Article

Multi-Dataset Hyper-CNN for Hyperspectral Image Segmentation of Remote Sensing Images

Li Liu [1], **Emad Mahrous Awwad** [2,*], **Yasser A. Ali** [3], **Muna Al-Razgan** [4], **Ali Maarouf** [2], **Laith Abualigah** [5,6,7,8,9] **and Azadeh Noori Hoshyar** [10]

1 Chongqing Creation Vocational College, Chongqing 402160, China
2 Electrical Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia
3 Department of Information Systems, College of Computer and Information Sciences, King Saud University, P.O. Box 51178, Riyadh 11543, Saudi Arabia
4 Department of Software Engineering, College of Computer and Information Sciences, King Saud University, P.O. Box 22452, Riyadh 11495, Saudi Arabia
5 Computer Science Department, Prince Hussein Bin Abdullah Faculty for Information Technology, Al Al-Bayt University, Mafraq 25113, Jordan
6 Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan
7 Faculty of Information Technology, Middle East University, Amman 11831, Jordan
8 Applied Science Research Center, Applied Science Private University, Amman 11931, Jordan
9 School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang 11800, Malaysia
10 Institute of Innovation, Science and Sustainability, Federation University Australia, Brisbane, QLD 4000, Australia
* Correspondence: 442106835@student.ksu.edu.sa

Abstract: This research paper presents novel condensed CNN architecture for the recognition of multispectral images, which has been developed to address the lack of attention paid to neural network designs for multispectral and hyperspectral photography in comparison to RGB photographs. The proposed architecture is able to recognize 10-band multispectral images and has fewer parameters than popular deep designs, such as ResNet and DenseNet, thanks to recent advancements in more efficient smaller CNNs. The proposed architecture is trained from scratch, and it outperforms a comparable network that was trained on RGB images in terms of accuracy and efficiency. The study also demonstrates the use of a Bayesian variant of CNN architecture to show that a network able to process multispectral information greatly reduces the uncertainty associated with class predictions in comparison to standard RGB images. The results of the study are demonstrated by comparing the accuracy of the network's predictions to the images.

Keywords: long short-term memory (LSTM); 3D-CNN; hyperspectral image segmentation (HSI)

1. Introduction

Hyperspectral remote sensing has matured into a trustworthy instrument for Earth observations in recent years [1]. Because HSIs can collect so much information in both the spectral and spatial domains, they have found use in many different fields [2]. These fields include agriculture, geology, food science, and even military target reconnaissance. The classification of hyperspectral images will have profound effects on the aforementioned areas of study. Improved hyperspectral imaging methods have made high-resolution HSIs available to the public, making it simpler for researchers to push the state of the art in HSI segmentation forward [2–5].

Several methods for recognizing and segmenting hyperspectral images have implemented standard approaches from the field of machine learning [2]. Several methods, including kernelized support vector machines (k-SVMs), Markov random fields (MRFs), sparse representation (SR), morphological transformations (MTs), and composite kernels or

spatial–spectral kernels [2,6,7], can be used to classify or segment images by combining the spectral and spatial information carried by the images. You can find tools to help with the categorization and division of datasets. Traditional methods have worked well and produced satisfying results; however, improving their classification performance has been challenging due to a lack of knowledge on how to best utilize the rich feature information of hyperspectral images.

Traditionally, hyperspectral image analyses have relied on methods such as principal component analysis (PCA) and linear discriminant analysis (LDA) to extract features from the images. These methods are based on linear algebra and are effective at reducing the dimensionality of the data, but they can fail to capture the non-linear relationships and complex patterns present in hyperspectral images. Additionally, traditional methods often involve the manual selection of features, which can be time-consuming and subjective. This can lead to the inclusion of irrelevant features and the exclusion of important ones, which can negatively impact the performance of the analysis. Furthermore, traditional methods, such as PCA and LDA, are not able to learn features from the data in the way that deep-learning models, such as CNNs, can. CNNs are able to automatically learn features from the images, which can lead to more accurate and robust results. This is particularly important in hyperspectral image analyses where the data is high-dimensional and complex. In summary, traditional methods for hyperspectral image analyses, such as PCA and LDA, are not able to fully exploit the rich feature information present in the data, which can limit the performance of analyses. Deep-learning models, such as CNNs, are better suited for this task as they can automatically learn features from the data, which can lead to more accurate and robust results [8–10]. This is especially true of deep learning, a field where technological capabilities have grown rapidly in recent years. Its ability to autonomously extract rich deep features is the key to its outstanding performance in image/video processing, speech recognition, and other domains. Natural language processing and understanding are two additional subfields. The use of deep-learning techniques in the evaluation of hyperspectral images is also being considered by researchers [11–13]. Convolutional neural networks, also known as CNNs, have shown to be especially useful in the field of computer vision. Conventional neural network architecture (CNN) has stood the test of time in the realm of deep learning. Over the past few years, CNNs have established themselves as reliable classification aids for HSI probes. Using CNNs that are constructed on two-dimensional convolution yields significantly more accurate results than more traditional machine-learning-based HSI classification techniques [10,14–16].

Several factors make hyperspectral image classification a difficult process, including overlap and layering, as well as high degrees of similarity and diversity among classes. Because spectral and spatial information are both necessary for HSI, two-dimensional convolutional neural networks are a useful classification method. Although 3D-CNNs can be computationally intensive, they appear to be a viable option for enhancing HSI's accuracy. This is due to the fact that HSI is multi-dimensional in both space and time. Furthermore, these models may fail to extract quality feature maps from areas with similar textures. The quality of the data presents one of the greatest obstacles for HSI. Refs. [17,18] in particular, hyperspectral data provide access to thousands of high-resolution spectral bands across the electromagnetic spectrum. The "curse of dimensionality" can occur when this situation is joined with a scarcity of properly labelled training data. This happens when there are more spectral bands in the data than there are labelled training samples. Because of this, supervised and semi-supervised learning approaches to HSI classification (HSI) have average or below-average predictive performance [19].

Either decreasing the number of dimensions or amassing a large number of labelled training samples would be beneficial, but this would be impractically time-consuming and costly [20]. Important geographical information associated with HSI may be lost in the dimensionality reduction process. The goal of this research is to find a computationally efficient solution to the aforementioned HSI problem that does not require a massive

amount of labelled training samples or the elimination of potentially useful information. Additionally, HSI cannot afford to lose any crucial information in the process of fixing this issue.

In this study, we use deep learning to classify hyperspectral images in a novel way. To review, a pixel's predicted class label is influenced by its spectral and spatial contexts. When scanning a pixel with different spectra, and one obtains spectral values that correspond to the label given to that pixel. Considering neighboring pixels' class labels is crucial for predicting a single pixel's class label [2,4,21,22]. Therefore, a good spectral factor and spatial factor should be considered in a classification method for hyperspectral images [22–24]. After praising the virtues of a 2D-CNN model for hyperspectral image classification, we turn to a 3D-CNN model. The idea behind a 3D-CNN is that it can also gain from the spectral context, which a 2D-CNN lacks [23,24]. In contrast, a 2D-CNN has only the spatial context in which to learn [25,26]. However, there are limitations on how much relevant data can be incorporated into the aforementioned models. As a result, we created a recurrent 2D-CNN classification model and a recurrent 3D-CNN classification model to deal with the problem of noisy spatial information [23–27].

We compared spectral data to the channels of conventional photography to show how it could be used. Hyperspectral images are classified by removing a tiny square from the center of each pixel. Like a multi-channel image, the patch is processed in the same way. After that, we used three 2D convolution layers and a full connection layer to create a 2D-CNN model for patch classification. The label of the pixel in the patch's geographic center is used as the patch's overall label. Maximum pooling layers and average pooling layers are two examples of pooling layers that can be used to reduce the dimensionality of feature maps and speed up the computation process. However, the classification performance of the network may be impacted by the use of pooling layers [28,29].

Because we want to preserve as much of the useful context information as possible in our 2D-CNN model, we opted to not use pooling layers. In this section, we discuss the various layers that make up a CNN, such as the convolution layer, the pooling layer, and the fully connected layer.

The 2D-CNN model can make use of spatial context, but it ignores spectral correlations. For this reason, we came up with a model of a three-dimensional convolutional neural network with seven convolutional layers and a single fully connected node layer to solve the problem. This model employs a 3D convolution operator in order to acquire knowledge about the spectral context in addition to the spatial context, in contrast to the 2D CNN's exclusive focus on the spatial context. The 3D-CNN model has the potential to be more effective than its 2D counterpart because of its ability to analyze the spectral correlations present within a hyperspectral image, despite having a larger number of network parameters. The 2D-CNN model only considers a small area cantered on the pixel when drawing conclusions about its categorization, so it is possible that the results are noisy because of this. The contributions of this study are as follows:

- We build a recurrent 2D-CNN model as a further step toward capitalizing on the spatial context (R-2DCNN). As the area of the patch gets smaller, the R-2D-CNN is able to focus more intently on the core pixel, which allows it to extract more meaningful information from it. Experiments show that the R-2D-CNN model performs significantly better than its predecessor, the 2D-CNN model.
- In order to solve the issue of noisy patches, we develop a spatially and spectrally aware recurrent 3D-CNN model, which we refer to as R-3DCNN. The R-3D-CNN model improves the functionality of the 3D-CNN architecture by reducing the patch size in an iterative manner. As a result, rather than relying on information about patches, the final classification of each pixel relies significantly on information about individual pixels. Experiments provide evidence that demonstrates that the R-3D-CNN model is superior to other models. Most notably, it offers the highest level of classification accuracy that is practical and converges at a faster rate than other methods.

- The proposed HSI classification approach outperforms state-of-the-art conventional and deep-learning-based HSI algorithms with less labelled samples, as shown by experiments conducted on the Indian Pines, Pavia University, and Salinas HSI benchmark datasets.

The paper is divided into five sections. Section 1 discusses the problems and contributions of this study. Section 2 discusses the related work, while Section 3 discusses the detailed methodology. Section 4 discusses the results, while Section 5 discusses the conclusions.

2. Literature Review

Hyperspectral imaging (HSI) can be used for numerous purposes. Noise, band correlations, and excessive dimensionality limit such data. ResNet, SSRN, and A2S2K are new deep-learning network topologies [2]. The last layer (the classification layer), a Softmax classifier, is not adjusted. Instead, a watershed classifier is recommended. Watershed classifiers outclasses mathematical morphology's typical watershed operator. This watershed classifier lacks trainable parameters. In this article, the authors present a novel way to train deep-learning networks to gain watershed-classifier-compatible representations. The watershed classifier exploits connection patterns to increase inference. The authors show that the triplet watershed may obtain state-of-the-art results in supervised and semi-supervised situations by using these properties. These results have been verified in datasets from IP, UP, KSC, and UH, all of which rely on a simple convent design with a quarter of the features of previous state-of-the-art networks (UH).

Zhang et al. [30] presented variable region-based CNN to encode semantic context-aware representation and create meaningful features. CNN-based representations are sensitive to spatial and spectral contexts that are required for effective pixel classification. The suggested method uses inputs that change by location to learn contextual interaction features, resulting in better discriminatory power. Rich spectral and spatial information is delivered to a softmax layer to predict pixel vector labels. The suggested method beats state-of-the-art classifiers and all existing deep-learning-based algorithms.

Gao et al. [31] used convolutional neural networks (CNNs) for hyperspectral image categorization (HSI). When trying to extract features from a dataset with few labelled samples and mixed pixels, overfitting the model becomes a problem. Improving the CNNs' extraction capabilities by increasing the model's depth and convolution kernel complexity is usual practice. This article suggests a spectral CNN for HSI (SFE-SCNN). SFE-SCNN uses spectral feature enhancement to make data reflect more discriminative spectral detail, reducing mixed-pixel interference. A lightweight sandwich convolution neural network considers the preprocessed data structure. First, re-extraction is utilized to fully extract spectral characteristics. The suggested technique enhances classification accuracy in three real-world hyperspectral datasets.

Hyperspectral pictures are unmatched at spotting items on the Earth's surface. Most item classifiers employ simply spectral data, ignoring geographical context. This research [4] uses a convolutional neural network to classify hyperspectral images by their spectral and spatial features (CNN). The hyperspectral image is patchy. CNN builds each patch's high-level spectral and spatial characteristics, and the multi-layer perceptron classifies features. Our simulations reveal that CNN classifies hyperspectral pictures most accurately.

Remote sensing researchers are studying the classification of hyperspectral images (HIC). Hyperspectral pictures create large data cubes that are difficult to acquire, store, transmit, and analyze. The authors of this study [7] present a deep-learning HIC approach that uses compressive measurements from coded-aperture snapshot spectrum imagers (CASSIs), rather than recreating the whole hyperspectral data cube. In this paper, we propose a 3D coded convolutional neural network (3D-CCNN) to efficiently tackle the classification problem by treating the hardware-based coded aperture as a pixel-wise linked network layer. A thorough training approach is designed to optimize the network parameters and periodic aperture topologies. Using deep learning and coded apertures, the

author improves classification accuracy. Many hyperspectral datasets are used to compare the recommended method against state-of-the-art HIC methods.

Liu et al. [32] described convolutional neural networks (CNNs) for dense, pixel-wise satellite image classification. We employ convolutional neural networks (CNNs) to classify input photos. The authors create convolutional architecture to solve dense classification. We propose a two-step training technique in which CNNs are first taught with a huge volume of reference data before being retrained with accurate labels. The author created a multi-scale neuron module that does not sacrifice recognition for localization. Experiments indicate that our networks use contextual information to build fine-grained categorization maps.

Kumar et al. [33] reviewed the research on deep-learning algorithms for HSI classification. The authors structured their literature analysis according to the five most common deep-learning models, summarizing the primary feature extraction strategies for each. This study may set guidelines for future research.

Gao et al. [34] suggested utilizing CNNs to categorize hyperspectral images (HSIs) due to their improved feature representation and performance. In this study, they used both convolutional neural networks (CNNs) and multiple feature learning to better predict HSI pixel class labels [29,30]. The CNN was trained using photo features [31,32]. The network feeds all feature maps it feels are needed to adequately represent the input into a concatenating layer, which outputs a single feature map [33,34]. The generated joint feature map is used to forecast hyperspectral pixel labels. The proposed approach takes advantage of CNNs' increased feature extraction and spectral and spatial information simultaneously [35,36]. A CNN-based multi-feature learning framework enhances classification accuracy in three benchmark datasets [37,38].

This research [39,40] uses four deep-learning models built on internal data, ignoring HSI data [41,42]. This type of data collecting via conveyor belts is prevalent in industrial settings. This study develops deep-learning-based HSI segmentation methods and provides methods to analyze line-scanner hyperspectral data [43,44]. Using deep-learning-trained hyperspectral imaging systems [45–47], semantic segmentation for automated food quality inspections may be possible. The results were validated using k-fold cross-validation [48–51]. After reviewing the previous studies, the following challenges may be drawn:

(a) Developing a robust and efficient method for automatic land cover classification using hyperspectral images. The problem addresses the need for a fast and accurate method for land cover classification, which is important for applications such as urban planning and natural resource management.

(b) Improving the performance of anomaly detection in hyperspectral images using deep-learning models. The problem addresses the challenge of detecting anomalies in high-dimensional and complex hyperspectral data, which is important for applications such as mineral exploration and environmental monitoring.

(c) Investigating the use of multi-dataset hyper-CNN for hyperspectral image segmentation of remote sensing images. The problem addresses the need for an effective method for segmentation of hyperspectral images, which is important for applications such as object detection and tracking in surveillance and monitoring.

(d) Investigating the use of 3D-CNN for hyperspectral image classification. The problem addresses the need for an efficient and accurate method for hyperspectral image classification, which is important for applications such as land cover mapping and mineral exploration. Table 1 displays a meta-analysis of prior state-of-the-art studies.

Table 1. Comparative Analysis.

Reference	Dataset	Techniques	Accuracy
Haut et al. [35]	PINE	VGG 16	89%
Lee et al. [36]	PAVIA	CNN	84.5%
Sharma et al. [1]	PINE	CNN + LSTM	83%
Li et al. [37]	PINE	ANN	82.3%
Hu et al. [38]	PAVIA	CNN	81%

3. Methodology

We present the proposed methodology in the form of a flowchart and describe the associated dataset. To classify hyperspectral images, we additionally detailed the unique structure of a 3D convolutional neural network (CNN). The research process is depicted in Figure 1:
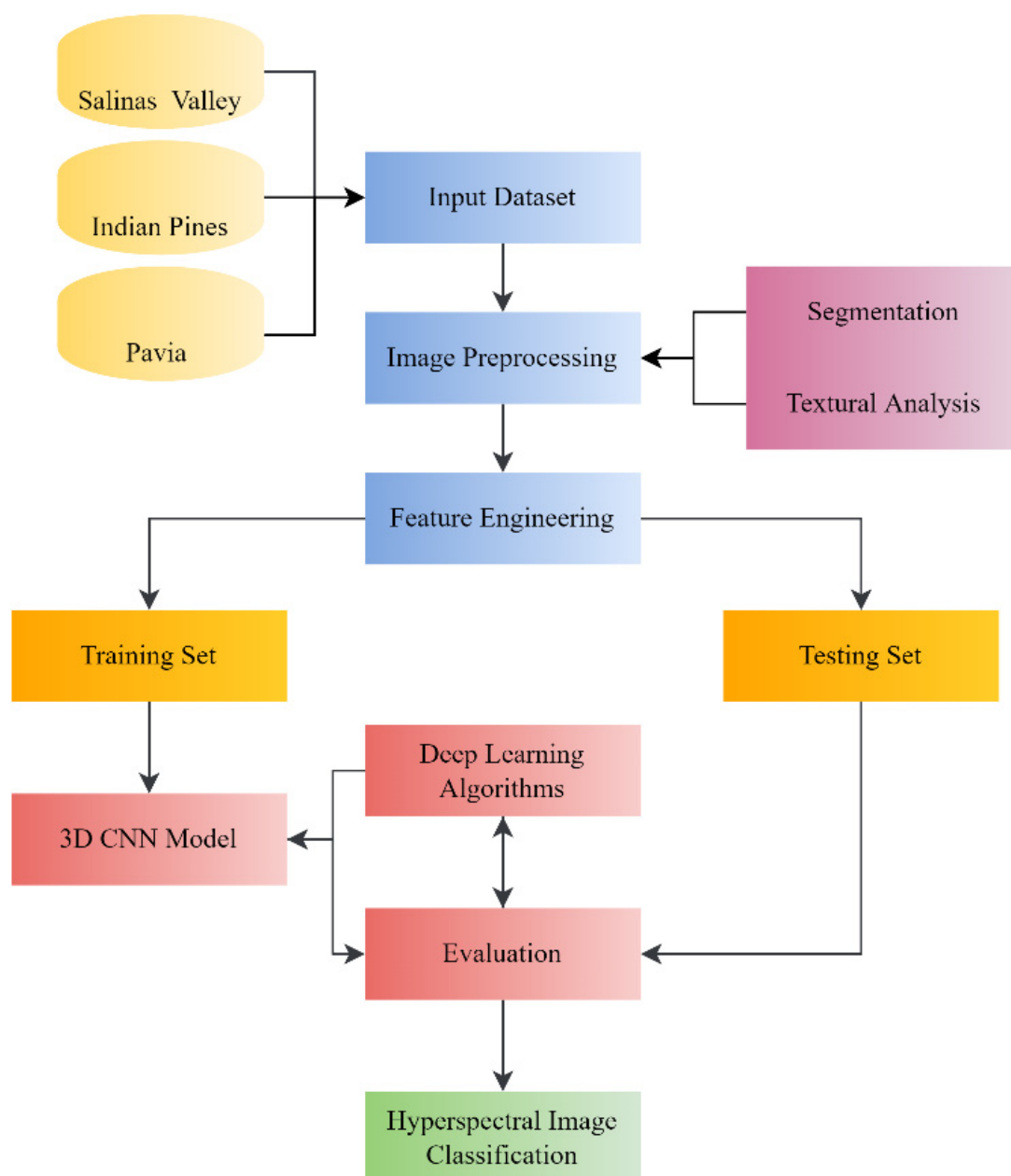
Figure 1. Proposed Workflow.

In this research, we used the proposed innovative architecture to conduct hyperspectral image classification on the following three datasets: Indian Pines; Salinas Valley; and Pavia.

3.1. Dataset Description

There are three datasets used in this investigation. There is a description and chart for each dataset shown below.

3.1.1. Indian Pines

The Indian Pines dataset (Kaggle.com, accessed on 13 August 2022) is used to practice segmentation techniques on hyperspectral images. A hyperspectral image of a single Indiana landscape (from the Indian Pines dataset) measures 145 × 145 pixels. There are 220 individual spectral reflectance bands, each corresponding to a specific wavelength in the electromagnetic spectrum, that are used to represent each pixel in the dataset. Training data for the Indian Pine dataset can be found in a numpy array and is freely accessible to anyone interested in using it. The dataset is very small (145 × 145 × 220) and serves as a fantastic introduction to hyperspectral remote sensing. Figure 2 displays some color-coded images from the Pines dataset.
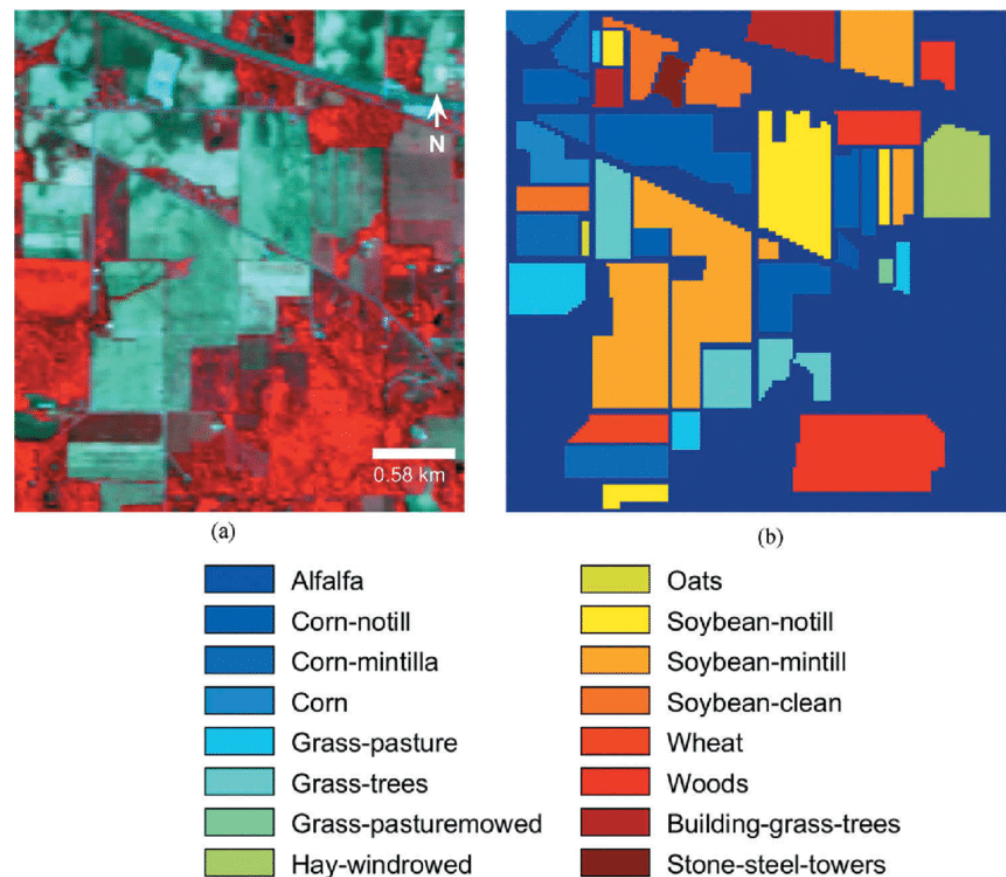

Figure 2. Classified Contrast of Indian Pines Dataset. (**a**) Segmented Image; (**b**) Detection.

3.1.2. Pavia Dataset

The images in the Pavia University dataset (Kaggle.com, accessed on 2 September 2022) were captured by a reflection optics system imaging spectrometer high above the Italian city of Pavia (ROSIS-3). With a resolution of 610 × 340 pixels, this image contains 115 individual color channels. The collection's 42,776 labelled samples are divided into nine categories. Asphalt, gravel, trees, metal sheet, bare soil, bitumen, brick, and shadow are all examples of these categories. Figure 3 shows the outcomes of the Pavia dataset's image classification.
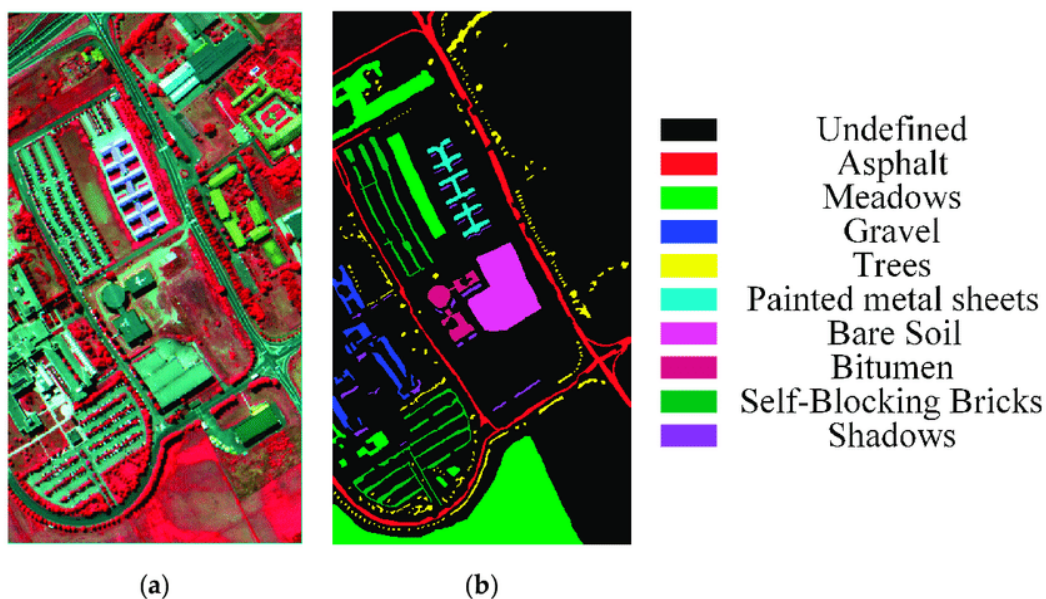
Figure 3. Pavia Hyperspectral Images Dataset (**a**) Segmented Image; (**b**) Ground truth labelled image.

3.1.3. Salinas Valley Dataset

The 224-band AVIRIS sensor in California's Salinas Valley (Kaggle.com, accessed on 5 October 2022) collected the data for the high-resolution hyperspectral Salinas Scene dataset (3.7-m pixels). Totaling 512 lines and 217 samples, the dataset is quite sizable. The water absorption ranges [108–112, 154–167] and [224] were disregarded. Only radiance data from the sensor were available, so this picture had to be sufficient. Plants, such as vegetables, weeds, and vining plants, are cultivated here as well. There are sixteen distinct varieties of Salinas ground truth architecture. An example of a correctly labelled image from the Salinas Valley dataset can be seen in Figure 4.
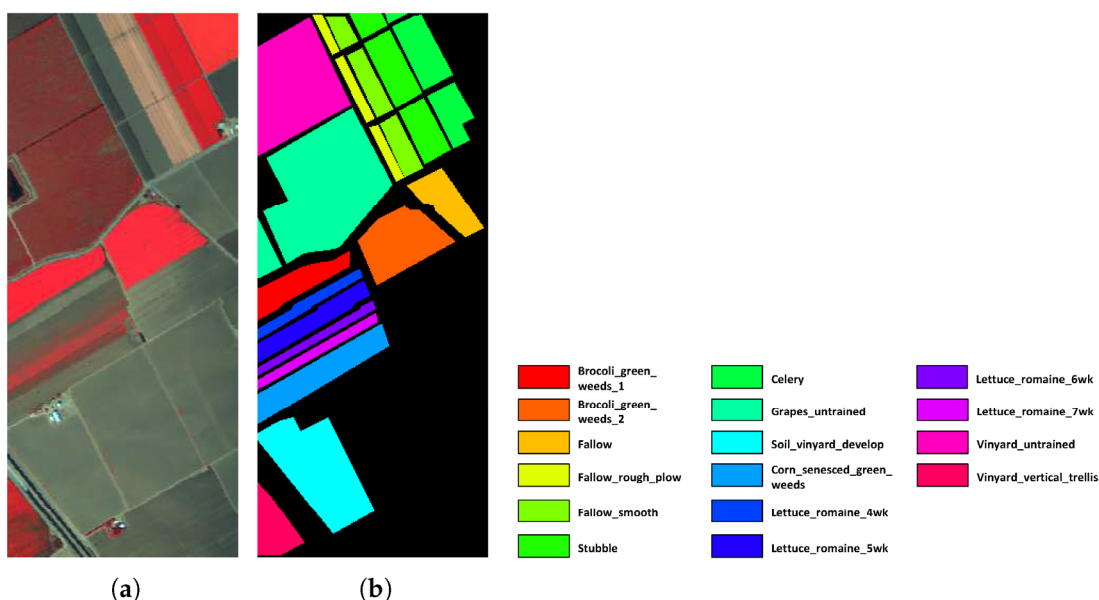

Figure 4. Images taken from Salinas Dataset: (**a**) Segmented Image; (**b**) Ground truth labelled image.

3.2. Architecture of 3D-CNN

Figure 5 shows the proposed novel architecture of 3D-CNN for hyperspectral image classification.
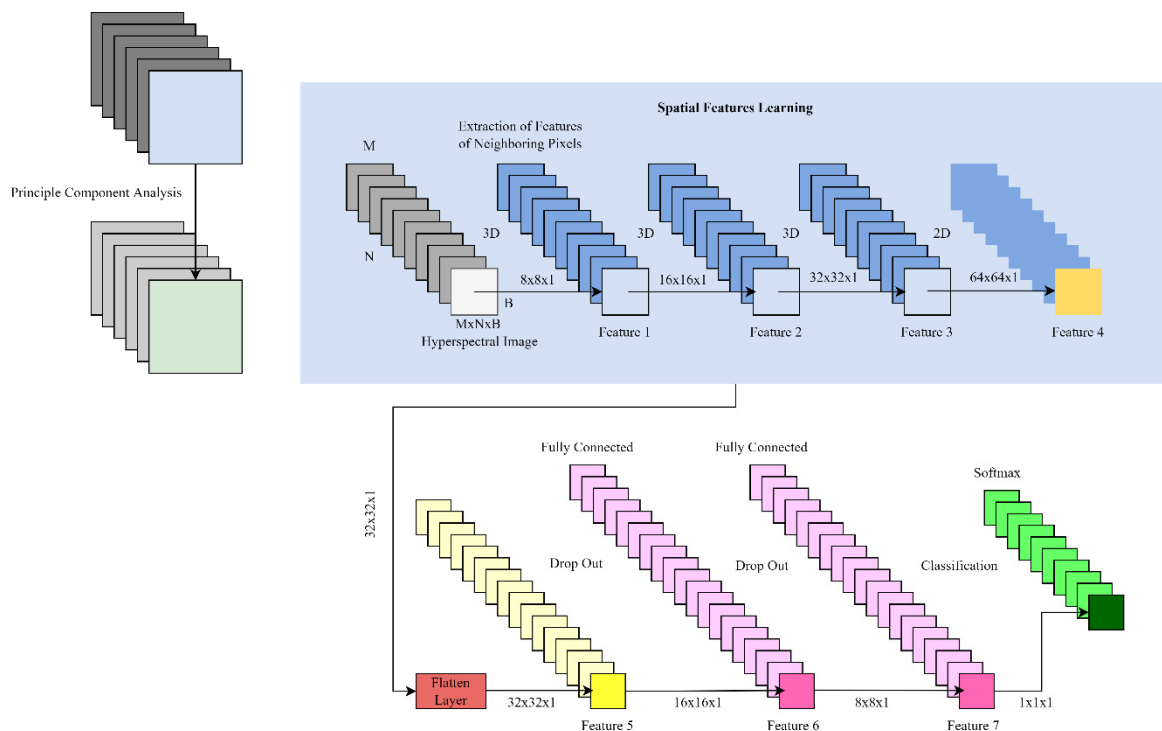
Figure 5. Proposed 3D-CNN Architecture.

A 3D convolutional neural network (CNN) is a type of deep-learning model that is designed to process 3D data, such as voxel data from 3D medical imaging or video data. The unique structure of a 3D-CNN is composed of several layers, including the following:

Input Layer: The input layer receives the 3D data, such as a 3D image or video, and passes it to the next layer.

Convolutional Layer: The convolutional layer applies a convolution operation to the input data using a set of 3D kernels. The convolution operation extracts features from the 3D data by sliding the kernels over the input and performing element-wise multiplication.

ReLU (Rectified Linear Unit) Layer: The ReLU layer applies an activation function to the output of the convolutional layer. The activation function is used to introduce non-linearity into the model and allows the CNN to learn more complex features from the data.

Pooling Layer: The pooling layer is used to down sample the output of the convolutional layer. This helps to reduce the dimensionality of the data and make the model more computationally efficient.

Fully Connected Layer: The fully connected layer is used to classify the features extracted by the previous layers. The output of the fully connected layer is passed to a softmax layer to produce a probability distribution over the classes.

Output Layer: The output layer produces the final classification or segmentation results.

These layers can be stacked multiple times to form a deep 3D-CNN. The 3D-CNN can learn the spatial context of the pixels and the spectral characteristics of the materials present in the scene. It can also learn the relationships between different bands in the image, which can improve classification performance.

3.3. Principle Component Analyses

In order to distinguish the copied area, principal component analyses (PCA) were used in the process of identifying duplicate images. Beginning with an M $\times$ N image, L $\times$ L overlapping chunks are created. By computing the PCA for each image block, features can be extracted. To find the PCA reconstruction Equation (1) with the least amount of loss, we have as the following:

PCA

$$e = \frac{1}{2} \sum_{i=K+!}^{N} \lambda_i \dots \quad (1)$$

To begin, we give a Nb-element array that contains the pixel values of an image block. The array containing the grayscale image's pixel values is L by L in size. When working with a color image, we have two options to compute its features using principal component analysis (PCA): (1) doing PCA on each color channel independently; or (2) flattening the image block pixel to a two-dimensional array 3b, where b is the side of the image block. When calculating an eigenvalue of a block, we may then define the number of dimensions. The next step is to construct a new one-dimensional array from the image blocks in order to find their primary components.

Next, the principal component of each image block is used to determine the order of the matrices via a lexicographical method. As a result, picture blocks with the same or comparable principal components are clustered closely together. Following this, the program generates an array and populates it with the xi, yi coordinates of the picture blocks (xj, yj). Offsets for each array element are then calculated by the algorithm. Each pair of coordinates in the array with an offset smaller than the threshold Nf is thrown out. We also exclude any sets of coordinates where the offset is less than Nd.

3.4. Feature Mapping

A 16 × 16 window is taken around each found key point, I, and further divided into 16 sub blocks, each of which is 4 × 4 in size, to produce extremely distinguishable descriptors against viewpoint and illumination shifts. With 8-dimensional histograms plotted against each 4 × 4 subblocks, the resulting feature descriptor has 128 dimensions. Gradients are used in the feature vector. When an image is rotated, the gradient directions likewise shift. We can make the gradients rotation invariant if we subtract the orientation of the key point from each gradient orientation. Therefore, the gradient direction is now a relative measure to the key point direction.

The parameters of the affine transformation that ties the model to the image are determined using a linear least-squares solution, and this approach is repeated for each found cluster to ensure accuracy. The affine transformation from model point $[x, y]$ T to its equivalent image point is shown below in Equation (2).

Model interpretation

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m1 & m2 \\ m3 & m4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} tx \\ ty \end{bmatrix} \dots \quad (2)$$

Corresponding image point represented by Equation (3) as shown below.

$$\begin{bmatrix} x & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & y \end{bmatrix} \begin{bmatrix} m1 \\ m2 \\ m3 \\ m4 \\ tx \\ ty \end{bmatrix} = \begin{bmatrix} u \\ v \\ . \\ . \end{bmatrix} \dots \quad (3)$$

In this equation, each additional match adds two rows to the first and last matrices, and the number of rows might grow infinitely large. If at least three identical moves are made, the answer is found.

Because the dimensions of the image $B(x, y)$ are ($m \times n$), denoting the row and column indices from which the block is extracted, the formula used to determine the extracted values is shown in the following Euations (4) and (5):

The row and column headers of the block

$$B(x, y) = f(x + c, y + r) \dots \quad (4)$$

$$N = (H - h + 1)x(W - w + 1)\ldots \tag{5}$$

As demonstrated, *W* can be sectioned off into *N* overlapping blocks.

The output is a coefficient matrix *C* with SIFT coefficients Equation (6), and its dimensions are $(m \times n)$.

In a matrix *C* of coefficients, the size is $(m \times n)$

$$C(p,q) = \propto_p \propto_q \sum_{x=0}^{h-1} \sum_{h=0}^{w-1} A \cos\left(\frac{\pi\,(2x + 1)}{2h}\right) \cos\left(\frac{\pi\,(2y + 1)}{2w}\right)\ldots \tag{6}$$

3.5. Flatten Layers

The flatten function cell is expressed as shown in Equation (7):

$$f = a\left(\left(b^f + x_t^{U^f} + h_{t-1}V^f\right)\right)\ldots \tag{7}$$

The product of the previous state with the flatten gate yields an expression of the form $\left(\left(b^f + x_t^{U^f} + h_{t-1}V^f\right)\right)$ as its output. Following the forget gate/state loop, the product is shown in Equation (8):

$$s_t = s_{t-1}x\,f\,x\,g\ldots \tag{8}$$

3.6. Output Layers

The output gate of 3D-CNN is expressed as Equation (9):

$$O = a\left(\left(b^o + x_t^{U^o} + h_{t-1}V^o\right)\right) \tag{9}$$

Finally. the product of all gates is represent by Equation (10):

$$h_i = \tan h\left(a\left(\left(b^{o+f+s+i} + x_t^{U^{o+f+s+i}} + h_{t-1}V^{o+f+s+i}\right)\right)\right)\ldots \tag{10}$$

We utilize measures of accuracy, confusion matrices, and true positive and false negative rates to assess the efficacy of our suggested approach.

The methods' efficacy was evaluated using criteria such as accuracy, precision, recall, and F1 score. This was categorized and reclassified numerous times, as the confusion matrix demonstrates [52]. Figure 6 below displays the investigation's metrics.

Figure 6. Description of confusion matrix.

4. Results

4.1. Image Segmentation

The results of our proposed architecture on all datasets are shown below in Figures 7–9. Segmenting the image and extracting features to make a feature map is the first stage by using Equation (11).

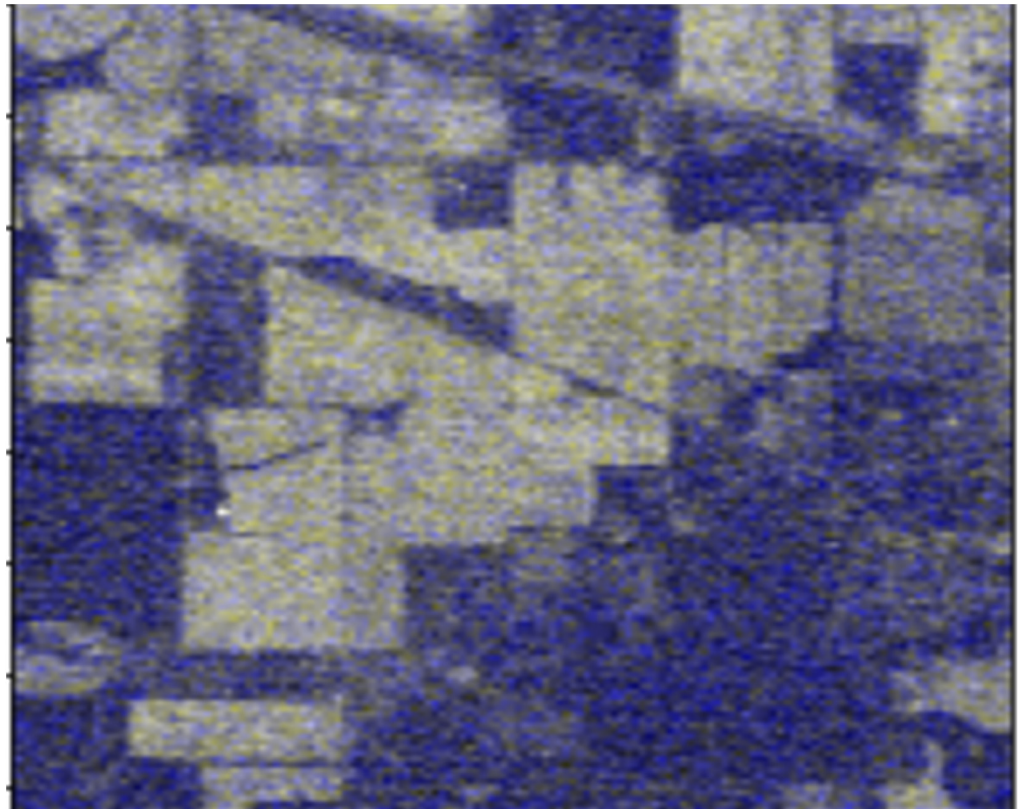$$X_o = F\left(\tan h\left(b^g + x_t^{U^g} + h_{t-1}V^g\right)\right) \tag{11}$$

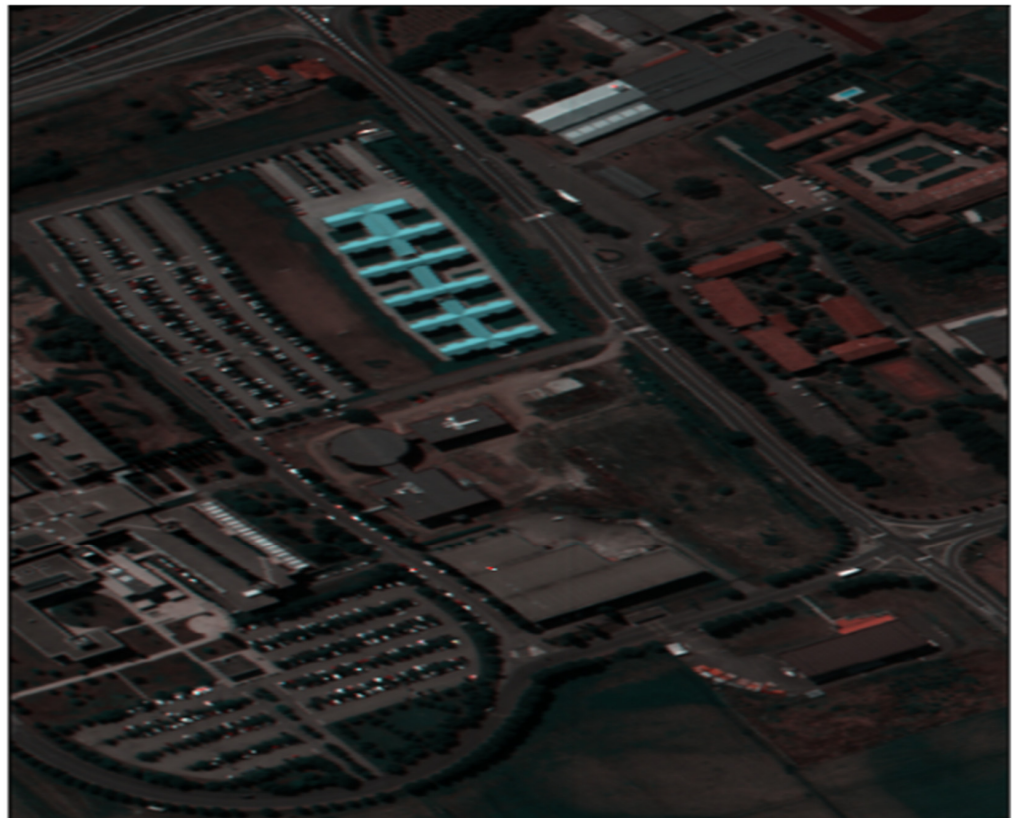**Figure 7.** Image segmentation for Indian Pines Dataset.

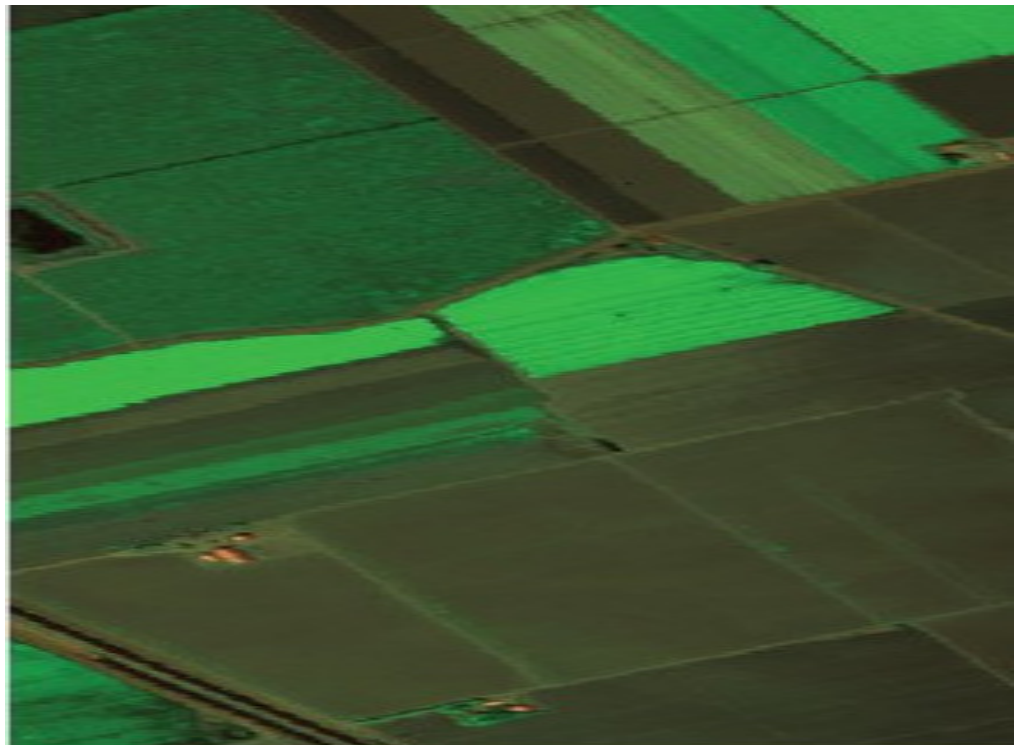**Figure 8.** Pavia Dataset Image Segmentation.

Figure 9. Image segmentation of Salinas Valley Dataset.

4.2. Detection

As mentioned in Figure 2, the Indian Pines dataset was analyzed using a CNN Model to detect and classify individual sections (Figure 10).
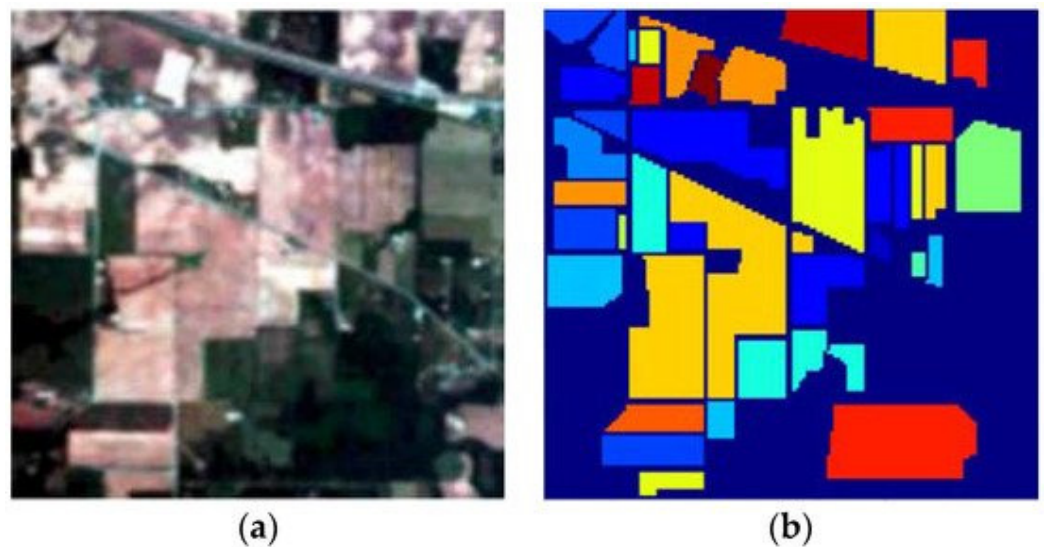

Figure 10. Detection of Areas in Indian Pines Dataset: (**a**) Segmented Image; (**b**) Detection.

Figure 3 of the Pavia dataset demonstrates the use of the CNN model for part detection and classification, and Figure 11 below illustrates this.
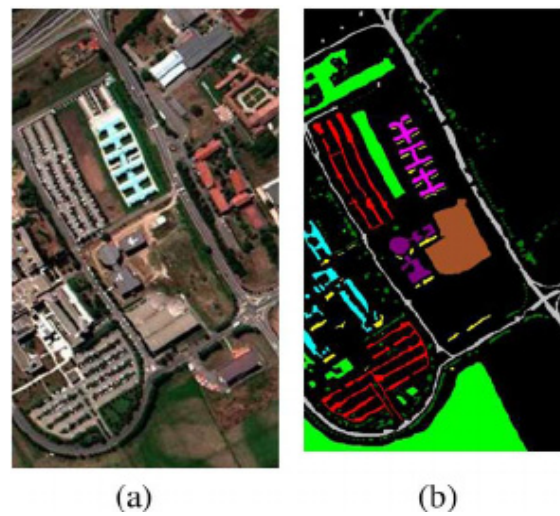
Figure 11. Detection of Areas in Pavia Dataset: (**a**) Segmented Image; (**b**) Detection.

Below in Figure 12 is a screenshot of the Salinas image dataset's CNN model being used to detect and classify portions.
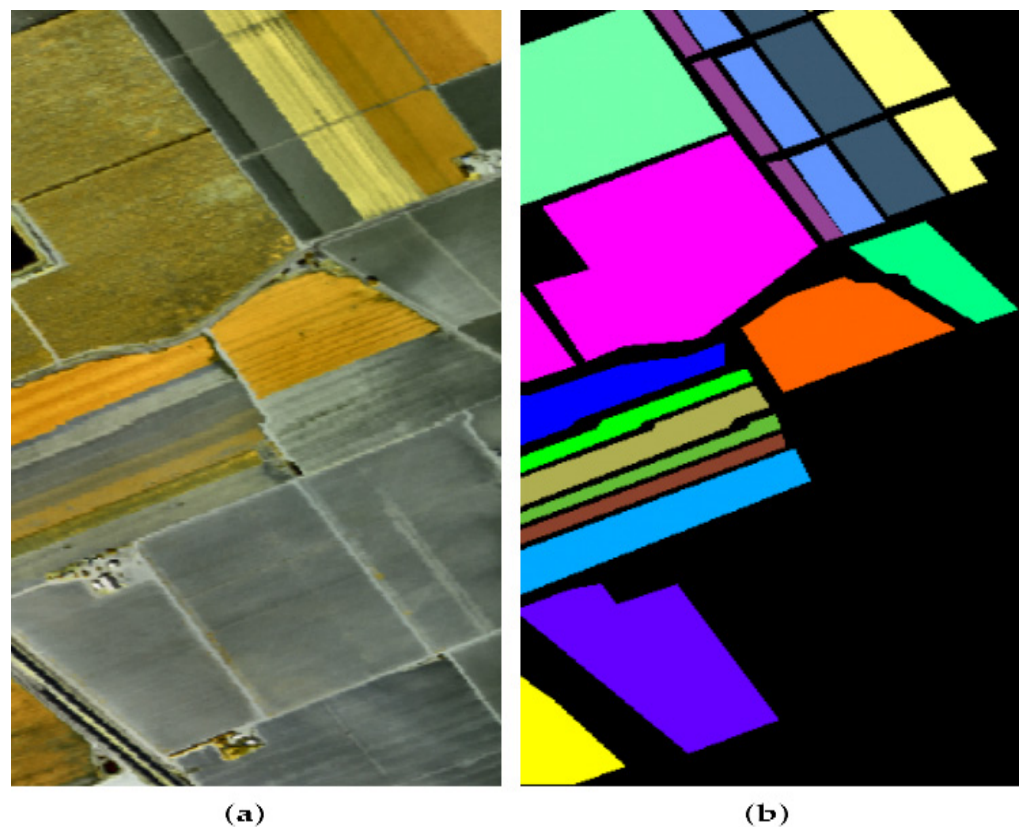

Figure 12. Detection of Areas in Salinas Dataset: (**a**) Segmented Image; (**b**) Detection.

4.3. Performance

On the test datasets of Indian Pine and Pavia, our model achieved 97.56 and 97.55 percent accuracy, respectively. The results obtained using CNN are shown in Figure 13 for various datasets.
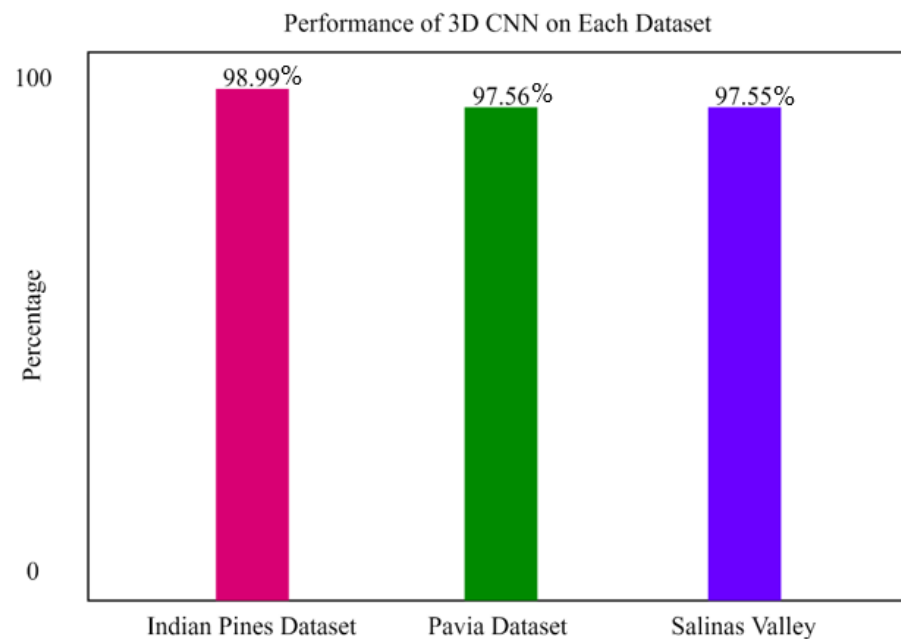
Figure 13. Performance in Each Dataset.

The performance of the proposed 3D-CNN in each dataset depends on several factors, including the architecture of the model, the quality of the dataset, and the specific evaluation metrics used. It is best to report the performance of the proposed 3D-CNN by using different evaluation metrics, such as accuracy, precision, recall, F1 score, and confusion matrix. These metrics give a clear picture of how well the model is performing in classifying the different classes of land cover present in the Indian Pines dataset.

It is also important to consider the training and testing dataset, if the model is overfitting or underfitting, and compare the results with other state-of-the-art methods for that specific dataset. In general, the performance of the proposed 3D-CNN would likely be influenced by factors such as the complexity of the dataset and the model's ability to effectively learn the spectral characteristics of each class of land cover. The results, shown in Table 2, are specific to the dataset and the experimental setup used, and it is crucial to compare the results with other methods for that specific dataset.

Table 2. Comparison with other methods.

Dataset	Techniques	Accuracy	Reference
Pines	CNN	94.6%	[12]
Pavia	CNN	96.55%	[4]
Pines	3D-CNN	98.9%	Our Proposed
Pavia	3D-CNN	97.56%	Our Proposed
Salinas	3D-CNN	97.55%	Our Proposed

The 3D-CNN is a type of CNN that can process three-dimensional data, such as videos or volumetric images. In the case of the Indian Pines dataset, a 3D-CNN could be used to classify the different types of land cover present in the images. The 3D-CNN can take advantage of the temporal information present in the dataset, making it more effective than traditional 2D-CNNs for this task.

If the proposed method of using a 3D-CNN for the classification of the Indian Pines dataset is found to produce better results than other methods, such as a traditional 2D-CNN, it could suggest that the 3D-CNN is better suited to the task of analyzing this type of hyperspectral data. This could be due to its ability to effectively process the additional temporal dimension present in the dataset.

It is important to note that the results will depend on several factors, such as the architecture of the model, the quality of the dataset, and the choice of evaluation metrics. A rigorous experimental setup should be followed to compare different methods and to support the conclusion.

5. Conclusions

There has been a disproportionate amount of focus on developing neural network architecture for RGB images, while multi- and hyperspectral photography have received comparatively less attention. We developed compact convolutional neural network (CNN) architecture that can classify multispectral images with 10 bands using fewer parameters than traditional deep designs. As a means to this end, we made use of the state of the art in compact, efficient convolutional neural networks (CNNs). The results show that the network's classification accuracy and sampling efficiency are better than those of a comparable network trained on RGB images. On top of that, we used a Bayesian variant of our CNN architecture to demonstrate how a network capable of processing multispectral information greatly diminishes uncertainty when making class predictions. A CNN helped us accomplish this task. Forward hyper-tuning and optimization of this model is possible.

Author Contributions: Conceptualization, L.L., L.A. and A.N.H.; methodology, L.L., L.A., E.M.A., Y.A.A. and M.A.-R.; software, L.L., Y.A.A., A.M. and E.M.A.; validation, L.L., L.A., A.N.H., E.M.A., A.M. and Y.A.A.; formal analysis, L.L., E.M.A., Y.A.A., A.M. and M.A.-R.; investigation, L.L., L.A., A.N.H., E.M.A., A.M. and Y.A.A.; resources, M.A.-R.; data curation, L.L., A.N.H., Y.A.A., E.M.A. and A.M.; writing—original draft preparation, L.L., L.A., A.N.H., Y.A.A., E.M.A. and A.M.; writing—review and editing, L.A., A.N.H., Y.A.A., E.M.A. and M.A.-R.; visualization, L.L., Y.A.A. and A.M.; supervision, L.A., A.N.H., Y.A.A. and M.A.-R.; project administration, A.N.H. and M.A.-R. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Publicly available datasets were analyzed in this study. We acquired data on kaggle.com, accessed on 5 October 2022.

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

References

1. Sharma, V.; Diba, A.; Tuytelaars, T.; Van Gool, L. *Hyperspectral CNN for Image Classification & Band Selection, with Application to Face Recognition*; Technical Report: KUL/ESAT/PSI/1604; KU LEUVEN: Leuven, Belgium, 2016.
2. Hameed, M.; Yang, F.; Bazai, S.U.; Ghafoor, M.I.; Alshehri, A.; Khan, I.; Baryalai, M.; Andualem, M.; Jaskani, F.H. Urbanization Detection Using LiDAR-Based Remote Sensing Images of Azad Kashmir Using Novel 3D-CNNs. *J. Sens.* **2022**, *2022*, 6430120. [CrossRef]
3. Kuras, A.; Brell, M.; Rizzi, J.; Burud, I. Hyperspectral and Lidar Data Applied to the Urban Land Cover Machine Learning and Neural-Network-Based Classification: A Review. *Remote. Sens.* **2021**, *13*, 3393. [CrossRef]
4. Cruz, P.H.A. Mapping Urban Tree Species in a Tropical Environment Using Airborne Multispectral and LiDAR Data. Master's Thesis, Universidade Nova de Lisboa, Lisbon, Portugal, 2021; p. 65.
5. Li, Y.; Ma, L.; Zhong, Z.; Liu, F.; Chapman, M.A.; Cao, D.; Li, J. Deep Learning for LiDAR Point Clouds in Autonomous Driving: A Review. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 3412–3432. [CrossRef] [PubMed]
6. Hameed, M.; Yang, F.; Bazai, S.U.; Ghafoor, M.I.; Alshehri, A.; Khan, I.; Ullah, S.; Baryalai, M.; Jaskani, F.H.; Andualem, M. Convolutional Autoencoder-Based Deep Learning Approach for Aerosol Emission Detection Using LiDAR Dataset. *J. Sens.* **2022**, *2022*, 3690312. [CrossRef]
7. Banerjee, B.P.; Spangenberg, G.; Kant, S. CBM: An IoT Enabled LiDAR Sensor for In-Field Crop Height and Biomass Measurements. *Biosensors* **2022**, *12*, 16. [CrossRef]
8. Zhang, L.; Shao, Z.; Liu, J.; Cheng, Q. Deep Learning Based Retrieval of Forest Aboveground Biomass from Combined LiDAR and Landsat 8 Data. *Remote. Sens.* **2019**, *11*, 1459. [CrossRef]

9. Guo, F.; Wang, S.; Yue, B.; Wang, J. A Deformable Configuration Planning Framework for a Parallel Wheel-Legged Robot Equipped with Lidar. *Sensors* **2020**, *20*, 5614. [CrossRef] [PubMed]

10. Marshall, M.R.; Hellfeld, D.; Joshi, T.H.Y.; Salathe, M.; Bandstra, M.S.; Bilton, K.J.; Cooper, R.J.; Curtis, J.C.; Negut, V.; Shurley, A.J.; et al. 3-D Object Tracking in Panoramic Video and LiDAR for Radiological Source–Object Attribution and Improved Source Detection. *IEEE Trans. Nucl. Sci.* **2021**, *68*, 189–202. [CrossRef]

11. Shinohara, T.; Xiu, H.; Matsuoka, M. FWNet: Semantic Segmentation for Full-Waveform LiDAR Data Using Deep Learning. *Sensors* **2020**, *20*, 3568. [CrossRef] [PubMed]

12. Melotti, G.; Premebida, C.; Goncalves, N. Multimodal Deep-Learning for Object Recognition Combining Camera and LIDAR Data. In Proceedings of the 2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Ponta Delgada, Portugal, 15–17 April 2020; pp. 177–182.

13. Männistö, S. Mapping and Classification of Urban Green Spaces with Object-Based Image Analysis and Lidar Data Fusion. October 2020, pp. 1–46. Available online: https://helda.helsinki.fi/handle/10138/322585 (accessed on 13 November 2022).

14. Senchuri, R.; Kuras, A.; Burud, I. Machine Learning Methods for Road Edge Detection on Fused Airborne Hyperspectral and LIDAR Data. In Proceedings of the 2021 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Amsterdam, The Netherlands, 24–26 March 2021.

15. Wei, Y.; Akinci, B. A vision and learning-based indoor localization and semantic mapping framework for facility operations and management. *Autom. Constr.* **2019**, *107*, 102915. [CrossRef]

16. Cui, X.; Avestruz, A.-T. Fast-Response Variable Frequency DC-DC Converters Using Switching Cycle Event-Driven Digital Control. *arXiv* **2022**, arXiv:2209.05272.

17. Raiyn, J. Data and Cyber Security in Autonomous Vehicle Networks. *Transp. Telecommun. J.* **2018**, *19*, 325–334. [CrossRef]

18. El-Rewini, Z.; Sadatsharan, K.; Sugunaraj, N.; Selvaraj, D.F.; Plathottam, S.J.; Ranganathan, P. Cybersecurity Attacks in Vehicular Sensors. *IEEE Sens. J.* **2020**, *20*, 13752–13767. [CrossRef]

19. Zhang, J. Multi-source remote sensing data fusion: Status and trends. *Int. J. Image Data Fusion* **2010**, *1*, 5–24. [CrossRef]

20. Young, J.R., III. An Evaluation of DEM Generation Methods Using a Pixel-Based Landslide Detection Algorithm. Ph.D. Thesis, Virginia Tech, Blacksburg, VA, USA, 2021.

21. Theodouli, A.; Moschou, K.; Votis, K.; Tzovaras, D.; Lauinger, J.; Steinhorst, S. Towards a Blockchain-based Identity and Trust Management Framework for the IoV Ecosystem. In Proceedings of the 2020 Global Internet of Things Summit (GIoTS), Dublin, Ireland, 3 June 2020; pp. 1–6. [CrossRef]

22. Kaushik, I.; Sharma, N. Black Hole Attack and Its Security Measure in Wireless Sensors Networks. In *Handbook of Wireless Sensor Networks: Issues and Challenges in Current Scenario's*; Springer: Cham, Switzerland, 2020; pp. 401–416. [CrossRef]

23. Onita, D.; Dinu, L.P.; Adriana, B. From Image to Text in Sentiment Analysis via Regression and Deep Learning. In Proceedings of the Recent Advances in Natural Language Processing, Varna, Bulgaria, 2–4 September 2019; pp. 862–868. [CrossRef]

24. El-Bakry, H.M. Fast Face Detection Using Neural Networks and Image Decomposition. In Proceedings of the 6th International Computer Science Conference, AMT 2001, Hong Kong, China, 18–20 December 2001; Lecture Notes in Computer Science; pp. 205–215. [CrossRef]

25. Bist, A.S.; Samriya, J.K.; Rawat, B. Hybrid Authentication Policy for Digital Image Processing in Big Data Platform. In Proceedings of the International Conference on Innovative Computing & Communication (ICICC), Delhi, India, 3 July 2021.

26. Li, K.; Pang, K.; Song, Y.-Z.; Hospedales, T.M.; Xiang, T.; Zhang, H. Synergistic Instance-Level Subspace Alignment for Fine-Grained Sketch-Based Image Retrieval. *IEEE Trans. Image Process.* **2017**, *26*, 5908–5921. [CrossRef]

27. Yang, Y.; Jiang, G.; Yu, M.; Qi, Y. Latitude and binocular perception based blind stereoscopic omnidirectional image quality assessment for VR system. *Signal Process.* **2020**, *173*, 107586. [CrossRef]

28. Chetouani, A.; Li, L. On the use of a scanpath predictor and convolutional neural network for blind image quality assessment. *Signal Process. Image Commun.* **2020**, *89*, 115963. [CrossRef]

29. Jeyalakshmi, S.; Radha, R. A Review on Diagnosis of Nutrient Deficiency Symptoms in Plant Leaf Image Using Digital Image Processing. *ICTACT J. Image Video Process.* **2017**, *7*, 1515–1524. [CrossRef]

30. Zhang, X.; Jiao, L.; Paul, A.; Yuan, Y.; Wei, Z.; Song, Q. Semisupervised Particle Swarm Optimization for Classification. *Math. Probl. Eng.* **2014**, *2014*, 832135. [CrossRef]

31. Gao, C.; Zeng, J.; Xia, X.; Lo, D.; Lyu, M.R.; King, I. Automating App Review Response Generation. In Proceedings of the 2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE), San Diego, CA, USA, 11–15 November 2019; pp. 163–175. [CrossRef]

32. Chen, L.; Shi, X.-J.; Liu, H.; Mao, X.; Gui, L.-N.; Wang, H.; Cheng, Y. Oxidative stress marker aberrations in children with autism spectrum disorder: A systematic review and meta-analysis of 87 studies (N = 9109). *Transl. Psychiatry* **2021**, *11*, 15. [CrossRef]

33. Kumar, S.M.; Majumder, D. Healthcare Solution based on Machine Learning Applications in IOT and Edge Computing. *Int. J. Pure Appl. Math.* **2018**, *119*, 1473–1484.

34. Yang, L.; Gao, H.; Yu, D.; Pan, S.; Zhou, Y.; Gai, Y. Design of a Novel Fully Automatic Ocean Spectra Acquisition and Control System Based on the Real-Time Solar Angle Analyzing and Tracking. *IEEE Access* **2020**, *9*, 4752–4768. [CrossRef]

35. Paoletti, M.E.; Haut, J.M.; Plaza, J.; Plaza, A. A new deep convolutional neural network for fast hyperspectral image classification. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 120–147. [CrossRef]

36. Lee, H.; Kwon, H. Going Deeper With Contextual CNN for Hyperspectral Image Classification. *IEEE Trans. Image Process.* **2017**, *26*, 4843–4855. [CrossRef]

37. Li, X.; Ding, M.; Pižurica, A. Group Convolutional Neural Networks for Hyperspectral Image Classification. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 639–643.

38. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep Convolutional Neural Networks for Hyperspectral Image Classification. *J. Sens.* **2015**, *2015*, 258619. [CrossRef]

39. Huang, X.; Wang, H.; Xue, W.; Xiang, S.; Huang, H.; Meng, L.; Ma, G.; Ullah, A.; Zhang, G. Study on time-temperature-transformation diagrams of stainless steel using machine-learning approach. *Comput. Mater. Sci.* **2020**, *171*, 109282. [CrossRef]

40. Huang, X.; Wang, H.; Xue, W.; Ullah, A.; Xiang, S.; Huang, H.; Meng, L.; Ma, G.; Zhang, G. A combined machine learning model for the prediction of time-temperature-transformation diagrams of high-alloy steels. *J. Alloys Compd.* **2020**, *823*, 153694. [CrossRef]

41. Geng, X.; Mao, X.; Wu, H.-H.; Wang, S.; Xue, W.; Zhang, G.; Ullah, A.; Wang, H. A hybrid machine learning model for predicting continuous cooling transformation diagrams in welding heat-affected zone of low alloy steels. *J. Mater. Sci. Technol.* **2022**, *107*, 207–215. [CrossRef]

42. Geng, X.; Wang, H.; Ullah, A.; Xue, W.; Xiang, S.; Meng, L.; Ma, G. Prediction of Continuous Cooling Transformation Diagrams for Ni-Cr-Mo Welding Steels via Machine Learning Approaches. *JOM* **2020**, *72*, 3926–3934. [CrossRef]

43. Sheng, H.; Cong, R.; Yang, D.; Chen, R.; Wang, S.; Cui, Z. UrbanLF: A Comprehensive Light Field Dataset for Semantic Segmentation of Urban Scenes. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 7880–7893. [CrossRef]

44. Liu, Y.; Zhang, Z.; Liu, X.; Wang, L.; Xia, X. Efficient image segmentation based on deep learning for mineral image classification. *Adv. Powder Technol.* **2021**, *32*, 3885–3903. [CrossRef]

45. Zhou, W.; Yu, L.; Zhou, Y.; Qiu, W.; Wu, M.-W.; Luo, T. Local and Global Feature Learning for Blind Quality Evaluation of Screen Content and Natural Scene Images. *IEEE Trans. Image Process.* **2018**, *27*, 2086–2095. [CrossRef]

46. Yang, M.; Wang, H.; Hu, K.; Yin, G.; Wei, Z. IA-Net: An Inception–Attention-Module-Based Network for Classifying Underwater Images from Others. *IEEE J. Ocean. Eng.* **2022**, *47*, 704–717. [CrossRef]

47. Shi, Y.; Xu, X.; Xi, J.; Hu, X.; Hu, D.; Xu, K. Learning to Detect 3D Symmetry from Single-View RGB-D Images with Weak Supervision. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, 1–15. [CrossRef]

48. Zhou, W.; Wang, H.; Wan, Z. Ore Image Classification Based on Improved CNN. *Comput. Electr. Eng.* **2022**, *99*, 107819. [CrossRef]

49. Liao, L.; Du, L.; Guo, Y. Semi-Supervised SAR Target Detection Based on an Improved Faster R-CNN. *Remote. Sens.* **2021**, *14*, 143. [CrossRef]

50. Liu, H.; Liu, M.; Li, D.; Zheng, W.; Yin, L.; Wang, R. Recent Advances in Pulse-Coupled Neural Networks with Applications in Image Processing. *Electronics* **2022**, *11*, 3264. [CrossRef]

51. Zheng, W.; Liu, X.; Yin, L. Research on image classification method based on improved multi-scale relational network. *PeerJ Comput. Sci.* **2021**, *7*, e613. [CrossRef]

52. Pykes, K. February 2022. Available online: https://towardsdatascience.com/confusion-matrix-un-confused-1ba98dee0d7f (accessed on 29 January 2023).