



# Compressive spectral imaging system for soil classification with three-dimensional convolutional neural network

YUE YU, TINGFA XU,\* ZIYI SHEN, YUHAN ZHANG, AND XI WANG

Key Laboratory of Photoelectronic Imaging Technology and System of Ministry of Education, School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China

\*[ciom\\_xtf1@bit.edu.cn](mailto:ciom_xtf1@bit.edu.cn)

**Abstract:** Compressive spectral imaging systems have promising applications in the field of object classification. However, for soil classification problem, conventional methods addressing this specific task often fail to produce satisfying results due to the tradeoff between the invariance and discrepancy of each soil. In this paper, we explore a liquid crystal tunable filters (LCTF)-based system and propose a three-dimensional convolutional neural network(3D-CNN) for soil classification. We first obtain a set of soil compressive measurements via a low spatial resolution detector, and soil hyperspectral images are reconstructed with improved resolution in spatial as well as spectral domains by a compressive sensing(CS) method. Furthermore, different from previous spectral-based object classification methods restricted to extract features from each type independently, on account of the potential of spectral property on individual solid, our method proposes to apply the principal component analysis(PCA) to achieve a dimensionality reduction in the spectral domain. Then, we explore a differential perception model for flexible feature extraction, and finally introduce a 3D-CNN framework to solve the multi-soil classification problem. Experimental results demonstrate that our algorithm not only is able to accelerate the ability of feature discriminability but also performs against conventional soil classification methods.

© 2019 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

## 1. Introduction

Spectral technique is introduced to learn an interaction between electromagnetic waves and ground objects. Visible-near infrared spectrometer is designed for separating multiplex light to spectrum lines in the visible and near-infrared bands, which is applied to efficiently measure the light reflected from the surface of an object [1, 2]. However, the spectroscopy is restricted to the simplex spectral information it obtains. Hyperspectral imaging(HSI) has been widely employed as a new technology combining imaging technology with spectroscopy, which makes up for the insufficiency of spectrometer by providing spatial information. It simultaneously obtains spatial and spectral information of the scene and forms a three-dimensional data cube. The data cube typically contains continuous and narrow spectral bands image with high spectral resolution. Pixels in the hyperspectral image provide spectrum that is utilized to identify the materials by reflectance intensity, which can be measured based on laboratory hyperspectral sensor observations [3, 4] or hyperspectral remote sensing satellite observations [5, 6]. However, conventional hyperspectral imaging is required to process and transmit large amounts of data. In addition, conventional hyperspectral imaging faces various tradeoffs, such as acquisition time, detector size and photon efficiency. Compressive sensing(CS) theory has emerged as a novel approach for high-dimensional signal acquisition, with the ability to overcome these drawbacks that conventional hyperspectral imaging faced. CS theory acquires certain sparse or compressible signals at a rate significantly below Shannon-Nyquist sampling theorem and achieves signal reconstruction accurately [7]. Since hyperspectral data is highly compressible, CS theory can be

applied to hyperspectral imaging. Recently, several systems have demonstrated the potential of CS theory in spectral imaging, such as coded aperture snapshot spectral imaging(CASSI) [8], code aperture agile spectral imaging (CAASI) [9] and digital-micromirror-device-based multishot snapshot spectral imaging (DMD-SSI) system [10].

Spectral technology is mainly applied to information processing and quantitative analysis aiming to object classification, object identification and object feature information extraction. Object classification plays a crucial role in application fields, which lays a solid foundation for subsequent object quantitative analysis. Spectral-based classification utilizes the spectral features of the object to distinguish the object type. For imaging spectroscopy, spatial features further enrich the classification features. Spectral-based classification has been applied in a wide variety of important fields, such as agriculture [11], geology [12], vegetation [13] and food quality [14]. Spectral classification methods are generally divided into two categories: spectral matching and statistical characteristics. The spectral matching method performs the similarity analysis between the unknown spectrum and the reference spectrum, such as spectral angle mapper(SAM) [15] and spectral information divergence(SID) [16]. However, the spectral matching methods highly rely on reference spectral data. Classification methods based on statistical characteristics address this problem and improve classification performance, such as k-nearest neighbor(KNN) [17], decision tree(DT) [18] and support vector machine(SVM) [19,20]. Soil classification has received considerable attention due to its significant role in soil resource evaluation. It provides a scientific basis to promote agricultural technology [21]. Several studies have focused on conventional classification methods for soil classification. In [22, 23],SAM method was conducted to classify the soil types, in which the angle between the unknown spectra and the reference spectra was calculated to determine the spectral similarity. Furthermore, Vibhute *et al.* demonstrated the effectiveness of SVM method for soil classification with a small set of training samples [24]. In order to take advantage of soil spatial information, Jia *et al.* proposed gray-level co-occurrence matrix (GLCM) method to extract soil texture features in the spatial domain. The effective wavelengths and texture features are selected as input variables for the SVM classification model, where the effective wavelength is obtained by the successive projection algorithm(SPA) [25]. However, these conventional classification methods are limited to manual feature extraction, which fail to interpret rich intrinsic information of hyperspectral data.

Recently, convolutional neural networks(CNNs) have obtained extensively attention in the field of object classification. Therefore, numerous object classification CNN architectures are introduced, such as AlexNet [26], GoogleNet [27] and ResNet [28]. More importantly, the classification of hyperspectral image based on CNN has made a breakthrough in classification performance. The CNN-based method with hyperspectral remote sensing technology is widely used in the classification of land-cover types. Hu *et al.* applied the one-dimensional CNN(1D-CNN) to classify the hyperspectral images in the spectral domain directly [29]. Two-dimensional CNN(2D-CNN) takes the spatial features of hyperspectral images into account. Since hyperspectral images contain numerous spectral bands, dimensionality reduction method in the spectral domain is usually performed before 2D-CNN feature extraction. Researches have confirmed the effectiveness of combining dimensionality reduction method and 2D-CNN for classification. Yue *et al.* employed principal component analysis(PCA) method along the spectral domain to reduce the dimension and utilized the 2D-CNN to extract the features [30]. Makantasis *et al.* introduced Randomized PCA (R-PCA) to condense the hyperspectral image before training [31]. In addition, Liang *et al.* combined PCA, 2D-CNN and sparse representation technology to achieve classification [32]. However, these classification methods based on 2D-CNN framework fail to take full advantage of spectral information.

In this paper, we focus on solving soil classification problem with three-dimensional CNN(3D-CNN)framework, which extracts spatial and spectral information simultaneously. Several classification methods based on 3D-CNN framework for hyperspectral remote sensing image

classification have been developed. In [33,34], 3D-CNN is exploited for feature extraction without dimensionality reduction pre-processing. However, existing classification methods independently extract features from each type. Considering the discrepancy of different soil types, we develop a differential perception model in the spectral domain for flexible feature extraction. Furthermore, we combine the PCA dimensionality reduction method with 3D-CNN, which increases the feature discrimination and improves computational efficiency. This CNN-based classification algorithm is abbreviated to 3D-CNN-SD-PCA.

In addition, this paper proposes a liquid crystal tunable filters(LCTF)-based compressive spectral imaging system for soil classification. In object classification applications based on compressive spectral imaging system, we only focus on achieving high classification accuracy, regardless of obtaining a high quality reconstruction. Solving object classification task with compressive spectral imaging system has added benefit of obtaining fewer measurements and reducing acquisition cost in practical applications. The LCTF is used to modulate the spectral images in the spectral domain. Furthermore, the digital micromirror device (DMD) generates coded aperture patterns for spatial modulation of spectral images. In order to improve image acquisition efficiency and perform soil classification quickly, we add the automatic control module in the imaging system which is capable of automatically acquiring compressive measurements at different coded aperture patterns. To build a classification model, we first obtain a set of soil compressive measurements via a low spatial resolution detector. According to CS method, the soil hyperspectral images are effectively reconstructed with improved resolution in both spatial and spectral domains. Then, the 3D-CNN-SD-PCA algorithm is used to train a classification model which can classify each pixel soil type of the reconstructed spectral images. Experimental results demonstrate that the 3D-CNN-SD-PCA algorithm achieves remarkable classification performance compared with other CNN-based and conventional classification methods.

The following part of this paper is organized as follows. Section 2 introduces the principle of imaging. Section 3 describes the 1D-CNN, 2D-CNN and 3D-CNN framework. Section 4 proposes five classification algorithms based on CNN. Section 5 presents the results of the soil reconstruction and classification. Section 6 provides the conclusion and discussion.

## 2. Principle of imaging

Fig. 1 is the schematic diagram of compressive spectral imaging system for soil classification. The LCTF-based compressive spectral imaging system is used for soil compressive measurements collection [35], which consists of an imaging lens, a LCTF, two relay lenses, a DMD and a complementary metal oxide semiconductor(CMOS) camera. In this paper, we remove a lens in order to optimize the structure of this imaging system. Therefore, the imaging system structure is more compact and simplified. The imaging system used in this paper consists of a LCTF, two imaging lenses, a DMD and a CMOS camera. The light emitted from the soil sample passes through the LCTF, the imaging lens, and then projected onto the DMD. The encoded spectral image is collected onto the CMOS camera through the second imaging lens.

LCTF and DMD are two main components to achieve spectral modulation and spatial modulation, respectively. The spectral image filtered by the LCTF is indeed a multi-spectral image in a narrow band near the center wavelength. The LCTF is used to modulate the spectral images in the spectral domain by the amplitudes of the transmission functions. Spatial modulation is implemented by loading the coded aperture patterns on the DMD. Since the compressive measurements under a single coded aperture pattern are not sufficient to accurately reconstruct the hyperspectral images, it is necessary to utilize different coded aperture patterns for spatial modulation.

Traditional acquisition loads different coded aperture patterns manually, which increases acquisition time. Due to the demand for rapid soil classification, we add automatic control module to automatically control spectral acquisition. The automatic control module loads multiple of

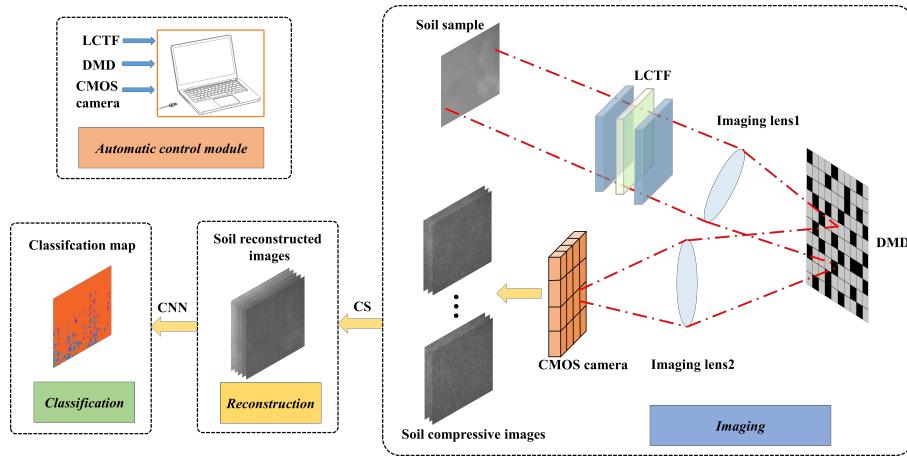


Fig. 1. The schematic diagram of compressive spectral imaging system for soil classification.

code aperture patterns into the random access memory (RAM) of the DMD before acquisition, and automatically transforms the coded aperture pattern displayed by the DMD when the termination wavelength is acquired, and finally obtains a set of compressive measurements using different coded aperture patterns. Excessive acquisition time may lead to system instability, the automatic control module reduces the acquisition time, and collects the compressive measurements quickly, conveniently and accurately. As a result, the increase of acquisition speed improves the efficiency of soil classification.

Suppose the dimension of the coded aperture is  $N_x \times N_y$ , and the number of hyperspectral bands is  $N_\lambda$ . The dimension of the detector is  $M_x \times M_y$ , where  $M_x = N_x/\delta$  and  $M_y = N_y/\delta$ ,  $\delta$  represents the ratio of pixel pitches between the detector and the coded aperture. Suppose the number of spectral channels is  $M_\lambda$  and the number of coded aperture patterns is  $M_k$ . The hyperspectral data is effectively reconstructed using the theory of CS. Suppose  $g \in \mathbb{R}^{M_k \cdot M_\lambda \cdot M_x \cdot M_y \times 1}$  represents the vector of the compressive measurements,  $f \in \mathbb{R}^{N_x \cdot N_y \cdot N_\lambda \times 1}$  represents the vector of the hyperspectral data cube. The imaging model of the system is represented by

$$g = \Phi f, \quad (1)$$

where  $g = [g_1^T, g_2^T, \dots, g_{M_x \cdot M_y}^T]^T$ , and  $g_i \in \mathbb{R}^{M_k \cdot M_\lambda \times 1}$  denotes the measurements on the  $i^{th}$  detector pixel across the  $M_\lambda$  spectral channels using  $M_k$  different coded apertures.  $\Phi \in \mathbb{R}^{(M_k \cdot M_\lambda \cdot M_x \cdot M_y) \times (N_x \cdot N_y \cdot N_\lambda)}$  represents the transmission matrix of the imaging system.  $f = [f_1^T, f_2^T, \dots, f_{N_x \cdot N_y}^T]^T$ , and  $f_i \in \mathbb{R}^{N_\lambda \times 1}$  represents the spectrum of the  $i^{th}$  pixel across the  $N_\lambda$  spectral bands. The transmission matrix of the system includes the combined effects of the LCTF and DMD, which is expressed as

$$\Phi = \Phi_{xy} \otimes \Phi_\lambda, \quad (2)$$

where  $\otimes$  denotes the Kronecker product,  $\Phi_{xy} \in \mathbb{R}^{(M_k \cdot M_x \cdot M_y) \times (N_x \cdot N_y)}$  denotes the spatial transmission matrix of the coded apertures. The structure of the matrix  $\Phi_{xy}$  is denoted as

$$\Phi_{xy} = \begin{bmatrix} \Phi_{xy}^1 & \mathbf{0}_{M_k \times \delta^2} & \cdots & \mathbf{0}_{M_k \times \delta^2} \\ \mathbf{0}_{M_k \times \delta^2} & \Phi_{xy}^2 & \cdots & \mathbf{0}_{M_k \times \delta^2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{M_k \times \delta^2} & \mathbf{0}_{M_k \times \delta^2} & \cdots & \Phi_{xy}^{M_x \times M_y} \end{bmatrix}, \quad (3)$$

where  $\mathbf{0}_{M_k \times \delta^2} \in \delta^{M_k \times \delta^2}$  is a zero matrix, and  $\Phi_{xy}^i \in \mathbb{R}^{M_k \times \delta^2}$  is the spatial transmission matrix of the  $i^{th}$  detector pixel. In this paper, a sparse random matrix is utilized as  $\Phi_{xy}^i$ . In Eq. (2),  $\Phi_\lambda \in \mathbb{R}^{M_\lambda \times N_\lambda}$  is the spectral transmission matrix of the LCTF. Suppose  $T_s^l(\lambda)$  represents the  $l^{th}$  channel transmission function of LCTF. The  $l^{th}$  row of  $\Phi_\lambda$  is generated by discretizing  $T_s^l(\lambda)$  into  $N_\lambda$  points. The overall compression ratio of the proposed system is given by

$$\gamma = \frac{N_x \cdot N_y \cdot N_\lambda}{M_k \cdot M_\lambda \cdot M_x \cdot M_y} = \frac{\delta^2 \cdot N_\lambda}{M_k \cdot M_\lambda}. \quad (4)$$

Based on the theory of CS, the hyperspectral data can be reconstructed from a set of compressive measurements under the sparse assumption. It is universally known that hyperspectral data can be sparsely represented on certain bases, which is given by

$$f = \Psi \theta, \quad (5)$$

where  $\Psi$  denotes the sparse basis,  $\theta$  denotes the sparse coefficient vector. In this paper, the sparse basis  $\Psi$  is expressed as  $\Psi = \Psi_1 \otimes \Psi_2 \otimes \Psi_3$ , where  $\Psi_1 \otimes \Psi_2 \in \mathbb{R}^{(N_x \cdot N_y) \times (N_x \cdot N_y)}$  is the two-dimensional Haar discrete wavelet transform(DWT) basis, and  $\Psi_3 \in \mathbb{R}^{N_\lambda \times N_\lambda}$  is the one-dimensional discrete cosine transform(DCT) basis. Considering the measurement noise on the CMOS,  $g$  is expressed as

$$g = \Phi \Psi \theta + \omega, \quad (6)$$

where  $\omega$  is the noise vector. The hyperspectral data is reconstructed by solving the following  $l_1$ -norm minimization problem

$$\hat{\theta} = \arg \min_{\theta} \|\theta\|_1 \text{ subject to } \|g - \Phi \Psi \theta\|_2 \leq \varepsilon, \quad (7)$$

where  $\varepsilon$  represents the bound of the noise. Several optimization algorithms can be used to solve the reconstruction problem, such as gradient projection for sparse reconstruction(GPSR), two-step iterative shrinkage/thresholding(TwIST) and alternating direction method of multipliers(ADMM). In this paper, the TwIST algorithm is used to solve the  $l_1$ -norm minimization problem in Eq. (7), because it is robust to the variation of parameters and computational efficient. We apply the reconstructed hyperspectral data to distinguish the soil type of each pixel.

### 3. CNN framework

In this section, we introduce basic operations of three soil classification CNN frameworks. In order to distinguish soil type, we use the 1D-CNN, 2D-CNN and 3D-CNN framework respectively. The 1D-CNN framework uses a single pixel spectral vector as input feature while the 2D-CNN and the 3D-CNN framework consider  $w \times w$  neighborhoods of a pixel spectral vector.

#### 3.1. 1D-CNN framework

Specifically, the 1D-CNN framework consists of three convolutional layers, three Rectified Liner Units (ReLU) layers, two pooling layers, a fully connected layer and a softmax layer. The 1D-CNN framework is shown in Fig. 2. The input layer is a pixel vector of hyperspectral data, and the output is the label of the pixel vector. The output of each layer is provided as input to the next layer. Firstly, we introduce the convolutional layer operation. The output of one-dimension convolutional layer is given by

$$Y_j^d = b_j + \sum_{l=1}^L \sum_{r=0}^{R-1} \omega_{jl}^r X_l^{d+r}, \quad (8)$$

where  $Y_j^d$  represents the output of  $j^{th}$  feature map at the position  $d$ ,  $b_j$  denotes the bias of  $j^{th}$  feature map,  $L$  is the number of feature maps in the previous layer,  $R$  is the size of the convolutional kernel in the spectral domain,  $\omega_{jl}^r$  represents the weight of  $j^{th}$  feature map at the position  $r$  connected to the  $l^{th}$  feature map in the previous layer,  $X_l^{d+r}$  represents the  $l^{th}$  feature map at the position  $(d+r)$  in the previous layer.

In order to increase the non-linear representations in the network, we choose ReLU as the activation layer after the convolutional layer [26], which is given by

$$Y_j^d = \max(X_j^d, 0), \quad (9)$$

where  $Y_j^d$  represents the output of  $j^{th}$  feature map at the position  $d$ ,  $X_j^d$  represents the  $j^{th}$  feature map at the position  $d$  in the previous layer.

The pooling layer reduces the size of previous layer output features, for its advantage of extracting main features and simplifying the computation complexity. The max pooling layer takes the maximum value in a small patch as the output value of the pooling layer.

After several convolutional layers, ReLU layers and pooling layers, the input pixel vector is converted into a feature vector. Then, the fully connected layer merges features that are obtained by previous layer. The output of the fully connected layer is connected to the softmax classifier to obtain the probability of different soil types. We choose the logistic loss(Log loss) function after the softmax classifier to calculate the error between the predicted label and the real label. The goal of our training is to minimize the Log loss function.

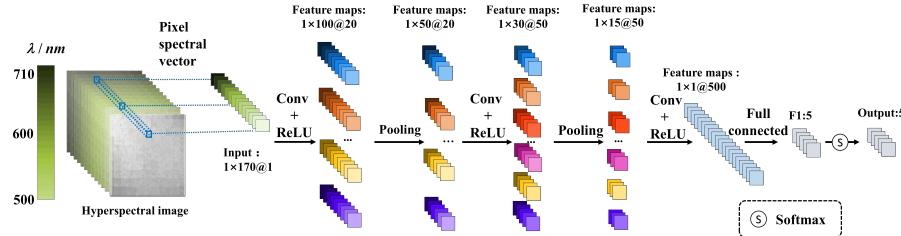


Fig. 2. 1D-CNN framework.

### 3.2. 2D-CNN framework

The 1D-CNN framework ignores rich spatial information of hyperspectral data. For this reason, we introduce the 2D-CNN framework in soil classification to exploit the spatial features. Similarly, the 2D-CNN framework consists of three convolutional layers, three ReLU layers, two pooling layers, a fully connected layer and a softmax layer. The 2D-CNN framework is shown in Fig. 3. The input layer of 2D-CNN framework is  $w \times w$  neighborhoods of a pixel spectral vector, which is different from 1D-CNN framework. The main difference between the 2D-CNN framework and the 1D-CNN framework is convolutional layer operation. The output of two-dimensional convolutional layer is given by

$$Y_j^{hw} = b_j + \sum_{l=1}^L \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \omega_{jl}^{pq} X_l^{(h+p)(w+q)}, \quad (10)$$

where  $Y_j^{hw}$  represents the output of  $j^{th}$  feature map at the position  $(h, w)$ ,  $b_j$  denotes the bias of  $j^{th}$  feature map,  $L$  is the number of feature maps in the previous layer,  $P$  and  $Q$  is the size of the convolutional kernel in the spatial domain,  $\omega_{jl}^{pq}$  represents the weight of  $j^{th}$  feature map at

the position  $(p, q)$  connected to the  $l^{th}$  feature map in the previous layer,  $X_l^{(h+p)(w+q)}$  represents the  $l^{th}$  feature map at the position  $(h + p, w + q)$  in the previous layer. In particular, when the convolution operation is applied to the input layer,  $L$  represents the number of spectral bands.

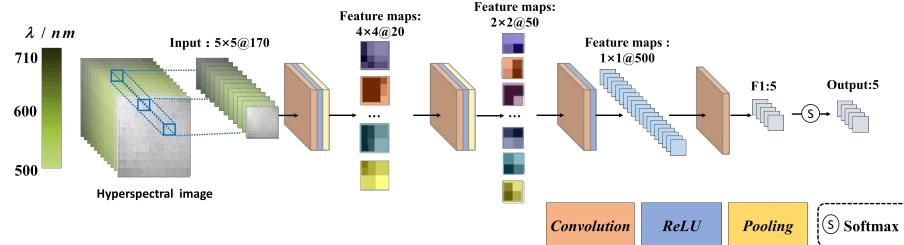


Fig. 3. 2D-CNN framework.

Two-dimensional max pooling layer performs in the similar way to the 1D-CNN framework. The max pooling layer takes the maximum value in a small two-dimensional patch as the output value of the pooling layer.

### 3.3. 3D-CNN framework

The 2D-CNN framework fails to take full advantage of spectral information. In order to preserve spectral information of the input data, we further introduce the 3D-CNN framework in soil classification to learn both spatial and spectral features simultaneously. Similarly, the 3D-CNN framework consists of three convolutional layers, three ReLU layers, two pooling layers, a fully connected layer and a softmax layer. The 3D-CNN framework is shown in Fig. 4. The input layer of 3D-CNN framework is  $w \times w$  neighborhoods of a pixel spectral vector. The main difference between the 3D-CNN framework and the 2D-CNN framework is convolutional layer operation. The output of three-dimensional convolutional layer is denoted as [36]

$$Y_j^{hwd} = b_j + \sum_{l=1}^L \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \sum_{r=0}^{R-1} \omega_{jl}^{pqr} X_l^{(h+p)(w+q)(d+r)}, \quad (11)$$

where  $Y_j^{hwd}$  represents the output of  $j^{th}$  feature map at the position  $(h, w, d)$ ,  $b_j$  denotes the bias of  $j^{th}$  feature map,  $L$  is the number of feature maps in the previous layer,  $P$  and  $Q$  is the size of the convolutional kernel in the spatial domain,  $R$  is the size of the convolutional kernel in the spectral domain,  $\omega_{jl}^{pqr}$  represents the weight of  $j^{th}$  feature map at the position  $(p, q, r)$  connected to the  $l^{th}$  feature map in the previous layer,  $X_l^{(h+p)(w+q)(d+r)}$  represents the  $l^{th}$  feature map at the position  $(h + p, w + q, d + r)$  in the previous layer. Three-dimensional max pooling layer performs in the spectral and spatial domains simultaneously, which takes the maximum value in a small three-dimensional cube as the output value of the pooling layer.

## 4. Training methodology for classification model

In this section, we comprehensively analyze four CNN-based algorithms and propose an optimal algorithm. The 1D-CNN, 2D-CNN and 3D-CNN algorithms train the features after mean subtraction pre-processing directly. Moreover, we put forward the 3D-CNN-SD algorithm based on differential perception model in the spectral domain. In order to increase feature discrimination and improve computational efficiency, we propose the 3D-CNN-SD-PCA algorithm finally. Fig. 5 shows the flow chart of five CNN-based algorithms.

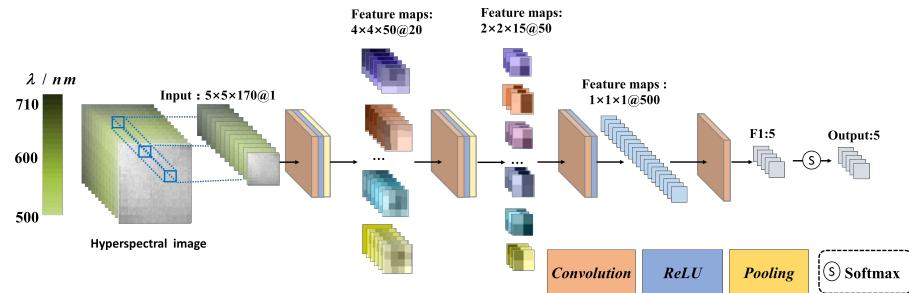


Fig. 4. 3D-CNN framework.

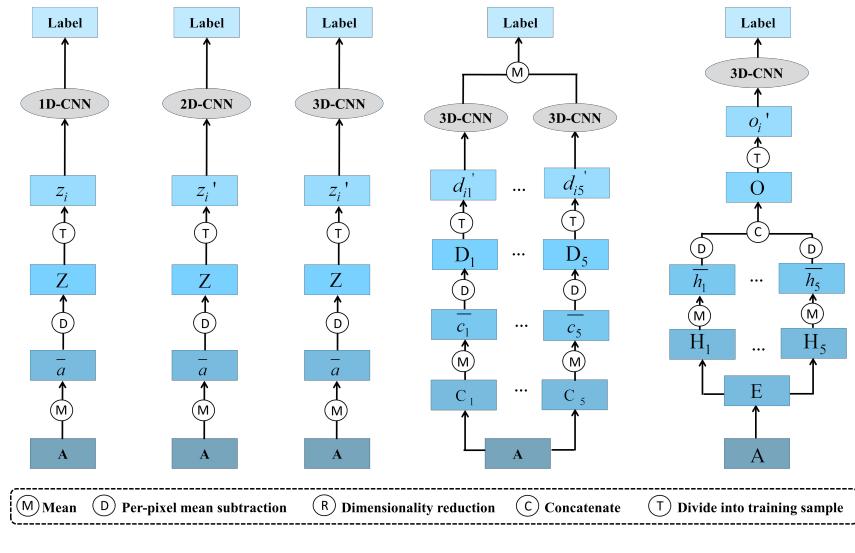


Fig. 5. The flow chart of five CNN-based algorithms:(a)1D-CNN;(b)2D-CNN;(c)3D-CNN;(d)3D-CNN-SD and (e)3D-CNN-SD-PCA.

#### 4.1. Analysis of several proposed CNN-based algorithms

Firstly, we introduce the training methodology of 1D-CNN algorithm for spectral feature extraction, as shown in Fig. 5(a). Assume we have to classify five soil types in total. We firstly convert the three-dimensional hyperspectral data cube to a two-dimensional matrix in which each row represents the spectral vector of one pixel. Suppose  $A = [a_1, a_2, \dots, a_m]^T \in \mathbb{R}^{m \times N_\lambda}$  is the spectral matrix of all training dataset pixels, where  $m$  represents the number of pixels,  $a_i \in \mathbb{R}^{1 \times N_\lambda}$  denotes the spectral vector of the  $i^{\text{th}}$  pixel.  $T = [t_1, t_2, \dots, t_m]^T \in \mathbb{R}^{m \times 1}$  is the category label of all training dataset pixels, where  $t_i \in \{1, 2, 3, 4, 5\}$  denotes the category label of the  $i^{\text{th}}$  pixel. In order to accelerate the convergence of training process and improve classification accuracy, it is necessary to perform mean subtraction pre-processing on the input pixels before CNN training.  $Z = [z_1, z_2, \dots, z_m]^T \in \mathbb{R}^{m \times N_\lambda}$  is the spectral matrix of all training dataset pixels after mean subtraction pre-processing, where  $z_i = a_i - \bar{a}$ ,  $\bar{a} \in \mathbb{R}^{1 \times N_\lambda}$  denotes the column mean vector of  $A$ . Then, we use the 1D-CNN framework to solve the multi-soil classification problem. The 1D-CNN algorithm directly uses a single pixel spectral vector as input layer, therefore  $z_i$  is the  $i^{\text{th}}$  pixel input layer.

1D-CNN algorithm has many advantages, including efficiency and simplicity. However, it fails to take full advantage of the three-dimensional data cube characteristics. In order to take spatial information of hyperspectral images into account, we propose a 2D-CNN algorithm, as shown in Fig. 5(b). Similarly, we perform mean subtraction pre-processing in the spectral domain before training. Then, we use the 2D-CNN framework to solve the multi-soil classification problem. Suppose  $z_i' \in \mathbb{R}^{w \times w \times N_\lambda}$  is the  $i^{\text{th}}$  pixel input layer, which represents the spectral vector in association with  $w \times w$  spatial neighborhoods of  $i^{\text{th}}$  pixel after mean subtraction in the Z.

However, the 2D-CNN algorithm fails to make full use of spectral information. In order to extract spatial and spectral features simultaneously, we further develop a 3D-CNN algorithm, as shown in Fig. 5(c). Similarly, we perform mean subtraction pre-processing in the spectral domain before training. Then, we use the 3D-CNN framework to solve the multi-soil classification problem. Suppose  $z_i' \in \mathbb{R}^{w \times w \times N_\lambda}$  is the  $i^{\text{th}}$  pixel input layer, which represents the spectral vector in association with  $w \times w$  spatial neighborhoods of  $i^{\text{th}}$  pixel in the Z.

Commonly, hyperspectral classification algorithms extract features from each type independently. We put forward a differential perception model for flexible feature extraction, as shown in Fig. 5(d). Differential perception is achieved by performing the difference operation between the spectral vector of each pixel and the reference spectral vector of each soil type. A can be divided into five matrices according to soil type, which is expressed as  $A = [C_1, C_2, C_3, C_4, C_5]^T$ , where  $C_k$  represents the  $k^{\text{th}}$  soil type spectral matrix.  $C_k$  contains all pixels spectral vector of the  $k^{\text{th}}$  soil type, in which each row represents one pixel spectral vector of the  $k^{\text{th}}$  soil type. The difference perception model extracts the reference spectral vector of each soil type firstly. Suppose  $\bar{c}_k \in \mathbb{R}^{1 \times N_\lambda}$  represents the  $k^{\text{th}}$  soil type reference spectral vector,  $\bar{c}_k$  is the column mean vector of  $C_k$ . Suppose  $D_k \in \mathbb{R}^{m \times N_\lambda} = [d_{1k}, d_{2k}, \dots, d_{mk}]^T$  is the spectral difference matrix with  $k^{\text{th}}$  soil type reference spectral vector, where  $d_{ik} = a_i - \bar{c}_k$ , and  $d_{ik}$  represents the difference between the  $i^{\text{th}}$  pixel spectral vector and the  $k^{\text{th}}$  soil type reference spectral vector. We utilize parallel structure to train the spectral difference feature. The parallel structure has multiple basic CNN structures that work in parallel. Each parallel CNN structure focuses on the spectral difference with a specific soil type, and the classification performance is improved by averaging the predicted probability of each parallel structure ultimately. Specifically, the parallel structure has five parallel CNNs with same structure and different weights, and the  $k^{\text{th}}$  network input layer represents the spectral difference with  $k^{\text{th}}$  soil type. We use the 3D-CNN framework to solve the multi-soil classification problem. The  $i^{\text{th}}$  pixel  $k^{\text{th}}$  network input layer can be expressed by  $d_{ik}' \in \mathbb{R}^{w \times w \times N_\lambda}$ , which represents the spectral difference vector in association with  $w \times w$  spatial neighborhoods of  $i^{\text{th}}$  pixel in the  $D_k$ . Each network has the ability to acquire the probability of five soil types after the softmax classifier. The mean layer is added after the softmax layer, which has the ability to average the five parallel network probabilities. The loss function is calculated using the average probability. The derivative of the loss function to the mean layer can be formulated as

$$\frac{\partial F}{\partial F_k} = \frac{1}{5} \times \frac{\partial F}{\partial F'}, \quad (12)$$

where  $F$  represents the final output of the loss function,  $F_k$  represents the output of the  $k^{\text{th}}$  network softmax layer, and  $F'$  represents the output of the mean layer. The parameters of five networks are updated respectively.

#### 4.2. An optimal CNN-based algorithm

The differential perception model extracts feature flexibly and improves the classification performance. However, this algorithm increases the number of input features, resulting in a reduction in computational efficiency. For CNN-based hyperspectral classification, the algorithm optimality criterion is high classification accuracy. In addition, network complexity, model generalization ability and computational efficiency are also the criteria for evaluating the algorithm

performance. Therefore, we propose an optimal algorithm based on the previous algorithms. This algorithm is abbreviated to 3D-CNN-SD-PCA, as shown in Fig. 5(e). Hyperspectral data usually contains hundreds of bands. A typical characteristic of hyperspectral data is that adjacent bands are highly correlated, so there exists spectral information redundancy. To reduce information redundancy, this optimal algorithm reduces the dimensionality of spectral data firstly and then performs differential perception model. Dimensionality reduction method in the spectral domain reduces the number of features while preserving original information. The dimensionality reduction methods are divided into linear methods and non-linear methods. We choose linear method for the reason that linear method can preserve spatial information intact. PCA is a typical unsupervised linear dimensionality reduction method with the theoretical basis of the largest variance between sample in the low-dimensional space.

Suppose the input data is a two-dimensional matrix, one dimension represents the pixel and one dimension represents the feature. The PCA method performs mean subtraction pre-processing for each feature firstly, and then calculates the covariance matrix, and obtains the eigenvalues and eigenvectors of the matrix. The  $s$  eigenvectors corresponding to the largest  $s$  eigenvalues form the projection matrix. Dimensionality reduction is achieved by projecting the mean subtraction processed data onto selected eigenvectors [37].

$A$  is the input matrix for PCA dimensionality reduction. Suppose  $E = [e_1, e_2, \dots, e_m]^T \in \mathbb{R}^{m \times s}$  represents the matrix after dimensionality reduction, where  $s$  represents the number of principal components which reserves 99.9% of original information,  $e_i \in \mathbb{R}^{1 \times s}$  denotes the spectral principal component vector of the  $i^{\text{th}}$  pixel. Differential perception is achieved by performing the difference operation between the spectral principal component vector of each pixel and the reference spectral principal component vector of each soil type.  $E$  can be divided into five matrices according to soil type, which is expressed as  $E = [H_1, H_2, H_3, H_4, H_5]^T$ , where  $H_k$  represents the  $k^{\text{th}}$  soil type spectral principal component matrix.  $H_k$  contains all pixels spectral principal component vector of the  $k^{\text{th}}$  soil type in which each row represents one pixel spectral principal component vector of the  $k^{\text{th}}$  soil type. The difference perception model extracts the reference spectral principal component vector of each soil type firstly. Suppose  $\bar{h}_k \in \mathbb{R}^{1 \times s}$  represents the  $k^{\text{th}}$  soil type reference spectral principal component vector,  $\bar{h}_k$  is the column mean vector of  $H_k$ . Suppose  $O_k \in \mathbb{R}^{m \times s} = [o_{1k}, o_{2k}, \dots, o_{mk}]^T$  is the spectral principal component difference matrix with  $k^{\text{th}}$  soil type reference spectral principal component vector, where  $o_{ik} = e_i - \bar{h}_k$ , and  $o_{ik}$  represents the difference between the  $i^{\text{th}}$  pixel spectral principal component vector and the  $k^{\text{th}}$  soil type reference spectral principal component vector. Then, we concatenate the spectral principal component differences with different soil types in the spectral domain. Suppose  $O = [O_1, O_2, O_3, O_4, O_5] \in \mathbb{R}^{m \times (5 \cdot s)}$  is the spectral principal component difference matrix of all training dataset pixels. To simplify the network structure, we use a single network to train the features. We use the 3D-CNN framework to solve the multi-soil classification problem. The  $i^{\text{th}}$  pixel input layer can be expressed by  $o'_i \in \mathbb{R}^{w \times w \times (5 \cdot s)}$ , which represents the spectral principal component difference vector in association with  $w \times w$  spatial neighborhoods of  $i^{\text{th}}$  pixel in the  $O$ .

## 5. Experiments and results

In this section, we introduce the approach to build the datasets using the LCTF-based compressive spectral imaging system. We illustrate the reconstruction results and spectral characteristics of different soil types. Additionally, in order to confirm the feasibility of this system in soil classification, we evaluate the CNN-based algorithms from the aspects of classification metrics and computation time. To further confirm the superiority of the proposed algorithm, we compare this algorithm with the conventional hyperspectral classification methods.

### 5.1. Building the datasets

A training dataset is required to train the proposed classification model and a test dataset is required to evaluate the performance. We distinguish five typical soil types of known real label: Red earths, Humid-thermo ferralitic, Purplish soil, Paddy soil and Chernozem. Fig. 6 is the imaging system in the laboratory. The soil sample is placed on a fixed platform. The light source is high intensity fiber coupled illuminator. The DMD (DLP9500) consists of  $1920 \times 1080$  micromirror arrays and the spatial resolution of CMOS is  $1024 \times 1024$ .

In the measurement stage, each soil type is imaged independently. Each soil type is divided into two samples, one soil sample is used to generate a training dataset, the other soil sample is used to generate a test dataset. There are 10 soil samples in total since we distinguish five soil types. We capture compressive measurements for each soil sample using the LCTF-based compressive spectral imaging system. The CMOS exposure time is set to 20ms. The wavelength range of the LCTF is 500nm-710nm and the scanning step of the LCTF is 10nm. The LCTF is switched for 22 times at each coded aperture pattern to complete the acquisition of different spectral channels. We use a coded aperture including  $280 \times 280$  pixels for spatial modulation, and the compressive measurements are obtained by combining  $8 \times 8$  pixels into one macro-pixel on the CMOS. Thus, we utilize  $35 \times 35$  macro-pixels on the detector to collect the compressive measurements. In order to conduct the spatial modulation, we transform the coded aperture pattern 20 times. Since the discretization precision of the LCTF transmittance function is set to 1.24 nm, the reconstructed hyperspectral images contain 170 spectral bands in the 500nm-710nm wavelength range, including  $280 \times 280$  spatial pixels. Thus, the overall compression ratio of the system is  $\gamma = \delta^2 \cdot N_\lambda / (M_k \cdot M_\lambda) = 8^2 \cdot 170 / (20 \cdot 22) \approx 24.73$ . In the reconstructed image, the rectangular area that covered soil is selected as the region of interest(ROI). We utilize the pixels in the ROI to generate the training dataset and the test dataset. For the training dataset, the selected ROI contains  $80 \times 80$  pixels. For the test dataset, the selected ROI contains  $50 \times 50$  pixels. Therefore, each soil type has 6400 training dataset pixels and 2500 test dataset pixels.

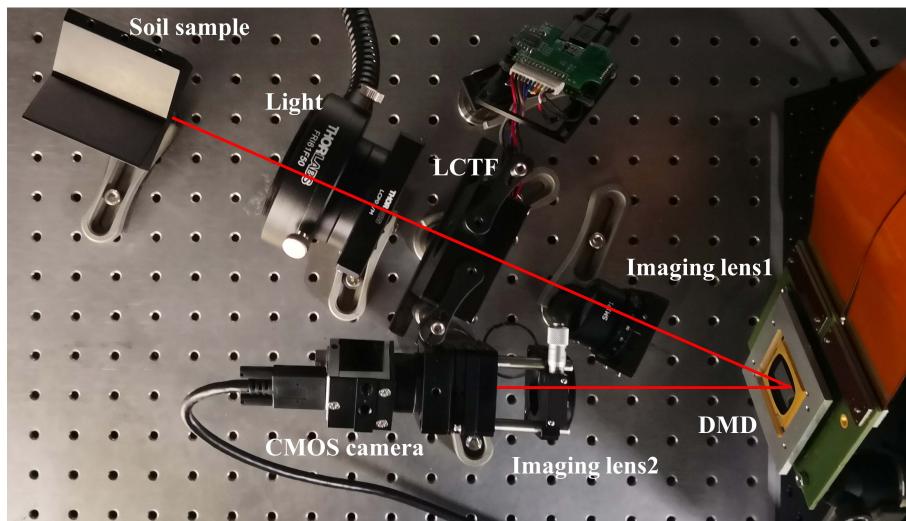


Fig. 6. The imaging system in laboratory.

### 5.2. Reconstruction results

The soil reconstructed hyperspectral images are obtained by CS theory. Fig. 7 shows the reconstructed hyperspectral images of the five soil types at the wavelength 698.63nm. The five

representative pixels are located at O1, O2, O3, O4 and O5 in the same position of the five soil types. In order to demonstrate the soil spectral characteristics at different wavelengths, Fig. 8 shows the average spectra of the five soil types in the range of 500-710 nm. Owing to differences in soil color, physical properties and other characteristics, different soil types have diverse spectral characteristics in the range of 500-710 nm, which provides a theoretical basis for soil classification.

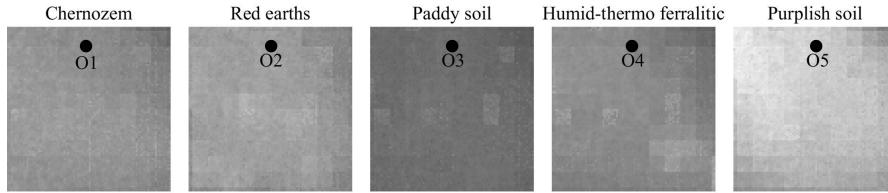


Fig. 7. The reconstructed hyperspectral images of the five soil types at the wavelength 698.63nm.

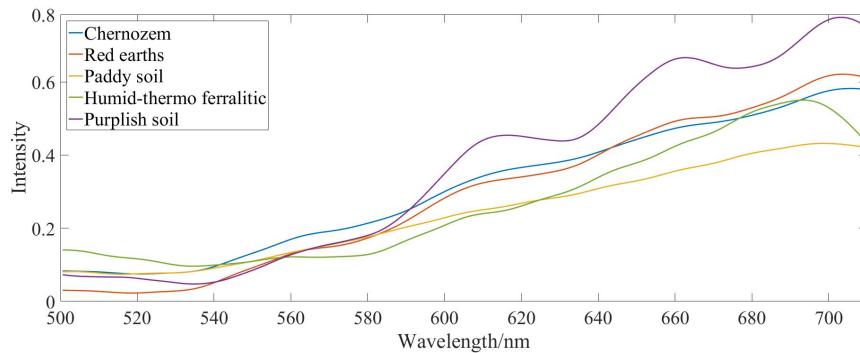


Fig. 8. The average spectra of the five soil types in the range of 500-710 nm.

The compressive measurements acquisition time and hyperspectral images reconstruction time influence the efficiency of soil classification. Since our classification is based on pixel-level classification, a large number of training dataset pixels are obtained by a single acquisition. In the case of using the automatic control module for acquisition, this system is capable of obtaining a large number of training dataset pixels rapidly. For each soil sample, the acquisition time of all compressive measurements using 20 different coded aperture patterns is 176 seconds, and the reconstruction computation time is 229 seconds based on an ordinary computer (Intel Core i5-7400 CPU and 8.00 GB RAM).

### 5.3. Classification results

All CNN-based algorithms are implemented based on Matconvnet[25], which lays emphasis on simplicity and flexibility. The parameter  $w$  is set to 5 to take  $5 \times 5$  spatial neighborhoods of each pixel into consideration in the 2D-CNN and 3D-CNN framework. The parameter  $s$  is set to 14 to preserve 99.9% of original information in the PCA dimensionality reduction method. For a fair comparison, the 3D-CNN algorithm network structure is consistent with the 1D-CNN algorithm in the spectral domain and consistent with the 2D-CNN algorithm in the spatial domain. Furthermore, the structure of each parallel network in the 3D-CNN-SD algorithm is consistent with the 3D-CNN algorithm network structure. In addition, the 3D-CNN-SD-PCA

algorithm network structure is similar to the 3D-CNN algorithm, except that the size of the three convolutional kernels in the spectral domain is different due to the different input layer sizes of these two algorithms. The number of feature maps generated by three convolutional layers is 20, 50, 500 respectively in all CNN-based algorithms. All algorithms are trained under the same training parameters with the learning rate of 0.0001 and the mini-batch size of 100. The stochastic gradient descent (SGD) algorithm with back propagation is applied to minimize the loss function in all algorithms. The training is stopped when the loss function finally converges after certain training epochs. In order to analyze the classification results deeply, we also carry out differential perception model experiments in the 1D-CNN and 2D-CNN frameworks, which are abbreviated to 1D-CNN-SD, 1D-CNN-SD-PCA, 2D-CNN-SD and 2D-CNN-SD-PCA respectively.

Fig. 9 shows the classification maps obtained by different CNN-based algorithms. The overall accuracy(OA), average accuracy(AA) and kappa coefficient( $\kappa$ ) are usually utilized to measure the classification accuracy in hyperspectral classification. However, the number of test dataset pixels for different soil types is same in our experiment. In that case, we only need to utilize OA as an evaluation metric to measure the classification accuracy. In order to further evaluate classifier performance, the area under curve(AUC) and Log loss are also used to quantitatively measure the classification performance. The AUC represents the area under the receiver operating characteristic(ROC) curve, which is commonly used to evaluate the binary classifier. Hand *et.al* [38] proposed a simple generalization of AUC to multiple classes. The Log loss is used to evaluate the probability output of the classifier. Fig. 10 shows the test dataset evaluation metrics as a function of the number of parameters. The algorithms based on the 2D-CNN framework improve the classification performance than the algorithms based on the 1D-CNN framework. In addition, the algorithms based on the 3D-CNN framework provide a boost in classification performance than the algorithms based on the 2D-CNN framework. Therefore, the simultaneous extraction of spatial and spectral features is more effective in soil classification. It is obvious that the 3D-CNN-SD-PCA algorithm performs against other CNN-based algorithms in terms of OA, AUC and Log loss. The algorithms based on the 3D-CNN framework increase the number of network parameters due to the increase of spatial information. The 3D-CNN-SD-PCA algorithm has the best classification performance and the least number of parameters in the algorithms based on the 3D-CNN framework, which confirms the superiority of this algorithm.

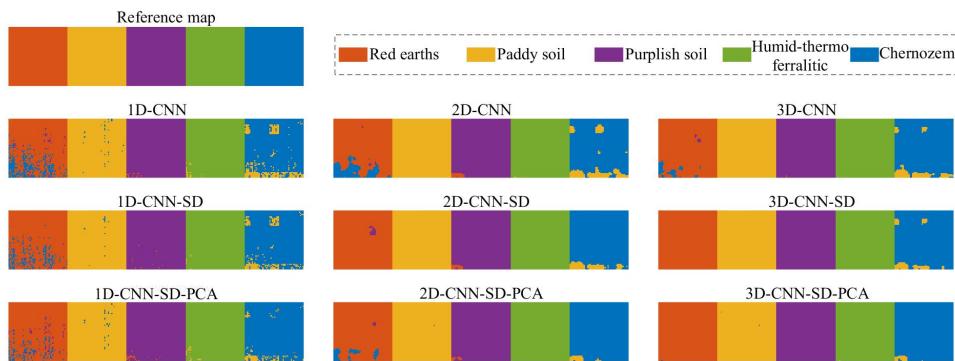


Fig. 9. The classification maps obtained by different CNN-based algorithms.

Further, we analyze the classification results. The 3D-CNN algorithm is superior to the 1D-CNN and 2D-CNN algorithms, because it takes full advantage of the three-dimensional data cube characteristics. In the following analysis, we analyze the single pixel spectral curve of five soil types using the five representative pixels O1, O2, O3, O4 and O5. The 3D-CNN-SD algorithm improves the classification performance compared with the 3D-CNN algorithm. Based

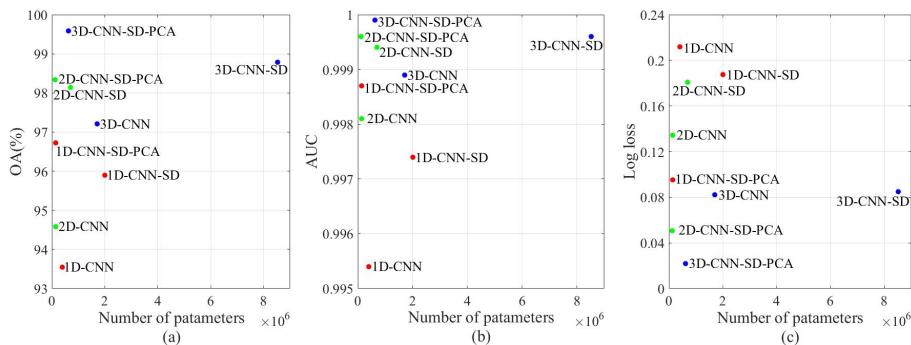


Fig. 10. The test dataset evaluation metrics as a function of the number of parameters:(a)OA,(b)AUC and (c)Log loss.

on the differential perception model, the 3D-CNN-SD algorithm flexibly extracts features in the spectral domain and uses five parallel networks for joint classification. Fig. 11(a) presents the spectral curve of a single pixel after mean subtraction pre-processing in the 3D-CNN algorithm. Figs. 11(b)-11(f) present the spectral difference curve between a single pixel with five soil types respectively in the 3D-CNN-SD algorithm. It is observed that the spectral difference curve trend between a single pixel with different soil types is diverse. Each parallel network focuses on the spectral difference with a specific soil type, joint classification ultimately achieves higher classification performance. Adding the PCA dimensionality reduction method significantly improves the classification performance. Hyperspectral data is highly correlated in adjacent bands, the PCA method reduces spectral information redundancy and maintains original information to the greatest extent. Fig. 12 presents the spectral principal component difference curve between a single pixel with five soil types in the 3D-CNN-SD-PCA algorithm. The horizontal coordinate indexes principal component bands number. The principal component score represents the value after dimensionality reduction. The vertical coordinate represents the difference between a single pixel principal component score with different soil types reference principal component score. According to the analysis of Fig. 12, the first several spectral principal component differences of O1, O2, O3, O4 and O5 are completely diverse, especially the first spectral principal component difference. Since the theoretical principle of PCA is that the variance between samples is largest in low-dimensional space, features are more discriminative in low-dimensional space. The increase in feature discrimination between different soil types is more beneficial to feature extraction in convolution and pooling operations. This algorithm has the advantage of reducing the over-fitting problem, thus improving the generalization ability of the model. Furthermore, this algorithm adds benefit of simple network structure, which reduces the network complexity in contrast to the 3D-CNN-SD algorithm. Moreover, this algorithm is capable of increasing the computational efficiency. In conclusion, the 3D-CNN-SD-PCA algorithm is optimal compared with other CNN-based algorithms. The SD-PCA method also improves the classification performance in the 1D-CNN and 2D-CNN frameworks, which further shows the superiority of this method.

Accordingly, we calculate the computation time for these algorithms based on the same computer as the reconstruction algorithm. Table 1 summarizes the training time and testing time of nine CNN-based algorithms. The input layer of the 3D-CNN algorithm is three-dimensional, thus increasing the computational cost of network layers such as convolutional layers and pooling layers. Therefore, the computation time is longer than 1D-CNN and 2D-CNN algorithms. The 3D-CNN-SD algorithm utilizes five parallel networks for joint classification. An increase in the number of features leads to an increase in computation time. The introduction of PCA dimensionality reduction method greatly improves computational efficiency. The testing time

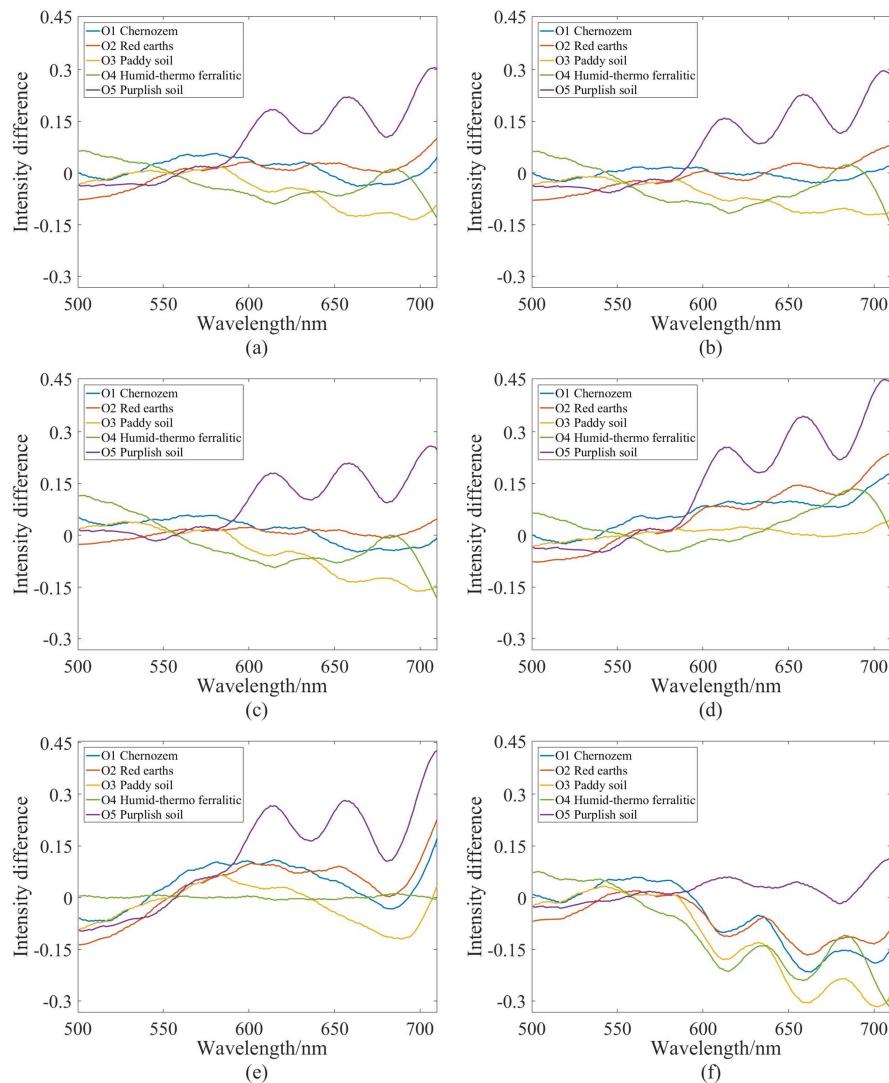


Fig. 11. The spectral difference curve between a single pixel with (a)all soil types,(b) Chernozem ,(c) Red earths, (d) Paddy soil,(e) Humid-thermo ferralitic and (f) Purplish soil.

with 12500 test dataset pixels in the 3D-CNN-SD-PCA algorithm is 55.89s, which promises its application in rapid soil classification. In summary, the 3D-CNN-SD-PCA algorithm is optimal in terms of classification performance, and the computation time is fastest in the algorithm based on the 3D-CNN framework. Therefore, we employ the 3D-CNN-SD-PCA algorithm for soil classification in practical applications.

Each soil type is imaged independently in the experiment mentioned above. In order to further demonstrate its practical application capabilities, we place different soil types at different locations during testing process. Fig. 13 shows the classification maps obtained by 3D-CNN-SD-PCA algorithm. It is seen that 3D-CNN-SD-PCA algorithm can achieve soil mapping in practical applications. Experiments demonstrate the feasibility of compressive spectral imaging system for soil classification rapidly and accurately using few compressive measurements.

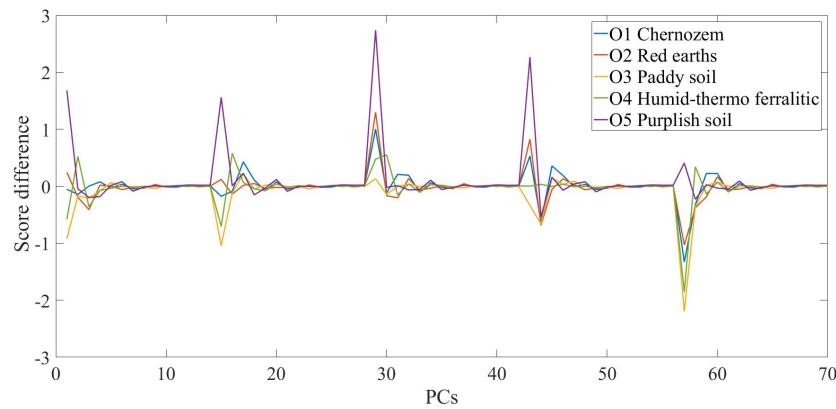


Fig. 12. The spectral principal component difference curve between a single pixel with five soil types in the 3D-CNN-SD-PCA algorithm.

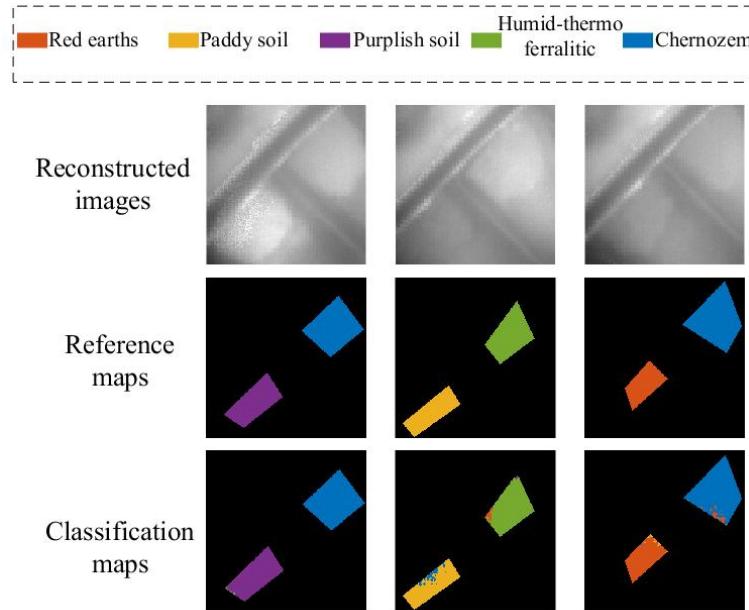


Fig. 13. (Top)the reconstructed spectral images,(middle)the reference maps, and(bottom)the classification maps generated by 3D-CNN-SD-PCA algorithm.

#### 5.4. Compare with conventional methods

In order to illustrate the comparative advantage of our algorithm, we compare the 3D-CNN-SD-PCA algorithm with other four conventional methods. The SAM, DT, SVM and KNN methods are also conducted to classify five soil types. The caption of Fig. 14 shows the details of each method. The classifier input feature can be divided into three categories: the full spectrum(F), the spectral principal components(P) and the spectral principal components and spatial texture features(PT). The GLCM method is the most effective method in spatial texture feature extraction. The direction of  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  and  $135^\circ$  and distance of one pixel are utilized in creating GLCM.

Table 1. The training time and testing time of nine CNN-based algorithms.

Method	1D-CNN	1D-CNN -SD	1D-CNN -SD-PCA	2D-CNN	2D-CNN -SD	2D-CNN -SD-PCA	3D-CNN	3D-CNN -SD	3D-CNN -SD-PCA
Training time (s/epoch)	14.14	68.76	7.14	22.48	112.5	11.47	1399.42	6920.89	249.11
Testing time(s)	8.01	44.56	6.42	10.51	51.35	9.3	263.23	1297.79	55.89

Four texture variables(energy, contrast, homogeneity and entropy) are extracted in each direction. The mean texture variables in four directions at the spectral principal components are used as spatial texture features. It is obvious that the 3D-CNN-SD-PCA algorithm outperforms all the other conventional methods, which can be seen from the histogram in Fig. 14. The SAM method is no need for a training model, but due to the similarity of different soil types spectra, it is impossible to obtain high classification accuracy relying on spectral angle alone. The DT method is simple to understand, however it may create over-complex trees that fail to generalize the data well. Apart from that, considering the spatial features reduces the classification accuracy of the DT method, which is also the disadvantage of this method. The SVM method can solve high dimensional and non-linear problems, however the classical SVM is a binary classifier. Therefore, multi-class classification requires a combination of multiple binary classifiers. The KNN method has better performance in multi-class classification problems and is insensitive to outliers. However, this method has high computational cost during testing. The most remarkable advantage of 3D-CNN is feature extraction in contrast to conventional methods, which can automatically and simultaneously extract spatial and spectral features. The proposed algorithm which combines 3D-CNN with differential perception model and PCA dimensionality reduction achieves superior classification performance.

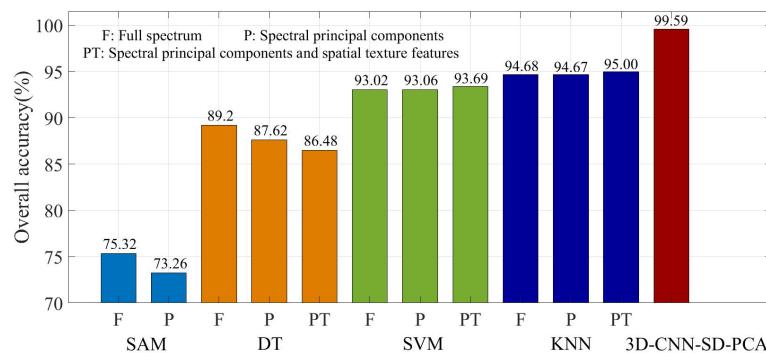


Fig. 14. Classification performance comparison between the 3D-CNN-SD-PCA algorithm with other conventional methods: SAM: the reference spectrum is each soil type mean spectral vector; SVM: the kernal is radial basis function(RBF), multi-class method is one to one, the penalty parameter and the kernel function parameter are obtained by grid search; DT: the Gini index is used to split measure in selecting the splitting attribute; KNN: the distance metric is euclidean distance, the number of neighbors is determined by cross-validation.

Accordingly, we compare the computation time of the proposed algorithm with the computation time of conventional methods based on joint spectral-spatial feature. Table 2 summarizes the training time and testing time for different machine learning classification methods. The KNN method is no need for a training model, therefore the training time represents the feature extraction time. Because of the need to train multiple epochs in the 3D-CNN-SD-PCA algorithm, the

**Table 2. The training time and testing time for different machine learning classification methods.**

Method	DT (PT)	SVM (PT)	KNN (PT)	3D-CNN -SD-PCA
Training time	983.8s	982.51s	972.96s	249.11s/epoch
Testing time	385.48s	388.14s	398.25s	55.89s

training time of the proposed algorithm is relatively long compared with conventional machine learning classification methods. However, we focus on the testing time in practical applications. It is noted that the testing time of DT, SVM and KNN classification methods based on joint spectral-spatial feature is 385.48s, 388.14s and 398.25s respectively, while the testing time of the 3D-CNN-SD-PCA algorithm is 55.89s. The proposed algorithm significantly accelerates the testing speed compared with other joint spectral-spatial feature classification methods, which is more competitive in practical applications.

## 6. Conclusion and discussion

In conclusion, we explore a LCTF-based compressive spectral imaging system and propose a 3D-CNN for soil classification. According to CS method, soil hyperspectral images are reconstructed with improved resolution in spatial as well as spectral domains. As a result, high resolution spectral data provides the potential to obtain more reflection intensity information about different soil types. Furthermore, the spatial resolution is improved to achieve a more refined soil classification. We apply the PCA method to reduce information redundancy in the spectral domain which has the benefit of increasing feature discrimination and improving computational efficiency. Then, we develop a differential perception model for flexible feature extraction, and finally introduce a 3D-CNN framework to classify each pixel soil type of the reconstructed hyperspectral images. Experimental results show that 3D-CNN combined with differential perception model and PCA dimensionality reduction has better performance in contrast to other CNN-based and conventional algorithms. Compressive spectral imaging system is capable of achieving soil classification effectively and accurately with potential benefits of reducing system size and acquisition cost.

As shown in this paper, it is practical to use the compressive spectral imaging system for soil classification. It would be a future work to apply the proposed algorithm to soil classification based on hyperspectral satellite observations. In addition, we hope to develop a novel soil classification approach with LCTF-based compressive imaging system from compressive measurements, without first reconstructing the hyperspectral data cube.

## Funding

National Natural Science Foundation of China (NSFC) (61527802, 61371132, 61471043).

## References

1. Z. Shi, Q. L. Wang, J. Peng, W. J. Ji, H. J. Liu, X. Li, and R. A. Viscarra Rossel, "Development of a national VNIR soil-spectral library for soil classification and prediction of organic matter concentrations," *Sci. China Earth Sci.* **57**, 1671–1680 (2014).
2. J. Tian and W. D. Philpot, "Soil directional (biconical) reflectance in the principal plane with varied illumination angle under dry and saturated conditions," *Opt. Express* **26**, 23883 (2018).
3. J. Li, W. Huang, X. Tian, C. Wang, S. Fan, and C. Zhao, "Fast detection and visualization of early decay in citrus using Vis-NIR hyperspectral imaging," *Comput. Electron. Agric.* **127**, 582–592 (2016).

4. B. Zhang, J. Li, S. Fan, W. Huang, C. Zhao, C. Liu, and D. Huang, "Hyperspectral imaging combined with multivariate analysis and band math for detection of common defects on peaches (*Prunus persica*)," *Comput. Electron. Agric.* **114**, 14–24 (2015).
5. Z. Lee, S. Shang, G. Lin, J. Chen, and D. Doxaran, "On the modeling of hyperspectral remote-sensing reflectance of high-sediment-load waters in the visible to shortwave-infrared domain," *Appl. Opt.* **55**, 1738 (2016).
6. D. R. Thompson, J. W. Boardman, M. L. Eastwood, and R. O. Green, "A large airborne survey of Earth's visible-infrared spectral dimensionality," *Opt. Express* **25**, 9186 (2017).
7. D. L. Donoho, "Compressed sensing," *IEEE Transactions Inf. Theory* **52**(4), 1289–1306 (2006).
8. W. Feng, H. Rueda, C. Fu, G. R. Arce, W. He, and Q. Chen, "3D compressive spectral integral imaging," *Opt. Express* **24**, 24859 (2016).
9. H. Arguello and G. Arce, "Code Aperture Agile Spectral Imaging (CAASI)," in *Imaging Systems Applications*, OSA Technical Digest (CD) (Optical Society of America, 2011), paper ITuA4.
10. Y. Wu, I. O. Mirza, G. R. Arce, and D. W. Prather, "Development of a digital-micromirror-device-based multishot snapshot spectral imaging system," *Opt. Lett.* **36**, 2692 (2011).
11. N. R. Rao, P. K. Garg, and S. K. Ghosh, "Development of an agricultural crops spectral library and classification of crops at cultivar level using hyperspectral data," *Precis. Agric.* **8**, 173–185 (2007).
12. D. Ramakrishnan and R. Bharti, "Hyperspectral remote sensing and geological applications," *Curr. Sci.* **108**, 879–891 (2015).
13. O. E. Adedipe, B. Dawson-Andoh A, J. Slahor, and L. Osborn A, "Classification of red oak (*Quercus rubra*) and white oak (*Quercus alba*) wood using a near infrared spectrometer and soft independent modelling of class analogies," *J. Near Infrared Spectrosc.* **16**, 49–57 (2008).
14. J. Zhao, Q. Chen, J. Cai, and Q. Ouyang, "Automated tea quality classification by hyperspectral imaging," *Appl. Opt.* **48**, 3557 (2009).
15. C. Yang, J. H. Everitt, and J. M. Bradford, "Yield Estimation from Hyperspectral Imagery Using Spectral Angle Mapper (SAM)," *Transactions ASABE* **51**, 729–737 (2013).
16. F. van der Meer, "The effectiveness of spectral similarity measures for the analysis of hyperspectral imagery," *Int. J. Appl. Earth Obs. Geoinformation* **8**, 3–17 (2006).
17. K. Huang, S. Li, X. Kang, and L. Fang, "Spectral-Spatial Hyperspectral Image Classification Based on KNN," *Sens. Imaging* **17**, 1–13 (2016).
18. V. M. Dolas and P. U. Joshi, "A Novel Approach for Classification of Soil and Crop Prediction," *Int. J. Comput. Sci. Mob. Comput.* **7**, 20–24 (2018).
19. G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogramm. Remote. Sens.* **66**, 247–259 (2011).
20. M. Pal and G. M. Foody, "Feature selection for classification of hyperspectral data by SVM," *IEEE Transactions on Geosci. Remote. Sens.* **48**, 2297–2307 (2010).
21. E. C. Brevik, C. Calzolari, B. A. Miller, P. Pereira, C. Kabala, A. Baumgarten, and A. Jordán, "Soil mapping, classification, and pedologic modeling: History and future directions," *Geoderma* **264**, 256–274 (2016).
22. Y. Ogen, N. Goldshleger, and E. Ben-Dor, "3D spectral analysis in the VNIR-SWIR spectral region as a tool for soil classification," *Geoderma* **302**, 100–110 (2017).
23. M. Steffens and H. Buddenbaum, "Laboratory imaging spectroscopy of a stagnic Luvisol profile - High resolution soil characterisation, classification and mapping of elemental concentrations," *Geoderma* **195–196**, 122–132 (2013).
24. A. D. Vibhute, K. V. Kale, R. K. Dhumal, and S. C. Mehrotra, "Soil type classification and mapping using hyperspectral remote sensing data," in *International Conference on Man and Machine Interfacing*, (IEEE, 2016), pp.1–4.
25. S. Jia, H. Li, Y. Wang, R. Tong, and Q. Li, "Hyperspectral imaging analysis for the classification of soil types and the determination of soil total nitrogen," *Sensors* **17**, 1–14 (2017).
26. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Neural Information Processing Systems*, (Academic, 2012), pp.1097–1105.
27. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (IEEE, 2015), pp.1–9.
28. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (IEEE, 2016), pp.770–778.
29. W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep Convolutional Neural Networks for Hyperspectral Image Classification," *J. Sensors* **2015**, 258619 (2015).
30. J. Yue, W. Zhao, S. Mao, and H. Liu, "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks," *Remote. Sens. Lett.* **6**(6), 468–477 (2015).
31. K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doula, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *IEEE International Geoscience and Remote Sensing Symposium*, (IEEE, 2015), pp. 4959–4962.
32. H. Liang and Q. Li, "Hyperspectral imagery classification using sparse representations of convolutional neural network features," *Remote. Sens.* **8**, 99 (2016).
33. Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep Feature Extraction and Classification of Hyperspectral Images

- Based on Convolutional Neural Networks," IEEE Transactions on Geosci. Remote. Sens. **54**, 6232–6251 (2016).
- 34. Y. Li, H. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network," Remote. Sens. **9**, 67 (2017).
  - 35. X. Wang, Y. Zhang, X. Ma, T. Xu, and G. R. Arce, "Compressive spectral imaging system based on liquid crystal tunable filter," Opt. Express **26**, 25226 (2018).
  - 36. S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional neural networks for human action recognition," IEEE Transactions on Pattern Analysis Mach. Intell. **35**, 221–231 (2013).
  - 37. A. A. Gowen, C. P. O. Donnell, M. Taghizadeh, P. J. Cullenb, and J. M. Frias, and G. Downey, "Hyperspectral imaging combined with principal component analysis for bruise damage detection on white mushrooms (*Agaricus bisporus*)," J. Chemom. **22**, 259-267 (2008).
  - 38. D. J. Hand and R. J. Till, "A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems," Mach. Learn. **45**, 171–186 (2001).