IET Computer Vision

Research Article

IET Journals
The Institution of Engineering and Technology

Fully-connected semantic segmentation of hyperspectral and LiDAR data

ISSN 1751-9632
Received on 2nd March 2018
Revised 5th September 2018
Accepted on 1st October 2018
doi: 10.1049/iet-cvi.2018.5067
www.ietdl.org

Hakan Aytaylan[1], Seniha Esen Yuksel[1] ✉

[1]*Department of Electrical and Electronics Engineering, Hacettepe University, Ankara, Turkey*

✉ *E-mail: eyuksel@ee.hacettepe.edu.tr*

Abstract: Semantic segmentation is an emerging field in the computer vision community where one can segment and label an object all at once, by considering the effects of the neighbouring pixels. In this study, the authors propose a new semantic segmentation model that fuses hyperspectral images with light detection and ranging (LiDAR) data in the three-dimensional space defined by Universal Transverse Mercator (UTM) coordinates and solves the task using a fully-connected conditional random field (CRF). First, the authors' pairwise energy in the CRF model takes into account the UTM coordinates of the data; and performs fusion in the real world coordinates. Second, as opposed to the commonly used Markov random fields (MRFs) which consider only the nearby pixels; the fully-connected CRF considers all the pixels in an image to be connected. In doing so, they show that these long-term interactions significantly enhance the results when compared to traditional MRF models. Third, they propose an adaptive scaling scheme to decide the weights of LiDAR and hyperspectral sensors in shadowy or sunny regions. Experimental results on the Houston dataset indicate the effectiveness of their method in comparison to the several MRF based approaches as well as other competing methods.

1 Introduction

By capturing information from hundreds of frequencies of light, hyperspectral imaging provides valuable information on the material of the object. The use of this information on remote sensing has been a highlighted topic of research in recent years. Even though the information on hundreds of spectral bands of a surface can be stored within one pixel, factors such as atmospheric effects and illumination may cause misclassification and decrease the overall performance in urban areas [1, 2]. To overcome this problem, hyperspectral data is commonly fused with light detection and ranging (LiDAR) data, which provides elevation information, and is less sensitive to atmospheric conditions [3].

The goal of this study is the semantic segmentation of hyperspectral and LiDAR datasets. Semantic segmentation refers to the joint segmentation and classification of an image, and it is an active research topic in the computer vision field [4–7]. Semantic segmentation aligns with the spatial-spectral segmentation studies in the hyperspectral community, which also jointly segment and classify the data while taking into account the neighbouring information. While the semantic segmentation studies from the computer vision field have been making a significant use of conditional random field (CRF)/MRF (Markov random field) models [6, 8–15]; the spatial-spectral segmentation studies in the hyperspectral imaging field have been predominantly using the MRF models [16–20] and not the CRF. More recently, MRF models were used for the fusion of hyperspectral and LiDAR data in [21]. However, to the best of our knowledge, there are no studies that fuse hyperspectral and LiDAR data under a fully-connected CRF model.

CRF and MRF are models that provide a graphical model for finding the maximum a posteriori (MAP) solutions. One major difference is that the MRFs are generative models, whereas the CRFs are discriminative [22]. On the other hand, both models typically consider only the statistical relationship between the 4 or 8 neighbouring pixels to make the inference tractable; which is solved using graph-cut or message passing techniques. Although considering only the close-neighbours permit efficient inference, the traditional MRFs and CRFs have restricted expressive power as they are unable to enforce the high-level structural dependencies between pixels [23].

In this study, we propose an alternative approach to the semantic segmentation of hyperspectral images fused with LiDAR data, in which fully-connected CRFs are used. As opposed to the traditional close neighbourhood CRFs, fully-connected CRFs [4] consider all the pixels in an image to be connected, and can, therefore, model the long-range dependencies. To the best of our knowledge, we are the first group to investigate the potential of fully-connected CRFs on the fusion of LiDAR and hyperspectral data.

Our proposed model is depicted in Fig. 1. On the left branch, spectral data is preprocessed using a 3D Gaussian Filter and insignificant features are reduced using the HySime [24] algorithm. Then, probability maps for spectral data are computed using probabilistic classifiers such as the probabilistic support vector machine (pSVM) or subspace Multinomial Logistic Regression (MLRsub) [5]. On the right branch, LiDAR elevation data is concatenated with the first return intensity, which is filtered with Gaussian and median filters. Then, extended morphological profile (EMAP) [25] features are extracted from the LiDAR data and these features are again fed to pSVM or MLRSub classifiers. These classifiers compute the pixel-wise probabilities which form the unary energies in our proposed CRF model. However, they lack a spatial smoothness term that would make each class coherent in itself. For that purpose, we propose a new pairwise energy for the fully-connected CRF, which uses the three dimensional physical distances of pixels. The first two dimensions (2Ds) of this distance are obtained from the Universal Transverse Mercator (UTM) coordinates of the hyperspectral data, and the 3D is the elevation information from the LiDAR data. Then, using our newly proposed pairwise energy for fully-connected CRF, the graphical model is solved using message-passing for inference. In addition, we propose a simple experimental solution to determine the weighted contribution of LiDAR to hyperspectral data in shadow and sunny regions.

The remaining of this paper is organised as follows. Section 2 explains some of the related studies. Section 3 formulates classification as an energy minimisation problem that is written as the sum of a unary energy term and a pairwise energy term. In Section 3.1, the unary energy term is defined and two of the unary classifiers that are compared in this paper, the pSVM and the MLRsub are briefly explained. Also, an algorithm is proposed to adjust the weights for the shadow and non-shadow regions when

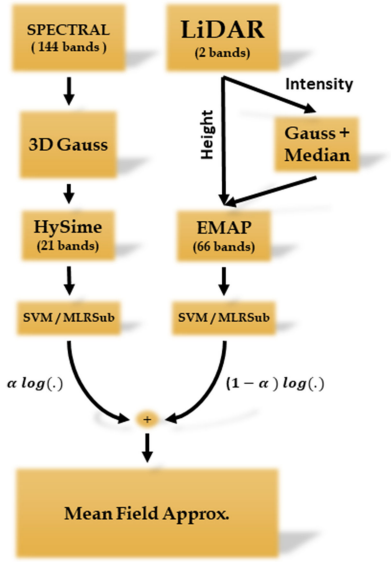

Fig. 1 *Flowchart of our proposed method*

combining the unary terms. Then, in Section 3.2, the pairwise energy is defined for the CRF framework and solved with mean field approximation. These approaches are tested on the Houston dataset as given in Section 4. First, data is passed through a preprocessing stage as explained in Section 4.1. Afterwards, learning results are given in Section 4.4; and the paper is concluded in Section 5. For the interested readers, the pSVM and the MLRsub are explained in Appendices 8.1 and 8.2.

2 Literature review

There are several studies that use random fields on hyperspectral datasets alone (where no fusion is considered). In [19], pSVM is used for spectral classification, and a spatial energy term is used for spatial information, which is solved via graph-cuts. In [20], support vector machine (SVM) is again used to extract the class combination maps for a hyperspectral image. Later, subspace projection-based MLRsub [5] is used in order to obtain global and local posterior probabilities. Classifier results are fused together within a MRF model. Since in both cases, pSVM and MLRSub are proven to be good candidates for the unary term of the energy function, we also use these two classifiers for the spectral term in our experiments.

MLR classifiers model the probability distribution of a classification problem as an MLR function [26]. MLRSub, as introduced in [5] assumes a linear mixture model of the hyperspectral image pixel. Using the eigenvalues of the training set, it extracts the subspace, i.e. the most important components of the pixel, and tries to minimise the effect of unrelated information. Then, logistic regression parameters are learned from the subspace of the pixels, and classification is performed. Several other studies including [3, 20] also mark that MLRSub classifier is good at dealing with pixels that have mixed data, due to its basic assumption that all pixels are noisy mixtures of data.

Another highly popular concept due to its success is the use of Morphological Attribute Profiles (MAPs) [3, 21, 27]. Introduced in [25], MAPs use the openings and closings of the image, exploiting the spatial attributes of the objects in an image such as area, standard deviation or moment of inertia. Similar regions with given parameters are filtered and their geometry is preserved. These attributes introduce new features to the observed data.

MAPs have been found useful also in the fusion of LiDAR and hyperspectral data as studied in [3, 27, 28]. Specifically, in [3], the authors use MAPs of LiDAR elevation and hyperspectral data and classify the data using MLRsub classifier and MRFs. The method in [27] extracts the clouded region using Attribute Profiles and applies a separate classification within this region. It assumes the area of the clouded region to be much larger than the objects on the image. Also, the authors [21, 28] propose a pairwise energy term that penalises the pixel in focus if any one of the spectral, spatial or morphological features is too different than the other two.

Different from these related approaches, instead of performing fusion during feature extraction, we combine the results of energy descriptors and jointly solve the labelling problem. In our proposed method, processing each sensor data with a separate unary classifier helps us overcome the Hughes Phenomenon [29], i.e. the curse of dimensionality issue which is frequently encountered in fusion problems. With our pairwise energy term, we guide the spectral classification results to be consistent with the spatial properties including elevation. Then, we utilise the fully-connected CRF method for the first time and make use of the long-range dependencies of the pixels. In doing so, we integrate the fusion task seamlessly into the energy model for semantic segmentation.

3 Proposed method: energy minimisation with fully-connected CRFs

In MRFs and CRFs, an image is represented as a graph $G = (V, E)$ where the vertices V correspond to the pixels, which are connected with edges E. In a segmentation problem, this graph connectivity is explained in terms of a probabilistic conditional dependency, where the labels associated with the pixels are considered as the hidden random variables. Then, a joint probabilistic model is defined over the pixel values and the hidden variable, which is used to understand the statistical dependencies between the hidden variables by putting them into groups. These groups are often pairs shown as edges in a graph [30]. When hidden variables are associated with the nodes in this way and connected in a graph structure, neighbouring sites can be adjusted to have the same label. If all the pixels in an image are linked together explicitly, one obtains a densely connected graph. Also called as fully-connected graphs, densely connected graphs are computationally expensive to solve. As a solution, Markov models explicitly represent only the association between neighbouring pixels and significantly decrease the computational cost. The fully-connected CRFs that is being used in this paper, however, takes into account all the long-range interactions and also provides a computationally feasible solution.

Recent years have seen the emergence of CRFs [13, 22], which are discriminative variants of MRFs and have the ability to model complex spatial dependencies. With this ability, they have shown quite a bit of success in the semantic segmentation of natural images. Motivated by these successes, we developed a novel segmentation approach for LiDAR and hyperspectral datasets via the use of fully-connected CRFs. In the rest of this section, we define the energy function for a fully-connected CRF for a semantic segmentation problem and investigate the unary and pairwise energies.

Semantic segmentation task is defined as assigning each pixel in an image to a class from a previously determined class set $C = \{C_1, C_2, C_3, \ldots, C_n\}$. In CRFs, the MAP labelling is commonly expressed as a Gibbs distribution. Differently put, the energy function is expressed in terms of the posterior as

$$E(y; \boldsymbol{x}) = -\ln(P(Y = y_i | X = \boldsymbol{x_i})) - \ln(Z) \tag{1}$$

where $\boldsymbol{x}$ is the observed image with M pixels such that $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M]$, and $\boldsymbol{x_i}$ is a k -dimensional vector such as hyperspectral or LiDAR, $\boldsymbol{y} = [y_1, y_2, \ldots, y_i, \ldots, y_M]$ is the assigned class and Z is a normalisation constant.

Therefore, the MAP inference of y can be done by maximising the posterior, or equivalently by minimising the energy

$$y^* = \underset{y \in C}{\operatorname{argmax}} P(Y = y_i \,|\, \boldsymbol{x_i}) = \underset{y \in C}{\operatorname{argmax}} E(y_i; \boldsymbol{x_i}) \tag{2}$$

The energy functions for many commonly used Markov models can be written as the sum of a unary term and a pairwise term [23]

$$E(y; \boldsymbol{x}) = E_{\text{unary}} + E_{\text{pairwise}} \tag{3}$$

Input: μ_t: Average reflectance of every pixel in an HSI
μ_p: Reflectance of one pixel
p: A reflectance vector of one pixel
for For every pixel p in image **do**
$\mu_p = \|p\|$
if ($\mu_p < \mu_t/4$) **then**
$\alpha_i = 0$ (Use LiDAR)
else
$\alpha_i = 0.5$ (Use both)
end if
end for
return α

Fig. 2 *Algorithm 1: Algorithm for adaptive selection*

For MRFs, the energy generally consists of potentials of degree one and two, i.e.

$$E(y;\boldsymbol{x}) = \sum_{i \in V} \psi_i(y_i;\boldsymbol{x}) + \sum_{(i,j) \in N} \psi_{ij}(y_i, y_j) \tag{4}$$

where N represents a neighbourhood, ψ_i is the clique potential and ψ_{ij} is the pairwise potential.

It is important to note that the pairwise potential ψ_{ij} does not depend on the image data. If we condition the pairwise potentials on the data, then we obtain the CRF model, which is defined as

$$E(y;\boldsymbol{x}) = \sum_{i \in V} \psi_i(y_i;\boldsymbol{x}) + \sum_{(i,j) \in N} \psi_{ij}(y_i, y_j;\boldsymbol{x}) \tag{5}$$

The unary term of the energy function is the probabilistic classification of every pixel to a certain class. The pairwise term is concerned about the spatial relation of the pixel with its neighbouring pixels. In Section 3.1, the classifiers that are used to compute the unary energy function, and our proposed energy functions for the pairwise term are discussed in further detail.

3.1 Unary energy

For the unargy energy term, we consider two separate unary probabilistic distributions, namely $P(C_i|\boldsymbol{x}_i)$ and $P(C_i|\boldsymbol{h}_i)$ for hyperspectral and LiDAR data. Here $\boldsymbol{x}_i$ are the hyperspectral data; and $\boldsymbol{h}_i$ are the intensity and elevation information, extracted from LiDAR data, which is expanded using Morphological Attributes. Therefore, our proposed unary probability distribution energy is as follows:

$$E_{\text{unary}} = -k[\alpha \ln P(C_i|\boldsymbol{x}_i) + (1-\alpha)\ln P(C_i|\boldsymbol{h}_i)] \tag{6}$$

where k is a constant for emphasising the weight of the unary term with respect to the pairwise term; and α is a constant that adjusts the weights between the two probability distributions. Here, there are two main benefits of using two separate classifiers; first, the curse of dimensionality is mitigated by separating the datasets, thus reducing the number of dimensions. Second, the adverse effects of both datasets on the overall classification can be reduced by adjusting the value of alpha.

Owing to their proven track record in [5, 19, 20], we use the pSVM and MLRSub classifiers to compute the posterior probabilities in (6). The pSVM [31] is a method that maps the traditional SVM outputs to posterior probabilities. MLRSub [5] on the other hand is an MLR method that works on the basic assumption that a pixel in an image contains a mixture of materials. The details for pSVM and MLRSub are given in Appendix 8.1 and 8.2. In the rest of this paper, we refer to the unary energy term as dSVM or dMLRsub depending on which function we use as the classifier. In our experiments, we found $k = 2.5$ to be a good value for preserving the edges.

In (6), the value of α is critical when one sensor is more reliable than the other. For instance, the area under the cloud gives a low reflectance, which makes the LiDAR more accountable and discriminating. To handle such cases, we propose an adaptive selection for α, in which, LiDAR is the only source when the reflectance of a pixel is lower than one-fourth of the average value of all the pixels in an HSI. This adaptive selection strategy is summarised in Algorithm 1 (see Fig. 2). This strategy significantly simplifies the search for α, which could be overwhelming were it done iteratively.

3.2 Pairwise energy with the fully-connected CRF

As opposed to the MRFs that are generative models, CRFs are discriminative methods. In the generative models, a joint probability over observations and labels are defined and enumerated for all possible observations, which becomes intractable when long-range dependencies of the observations are considered. Therefore, the CRFs provide an alternative by looking at the conditional models instead and model the probabilities of possible label sequences given an observation sequence [22]. Although both MRFs and CRFs are written as the sum of a unary and a pairwise energy function, one major difference is that the CRFs learn the parameters of the pairwise energy, giving them the discriminating power.

CRFs have the advantage of incorporating smoothness terms that maximise the label agreement between similar pixels, and also modelling the contextual relationships between object classes. However, pairwise potentials of the CRFs are typically defined over neighbouring pixels and patches; which may not adequately model long-range interactions within the image and may over-smooth the object boundaries. To model the long-range interactions, the fully-connected CRF establishes pairwise potentials on all pairs of the pixels in the image. The computational burden to do inference on the fully-connected CRF has been resolved when Krähenbühl and Koltun [4] developed a highly efficient inference algorithm based on the mean field approximation to the CRF distribution.

In the fully-connected CRF model, the energy is also composed of a unary term and a pairwise term as was given in (5). However, the pairwise term computes the relation between all the pixels in the image, as opposed to taking just a neighbourhood. The pairwise potentials are defined over all the pairwise cliques in the following form:

$$E_{\text{pairwise}} = \sum_{i<j} \mu(C_i, C_j) k(\boldsymbol{x_i}, \boldsymbol{x_j}) \tag{7}$$

where μ is a label compatibility function which takes into account the interaction between the labels, and $k(x_i, x_j)$ is a Gaussian kernel that is the weighted addition of an appearance term and a smoothness term

$$k(\boldsymbol{x_i}, \boldsymbol{x_j}) = w^{(1)} \underbrace{\exp(-B)}_{\text{appearance}} + w^{(2)} \underbrace{\exp(-A)}_{\text{smoothness}} \tag{8}$$

where

$$A = \left(\frac{|p_{x_i} - p_{x_j}|^2}{2\theta_\alpha^2} + \frac{|p_{y_i} - p_{y_j}|^2}{2\theta_\alpha^2} + \frac{|p_{h_i} - p_{h_j}|^2}{2\theta_\alpha^2} + \frac{|\boldsymbol{x}_i - \boldsymbol{x}_j|^2}{2\theta_\beta^2} \right) \tag{9}$$

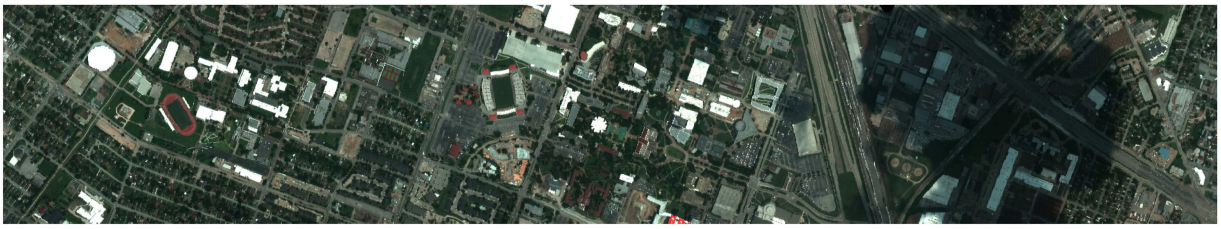

Fig. 3 *Houston data shown as a false RGB image. The shadow on the right-hand is a cloud that covers a large area of the ground*

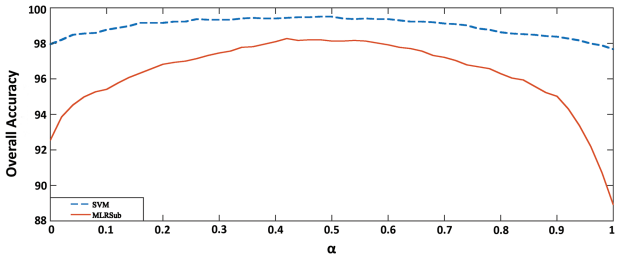

Fig. 4 *Impact of α. The best value of alpha for SVM is* $\alpha = 0.5$ *and for MRLSub* $\alpha = 0.42$

and

$$B = \left(\frac{\left| p_{x_i} - p_{x_j} \right|^2}{2\theta_\gamma^2} + \frac{\left| p_{y_i} - p_{y_j} \right|^2}{2\theta_\gamma^2} + \frac{\left| p_{h_i} - p_{h_j} \right|^2}{2\theta_\gamma^2} \right) \tag{10}$$

Here w are the weights and p_i are the pixel locations in 3D space using UTM coordinates. In doing so, we consider both the hyperspectral sensor resolution in the x–y direction, the height of each pixel as measured by LiDAR and the spectral differences of two pixels as measured by the spectral sensor. As a result, the energy is lower for pixels that are distant in any dimension.

In (8), the appearance term assumes that nearby pixels with similar features are likely to be in the same class. The smoothness term removes the small isolated parts within the class regions. Also, the label compatibility function $\boldsymbol{m}$ is learned during training.

Once the Gibbs energy E of a labelling is determined with the unary and pairwise terms, the CRF can be characterised by a Gibbs distribution P formed from these energies as it was given in (1). Since computing, the distribution P for all pixels in an image would be very time consuming, Krähenbühl and Koltun use mean field approximation which computes an approximate probability distribution Q by minimising the KL-divergence between P and Q. The details of the mean field approximation on the fully-connected CRF can be found in [4, 32].

4 Experimental analysis

In this study, the University of Houston dataset provided by the 2013 GRSS Data Fusion Contest [33] has been used. The dataset has both LiDAR and hyperspectral data, contains 15 classes, and around 200 training and 1000 test pixels for each class. A false RGB image of the dataset is shown Fig. 3. Note that on this image, there is a cloud that covers most of the commercial, railway and highway class test pixels. Each sensor's data is first separately preprocessed and its features are computed. Then, each pixel is classified with the unary classifiers and the results are combined using the fully-connected CRF model, which introduces a smoothness over the unary classifiers as explained in the following sections.

4.1 HSI preprocessing and feature extraction

The hyperspectral image has 144 spectral bands. The data is first filtered with a 3D Gaussian filter with $\sigma = 0.1$, each feature is scaled between [0,1] and then processed with HySime [24] in order to reduce the number of spectral bands. HySime performs dimension reduction without any supervision. The resulting data usually has 18–22 spectral bands.

4.2 LiDAR preprocessing and feature extraction

The LiDAR dataset includes both the intensity and elevation information. The first, last and average return intensities are provided, however only the first return intensity is considered in this study. Intensity is the first median filtered and then Gaussian filtered in order to decrease the effect of the saturated and noisy areas. The elevation and intensity images are used to extract the EMAP of the LiDAR data. These EMAPs are obtained by combining area and standard deviation attribute profiles of the image. For area attributes, parameters were selected to be between 100 and 500 with increments of 50. For standard deviation attribute, the values selected to be between 2.5 and 20 with increments of 2.5. This resulted in a 66-dimensional processed-LiDAR data, ready to be classified.

4.3 Impact of the parameters

The preprocessed HSI and LiDAR data are used in training pSVM and MLRSub classifiers. Then, the weighted results of the classifiers are computed based on the α parameter. One way to learn α is through an exhaustive search on a validation set, as shown in Fig. 4. It can be observed that the value of α has a dramatic effect on the overall accuracy. A second way to select α, which we proposed in this study in Algorithm 1 (Fig. 2), is to make it adaptive based on shadow information. With our proposed method, not only the classifier performance is increased in shadowy areas, but also the time consuming iterative search for α is eliminated.

For the other parameters in (8), we used $\omega^{(1)} = \omega^{(2)} = 4$, $\theta_\alpha = \theta_\beta = \theta_\gamma = 10$; and chose the minimum pairwise cost to be 0. Higher Gaussian and bilateral weights lead to too much smoothness, whereas a higher standard deviation parameter would have the opposite effect on the resulting image.

4.4 Results and discussion

In this section, we first compare the alternative classifiers in Table 1 and also evaluate their use as our unary term. In the first four columns, we show the classification results of pSVM and MLRSub on HSI-only and LiDAR-only data. Here, the naming convention is such that the classifier name is followed by the dataset. For example, pSVM-Spectral refers to training a pSVM classifier on the hyperspectral data; and MLRSub-LiDAR refers to training an MLRSub classifier on the LiDAR data. The numbers within the columns represent the accuracy percentage of the specified class. From these results, it can be seen that pSVM is a better classifier than MLRsub both in the spectral domain and the LiDAR domain. Then we use (6) and compute the classification results of our unary term as given in the fifth and sixth columns,

Table 1 Classification accuracies and kappa values for classical methods versus our methods (in %)

	HSI only		LiDAR only		Our proposed unary term		Our fully-connected CRF results		
	pSVM-spectral	MLRsb-spectral	pSVM-LiDAR	MLRSb-LiDAR	dSVM	dMLRSb	dSVM-CRF	dSVM-CRF-α	dMLRSb-CRF
healthy grass	81.671	80.342	50.427	72.270	80.057	81.766	80.057	83.096	83.096
stressed grass	82.707	80.357	67.011	45.395	86.654	83.835	85.244	85.245	77.820
synth grass	98.416	98.020	100	99.802	99.604	99.604	100	100	99.604
tree	91.761	96.875	74.716	32.765	94.318	91.193	96.307	98.769	94.697
soil	98.579	96.117	83.428	75.947	99.811	99.621	100	100	99.716
water	88.112	77.622	76.923	74.126	93.007	90.21	90.21	91.608	88.811
residential	81.810	67.444	75.373	54.478	86.194	64.366	90.112	88.619	74.067
commercial	59.069	51.757	83.001	92.213	88.699	94.777	88.889	91.453	86.989
road	76.298	62.606	66.855	36.544	88.763	62.229	95.845	96.034	83.758
highway	81.081	45.656	75.579	73.745	85.039	69.884	84.846	85.328	58.494
railway	85.579	73.814	83.397	82.163	88.235	85.389	89.943	90.417	96.395
parking lot 1	81.940	46.205	63.785	61.671	82.997	73.103	92.507	92.603	90.298
parking lot 2	65.263	52.631	62.807	92.631	78.596	83.86	69.825	70.526	74.737
tennis court	97.571	97.166	98.380	98.380	98.785	98.785	100	100	98.380
running track	94.715	95.983	68.922	89.429	99.366	97.463	99.154	99.366	97.886
OA	83.217	72.575	73.731	66.787	89.055	82.61	90.834	**91.547**	85.791
κ	0.818	0.7023	0.715	0.641	0.881	0.811	0.9	**0.908**	0.846

named as dSVM and dMLRSub. Upon observing the overall accuracies (OAs), we see that dSVM gives better results than dMLRSub. The accuracy of 83.217% on HSI-only data and 73.731% on LiDAR-only data is increased to 89.055% using our proposed unary term. Finally, we use these unary classifiers and introduce our proposed pairwise term in a fully-connected CRF, the results of which are given in the last three columns. Here, dSVM-CRF and dMLRSb-CRF are the results of the classifiers when fully-connected CRF is applied and where the LiDAR and hyperspectral have equal weights. If we use our algorithm for α selection, which provides the weighted addition of LiDAR and hyperspectral data for shadowy and sunny areas, we obtain the result in dSVM-CRF-α; which has the best OA with 91.547%.

Visual results are given in Fig. 5. In Figs. 5*a* and *b*, the unary classification using SVM classifier results are given for hyperspectral and LiDAR only data, and it can be observed that the results are not smooth. Also, it can be observed that there are many wrongly classified pixels on the right of the hyperspectral data due to the cloud coverage; however, the LiDAR information seems not to be affected from the clouds. Using both datasets, we removed the effect of the shadow on the image which can be seen in Fig. 5*c*. Fig. 5*d* shows fully-connected CRF results. In Fig. 5*e*, we show the misclassified points of the test data in red, and the correctly classified test points in black.

Next, we compared our results to competing approaches in Table 2. Here, Potts SVM-MRF represents the classical MRF which uses a single probability distribution without any dimension reduction for the unary term and Potts model for the pairwise term as in [37]. SLRCA [34] uses random forests on features extracted with sparse and low-rank component analysis; DFC is the winning method in the Data Fusion Contest [28]; and GBFF [35] uses deep convolutional neural networks on extinction profiles. Further, dSVM-MRF and dMLRSub-MRF are the results of SVM and MLRSub classifiers combined in an MRF framework as proposed by us in an earlier study in [36]. Upon comparing our method with other state-of-the-art approaches in Table 2; it can be observed that, dSVM-CRF-α outperforms all the methods presented here.

The confusion matrix for the dSVM-CRF-α is given in Table 3, in which red entries in each column denote the class with the highest confusion rate for that class. For example, it can be seen that the worst classified class is the parking lot 2; and it is mostly confused with parking lot 1. This is a very reasonable error considering how difficult it would be to differentiate two parking lots. Next, the lower accuracy of the highway and residential classes can be explained by most samples of these classes being under the cloudy region of the image. Further, there is a confusion between the healthy grass, stressed grass and the soil class, as a mixture of these classes within the same pixels can usually be found together.

For comparison purposes, the confusion matrix from the dSVM-MRF method is given in Table 4. Comparing Tables 3 and 4, it can be observed that the distribution of misclassifications has narrowed down in Table 3, and the misclassifications were more accurate on the closest classes. For instance, the biggest confuser for healthy grass (first column) is the soil class; and the biggest confuser for the railway class (column 11) is the parking lot where pebbles can be found on both surfaces. In doing so, our dSVM-α-CRF method reduced the variation of misclassification in most classes, which adds consistency to the classifier.

It is hard to make a detailed comparison to the SLCRA and DFC methods as confusion matrices were not provided in [34] or [35]. In addition to an overall better accuracy and a better kappa value, a general look indicates that our dSVM-α-CRF method is particularly good at detecting the highway class; whereas the SLCRA and DFC are better at detecting stressed grass. Also, dSVM-α-CRF performs better for targets under the cloudy region whereas the SLRCA and GBFF perform better for classes where training samples were taken from environments where the classes are closely mixed together such as the parking lots 1 and 2.

5 Conclusion

In this paper, we introduced a novel semantic segmentation approach using the fully-connected CRF, where we can both segment and label pixels at the same time. The fully-connected CRF did not have an efficient implementation until very recently, and to the best of our knowledge, this is the first study to use fully-connected CRFs for hyperspectral and LiDAR data. With the mean field inference of the fully-connected CRF, the algorithm is much faster; and as compared to the SLRCA which uses random forests, and to the GBFF which uses deep convolutional neural networks, there are very few parameters in the framework to tweak. Further, we not only use this implementation and introduce it to our community for the first time; we also proposed both a novel unary term and a pairwise term in the 3D domain using UTM coordinates; and adjusted the fc-CRF for the fusion of hyperspectral and Lidar data.

The fusion in UTM coordinates is skipped in many studies, but actually makes a lot of sense to combine them in the physical world and not in the pixel coordinates. In our pairwise term, it can be seen that the higher the height or the spatial distance between pixels of differently classified pixels; the energy is higher, hence

the probability is lower. To the best of our knowledge, our proposed unary term which uses probability distributions obtained from both hyperspectral and LiDAR data classifications and weighs them according to an alpha parameter is also a first. It can be seen that this newly proposed unary term increases the classification accuracy quite a bit.

The fully-connected CRF approach showed excellent outcome in both performance and preservation of the edges. We concluded that fusion of two unary terms and the pairwise term improve the

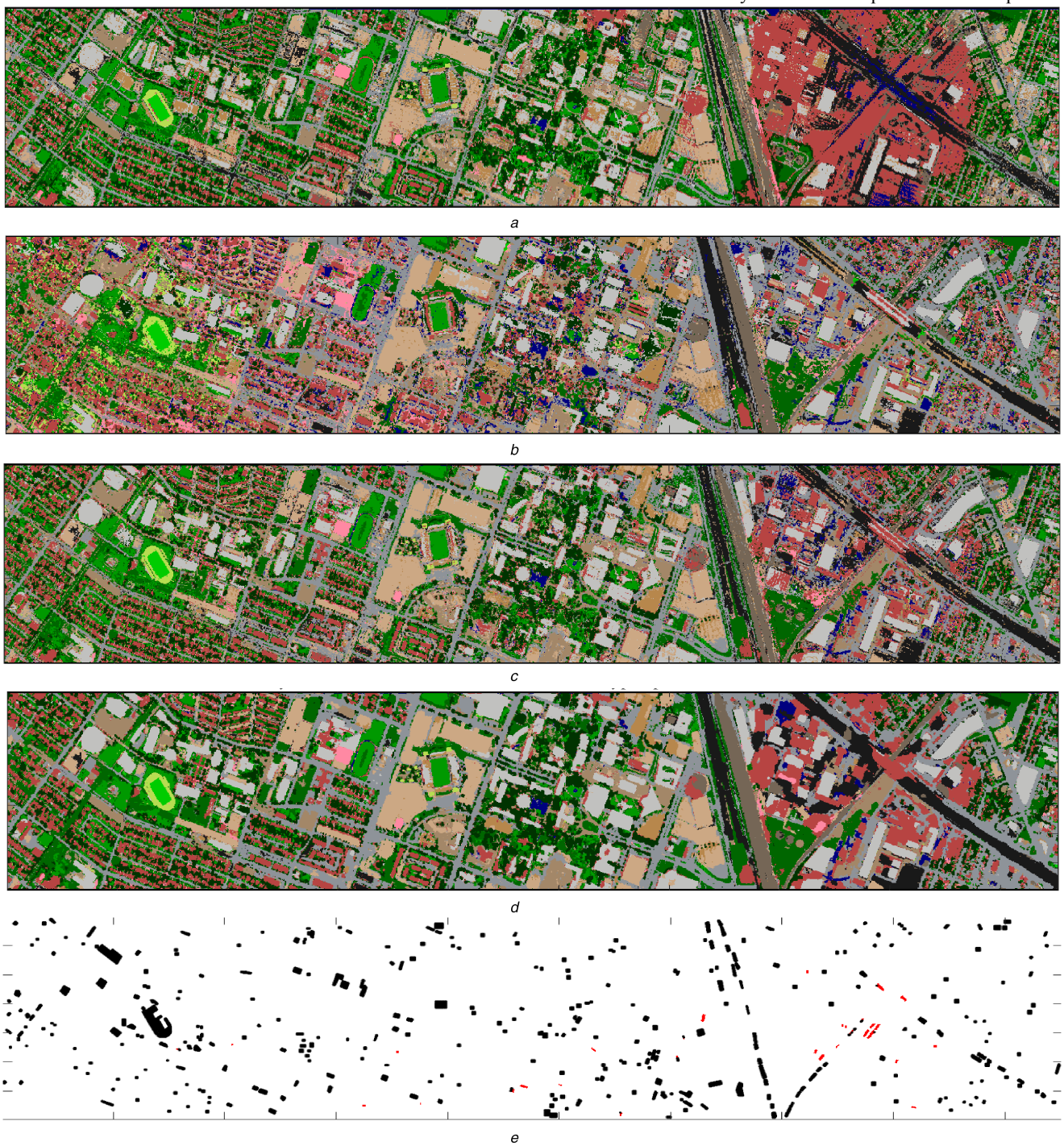

Fig. 5 *Visual results of our proposed approach where the colour representations of the classes are given in Fig. 6*
(a) Unary classification results on hyperspectral data, ***(b)*** Unary classification results on LiDAR data, ***(c)*** Unary classification results on the fusion of hyperspectral and LiDAR data, ***(d)*** Fully-connected CRF results with α correction, ***(e)*** Classification results on the test data. Misclassified points are shown in red

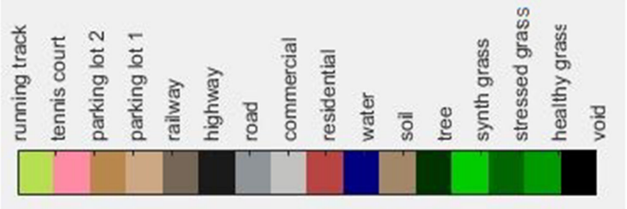

Fig. 6 *Colour representation of classes*

Table 2 Comparison of our proposed approach against competing methods (classification accuracies in %)

	Potts SVM-MRF	SLRCA [34]	DFC [28]	GBFF [35]	dSVM-MRF [36]	dMLRsb-MRF [36]	dSVM-CRF-α
healthy grass	82.431	81.58	73.31	78.73	80.722	81.197	83.096
stressed grass	82.989	99.44	97.84	94.92	86.936	90.132	85.245
synth grass	98.218	98.61	100.00	100	99.604	99.604	100
tree	92.898	96.12	97.82	99.34	93.655	91.477	98.769
soil	98.58	99.72	99.24	99.62	100	99.716	100
water	88.112	98.60	99.30	95.80	94.406	90.21	91.608
residential	78.078	90.39	88.15	87.87	88.153	65.205	88.619
commercial	44.634	95.73	96.20	95.25	89.459	94.587	91.453
road	76.676	98.21	86.59	89.71	91.407	71.86	96.034
highway	64.865	63.42	76.83	81.18	86.004	69.981	85.328
railway	80.93	90.70	92.41	86.34	89.089	89.564	90.417
parking lot 1	75.793	91.07	85.69	92.70	85.783	74.256	92.603
parking lot 2	60.702	76.49	76.49	87.02	78.246	86.667	70.526
tennis court	97.166	100.00	100.00	99.19	98.785	98.381	100
running track	94.503	99.15	99.58	89.64	99.789	97.886	99.366
OA	79.429	91.30	90.30	91.02	89.981	84.578	**91.547**
κ	0.778	0.9056	0.895	0.9033	0.891	0.833	**0.908**

Table 3 Confusion matrix for fully connected CRF (classification accuracies in %)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
healthy grass (1)	**83.1**	0	0	0	0	3.5	0.1	0	0	0	0	0	0	0	0
stressed grass (2)	0	**85.2**	0	0	0	0	0.7	0	0.6	0	0.5	0	0	0	0
synth grass (3)	0	0	**100**	0	0	0	0.4	0	0	0	0	0	0	0	0.2
Tree (4)	1.5	0	0	**98.8**	0	0	1.2	0	0	0	0	0	0	0	0
Soil (5)	9.1	11.7	0	0	**100**	0	0.1	0	2.0	0	0	0	0	0	0
Water (6)	0	0	0	0	0	**91.6**	0.6	0	0	0	0	0	0	0	0
Residential (7)	0.9	3.0	0	1.0	0	0	**88.6**	2.4	0	12.2	0.3	2.2	0	0	0
Commercial (8)	0	0	0	0	0	0	6.5	**91.4**	0	0	0	0	0	0	0
Road (9)	0	0	0	0	0	0	0.7	0	**96.0**	1.8	0.1	1.3	0.3	0	0
Highway (10)	0	0	0	0	0	0.7	0	4.3	0.9	**85.3**	0.5	0	0	0	0
Railway (11)	0	0	0	0	0	0	1.1	0	0	0.7	**90.4**	3.9	0	0	0
parking lot 1 (12)	0	0	0	0.1	0	0	0	0	0.6	0	7.8	**92.6**	29.1	0	0
parking lot 2 (13)	0	0	0	0	0	0	0	0	0	0	0	0	**70.5**	0	0
tennis court (14)	5.4	0	0	0.1	0	4.2	0	1.9	0	0	0.5	0	0	**100**	0.4
running track (15)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**99.4**

Table 4 Confusion matrix for dSVM-MRF for comparison purposes (classification accuracies in %)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
healthy grass (1)	**76.5**	0.5	0	0	0	2.8	0.1	0	0.1	0	0	0	0	0	0
stressed grass (2)	6.7	**86.8**	0	0	0	1.4	0.7	0	1.5	0	0.2	0	0	1.2	0
synth grass (3)	0	0	**99.8**	0	0	0	0.1	0	0	0	0	0	0	0	1.1
tree (4)	0.7	0.1	0	**97.9**	0	2.1	1.9	0	0	0	0	0	0	0	0
soil (5)	8.9	9.7	0	0	**99.6**	2.8	0.1	0	3.5	0	0	0	0	0	0
water (6)	0	0	0	0	0	**85.3**	0.9	0	0	0	0.0	0	0	0	0
residential (7)	2.5	3.0	0	2.0	0	0	**87.1**	4.9	0	9.7	0.9	1.7	0	0	0
commercial (8)	0	0	0	0	0	0	5.4	**88.9**	0.1	1.1	0	0	0.3	0	0
road (9)	0	0	0	0	0	0.7	0.9	0	**91.6**	1.8	0.5	1.7	0	0	0
highway (10)	0	0	0	0	0	0.7	0	4.3	2.0	**85.9**	3.0	1.0	0	0	0
railway (11)	0	0	0	0	0	0	2.1	0	0.2	1.3	**88.5**	4.1	0.3	0	0
parking lot 1 (12)	0	0	0	0.1	0.3	0	0.1	0	0.9	0.1	6.0	**89.1**	24.6	0	0
parking lot 2 (13)	0	0	0	0	0	0	0	0	0.1	0	0	2.3	**74.7**	0	0
tennis court (14)	4.7	0	0	0	0	4.2	0.1	1.9	0	0	0.8	0	0	**98.8**	0.4
running track (15)	0	0	0.2	0	0.1	0	0.4	0	0	0	0	0	0	0	**98.5**

overall accuracy when trained with fully-connected CRFs. In the future, each dataset can be classified with alternative classifiers, such as deep learning, in order to achieve better results.

6 Acknowledgment

The authors thank NCALM for making the University of Houston dataset public. They also thank Dr. Erkut Erdem for fruitful discussions. This work was funded by TUBITAK project no. 115E318.

7 References

[1] Richards, J.A., Xiuping, J.: '*Remote sensing digital image analysis, an introduction*' (Springer-Verlag, Berlin, Heidelberg, 2013)

[2] Qian, Y., Yao, F., Jia, S.: 'Band selection for hyperspectral imagery using affinity propagation', *IET Comput. Vis.*, 2009, **3**, (4), pp. 213–222

[3] Khodadadzadeh, M., Li, J., Prasad, S., *et al.*: 'Fusion of hyperspectral and LiDAR remote sensing data using multiple feature learning', *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 2015, **8**, (6), pp. 2971–2982

[4] Krähenbühl, P., Koltun, V.: 'Efficient inference in fully connected CRFs with Gaussian edge potentials'. Advances in Neural Information Processing Systems 24, Granada, Spain, 2011

[5] Li, J., Bioucas-Dias, J.M., Plaza, A.: 'Spectral-spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields', *IEEE Trans. Geosci. Remote Sens.*, 2012, **50**, (3), pp. 809–823

[6] Paisitkriangkrai, S., Sherrah, J., Janney, P., *et al.*: 'Effective semantic pixel labelling with convolutional networks and conditional random fields'. CVPR Workshops, Boston, MA, USA, 2015, pp. 36–43

[7] Marmanis, D., Wegner, J.D., Silvano, G., *et al.*: 'Semantic segmentation of aerial images with an ensemble of CNNs', *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, 2016, **3**, pp. 473–480

[8] Glenn, T.: 'Context-dependent detection in hyperspectral imagery'. Ph.D. dissertation, University of Florida, 2013

[9] Cross, G.R., Jain, A.K.: 'Markov random field texture models', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1983, **5**, (1), pp. 25–39

[10] Vicente, S., Kolmogorov, V., Rother, C.: 'Joint optimization of segmentation and appearance models'. Int. Conf. Computer Vision (ICCV), Kyoto, Japan, 2009, pp. 755–762

[11] Vicente, S., Kolmogorov, V., Rother, C.: 'Graph cut based image segmentation with connectivity priors'. Computer Vision and Pattern Recognition (CVPR), Alaska, USA, 2008

[12] Shotton, J., Winn, J., Rother, C., *et al.*: 'Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation'. European Conf. Computer Vision (ECCV), Graz, Austria, 2006, pp. 1–15

[13] Yu, L., Xie, J., Chen, S.: 'Conditional random field-based image labelling combining features of pixels, segments and regions', *IET Comput. Vis.*, 2012, **6**, (5), pp. 459–467

[14] Shotton, J., Winn, J., Rother, C., *et al.*: 'Textonboost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context', *Int. J. Comput. Vis.*, 2009, **81**, (1), pp. 2–23

[15] Salamati, N., Larlus, D., Csurka, G., *et al.*: 'Semantic image segmentation using visible and near-infrared channels'. European Conf. Computer Vision (ECCV), Florence, Italy, 2012, pp. 461–471

[16] Farag, A.A., Mohamed, R.M., El-Baz, A.: 'A unified framework for map estimation in remote sensing image segmentation', *IEEE Trans. Geosci. Remote Sens.*, 2005, **43**, (7), pp. 1617–1634

[17] Bai, J., Xiang, S., Pan, C.: 'A graph-based classification method for hyperspectral images', *IEEE Trans. Geosci. Remote Sens.*, 2013, **51**, (2), pp. 803–817

[18] Tarabalka, Y., Rana, A.: 'Graph-cut-based model for spectral-spatial classification of hyperspectral images'. IEEE Int. Geoscience and Remote Sensing Symp., Quebec, July 2014

[19] Tarabalka, Y., Fauvel, M., Chanussot, J., *et al.*: 'SVM and MRF based method for accurate classification of hyperspectral images', *IEEE Geosci. Remote Sens. Lett.*, 2010, **7**, (4), pp. 736–740

[20] Khodadadzadeh, M., Li, J., Plaza, A., *et al.*: 'Spectral-spatial classification for hyperspectral data using SVM and subspace MLR', *IEEE Trans. Geosci. Remote Sens.*, 2010, **48**, (10), pp. 809–823

[21] Liao, W., Pizurica, A., Bellens, R., *et al.*: 'Generalized graph-based fusion of hyperspectral and LiDAR data using morphological features', *IEEE Geosci. Remote Sens. Lett.*, 2015, **12**, (3), pp. 552–556

[22] Lafferty, J., McCallum, A., Pereira, F.C.N.: 'Conditional random fields: probabilistic models for segmenting and labeling sequence data'. Proc. Eighteenth Int. Conf. Machine Learning, ICML '01, Williamstown, MA, USA, June 2001, pp. 282–289

[23] Kohli, P., Rohter, C.: 'Higher-order models in computer vision', in Lezoray, O., Grady, L. (Eds.): '*Image Processing and Analysis with Graphs: Theory and Practice*', (CRC Press, 2012, 1st edn.), pp. 1–28

[24] Bioucas-Dias, J.M., Nascimento, J.M.P.: 'Hyperspectral subspace identification', *IEEE Trans. Geosci. Remote Sens.*, 2008, **46**, (8), pp. 2435–2445

[25] Dalla, M.M., Benediktsson, J.A., Waske, B., *et al.*: 'Morphological attribute profiles for the analysis of very high resolution images', *IEEE Trans. Geosci. Remote Sens.*, 2010, **48**, (10), pp. 3747–3762

[26] Bohning, D.: 'Multinomial logistic regression algorithm', *Ann. Inst. Stat. Math.*, 1992, **44**, (1), pp. 197–200

[27] Luo, R., Liao, W., Zhang, H., *et al.*: 'Classification of cloudy hyperspectral image and LiDAR data based on feature fusion AND decision fusion'. IEEE Int. Geoscience and Remote Sensing Symp., Beijing, 2016

[28] Debes, C., Merentitis, A., Heremans, R., *et al.*: 'Hyperspectral and LiDAR data fusion: outcome of the 2013 data fusion contest', *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 2014, **7**, (6), pp. 2405–2417

[29] Hughes, G.: 'On the mean accuracy of statistical pattern recognizers', *IEEE Trans. Inf. Theory*, 1968, **14**, (1), pp. 55–63

[30] Blake, A., Pushmeet, K., Carsten, R.: '*Markov random fields for vision and image processing*' (MIT Press, Massachusetts, USA, 2011)

[31] Platt, J.: 'Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods', *Adv. Large Margin Classifiers*, 1999, **10**, (3), pp. 61–74

[32] Krähenbühl, P., Koltun, V.: 'Code for efficient inference in fully connected CRFs with Gaussian edge potentials'. Available at http://graphics.stanford.edu/projects/densecrf/

[33] '2013 IEEE GRSS data fusion contest'. Available at http://www.grss-ieee.org/community/technical-committees/data-fusion/

[34] Rasti, B., Ghamisi, P., Plaza, J., *et al.*: 'Fusion of hyperspectral and LiDAR data using sparse and low-rank component analysis', *IEEE Trans. Geosci. Remote Sens.*, 2017, **55**, (11), pp. 6354–6365

[35] Ghamisi, P., Hofle, B., Zhu, X.X.: 'Hyperspectral and LiDAR data fusion using extinction profiles and deep convolutional neural network', *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 2017, **10**, (6), pp. 3011–3024

[36] Aytaylan, H., Yuksel, S.E.: 'Semantic segmentation of hyperspectral images with the fusion of LiDAR data'. IEEE Int. Geoscience and Remote Sensing Symp., Beijing, 2016

[37] Aytaylan, H.: 'Fusion and classification of using conditional random fields'. Master thesis, Hacettepe University, Department of Electrics and Electronics Engineering, January 2017

[38] Cheng, G., Wang, Y., Gong, Y., *et al.*: 'Urban road extraction via graph cuts based probability propagation'. 2014 IEEE Int. Conf. Image Processing (ICIP), Paris, France, 2014, pp. 5072–5076

[39] Chang, C.-C., Lin, C.-J.: 'LIBSVM: A library for support vector machines', *ACM Trans. Intell. Syst. Technol. (TIST)*, 2011, **2**, (3), p. 27

8 Appendix

8.1 Probabilistic SVM

The pSVM [31] is a method that maps the traditional SVM outputs to posterior probabilities. Consider the two-class case where for an input x_i, the label is $y_i \in \{-1, 1\}$, and the standard SVM output is f_i. From a given training set $\{(x_i, y_i)\}$, a new set is defined as $\{(f_i, t_i)\}$ where t_i is the targeting probability representation given as $t_i = (y_i + 1)/2$. Then, given the SVM outputs, the posterior probability of having a label $y_i = 1$ is computed as [17, 38]

$$p_i = 1/\exp(Af_i + B) \tag{11}$$

where A and B are the unknown parameters that are estimated from the minimisation of the cross entropy

$$\min \sum_i t_i \log(p_i) + (1 - t_i)\log(1 - p_i) \tag{12}$$

In the case of the multi-class SVM, the LIBSVM library [39] employs a one-against-one approach which is based on building multiple classifiers and using a voting strategy. With that, first, the pairwise class probabilities r_{ij} are estimated using (12) considering only the i_{th} and j_{th} classes. Then, it solves the following optimisation problem:

$$\min_p 0.5 \sum_{i=1}^{k} \sum_{j:j \neq i} (r_{ji} p_i - r_{ij} p_j)^2 \tag{13}$$

subject to $p_i \leq 0$ and $\sum_{i=1}^{k} p_i = 1$; and where k is the number of data points, and $r_{ji} = 1 - r_{ij}$.

In this study, LibSVM library [39] was used and cross-validation strategies were employed to estimate the parameters of the SVM.

8.2 MLRsub

The strength of MLRSub comes from the MLR classifiers' basic assumption that a pixel in an image contains a mixture of materials within. The mixture model of MLR is defined in [5] as

$$\boldsymbol{x}_i = m\gamma_i + n_i \tag{14}$$

where $\boldsymbol{m} = [m^{(1)}, \ldots, m^{(k)}]$ is a mixing matrix of spectral endmembers, n_i is the noise and $\gamma_i = [\gamma_i^{(1)}, \ldots, \gamma_i^{(K)}]$ is the fractional abundances of the endmembers of the pixel $\boldsymbol{x}_i$. As $p(\gamma_i)$ is unknown, computation of $p(\boldsymbol{x}_i | y_i = k)$ is impossible without a generative model.

According to the proposal in [5], the observation model for the class k can also be written as

$$\boldsymbol{x}_i^{(k)} = \boldsymbol{U}^{(k)}\boldsymbol{z}_i^{(k)} + \boldsymbol{n}_i^{(k)} \tag{15}$$

where $z_i^{(k)}$ is the coordinates of the pixel $x_i^{(k)}$ with respect to the basis $\boldsymbol{U}^{(k)} = [\boldsymbol{u}_1^{(k)}, \ldots, \boldsymbol{u}_{r^{(k)}}^{k}]$, which is a set of $r^{(k)}$-dimensional basis vectors for the subspace associated with class k. If $z_i^{(k)}$ and $n_i^{(k)}$ are assumed to be Gaussian distributed, the following generative model can be obtained:

$$p(\boldsymbol{x_i}\Big| y_i = k)\mathcal{N}(0, \alpha^{(k)}\boldsymbol{U}^{(k)}\boldsymbol{U}^{(k)^{\mathrm{T}}} + \sigma^{(k)^2}\boldsymbol{I}) \tag{16}$$

Here, $r^{(k)}$ can be found by computing the eigendecomposition of the matrix $\boldsymbol{R}^{(k)} = \langle \boldsymbol{x}_{l^{(k)}}^{(k)}\boldsymbol{x}_{l^{(k)}}^{(k)^{\mathrm{T}}} \rangle$. The first $r^{(k)}$ eigenvalues form the subspace of the pixel according to the following relation:

$$r^{(k)} = \min\left\{ r^{(k)} : \sum_{i=1}^{r^{(k)}} \lambda_i^{(k)} \geq \sum_{i=1}^{d} \lambda_i^{(k)} \times \tau \right\} \tag{17}$$

With the help of some algebraic operations and the definition of Gaussian Distribution Function, the following equation can be achieved, which is an MLR [26]:

$$p(y_i = k\Big| \boldsymbol{x}_i, \boldsymbol{w}) = \frac{\exp(\boldsymbol{w}^{(k)}\boldsymbol{f}^{(k)}(\boldsymbol{x}_i))}{\sum_{k=1}^{K}\exp(\boldsymbol{w}^{(k)}\boldsymbol{f}^{(k)}(\boldsymbol{x}_i))} \tag{18}$$

The $\boldsymbol{w}$ in the above equation is defined as $\boldsymbol{w} \equiv [\boldsymbol{w}^{(1)^{\mathrm{T}}}, \ldots, \boldsymbol{w}^{(K)^{\mathrm{T}}}]$ where $\boldsymbol{w}^{(k)} \equiv [\boldsymbol{w}_1^{(k)}\boldsymbol{w}_2^{(k)}]^{\mathrm{T}}$, $\boldsymbol{w}_1^{(k)} \equiv -(1/2\sigma^{(k)^2})$ and $\boldsymbol{w}_2^{(k)} \equiv (1/2\sigma^{(k)^2})(\alpha^{(k)}/\alpha^{(k)} + \alpha^{(k)^2})$. This MLR model then can be solved using the methods proposed in [5].