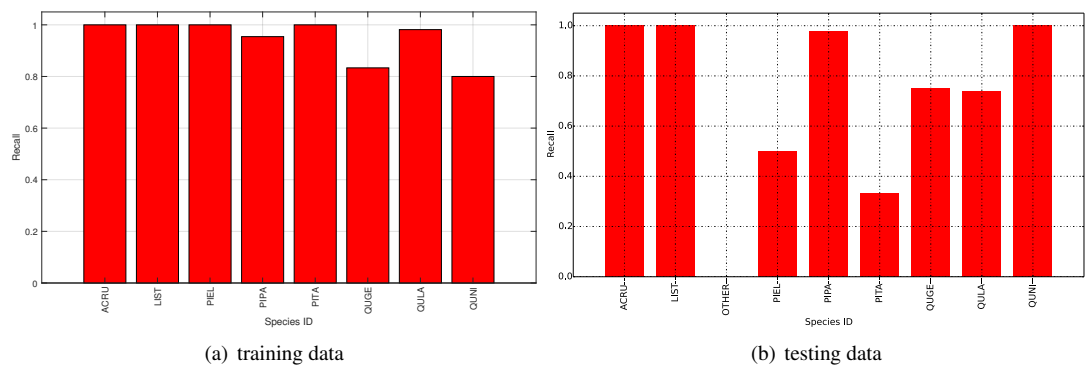



Figure 11. Recall (Per-Class) for training data (a) and testing data (b - provided by competition)

SUMMARY AND FUTURE WORK

224

225 A one-vs-one version of MI-ACE is proposed in the work to address the hyperspectral tree crown
 226 classification problem. The proposed method achieved a 86.4% overall classification accuracy on a
 227 blind testing dataset. Certainly, there are many **improvements can be investigated** in the future such as
 228 mechanisms to identify outliers and label them as members of the OTHERS class and estimate a likelihood
 229 of belonging to each class (as opposed to binary classification labels).

In the current implementation, only crisp binary classification results are estimated. However, competition organizers evaluated results using the cross entropy evaluation metric assuming probabilities of belonging to each class are estimated, 

$$\text{cost} = - \frac{\sum_{n,k} \ln p_{n,k} \delta(g_n, k)}{N} \quad (6)$$

230 where g_n is the ground truth class of crown n , $p_{n,k}$ is the probability assigned that crown n belongs to
 231 class k . Class probabilities given the one-vs-one scheme can be estimated in the future using approaches
 232 such as those proposed by Wu et al. (2004). Furthermore, even if individual probabilities per data are
 233 not computed, an overall uncertainty value can be estimated from the training data. In other words, as
 234 opposed to assigning 0-1 probabilities for the crisp class labels. In our implementation data points were
 235 assigned to the estimated class label with probability 1 and all others with probability 0. Instead, we
 236 could pre-compute an optimal epsilon value, ϵ^* , to add to the '0' probabilities and subtract from the '1'
 237 probabilities to ensure values sum to one across classes to minimize cross entropy on the training data.
 238 For instance, we found that when $\epsilon^* = 0.017$, the cross entropy for our results comes to 0.68, which is
 239 a smaller (i.e., better) than the cross entropy of 0.94 obtained using crisp labels and calculated by the
 240 competition organizers. The relationship between the cross entropy and epsilon value for the training data

241 provided shown in Figure 12.

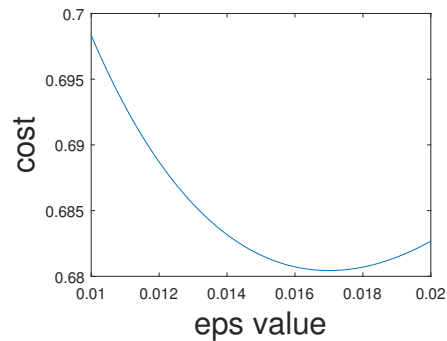


Figure 12. Cross entropy vs optimum epsilon value

242 In addition, in the current proposed framework, each one-vs-one classifier is equally weighted in
 243 final voting. However, the classification accuracies and the applicability of each classifier varies. For
 244 instance, if a tree crown is in class PI, the PI-vs-QU classifier should be more heavily weighted than
 245 the AC-vs-LI classifier. Investigation into whether this could be determined by considering the average
 246 confidence values estimated from the individual ACE detectors is needed. Furthermore, since some of
 247 the classes are more spectrally distinct, some one-vs-one classifiers have better prediction performances.
 248 One possible solution is to weight the classifiers based on the difference between the average confidence
 249 values of target and background classes for training data. Another possible solution is to weight based on
 250 the difference between the confidence value of testing point and threshold value of the classifier. In some
 251 scenarios, these two solutions might be equivalent. Finally, data fusion is also a promising approach for
 252 boosting the classification performance. For example, height information from Lidar data could be also
 253 incorporated into the training phase since different species generally have different average heights. In the
 254 current implementation, only hyperspectral information was leveraged.

255 ACKNOWLEDGEMENTS

256 This material is based upon work supported by the National Science Foundation under Grant IIS-1723891-
 257 CAREER: Supervised Learning for Incomplete and Uncertain Data. The National Ecological Observatory
 258 Network is a program sponsored by the National Science Foundation and operated under cooperative
 259 agreement by Battelle Memorial Institute. This material is based in part upon work supported by the
 260 National Science Foundation through the NEON Program. The ECODSE competition was supported, in
 261 part, by a research grant from NIST IAD Data Science Research Program to D.Z. Wang, E.P. White, and
 262 S. Bohlman, by the Gordon and Betty Moore Foundation's Data-Driven Discovery Initiative through grant
 263 GBMF4563 to E.P. White, and by an NSF Dimension of Biodiversity program grant (DEB-1442280) to S.
 264 Bohlman.

265 REFERENCES

- 266 Anderson, C. B. (2018). The CCB-ID approach to tree species mapping with airborne imaging spec-
 267 troscopy. *PeerJ Preprints*, 6:e26972v1. <https://doi.org/10.7287/peerj.preprints.26972v1>.
- 269 Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM.
- 271 Cochrane, M. (2000). Using vegetation reflectance variability for species level classification of hyperspec-
 272 tral data. *International Journal of Remote Sensing*, 21(10):2075–2087.
- 273 Dalponte, M., Frizzera, L., and Gianelle, D. (2018). NEON NIST data science evaluation challenge:
 274 methods and results of team fem. *PeerJ Preprints*, 6:e26973v1. <https://doi.org/10.7287/peerj.preprints.26973v1>.
- 276 Kraut, S., Scharf, L., and McWhorter, T. (2001). Adaptive subspace detectors. *IEEE Transactions on signal processing*, 49(1):1–16.

- 278 Marconi, S., Graves, S. J., Gong, D., Nia, M. S., Le Bras, M., Dorr, B. J., Fontana, P., Gearhart, J.,
279 Greenberg, C., Harris, D. J., et al. (2018). A data science challenge for converting airborne remote
280 sensing data into ecological information. *PeerJ Preprints*, 6:e26966v1. [https://doi.org/10.](https://doi.org/10.7287/peerj.preprints.26966v1)
281 [7287/peerj.preprints.26966v1](https://doi.org/10.7287/peerj.preprints.26966v1).
- 282 Maron, O. and Lozano-Pérez, T. (1998). A framework for multiple-instance learning. In *Advances in*
283 *neural information processing systems*, pages 570–576.
- 284 Sumsion, G. R., Bradshaw, M. S., Hill, K. T., Pinto, L. D., and Piccolo, S. R. (2018). Remote sensing tree
285 classification with a multilayer perceptron. *PeerJ Preprints*, 6:e26971v1. [https://doi.org/10.](https://doi.org/10.7287/peerj.preprints.26971v1)
286 [7287/peerj.preprints.26971v1](https://doi.org/10.7287/peerj.preprints.26971v1).
- 287 Wu, T.-F., Lin, C.-J., and Weng, R. C. (2004). Probability estimates for multi-class classification by
288 pairwise coupling. *Journal of Machine Learning Research*, 5(Aug):975–1005.
- 289 Zare, A., Glenn, T., and Gader, P. (2018a). Gatorsense/hsi_toolkit (version v0.1). [https://zenodo.](https://zenodo.org/record/1260272)
290 [org/record/1260272](https://zenodo.org/record/1260272).
- 291 Zare, A., Jiao, C., and Glenn, T. (2018b). Discriminative multiple instance hyperspectral target characteri-
292 zation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. In Press.