remote sensing

MDPI

Article

An Efficient Multi-Sensor Remote Sensing Image Clustering in Urban Areas via Boosted Convolutional Autoencoder (BCAE)

Maryam Rahimzad [1], Saeid Homayouni [2,*], Amin Alizadeh Naeini [3] and Saeed Nadi [1,4]

[1] Department of Geomatics Engineering, Faculty of Civil Engineering and Transportation, University of Isfahan, Isfahan 8174673441, Iran; maryamrahimzad74@trn.ui.ac.ir (M.R.); saeednadi@cmail.carleton.ca (S.N.)

[2] Centre Eau Terre Environnement, Institut National de la Recherche Scientifique, Québec, QC G1K 9A9, Canada

[3] Department of Earth and Space Science and Engineering, York University, Toronto, ON M3J 1P3, Canada; naeini@yorku.ca

[4] Department of Civil and Environmental Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada

* Correspondence: saeid.homayouni@ete.inrs.ca

Abstract: High-resolution urban image clustering has remained a challenging task. This is mainly because its performance strongly depends on the discrimination power of features. Recently, several studies focused on unsupervised learning methods by autoencoders to learn and extract more efficient features for clustering purposes. This paper proposes a Boosted Convolutional AutoEncoder (BCAE) method based on feature learning for efficient urban image clustering. The proposed method was applied to multi-sensor remote-sensing images through a multistep workflow. The optical data were first preprocessed by applying a Minimum Noise Fraction (MNF) transformation. Then, these MNF features, in addition to the normalized Digital Surface Model (nDSM) and vegetation indexes such as Normalized Difference Vegetation Index (NDVI) and Excess Green (ExG(2)), were used as the inputs of the BCAE model. Next, our proposed convolutional autoencoder was trained to automatically encode upgraded features and boost the hand-crafted features for producing more clustering-friendly ones. Then, we employed the Mini Batch K-Means algorithm to cluster deep features. Finally, the comparative feature sets were manually designed in three modes to prove the efficiency of the proposed method in extracting compelling features. Experiments on three datasets show the efficiency of BCAE for feature learning. According to the experimental results, by applying the proposed method, the ultimate features become more suitable for clustering, and spatial correlation among the pixels in the feature learning process is also considered.

Keywords: clustering; deep learning; unsupervised learning; convolutional autoencoder; feature extraction; hand-crafted features; multi-sensor data

check for updates

Citation: Rahimzad, M.; Homayouni, S.; Alizadeh Naeini, A.; Nadi, S. An Efficient Multi-Sensor Remote Sensing Image Clustering in Urban Areas via Boosted Convolutional Autoencoder (BCAE). *Remote Sens.* **2021**, *13*, 2501. https://doi.org/10.3390/rs13132501

Academic Editor: Danfeng Hong

Received: 16 May 2021
Accepted: 24 June 2021
Published: 26 June 2021

1. Introduction

Satellite and airborne image classification is one of the most demanding remote sensing (RS) applications [1]. In general, image classification can be categorized as supervised and unsupervised approaches [2]. Although supervised algorithms perform better than unsupervised ones, they require labeled or training samples. As a result, the unavailability of such high-quality and high-quantity training data justifies the use of clustering algorithms [3]. Another advantage of unsupervised methods is that they carry out the classification task by extracting valuable information from the data without any a priori knowledge or specific assumption about data.

Clustering has been one of the fundamental research topics in data-mining studies [4,5]. It is usually referred to as a subcategory of an unsupervised algorithm employed for dividing data into categories of a similar pattern without using training samples [6]. Furthermore, the efficiency of machine learning algorithms highly relies on the representation or feature selection from data [7,8]. In other words, choosing proper feature representation makes classification more efficient and accurate. Thus, proper feature representation for classification is an open question that has recently received much scientific attention [9].

Feature extraction methods try to transform the data space into a new feature space that is more compatible with clustering tasks [10]. Depending on the used features, land-cover identification methods can generally be based on hand-crafted features, unsupervised-learned features, and deep automated extracted features [11].

The first generation of research works on scene classification have been mostly based on hand-crafted features, requiring a considerable amount of domain-expert knowledge and time-consuming hand-tuning [12]. Traditionally, various hand-crafted feature extraction methods, such as scale-invariant feature transform (SIFT) [13], local binary pattern (LBP) [14], non-negative matrix factorization (NMF) [15], and complex wavelet structural similarity (CW-SSIM) [16], have been applied to extract the helpful information.

Automatic feature learning methods are considered a more helpful approach to overcome the limitations of hand-crafted based feature extraction methods [17]. A typical and commonly used unsupervised feature learning method is principal component analysis (PCA) [18]. However, such methods fail to model the nonlinear data structures [19]. Compared to traditional unsupervised feature-learning techniques, the deep-learning-based method includes several processing units and layers and can extract more abstract representations from data with the hierarchy of complexity levels. Furthermore, deep-feature learning approaches effectively detect complicated patterns and hidden information in high-dimensional data. These make it very promising to use deep-learning-based feature extraction method for classification tasks [20].

The autoencoder (AE) deep neural network produces a nonlinear mapping function through a learning process during iterations. The encoders map the input data to their corresponding feature representation, and then the decoder regenerates input from the features extracted by the encoding procedure. The learning procedure is iterative, ensuring that the mapping function is efficient for the input data [21]. Clustering based on independent features extracted from AEs has been studied in recent years. Song et al. [22] have developed one of the first AE-based clustering models. They obtained a practical and abstract feature, which is more informative for clustering purposes. They showed that their proposed deep network could learn a nonlinear mapping by effectively partitioning the transformed feature space [23]. Huang et al. [24] proposed a deep embedded network to use a multilayer Gaussian restricted Boltzmann machine (GRBM) for feature extraction with preserving spatial locality and group sparsity constraints, enabling the model to learn more robust representations for clustering tasks. Tian et al. [25] proposed a graph-based encoder method to use deep sparse AEs for clustering and obtained better accuracy than spectral ones.

Instead of using AE as a preprocessing, joint AE and clustering were also considered in the literature [26]. In Reference [27], a deep embedded clustering (DEC), as a jointly optimized algorithm that learns feature and clusters data simultaneously, is presented. The DEC works by mapping the data space to a new feature space through iteratively optimizing a clustering objective. Guo et al. [28] argued that the feature space could be adversely affected by clustering loss of DEC; accordingly, improved deep embedded clustering (IDEC) was presented to integrate the AE clustering loss and reconstruction loss and improve the performance by preserving the local-spatiality of data. Guo et al. argued the inefficiency of dense layers for clustering tasks and then used a convolutional autoencoder (CAE). Their results demonstrated improvement in both DEC and IDEC accuracies [29]. Thus, the method is introduced as deep convolutional embedded clustering (DCEC).

Recently, several research works have utilized AEs for clustering applications and demonstrated notable performances over clustering using only hand-crafted features [8,30,31]. In the multilayer structure of deep learning models, the feature of one level is transformed into a higher and more abstract level [32]. However, AE had little contribution because it cannot guarantee that similar input data obtain similar representations in the latent space, which is essential for clustering purposes [33]. On the other hand, several studies, including References [34–37], have indicated that hand-crafted features, despite their simplicity, contain information independent of the deep features.

Moreover, in remote-sensing image classification, single-source data generally cannot achieve high accuracy due to the lack of rich and diverse information to distinguish different land-cover classes [38]. Multisource data, such as optical, radar, lidar, thermal, etc., have been used to overcome these limitations and significantly increased the discrimination among the land-cover classes [39–41]. However, the limited representation of the hand-crafted features extracted from multi-sensor data remains a significant challenge and needs further investigation. This leads to the overlapped classes and corrupts the performance of classification [42]. Therefore, to simultaneously use the advantages of deep learning methods and available information in multi-sensor features in the proposed method in this article, instead of using raw images, hand-crafted features were used for training the CAE. The hand-crafted features applied are normalized difference surface model (nDSM), normalized difference vegetation index (NDVI) or Excess Green (ExG(2)), and components of minimum noise fraction (MNF) transformation.

To the best of our knowledge, no study has yet explored the possibility of CAEs for land cover clustering in urban areas. In this paper, a new boosted convolutional autoencoder (BCAE) with hand-crafted features is proposed to extract more effective deep features for clustering RS images automatically. The proposed model uses functional mid-level features to train a light-weighted network to increase the separation of clusters in feature space and boost the clustering results instead of applying raw images as input or employing complex network architectures. The model results are compared with the three most commonly used sets of features to prove the efficiency of the proposed method in extracting robust features. In addition, three different datasets were used to verify the performance of our proposed method compared to the competing sets. Experimental results demonstrate the effectiveness of the proposed method over the other three competing feature sets in terms of required processing time and accuracy of clustering applied to datasets.

2. Materials and Methods

2.1. Remotely Sensed Data

To evaluate the performance of the proposed BCAE method, we perform several experiments on three datasets, namely the Tunis, University of Houston, and ISPRS Vaihingen. Datasets descriptions are as follows:

1. Tunis: These data are an improved satellite image in terms of spatial resolution by the Gram–Schmidt technique and includes eight spectral bands with a spatial resolution of 50 cm acquired by a WorldView-2 sensor. In terms of dimensions, this image is 809 × 809 pixels. This dataset includes digital terrain model (DTM) and digital surface model (DSM) of the study area;
2. University of Houston: The imagery was acquired by National Center for Airborne Laser Mapping (NCALM) on 16 February 2017. The recording sensors consist of an Optech Titan MW (14SEN/CON340) with an integrated camera (a LiDAR imager operating at 1550, 1064, and 532 nm) and a DiMAC ULTRALIGHT + (a very high-resolution color sensor) with a 70 mm focal length. We produced DTM and DSM from this dataset from multispectral LiDAR point cloud data at a 50 cm ground sample distance (GSD) and a very high-resolution RGB image at a 5 cm GSD. The data cube used in the study includes a crop of the original data with a width and height of 1500 × 1500 pixels, in 5 layers (blue, green, and red bands with DSM and DTM);
3. ISPRS Vaihingen: The German Association of Photogrammetry, Remote Sensing, and Geoinformation produced the dataset. It consists of 33 image tiles in infra-red, red, and green wavelength and GSD of 9 cm/pixel that there is ground truth for 16 of them. This imagery also contains DSM extracted from dense LiDAR data. We used a 1500 × 1500 pixel cube from the 26th image patch in the dataset in 4 layers (IR-R-G with nDSM). It should be noted that we used the nDSM generated by Gerke [43].

2.2. *Methodology*

This paper proposes a two-step framework feature learning and clustering of RS images, as illustrated in Figure 1. The proposed CAE was trained with hand-crafted features to extract more effective deep features in the first step. The extracted deep features are fed into the Mini Batch K-Means clustering algorithm to identify clusters in the second step. The central core of this framework is the proposed BCAE, which generates deep features boosted by using hand-crafted features.

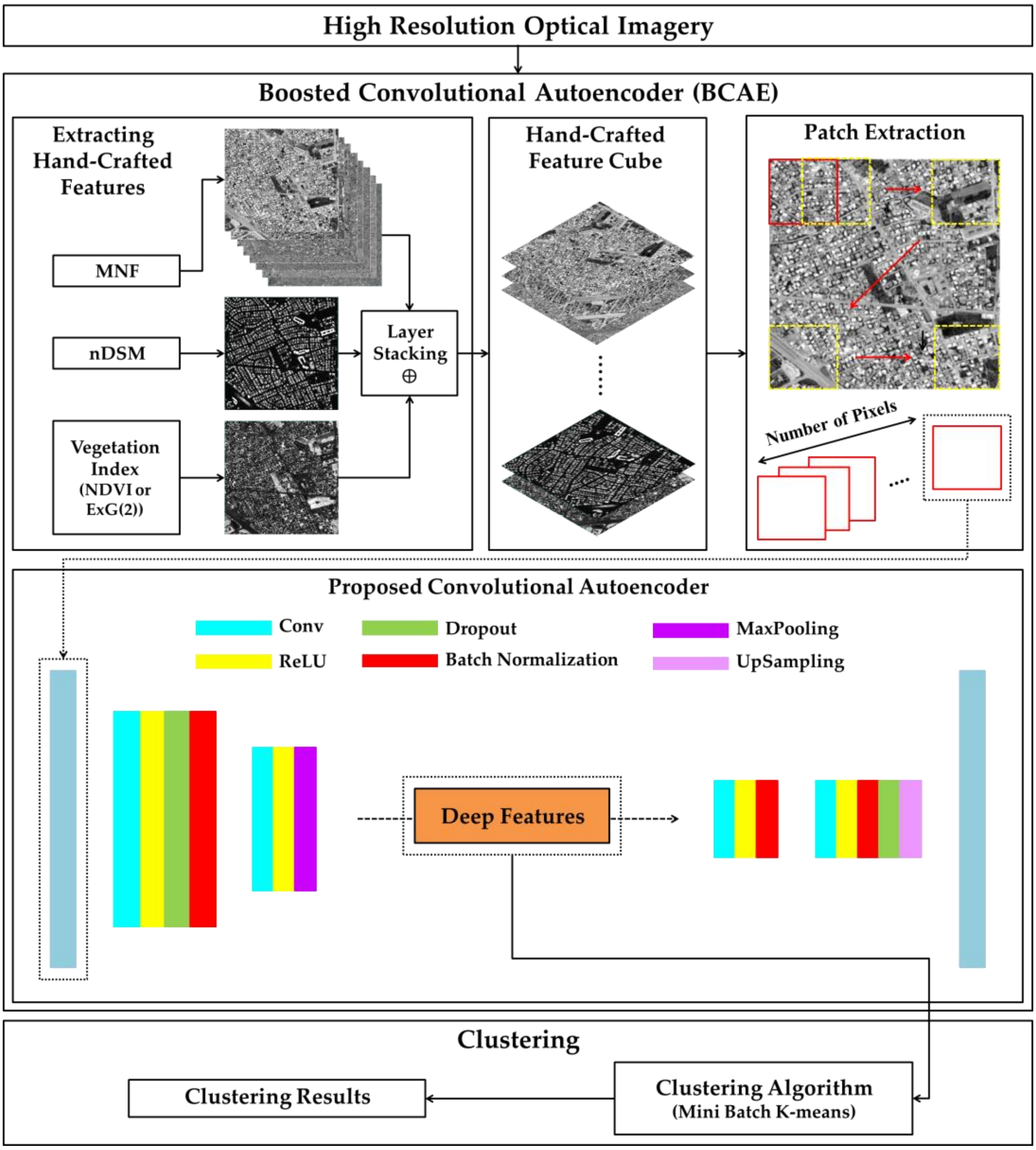

High Resolution Optical Imagery
Boosted Convolutional Autoencoder (BCAE)
Extracting Hand-Crafted Features
MNF
nDSM
Vegetation Index (NDVI or ExG(2))
Layer Stacking ⊕
Hand-Crafted Feature Cube
Patch Extraction
Number of Pixels
Proposed Convolutional Autoencoder
Conv
ReLU
Dropout
Batch Normalization
MaxPooling
UpSampling
Deep Features
Clustering
Clustering Results
Clustering Algorithm (Mini Batch K-means)

Figure 1. An overall overview of the proposed clustering method.

2.2.1. Boosted Convolutional Autoencoder (BCAE)

In this paper, we propose to boost CAE by using hand-crafted features. In our model, a preprocessing step was performed to create more robust representations through our proposed CAE. In this phase, two practical hand-crafted features (i.e., normalized digital surface model (nDSM) and vegetation indexes, such as NDVI and ExG(2) for urban-scene classification, were used. We also used minimum noise fraction (MNF) [44] as another boosting feature. This workflow upgraded our deep features after unsupervised learning by CAE.

CAE [45] is an unsupervised feature learning method that recently attracted scientific attention. CAE is a multilevel feature extraction model aimed at discovering the inner information of images [46]. Compared with conventional dense AE, CAE utilizes the spatial-locality of the original images, which is critical for image clustering and decreases the possibility of overfitting caused by parameter redundancy [30].

As illustrated in Figure 2, the proposed CAE includes two main blocks of encoder and decoder. The transformation from the original image into the hidden layer is named encoding. In contrast, the transforming feature map from the hidden layer toward the output image is described as a decoding procedure. The target of the encoder is to mine the inner information encapsulated in the input data and extracts them as constructive features, and the goal of the decoder block is to reconstruct the input data from the extracted features [46].

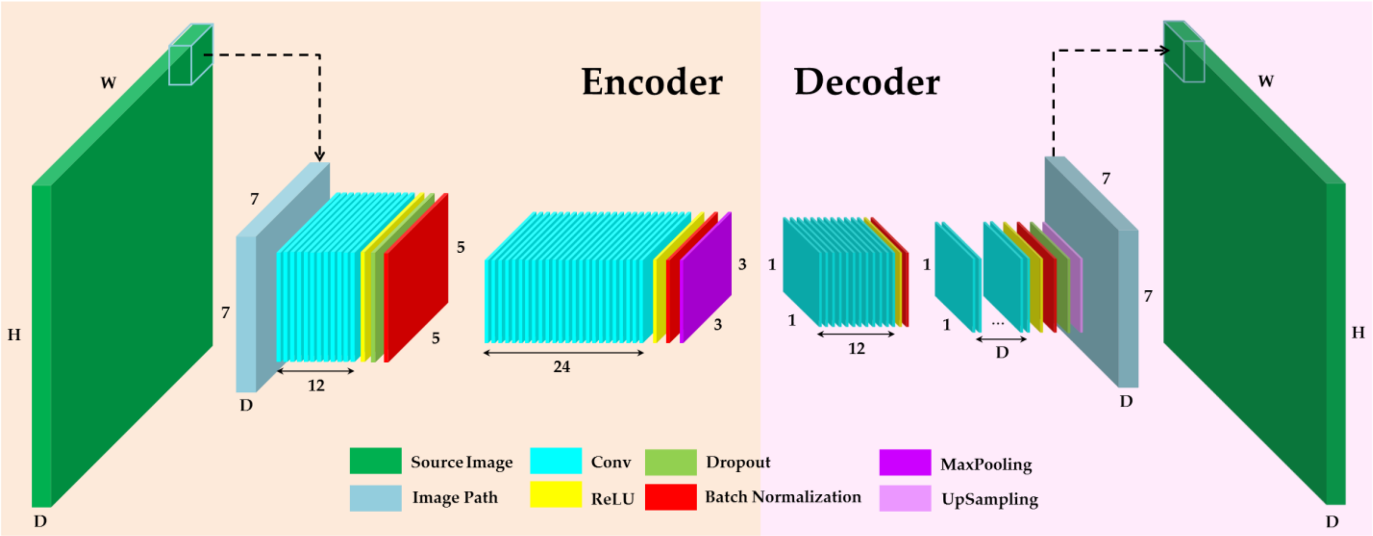


Figure 2. The proposed architecture of the CAE.

In the proposed CAE, $X = \{x_1, x_2, \ldots, x_n\} \in R^{H \times W \times D}$ is used as the input tensor, where D, W, and H indicate the depth (i.e., number of bands), width, and height of the input image, respectively, and n is the number of pixels. The X consists of the image patches $\left(x_i^*\right)$ with the size of $7 \times 7 \times D$ ($x_i^* \in R^{7\times7\times D}$) which are extracted from the input image. In the following, each patch is fed into the encoder Block. For the input x_i^*, the hidden layer mapping (latent representation) of the k^{th} feature map is given by Equation (1) [47]:

$$h^k = \sigma\left(x_i^* * W^k + b^k\right) \tag{1}$$

where b is the bias, σ is an activation function (in this work, the rectified linear unit (ReLU)), and the symbol * corresponds to the 2D convolution. Then, the reconstruction is obtained by using Equation (2):

$$y = \sigma\left(\sum\nolimits_{k \in H} h^k * \widetilde{W}^k + \widetilde{b}^k\right) \tag{2}$$

where $\widetilde{b}$ represents the bias for each input channel, and h denotes the encoded feature maps. $\widetilde{W}$ is the transposition of W, and y is the predicted value [48]. To calculate the parameter vector $\theta_{CAE} = \left\{\mathrm{W}^{\mathrm{k}}, \widetilde{W}^{\mathrm{k}}, \mathrm{b}^{\mathrm{k}}, \widetilde{b}^k\right\}$, the following loss function should minimize [49]:

$$E(\theta) = \frac{1}{n} \sum\nolimits_{i=1}^{n} \|x_i^* - y_i\|_2^2 \tag{3}$$

In order to minimize the loss function (Equation (3)), its gradient with respect to the convolution window parameters $\left(W, \widetilde{W},\ b,\ \widetilde{b}\right)$ should be calculated as Equation (4) [48]:

$$\begin{gathered}\frac{\partial E(\theta)}{\partial W^k} = x^* * \delta h^k + h^k * \delta y \\ \frac{\partial E(\theta)}{\partial b^k} = \delta h^k + \delta y\end{gathered} \tag{4}$$

where δh and δy are the deltas of the hidden states and the reconstruction, respectively. The weights are then updated by using adaptive learning rate methods (ADAMs) [50]. Finally, the ultimate parameters of CAE are estimated once the cost function converges. The output feature maps of the encoder block are considered deep features.

In this work, the dropout [51] strategy is added to improve the computational efficiency and reduce the overfitting of CAE [52]. In addition, the Batch Normalization (BN) [53] is also applied to improve the network's performance. BN helps networks learn faster, as well as increase accuracy [54].

2.2.2. Boosting Deep Representations with Hand-crafted Features

The classification of high-spatial-resolution RS images has become challenged by technology development in pixel size due to the high spectral mixture among different classes, and multispectral images are insufficient for such classification tasks [55]. Light detection and ranging (LiDAR) data like nDSM is a crucial component that provides high-accuracy data about absolute elevation of objects which is almost perfect to distinguish objects of different heights, such as buildings and roads in scene classification [56]. Huang et al. [55] also emphasized the effectiveness of using this elevation data in extracting buildings. Specifically, nDSM is advantageous for separating high and low vegetation. Recently, many building detection approaches that used aerial imagery and LIDAR data have shown that best correctness and completeness are achieved through spectral and elevation information [57,58]. Nowadays, the urban scene classification task mainly relies on elevation information rather than near-infrared (NIR) spectral bands [55].

NDVI, the popular vegetation index, quantifies the vegetation by the difference in photosynthetic response to red-light absorption and near-infrared reflectance. The addition of an NDVI image layer led to a more accurate clustering [59]. It was also added to detect vegetation [60]. According to MacFaden et al. [61], the use of NDVI and nDSM is critical to extract vegetation (in particular trees) next to buildings. The NDVI is based on the spectral response in the NIR spectrum, which does not exist in the RGB data captured by UAVs. Torres-Sánchez et al. [62] investigated several vegetation indices calculated from RGB bands and showed that the excess green (ExG(2)) index [63] obtained the best result for vegetation mapping in UAV data.

The MNF transformation, a modified version of PCA, aims to minimize the correlation between bands and reduce systematic noise in the image [56]. The approach is a conventional dimensionality reducing approach to determine the inherent dimensionality, reduce computational requirements, and segregate data noise [64]. The hand-crafted features applied in this study are nDSM, NDVI or ExG(2), and MNF components (Table 1).

Table 1. Extraction of hand-crafted features.

Feature	Definition
nDSM	nDSM = DSM − DTM where DSM is the digital surface model, and DTM is the digital terrain model.
NDVI	$NDVI = (\rho_{NIR} - \rho_{red})/(\rho_{NIR} + \rho_{red})$ where ρ_{NIR} is the reflectance of the near-infrared wavelength band and ρ_{red} is the reflectance of the red wavelength band.
ExG(2)	$\mathrm{ExG}(2) = 2\mathrm{g} - \mathrm{r} - \mathrm{b}$ where r, g, and b are the color band divided by the sum of three bands per pixel [60].

Table 1. *Cont.*

Feature	Definition
MNF	Considering noisy data as x with n-bands in the form of $x = s + e$, where s and e are the signals and noise parts of x, the covariance matrices of s and e can be calculated as follow: $Cov\{x\} = \sum = \sum_s + \sum_e$ Then, the noise variance of the i^{th} band with respect to the variance for i^{th} band can be described as: $Var\{e_i\}/Var\{x_i\}$ In the following, the MNF transform is considered as a linear transformation: $y = A^T x$ where, y is a produced dataset with n bands, which is a transformation of the original bands, the unknown coefficients (A) are obtained by calculating the eigenvectors associated with sorted eigenvalues: $A\sum_e \sum^{-1} = \Lambda A$ where, Λ is eigenvalue matrix (λ_i), each eigenvalue associated with a_i is the noise ratio in y_i, $i = 1, 2, \ldots, n$ [65].

2.3. Accuracy Assessment

Cluster validity indices (CVIs) are applied to evaluate the performance of clustering algorithms. Unfortunately, most of CVIs, including the Davies–Bouldin index (DBI) and Xie–Beni index (XBI), were not suitable for RS images due to considerable between-cluster overlaps [66]. Thus, in order to estimate the clustering accuracy, the confusion matrix was used, and the overall accuracy (OA), producer's accuracy (PA), and Kappa coefficient (κ) were calculated by using this matrix.

It is noticeable that, due to the dependence of the Mini Batch K-Means algorithm on the initial values and to eliminate its random influence and, also, because k-means may get stuck in local optima, we run Mini Batch K-Means 5 times and display the clustering with the smallest error [67]. Tables 2–4 describe the ground truth data used for each dataset.

Table 2. Ground truths of the Tunis dataset.

Class	Ground Truth (Pixel)
Bare Land	39,052
Building	48,909
Vegetation	33,499
Total	121,460

Table 3. Ground truths of the UH dataset.

Class	Ground Truth (Pixel)
Bare Land	131,061
Building	133,856
Low Vegetation	135,513
Tree	137,151
Total	537,581

Table 4. Ground truths of the Vaihingen dataset.

Class	Ground Truth (Pixel)
Bare Land	719,721
Building	534,190
Low Vegetation	128,787
Tree	508,345
Water	358,957
Total	2,250,000

2.4. Parameter Setting

The encoder block of the proposed CAE framework consists of two convolutional layers (CNN1 and CNN2) having 12 and 24 filters, respectively. The kernel size of these layers is set to be 3 × 3. There are also two convolutional layers (CNN3 and CNN4) with the kernel size of 1 × 1, making it applicable to completely use the spatial information from the input datasets without considering the neighborhood and extracting features based more on the depth of the data. In this block, we choose 12 and D to output convolutional layers (CNN3 and CNN4) in our proposed model, where D is the depth of the dataset in use. Based on trial and error, the learning rate and batch size were chosen to be 0.01 and 10, respectively. Regularization is also used to avoid overfitting. The BN is added to the third dimension of each activation map of the convolutional layer to overcome the internal covariant shift problem. A 30% dropout is applied to the CNN1 and CNN4 layers to enhance the generalization ability by ignoring random connections [68]. In the training process, the Adam optimizer optimized the mean squared error (MSE) cost function. Table 5 summarizes the specifications of the layers in the proposed BCAE framework.

Table 5. The configuration of BCAE for the feature learning of the image path dataset with a 7 × 7 × D window size for the input cube.

Block	Unit	Input Shape	Kernel Size	Regularization	Output Shape
Encoder	CNN1 + ReLU + BN	7 × 7 × D	3 × 3	Dropout (30%)	5 × 5 × 12
	CNN2 + ReLU + BN	5 × 5 × 12	3 × 3	-	3 × 3 × 24
	MaxPooling	3 × 3 × 24	2 × 2	-	1 × 1 × 24
Decoder	CNN3 + ReLU + BN	1 × 1 × 24	1 × 1		1 × 1 × 12
	CNN4 + ReLU + BN	1 × 1 × 12	1 × 1	Dropout (30%)	1 × 1 × D
	UpSampling	1 × 1 × D	7 × 7	-	7 × 7 × D

2.5. Competing Features

In order to evaluate the performance of the BCAE method, three sets of features were considered through three scenarios of input data. The configurations of these features are as follows. The first feature set contained spectral features, including original spectral bands from optical imageries. The second feature set contained spectral and spatial features, including nDSM, NDVI, or ExG(2). Finally, the third feature set contained deep features extracted by training proposed CAE over raw spectral information of each band.

- MS (Multispectral features);
- MDE (MNF + nDSM + ExG(2)) for UH dataset and MDN (MNF + nDSM + NDVI) for Tunis and Vaihingen datasets;
- CAE_MS.

2.6. Mini Batch K-Means

The Mini Batch K-Means [69] clustering algorithm is a modified version of the K-means algorithm that is considered faster, simpler to implement, and generally used for large datasets [70]. The main idea of this algorithm is to make batches of a fixed size randomly in each iteration and update the clusters. A learning rate is defined by inversing of samples to decrease the number of iterations. Therefore, as the number of samples increases, the effect of new samples is reduced [71]. In this study, Mini Batch K-Means with a batch size of 10 was applied to cluster features extracted by BCAE. First, the number of clusters was determined as the number of classes in each dataset. Then, we labeled the clusters through visual inspection.

3. Experimental Results

3.1. Preprocessing

Two preprocessing steps of feature extraction and resampling were applied to the datasets, as summarized in Table 6. The first one was to extract MNF, nDSM, NDVI, and ExG(2) as hand-crafted features, and the second one was to make the resolution of the dataset consistent.

Table 6. Hand-crafted features for all datasets.

Preprocessing		**Dataset**		
		UH	**Tunis**	**Vaihingen**
Features Extraction	MNF	☑	☑	☑
	nDSM	☑	☑	☑
	NDVI	-	☑	☑
	ExG(2)	☑	-	-

The MNF transformation was applied for the preprocessing phase. First, we carried out MNF through Python programming, using Spectral Python (Spy) 0.21 package, leading to a list of corresponding images from informative bands to noisy ones (Table 7). It is evident from (Table 7) that in Tunis, after four first bands (a–d), the others are noisy and contain no clear feature. On the other hand, considering the statistical information about transformed components (Table 8), we noticed that the first layer in UH has the highest eigenvalue with a meaningful interval. It is also true for the Vaihingen dataset. Accordingly, for optimum result and reducing the redundancy, the first three bands in Tunis and the first band in UH and Vaihingen were selected. Then, selected MNF bands were stacked to the nDSM and the NDVI or ExG(2) features. Next, a dataset composed of patches (equal to the number of pixels in the dataset) taken from the stacked cube was generated. Finally, the proposed CAE was trained over the patches, and through unsupervised learning, our deep representations were extracted. It should be noted that all the experiments here were conducted on an Intel Core i3-3120M CPU. Moreover, CAE has been implemented in the Python 3.7 language, which has utilized Tensorflow 2.1.0.

Table 7. MNF Components.

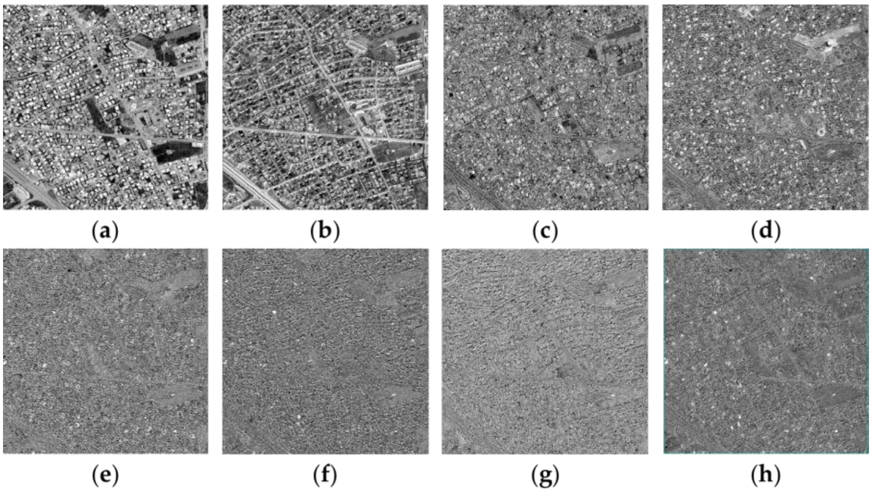
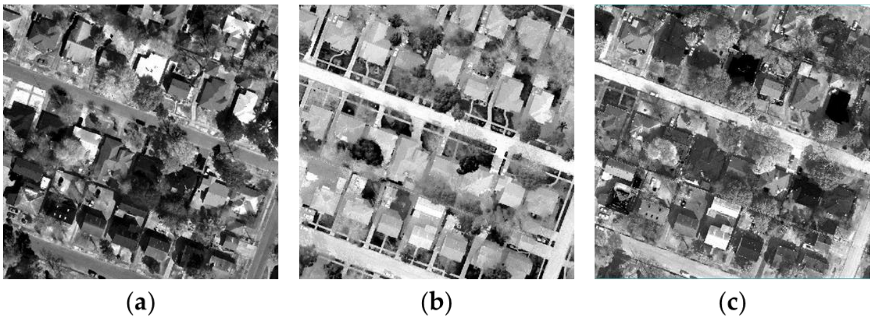
Dataset	**MNF Transformation Results**
Tunis	
UH	

Table 7. *Cont.*

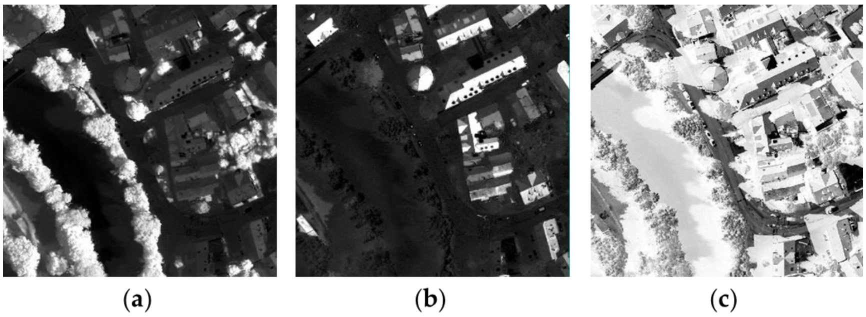
Dataset	MNF Transformation Results
Vaihingen	

Table 8. Statistics of MNF bands.

Dataset	MNF Band	Eigenvalue	Variance	
			Per Band (%)	Accumulative (%)
Tunis	1	85.4706	34.99	34.9
	2	53.9577	22.09	57.08
	3	27.5616	11.29	68.37
	4	22.4381	9.18	77.55
	5	17.8637	7.32	84.87
	6	16.7731	6.86	91.73
	7	14.9745	6.13	97.86
	8	5.2181	2.14	100.00
UH	1	97.7494	46.72	46.72
	2	64.9348	31.03	77.75
	3	46.5552	22.25	100.00
Vaihingen	1	219.6756	55.69	55.69
	2	128.2491	34.10	89.79
	3	39.5643	10.21	100.00

3.2. Clustering Results

We performed the BCAE method on the three datasets and applied the Mini Batch K-Means algorithm to the encoded features. The clustering maps (Figures 3–5) and quantitative results (Tables 9–11) are listed below.

3.2.1. Tunis Dataset

Figure 3 shows the clustering maps for different feature sets. Visually, the performance of the boosted deep features (i.e., BCAE) is similar to spectral and spatial features (i.e., MDN), and they both outperform two other ones (i.e., MS and CAE–MS). It should be mentioned that BCAE leads to compensate noise and extract the correct boundaries of the vast majority of classes. Bare land had a similar spectral representation to some buildings. Hence, spectral features (MS) alone were insufficient to separate these land-covers. Even though the CAE–MS features perform slightly better than MS, summarizing these results makes it possible to prove that the information in its raw form for use as features will significantly impact reducing clustering performance. Table 9 presents the clustering accuracies for different feature sets. MDN, i.e., spatial and spectral features (i.e., MNF, nDSM, and NDVI), significantly improved OA by 16% compared to the first feature set scenario, i.e., using only spectral features (MS). BCAE led to superior performance to the competing features. The corresponding OA and Kappa were 20% and 30% higher than the MS ones, 4% and 6% higher than the MDN ones, 14% and 21% higher than the CAE–MS ones. The improved accuracy over MDN, particularly for building class (6%), indicates that the proposed BCAE method boosted spatial and spectral features. Based on the comparing results in Table 9, the challenging class, i.e., building, demonstrates low clustering accuracy over MS and CAE–MS, while the BCAE has remarkably improved the accuracy. It increased about 6% in OA in comparison with the MDN feature set.

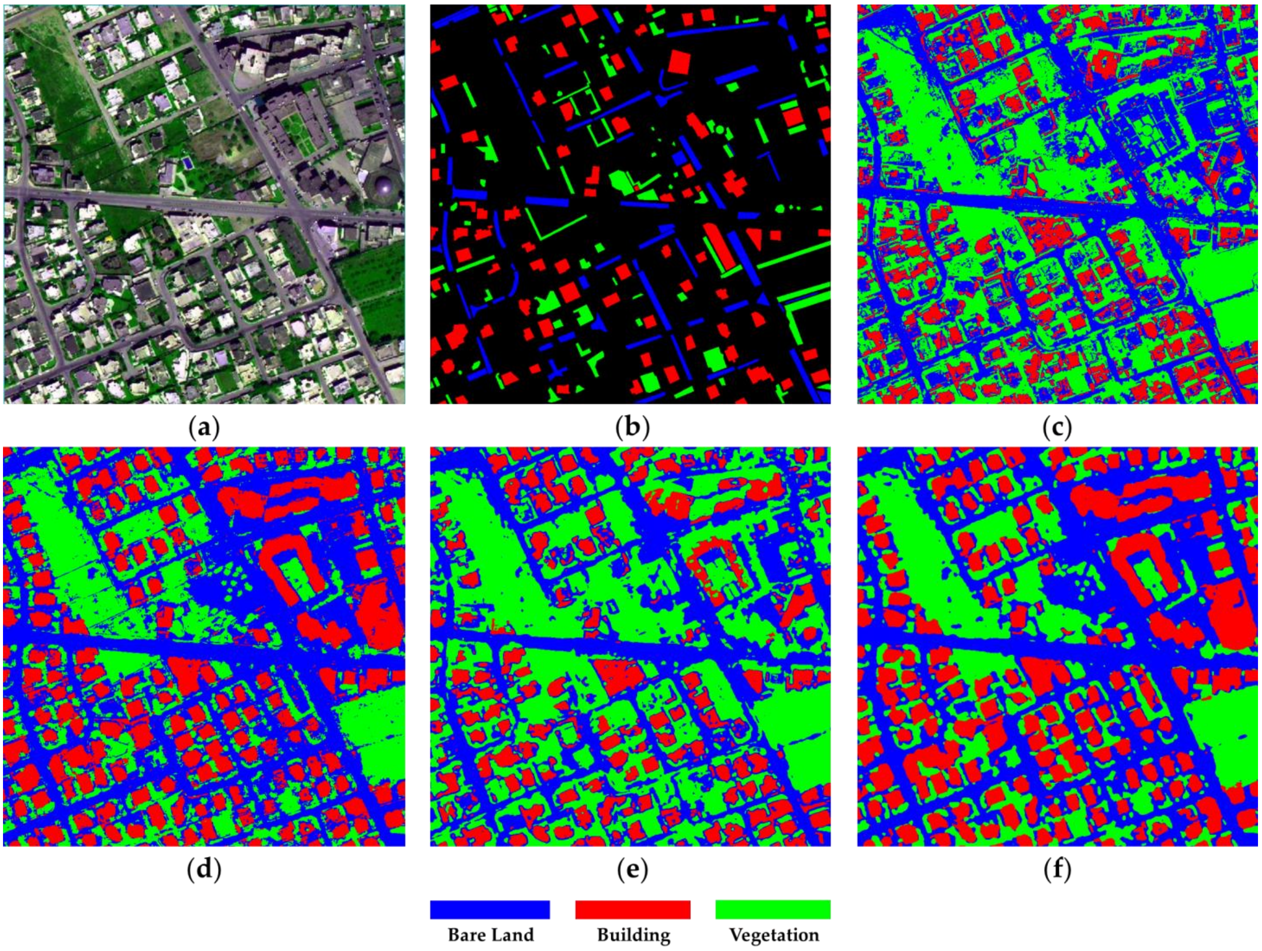


Figure 3. Clustering maps of Tunis dataset. (**a**) The true color composite, (**b**) the ground truth map, (**c**) MS, (**d**) MDN, (**e**) CAE–MS, and (**f**) BCAE.

Table 9. A comparison between the clustering results of the Tunis dataset.

Class (Producer's/User's Accuracy (%))	Method			
	MS	MDN	CAE–MS	BCAE
Bare Land	90.77/63.40	97.46/77.10	92.02/77.72	98.32/85.96
Building	41.90/92.19	81.71/99.43	55.93/99.17	87.94/99.54
Vegetation	98.32/76.02	91.33/95.91	98.42/69.21	97.83/97.59
Overall Accuracy	73.17	89.43	79.25	**94.01**
Kappa Coefficient (×100)	60.54	84.07	69.40	**90.95**

3.2.2. UH Dataset

The UH dataset has two important spectral mixings in land covers, including spectral similarity between trees and low vegetation and bare land and building. In addition, there are some difficulties in the classification of these two classes for trees near buildings or trees without green leaves. The MS and CAE–MS were built on the spectral information only; thus, they are insufficient to address clustering in this dataset and show poor OA and Kappa values. On the other hand, compared to the MS, clustering with the CAE–MS leads to about 2.84% and 3% improvements in OA and Kappa, respectively. According to a recent comparison, the use of CAE has a positive effect on increasing cluster accuracy by extracting more distinct features. In the case of MDE, even with the significant improvement in three classes, compared to MS and CAE–MS feature sets, there was no noticeable improvement for building class; however, we have to note the fact that the use of MNF, nDSM, and ExG(2) features lead to 12% and 16% improvements in the OA and κ compared to the MS, respectively. In particular, MDE used the height information alongside ExG(2) and facilitated tree class distinguishing. The BCAE approach had a visually highlighted enhancement (Figure 4). Moreover, the BCAE method achieved better OA of 30% and κ of 40%. For the most challenging class (i.e., tree) in the UH dataset, the results (Table 10) show poor clustering performance over competing features, while the BCAE has a considerably improved accuracy of 77.37%.

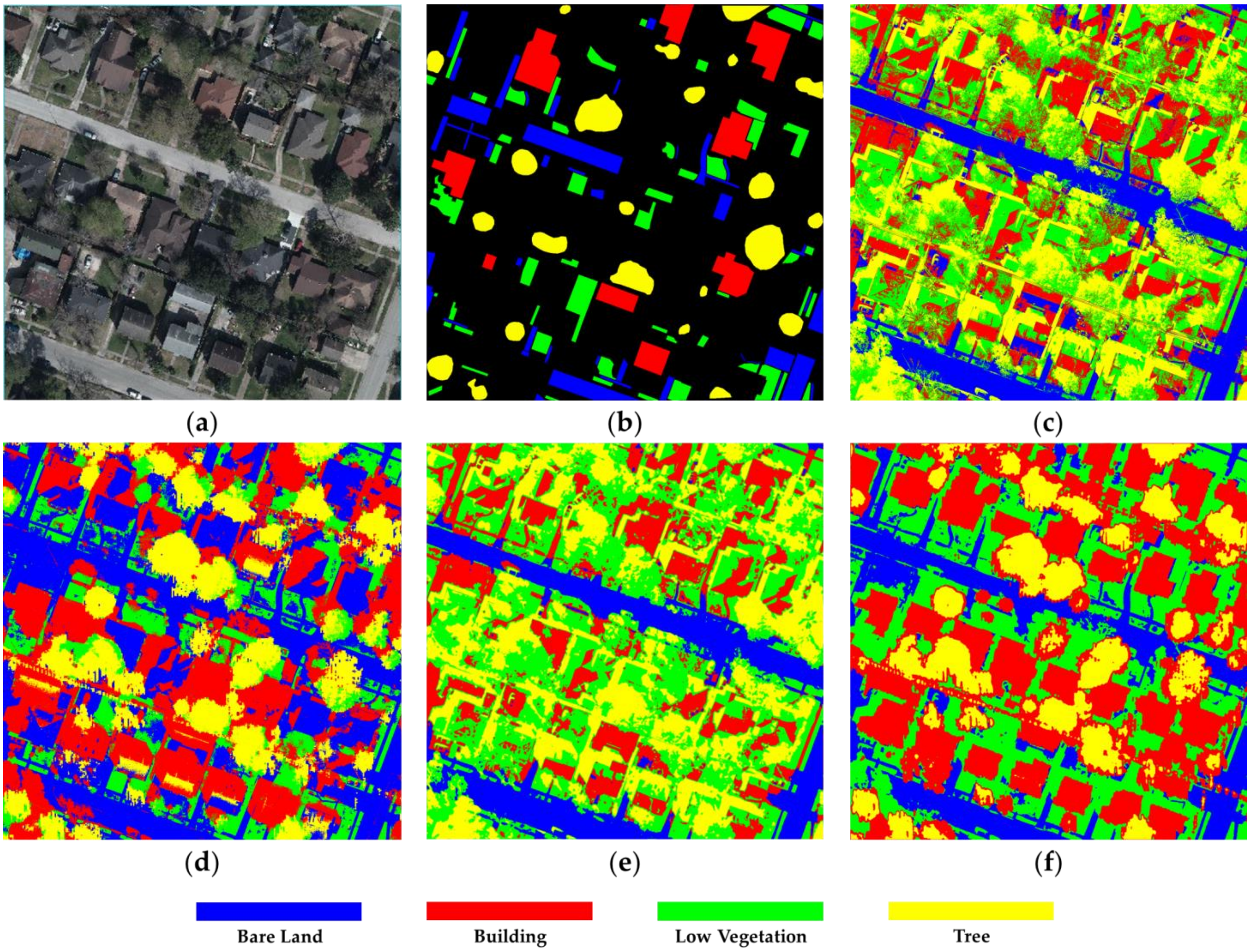


Figure 4. Clustering maps of UH dataset. (**a**) The true color composite, (**b**) the ground truth map, (**c**) MS, (**d**) MDE, (**e**) CAE–MS, and (**f**) BCAE.

Table 10. A comparison between the clustering results of the UH dataset.

Class (Producer's/User's Accuracy (%))	**Method**			
	MS	**MDN**	**CAE–MS**	**BCAE**
Bare Land	79.43/93.56	94.35/79.18	75.65/93.01	94.29/96.70
Building	45.35/54.82	50.09/89.04	54.54/59.88	96.59/77.84
Low Vegetation	66.82/43.40	77.33/64.06	71.63/43.76	94.21/94.58
Tree	38.76/49.71	68.27/39.39	32.93/51.77	77.37/97.61
Overall Accuracy	55.64	67.87	58.48	**90.53**
Kappa Coefficient (×100)	40.85	57.23	44.62	**87.37**

3.2.3. Vaihingen Dataset

Figure 5 and Table 11 demonstrate the clustering maps and accuracies, respectively. In this dataset, we observe that CAE–MS performance is worse than MS by 3% and 2% in terms of κ and OA, respectively. Generally, the spectral overlap between low vegetation and tree classes leads to the worst accuracy among MS and CAE–MS. In this case, the features learned directly from the raw data by the proposed ACE had a negative effect on accuracy. These results show that the clustering of images by using only spectral information in raw data cannot lead to acceptable performance. As shown in Figure 5, although the OA varies from 45 to 54% in three competing feature sets, the clustering maps do not represent the accurate content, especially in building class and bare land class affected by shadow. For very high-resolution imagery, there is no success with clustering on only the spectral data at all. There is a significant improvement for MDN compared to MS and CAE–MS (i.e., OA of 6–8% and Kappa of 9–12%). This is not surprising because the DSM can help in efficiently classifying houses and trees, while the infrared band can discriminate vegetation better. However, the water class is not classified correctly. Accordingly, to achieve high accuracy in such clustering tasks, spatial information is not representative enough. In BCAE features, all classes achieve remarkably better accuracy by using deep-boosted features than other comparing feature sets.

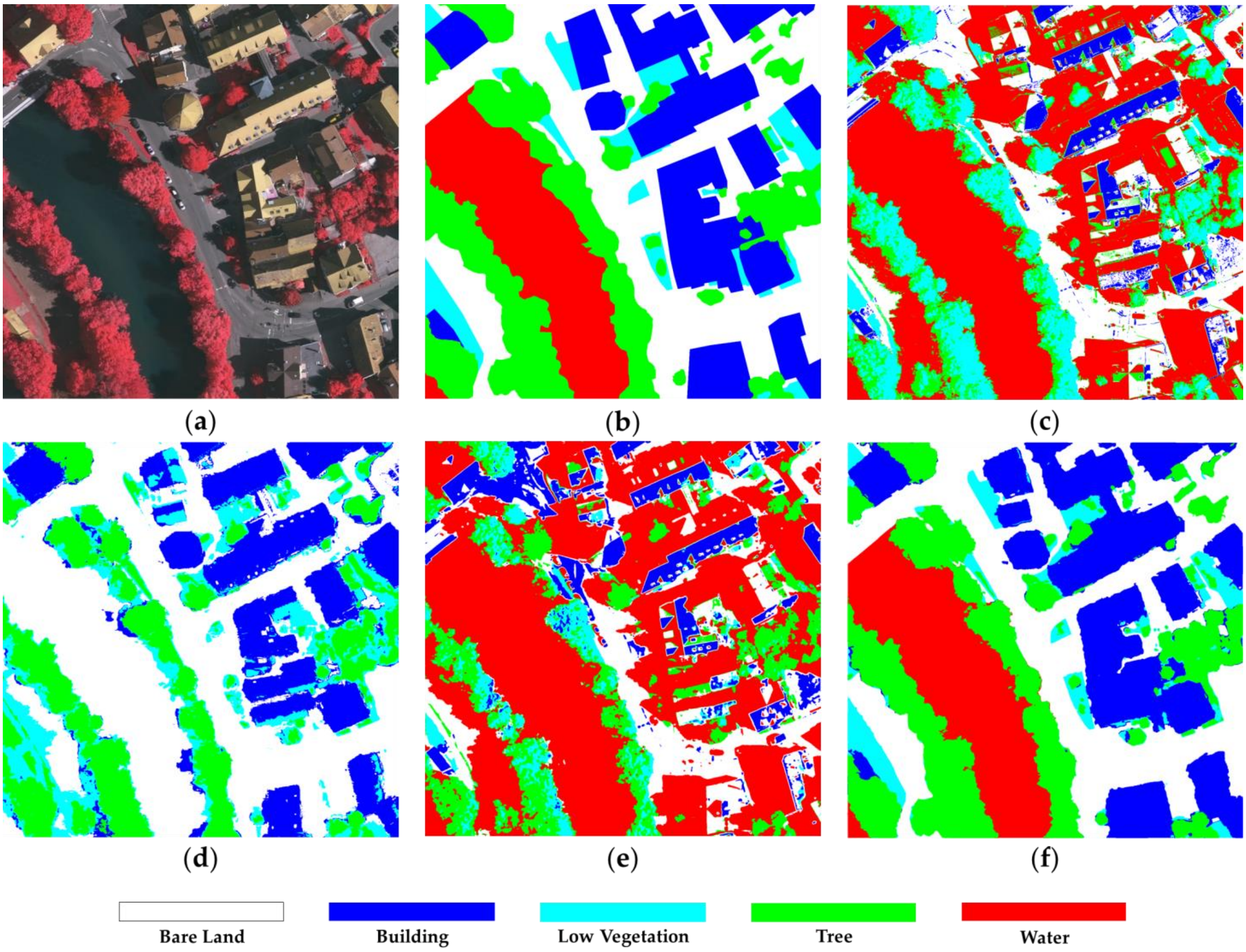


Figure 5. Clustering maps of the Vaihingen dataset. (**a**) The true color composite, (**b**) the ground truth map, (**c**) MS, (**d**) MDN, (**e**) CAE–MS, and (**f**) BCAE.

Table 11. A comparison between the clustering results of the Vaihingen dataset.

Class (Producer's/User's Accuracy (%))	**Method**			
	MS	**MDN**	**CAE–MS**	**BCAE**
Bare Land	40.28/65.12	94.16/32.46	34.21/61.60	91.63/97.22
Building	23.46/85.63	76.23/86.03	23.27/60.57	94.96/90.48
Low Vegetation	51.79/85.97	50.53/21.56	9.34/8.72	74.67/77.24
Tree	10.82/5.08	75.17/88.12	51.09/81.75	93.73/90.79
Water	98.00/32.64	00.00/00.00	98.24/29.65	97.52/96.79
Overall Accuracy	46.41	53.00	44.22	**92.86**
Kappa Coefficient (×100)	33.60	43.03	30.43	**90.65**

3.2.4. Running Time Comparison

The execution time of the K-Means-based algorithm increases with the growth of the size and dimension of the dataset. Hence clustering datasets usually is not a time-efficient task [72]. Therefore, it is necessary to accelerate these algorithms with dimension reduction and efficiently reduce data size. As shown in Table 12, it can be seen that the clustering of features extracted by BCAE has performed well compared to the three competing feature sets in terms of running time, and the clusters have converged faster. As a result, in addition to better accuracy, the proposed method is time-efficient.

Table 12. The computational time of the proposed and competing methods for the three datasets.

Method	**UH (s)**	**Tunis (s)**	**Vaihingen (s)**
MS	34.542	11.063	40.146
MDE/MDN	33.379	10.486	48.911
CAE–MS	39.755	9.917	37.345
BCAE	**31.292**	**9.864**	**35.308**

4. Discussion

The main issue associated with image classification from using supervised methods is the requirement of a large set of labeled training samples. The lack of sufficient training data could highlight the importance of using unsupervised approaches. Clustering of high-resolution urban scenes in remote sensing, considering high complexity and inter-class diversity of land cover, is faced with low performances. Therefore, one solution to overcome this problem would be extracting discriminative hand-crafted features, which requires a considerable amount of experience and time.

In this study, we used an unsupervised feature learning method to produce high-level features automatically. We also used multi-sensor data to boost deep features discrimination power by complementary spatial information and reducing dimensionality. The experimental results from the three datasets validated the efficiency and versatility of the BCAE for image clustering. The deep features outperform the manually designed ones. It turns out to work surprisingly well with clustering algorithm by use of the following factors: (1) the convolutional structure for local spatial-preserving and reducing data redundancy, (2) the BN for decreasing interval covariate shift of network and dropout for computational efficiency and increasing the generalizability, (3) fusion of complementary spatial information (i.e., nDSM) with spectral features (i.e., MNF and NDVI), and (4) no need for large labeled data and using patch learning with the scene image. In addition, since our proposed network is light-weighted, its run-time is short. The numerical results demonstrate the efficiency and superiority of the proposed method in terms of OA and κ, compared with three competing features that rely only on the hand-crafted features or only deep features learned from raw data. On the other hand, the low potential of clustering methods, specifically the K-Means-based algorithm, limits the number of correctly identified classes.

5. Conclusions

Many researchers prove that feature extraction plays a vital role in data processing; thus, in this article, we proposed a practical approach (BCAE) to learn more discriminative features based on CAE and hand-crafted (spectral and spatial) features for clustering RS images. The BCAE is constructed by stacking convolution layers in an AE form that learns features from hand-crafted features as its input, and our network boosts the discrimination of hand-crafted features. The feature maps of the last layer in the encoder part of BCAE that reflect the RS data's essential information can be used to input the clustering algorithm. The extracted features have more separable and compact patterns than the hand-crafted input features, which helps the clustering purpose. Our goal is to map data to a latent space where Mini Batch K-Means is a suitable tool for clustering. The learned deep feature representation is highly discriminative.

We will apply the proposed method to a broader range of data in our future work to establish a more global one. In addition, investigating various hand-crafted features and designing a robust autoencoder model and segment-based instead of pixel-wise clustering would be an open issue for future studies.

Author Contributions: Conceptualization, M.R. and A.A.N.; methodology, A.A.N. and M.R.; programming, M.R.; validation, all; formal analysis, all; writing—original draft preparation, M.R.; writing—review and editing, all; supervision, A.A.N., S.H., and S.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. These datasets can be found here: https://www2.isprs.org/commissions/comm2/wg4/benchmark/ (accessed on 24 April 2021), and https://ieee-dataport.org/open-access/2018-ieee-grss-data-fusion-challenge-%E2%80%93-fusion-multispectral-lidar-and-hyperspectral-data (accessed on 23 August 2018).

Acknowledgments: The authors would like to thank NCALM at the University of Houston for providing the dataset used in this paper, their support group for preparing the data. They also thank the ISPRS and the German Society for Photogrammetry, Remote Sensing, and Geoinformation (DGPF) for providing the Vaihingen benchmark dataset and [43] for making access to nDSM available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Basu, S.; Ganguly, S.; Mukhopadhyay, S.; DiBiano, R.; Karki, M.; Nemani, R. Deepsat: A learning framework for satellite imagery. In Proceedings of the 23rd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 3–6 November 2015; pp. 1–10.
2. Sheikholeslami, M.M.; Nadi, S.; Naeini, A.A.; Ghamisi, P. An Efficient Deep Unsupervised Superresolution Model for Remote Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 1937–1945. [CrossRef]
3. Naeini, A.A.; Babadi, M.; Homayouni, S. Assessment of Normalization Techniques on the Accuracy of Hyperspectral Data Clustering. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, Tehran, Iran, 7–10 October 2017.
4. Ghasedi Dizaji, K.; Herandi, A.; Deng, C.; Cai, W.; Huang, H. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5736–5745.
5. Fatemi, S.B.; Mobasheri, M.R.; Abkar, A.A. Clustering multispectral images using spatial-spectral information. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1521–1525. [CrossRef]
6. Mousavi, S.M.; Zhu, W.; Ellsworth, W.; Beroza, G. Unsupervised clustering of seismic signals using deep convolutional autoencoders. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1693–1697. [CrossRef]
7. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [CrossRef]
8. Fard, M.M.; Thonet, T.; Gaussier, E. Deep k-means: Jointly clustering with k-means and learning representations. *Pattern Recognit. Lett.* **2020**, *138*, 185–192. [CrossRef]

9. Tao, C.; Pan, H.; Li, Y.; Zou, Z. Unsupervised spectral-spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2438–2442.
10. Song, C.; Huang, Y.; Liu, F.; Wang, Z.; Wang, L. Deep auto-encoder based clustering. *Intell. Data Anal.* **2014**, *18*, S65–S76. [CrossRef]
11. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [CrossRef]
12. Hu, F.; Xia, G.-S.; Wang, Z.; Huang, X.; Zhang, L.; Sun, H. Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2015–2030. [CrossRef]
13. Van De Sande, K.; Gevers, T.; Snoek, C. Evaluating color descriptors for object and scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *32*, 1582–1596. [CrossRef]
14. Guo, Z.; Zhang, L.; Zhang, D. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **2010**, *19*, 1657–1663. [PubMed]
15. Hong, S.; Choi, J.; Feyereisl, J.; Han, B.; Davis, L.S. Joint image clustering and labeling by matrix factorization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 1411–1424. [CrossRef] [PubMed]
16. Sampat, M.P.; Wang, Z.; Gupta, S.; Bovik, A.C.; Markey, M.K. Complex wavelet structural similarity: A new image similarity index. *IEEE Trans. Image Process.* **2009**, *18*, 2385–2401. [CrossRef]
17. Kemker, R.; Salvaggio, C.; Kanan, C. Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 60–77. [CrossRef]
18. Pearson, K. LIII. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dubl. Phil. Mag.* **1901**, *2*, 559–572. [CrossRef]
19. Xing, C.; Ma, L.; Yang, X. Stacked denoise autoencoder based feature extraction and classification for hyperspectral images. *J. Sens.* **2016**, *2016*, 3632943. [CrossRef]
20. Opochinsky, Y.; Chazan, S.E.; Gannot, S.; Goldberger, J. K-autoencoders deep clustering. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 4037–4041.
21. Hamidi, M.; Safari, A.; Homayouni, S. An auto-encoder based classifier for crop mapping from multitemporal multispectral imagery. *Int. J. Remote Sens.* **2021**, *42*, 986–1016. [CrossRef]
22. Song, C.; Liu, F.; Huang, Y.; Wang, L.; Tan, T. Auto-encoder based data clustering. In Proceedings of the 2013 Iberoamerican Congress on Pattern Recognition, Havana, Cuba, 20–23 November 2013; pp. 117–124.
23. Chen, P.-Y.; Huang, J.-J. A hybrid autoencoder network for unsupervised image clustering. *Algorithms* **2019**, *12*, 122. [CrossRef]
24. Huang, P.; Huang, Y.; Wang, W.; Wang, L. Deep embedding network for clustering. In Proceedings of the 2014 22nd International Conference on Pattern Recognition, Stockholm, Sweden, 24–28 August 2014; pp. 1532–1537.
25. Tian, F.; Gao, B.; Cui, Q.; Chen, E.; Liu, T.-Y. Learning deep representations for graph clustering. In Proceedings of the 28th AAAI Conference on Artificial Intelligence, Québec City, QC, Canada, 27–31 July 2014; pp. 1293–1299.
26. Yang, B.; Fu, X.; Sidiropoulos, N.D.; Hong, M. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 3861–3870.
27. Xie, J.; Girshick, R.; Farhadi, A. Unsupervised deep embedding for clustering analysis. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 478–487.
28. Guo, X.; Gao, L.; Liu, X.; Yin, J. Improved deep embedded clustering with local structure preservation. In Proceedings of the IJCAI 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 1753–1759.
29. Guo, X.; Liu, X.; Zhu, E.; Yin, J. Deep clustering with convolutional autoencoders. In Proceedings of the 24th International Conference on Neural Information Processing, Guangzhou, China, 14–18 November 2017; pp. 373–382.
30. Li, F.; Qiao, H.; Zhang, B. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognit.* **2018**, *83*, 161–173. [CrossRef]
31. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
32. Wang, S.; Cao, J.; Yu, P. Deep learning for spatio-temporal data mining: A survey. *IEEE Trans. Knowl. Data Eng.* **2020**. [CrossRef]
33. Affeldt, S.; Labiod, L.; Nadif, M. Spectral clustering via ensemble deep autoencoder learning (SC-EDAE). *Pattern Recognit.* **2020**, *108*, 107522. [CrossRef]
34. Paisitkriangkrai, S.; Sherrah, J.; Janney, P.; Van Den Hengel, A. Semantic labeling of aerial and satellite imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 2868–2881. [CrossRef]
35. Nijhawan, R.; Das, J.; Raman, B. A hybrid of deep learning and hand-crafted features based approach for snow cover mapping. *Int. J. Remote Sens.* **2019**, *40*, 759–773. [CrossRef]
36. Majtner, T.; Yildirim-Yayilgan, S.; Hardeberg, J.Y. Combining deep learning and hand-crafted features for skin lesion classification. In Proceedings of the 2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), Oulu, Finland, 12–15 December 2016; pp. 1–6.

37. TaSci, E.; Ugur, A. Image classification using ensemble algorithms with deep learning and hand-crafted features. In Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2–5 May 2018; pp. 1–4.
38. Hong, D.; Gao, L.; Yokoya, N.; Yao, J.; Chanussot, J.; Du, Q.; Zhang, B. More diverse means better: Multimodal deep learning meets remote-sensing imagery classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 4340–4354. [CrossRef]
39. Chen, Y.; Li, C.; Ghamisi, P.; Jia, X.; Gu, Y. Deep fusion of remote sensing data for accurate classification. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1253–1257. [CrossRef]
40. Hang, R.; Liu, Q.; Hong, D.; Ghamisi, P. Cascaded recurrent neural networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5384–5394. [CrossRef]
41. Ienco, D.; Gbodjo, Y.J.E.; Gaetano, R.; Interdonato, R. Generalized Knowledge Distillation for Multi-Sensor Remote Sensing Classification: AN Application to Land Cover Mapping. In Proceedings of the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Nice, France, 31 August-2 September 2020; pp. 997–1003.
42. Aghdam, H.H.; Heravi, E.J. *Guide to Convolutional Neural Networks*; Springer: New York, NY, USA, 2017; Volume 282, p. 7.
43. Gerke, M. *Use of the Stair Vision Library within the ISPRS 2D Semantic Labeling Benchmark (Vaihingen)*; Technical Report; University of Twente: Enschede, The Netherlands, 2014.
44. Green, A.A.; Berman, M.; Switzer, P.; Craig, M.D. A transformation for ordering multispectral data in terms of image quality with implications for noise removal. *IEEE Trans. Geosci. Remote Sens.* **1988**, *26*, 65–74. [CrossRef]
45. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In Proceedings of the International Conference on Artificial Neural Networks, Espoo, Finland, 14–17 June 2011; pp. 52–59.
46. Tang, X.; Zhang, X.; Liu, F.; Jiao, L. Unsupervised deep feature learning for remote sensing image retrieval. *Remote Sens.* **2018**, *10*, 1243. [CrossRef]
47. Ribeiro, M.; Lazzaretti, A.E.; Lopes, H.S. A study of deep convolutional auto-encoders for anomaly detection in videos. *Pattern Recognit. Lett.* **2018**, *105*, 13–22. [CrossRef]
48. Zhao, W.; Guo, Z.; Yue, J.; Zhang, X.; Luo, L. On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery. *Int. J. Remote Sens.* **2015**, *36*, 3368–3379. [CrossRef]
49. Othman, E.; Bazi, Y.; Alajlan, N.; Alhichri, H.; Melgani, F. Using convolutional features and a sparse autoencoder for land-use scene classification. *Int. J. Remote Sens.* **2016**, *37*, 2149–2167. [CrossRef]
50. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
51. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
52. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.-S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]
53. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
54. Chen, Y.; Wang, Y.; Gu, Y.; He, X.; Ghamisi, P.; Jia, X. Deep learning ensemble for hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, *12*, 1882–1897. [CrossRef]
55. Huang, M.-J.; Shyue, S.-W.; Lee, L.-H.; Kao, C.-C. A knowledge-based approach to urban feature classification using aerial imagery with lidar data. *Photogramm. Eng. Remote Sens.* **2008**, *74*, 1473–1485. [CrossRef]
56. Man, Q.; Dong, P.; Guo, H. Pixel-and feature-level fusion of hyperspectral and lidar data for urban land-use classification. *Int. J. Remote Sens.* **2015**, *36*, 1618–1644. [CrossRef]
57. Rottensteiner, F.; Sohn, G.; Gerke, M.; Wegner, J.D.; Breitkopf, U.; Jung, J. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 256–271. [CrossRef]
58. Rafiezadeh Shahi, K.; Ghamisi, P.; Rasti, B.; Jackisch, R.; Scheunders, P.; Gloaguen, R. Data Fusion Using a Multi-Sensor Sparse-Based Clustering Algorithm. *Remote Sens.* **2020**, *12*, 4007. [CrossRef]
59. Hartfield, K.A.; Landau, K.I.; Van Leeuwen, W.J. Fusion of high resolution aerial multispectral and LiDAR data: Land cover in the context of urban mosquito habitat. *Remote Sens.* **2011**, *3*, 2364–2383. [CrossRef]
60. Gevaert, C.; Persello, C.; Sliuzas, R.; Vosselman, G. Informal settlement classification using point-cloud and image-based features from UAV data. *ISPRS J. Photogramm. Remote Sens.* **2017**, *125*, 225–236. [CrossRef]
61. MacFaden, S.W.; O'Neil-Dunne, J.P.; Royar, A.R.; Lu, J.W.; Rundle, A.G. High-resolution tree canopy mapping for New York City using LIDAR and object-based image analysis. *J. Appl. Remote Sens.* **2012**, *6*, 063567. [CrossRef]
62. Torres-Sánchez, J.; Pena, J.M.; de Castro, A.I.; López-Granados, F. Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from UAV. *Comput. Electron. Agric.* **2014**, *103*, 104–113. [CrossRef]
63. Woebbecke, D.M.; Meyer, G.E.; Von Bargen, K.; Mortensen, D.A. Color indices for weed identification under various soil, residue, and lighting conditions. *Trans. ASAE* **1995**, *38*, 259–269. [CrossRef]
64. Denghui, Z.; Le, Y. Support vector machine based classification for hyperspectral remote sensing images after minimum noise fraction rotation transformation. In Proceedings of the 2011 International Conference on Internet Computing and Information Services, Hong Kong, China, 17–18 September 2011; pp. 132–135.
65. Lixin, G.; Weixin, X.; Jihong, P. Segmented minimum noise fraction transformation for efficient feature extraction of hyperspectral images. *Pattern Recognit.* **2015**, *48*, 3216–3226. [CrossRef]

66. Li, H.; Zhang, S.; Ding, X.; Zhang, C.; Dale, P. Performance evaluation of cluster validity indices (CVIs) on multi/hyperspectral remote sensing datasets. *Remote Sens.* **2016**, *8*, 295. [CrossRef]
67. Hanhijärvi, S.; Ojala, M.; Vuokko, N.; Puolamäki, K.; Tatti, N.; Mannila, H. Tell me something I don't know: Randomization strategies for iterative data mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; pp. 379–388.
68. Seydgar, M.; Alizadeh Naeini, A.; Zhang, M.; Li, W.; Satari, M. 3-D convolution-recurrent networks for spectral-spatial classification of hyperspectral images. *Remote Sens.* **2019**, *11*, 883. [CrossRef]
69. Sculley, D. Web-scale k-means clustering. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010; pp. 1177–1178.
70. Feizollah, A.; Anuar, N.B.; Salleh, R.; Amalina, F. Comparative study of k-means and mini batch k-means clustering algorithms in android malware detection using network traffic analysis. In Proceedings of the 2014 International Symposium on Biometrics and Security Technologies (ISBAST), Kuala Lumpur, Malaysia, 26–27 August 2014; pp. 193–197.
71. Béjar Alonso, J. *K-Means vs Mini Batch K-Means: A Comparison*; (Technical Report); Universitat Poiltecnica de Catalunya: Barcelona, Spain, 2013.
72. Yan, B.; Zhang, Y.; Yang, Z.; Su, H.; Zheng, H. DVT-PKM: An improved GPU based parallel k-means algorithm. In Proceedings of the 2014 International Conference on Intelligent Computing, Taiyuan, China, 3–6 August 2014; pp. 591–601.