

A Comprehensive Survey of Deep Learning in Remote Sensing: Theories, Tools and Challenges for the Community

John E. Ball^{a,*}, Derek T. Anderson^a, Chee Seng Chan^b

^aMississippi State University, Department of Electrical and Computer Engineering, 406 Hardy Rd., Mississippi State, MS, USA, 39762

^bUniversity of Malaya, Faculty of Computer Science and Information Technology, 50603 Lembah Pantai, Kuala Lumpur, Malaysia

Abstract. In recent years, deep learning (DL), a re-branding of neural networks (NNs), has risen to the top in numerous areas, namely computer vision (CV), speech recognition, natural language processing, etc. Whereas remote sensing (RS) possesses a number of unique challenges, primarily related to sensors and applications, inevitably RS draws from many of the same theories as CV; e.g., statistics, fusion, and machine learning, to name a few. This means that the RS community should be aware of, if not at the leading edge of, of advancements like DL. Herein, we provide the most comprehensive survey of state-of-the-art RS DL research. We also review recent new developments in the DL field that can be used in DL for RS. Namely, we focus on theories, tools and challenges for the RS community. Specifically, we focus on unsolved challenges and opportunities as it relates to (i) inadequate data sets, (ii) human-understandable solutions for modelling physical phenomena, (iii) Big Data, (iv) non-traditional heterogeneous data sources, (v) DL architectures and learning algorithms for spectral, spatial and temporal data, (vi) transfer learning, (vii) an improved theoretical understanding of DL systems, (viii) high barriers to entry, and (ix) training and optimizing the DL.

Keywords: Remote Sensing, Deep Learning, Hyperspectral, Multispectral, Big Data, Computer Vision.

*John E. Ball, jeball@ece.msstate.edu

1 Introduction

In recent years, deep learning (DL) has led to leaps, versus incremental gain, in fields like computer vision (CV), speech recognition, and natural language processing, to name a few. The irony is that DL, a surrogate for neural networks (NNs), is an age old branch of artificial intelligence that has been resurrected due to factors like algorithmic advancements, high performance computing, and Big Data. The idea of DL is simple; the machine is learning the features and decision making (classification), versus a human manually designing the system. The reason this article exists is remote sensing (RS). The reality is, RS draws from core theories such as physics, statistics, fusion, and machine learning, to name a few. This means that the RS community should be aware of, if not at the leading edge of, advancements like DL. The aim of this article is to provide resources with respect to theory, tools and challenges for the RS community. Specifically, we focus on unsolved challenges and opportunities as it relates to (i) inadequate data sets, (ii) human-understandable solutions for modelling physical phenomena, (iii) Big Data, (iv) non-traditional heterogeneous data sources, (v) DL architectures and learning algorithms for spectral, spatial and temporal data, (vi) transfer learning, (vii) an improved theoretical understanding of DL systems, (viii) high barriers to entry, and (ix) training and optimizing the DL.

Herein, RS is a technological challenge where objects or scenes are analyzed by remote means. This includes the traditional remote sensing areas, such as satellite-based and aerial imaging. This definition also includes non-traditional areas, such as unmanned aerial vehicles (UAVs), crowd-sourcing (phone imagery, tweets, etc.), advanced driver assistance systems (ADAS), etc. These

types of remote sensing offer different types of data and have different processing needs, and thus also come with new challenges to algorithms that analyze the data.

The contributions of this paper are as follows:

1. *Thorough list of challenges and open problems in DL RS.* We focus on unsolved challenges and opportunities as it relates to (i) inadequate data sets, (ii) human-understandable solutions for modelling physical phenomena, (iii) Big Data, (iv) non-traditional heterogeneous data sources, (v) DL architectures and learning algorithms for spectral, spatial and temporal data, (vi) transfer learning, (vii) an improved theoretical understanding of DL systems, (viii) high barriers to entry, and (ix) training and optimizing the DL. These observations are based on surveying RS DL and feature learning literature, as well as numerous RS survey papers. This topic is the majority of our paper and is discussed in section 4.
2. *Thorough literature survey.* Herein, we review 207 RS application papers, and 57 survey papers in remote sensing and DL. In addition, many relevant DL papers are cited. Our work extends the previous DL survey papers¹⁻³ to be more comprehensive. We also cluster DL approaches into different application areas and provide detailed discussions of some relevant example papers in these areas in section 3.
3. *Detailed discussions of modifying DL architectures to tackle RS problems.* We highlight approaches in DL in RS, including new architectures, tools and DL components, that current RS researchers have implemented in DL. This is discussed in section 4.5.
4. *Overview of DL.* For RS researchers not familiar with DL, section 2 provides a high-level overview of DL and lists many good references for interested readers to pursue.
5. *Deep learning tool list.* Tools are a major enabler of DL, and we review the more popular DL tools. We also list pros and cons of several of the most popular toolsets and provide a table summarizing the tools, with references and links (refer to Table 2). For more details, see section 2.4.4.
6. *Online summaries of RS datasets and DL RS papers reviewed.* First, an extensive online table with details about each DL RS paper we reviewed: sensor modalities, a compilation of the datasets used, a summary of the main contribution, and references. Second, a dataset summary for all the DL RS papers analyzed in this paper is provided online. It contains the dataset name, a description, a URL (if one is available) and a list of references. Since the literature review for this paper was so extensive, these tables are too large to put in the main article, but are provided online for the readers' benefit. These tables are located at <http://www.cs-chan.com/source/FADL/>.

As an aid to the reader, Table 1 lists acronyms used in this paper.

Table 1: Acronym list.

Acronym	Meaning	Acronym	Meaning
ADAS	Advanced Driver Assistance System	AE	AutoEncoder
ANN	Artificial Neural Network	ATR	Automated Target Recognition
AVHRR	Advanced Very High Resolution Radiometer	AVIRIS	Airborne Visible / Infrared Imaging Spectrometer
BP	Backpropagation	CAD	Computer Aided Design
CFAR	Constant False Alarm Rate	CG	Conjugate Gradient
ChI	Choquet Integral	CV	Computer Vision
CNN	Convolutional Neural Network	DAE	Denoising AE
DAG	Directed Acyclic Graph	DBM	Deep Boltzmann Machine
DBN	Deep Belief Network	DeconvNet	DeConvolutional Neural Network
DEM	Digital Elevation Model	DIDO	Decision In Decision Out
DL	Deep Learning	DNN	Deep Neural Network
DSN	Deep Stacking Network	DWT	Discrete Wavelet Transform
FC	Fully Connected	FCN	Fully Convolutional Network
FC-CNN	Fully Convolutional CNN	FC-LSTM	Fully Connected LSTM
FIFO	Feature In Feature Out	FL	Feature Learning
GBRCN	Gradient-Boosting Random Convolutional Network	GIS	Geographic Information System
GPU	Graphical Processing Unit	HOG	Histogram of Ordered Gradients
HR	High Resolution	HSI	HyperSpectral Imagery
ILSVRC	ImageNet Large Scale Visual Recognition Challenge	L-BGFS	Limited Memory BGFS
LBP	Local Binary Patterns	LiDAR	Light Detection and Ranging
LR	Low Resolution	LSTM	Long Short-Term Memory
LWIR	Long-Wave InfraRed	MKL	Multi-Kernel Learning
MLP	Multi-Layer Perceptron	MSDAE	Modified Sparse Denoising Autoencoder
MSI	MultiSpectral Imagery	MWIR	Mid-wave InfraRed
NASA	National Aeronautics and Space Administration	NN	Neural Network
NOAA	National Oceanic and Atmospheric Administration	PCA	Principal Component Analysis
PGM	Probabilistic Graphical Model	PReLU	Parametric Rectified Linear Unit
RANSAC	RANdom SAMple Concesus	RBM	Restricted Boltzmann Machine
ReLU	Rectified Linear Unit	RGB	Red, Green and Blue image
RGBD	RGB + Depth image	RF	Receptive Field
Continued on next page			

Table 1 – continued from previous page

Acronym	Meaning	Acronym	Meaning
RICNN	Rotation Invariant CNN	RNN	Recurrent NN
RS	Remote Sensing	R-VCANet	Rolling guidance filter Vertex Component Analysis NETwork
S-MSDAE	Stacked MSDAE	SAE	Stacked AE
SAR	Synthetic Aperture Radar	SDAE	Stacked DAE
SIDO	Signal In Decision Out	SIFT	Scale Invariant Feature Transform
SISO	Signal In Signal Out	SGD	Stochastic Gradient Descent
SPI	Standardized Precipitation Index	SSAE	Stacked Sparse Autoencoder
SVM	Support Vector Machine	UAV	Unmanned Aerial Vehicle
VGG	Visual Geometry Group	VHR	Very High Resolution

This paper is organized as follows. Section 2 discusses related work in CV. This section contrasts deep and “shallow” learning, and discusses DL architectures. The main reasons for success of DL are also discussed in this section. Section 3 provides an overview of DL in RS, highlighting DL approaches in many disparate areas of RS. Section 4 discusses the unique challenges and open issues in applying DL to RS. Conclusions and recommendations are listed in section 5.

2 Related work in CV

CV is a field of study trying to achieve visual understanding through computer analysis of imagery. In the past, typical approaches utilized a processing chain which usually started with image denoising or enhancement, followed by feature extraction (with human coded features), a feature optimization stage, and then processing on the extracted features. These architectures were mostly “shallow”, in the sense that they usually had only one to two processing layers between the features and the output. Shallow learners (Support Vector Machines (SVMs), Gaussian Mixture Models, Hidden Markov Models, Conditional Random Fields, etc.) have been the backbone of traditional research efforts for many years². In contrast, DL usually has many layers (the exact demarcation between “shallow” and “deep” learning is not a set number), which allows a rich variety of highly complex, nonlinear and hierarchical features to be learned from the data. The following sections contrast deep and shallow learning, discuss DL approaches and DL enablers, and finally discuss DL success in domains other than RS.

2.1 Deep vs. shallow learning

Shallow learning is a term used to describe learning networks that usually have at most one to two layers. Examples of shallow learners include the popular SVM, Gaussian mixture models, hidden Markov models, conditional random fields, logistic regression models, and the extreme learning machine². Shallow learning models usually have one or two layers that compute a linear or non-linear function of the input data (often hand-designed features). DL, on the other hand, usually means a deeper network, with many layers of (usually) non-linear transformations. Although

there is no universally accepted definition of how many layers constitute a “deep” learner, typical networks are typically at least four or five layers deep. Three main features of DL systems are that DL systems (1) can learn features directly from the data itself, versus human-designed features, (2) can learn hierarchical features which increase in complexity through the deep network, and (3) can be more generalizable and more efficiently encode the model compared to a shallower NN approach; that is, a shallow system will require exponentially more neurons (and thus more free parameters) and more training data^{4,5}. An interesting study on deep and shallow nets is given by Ba and Caruana⁶, where they perform model compression, by training a Deep NN (DNN). The unlabeled data is then evaluated by the DNN and the scores produced by that model are used to train a compressed (shallower) model. If the compressed model learns to mimic the large model perfectly it makes exactly the same predictions and mistakes as the complex model. The key is the compressed model has to have enough complexity to regenerate the more complex model output.

DL systems are often designed to loosely mimic human or animal processing, in which there are many layers of interconnected components, e.g. human vision. So there is a natural motivation to use deep architectures in CV-related problems. For the interested reader, we provide some useful survey paper references. Arel et al. provide a survey paper on DL⁷. Deng et al.² provide two important reasons for DL success: (1) Graphical Processing Unit (GPU) units and (2) recent advances in DL research. They discuss generative, discriminative, and hybrid deep architectures and show there is vast room to improve the current optimization techniques in DL. Liu et al.⁸ give an overview of the autoencoder, the CNN, and DL applications. Wang et al. provide a history of DL⁴. Yu et al.⁹ provide a review of DL in signal and image processing. Comparisons are made to shallow learning, and DL advantages are given. Two good overviews of DL are the survey paper of Schmidhuber et al.¹⁰ and the book by Goodfellow et al.¹¹ Zhang et al.³ give a general framework for DL in remote sensing, which covers four RS perspectives: (1) image processing, (2) pixel-based classification, (3) target recognition, and (4) scene understanding. In addition, they review many DL applications in remote sensing. Cheng et al. discuss both shallow and DL methods for feature extraction.¹ Some good DL papers are the introductory DL papers of Arnold et al.¹² and Wang et al.⁴, the DL book by Goodfellow et al.¹¹, and the DL survey papers^{2,4,5,8,10,13-16}.

2.2 Traditional Feature Learning methods

Traditional methods of feature extraction involve hand-coded features to extract information based on spatial, spectral, textural, morphological content, etc. These traditional methods are discussed in detail in the following references, and we will not give extensive algorithmic details herein. All of these hand-derived features are designed for a specific task, e.g. characterizing image texture. In contrast, DL systems derive complicated, (usually) non-linear and hierarchical features from the data itself.

Cheng et al.¹ discuss traditional handcrafted features such as the Histogram of Ordered Gradients (HOG), the Scale-Invariant Feature Transform (SIFT) and SIFT variants, color histograms, etc. They also discuss unsupervised FL methods, such as principal components analysis, k -means clustering, sparse coding, etc. Other good survey papers discuss hyperspectral image (HSI) data analysis¹⁷, kernel-based methods¹⁸, statistical learning methods in HSI¹⁹, spectral distance functions²⁰, pedestrian detection²¹, multi-classifier systems²², spectral-spatial classification²³, change detection^{24,25}, machine learning in RS²⁶, manifold learning²⁷, transfer learning²⁸, endmember extraction²⁹, and spectral unmixing³⁰⁻³⁴.

2.3 DL Approaches

To date, the auto-encoder (AE), the CNN, Deep Belief Networks (DBNs), and the Recurrent NN (RNN), have been the four mainstream DL architectures. The deconvolutional NN (DeconvNet) is a relative newcomer to the DL community. The following sections discuss each of these architectures at a high level. Many good references are provided for the interested reader.

2.3.1 Autoencoder (AE)

An AE is a network designed to learn useful features from unsupervised data. One of the first applications of AEs was dimensionality reduction, which is required in many RS applications. By reducing the size of the adjacent layers, the AE is forced to learn a compact representation of the data. The AE maps the input through an encoder function f to generate an internal (latent) representation, or code, h , that is, $h = f(\mathbf{x})$. The autoencoder also has a decoder function, g that maps h to the output $\hat{\mathbf{x}}$. In general, the AE is constrained, either through its architecture, or through a sparsity constraint (or both), to learn a useful mapping (but not the trivial identity mapping). A loss function L measures how close the AE can reconstruct the output: L is a function of \mathbf{x} and $\hat{\mathbf{x}} = g(f(\mathbf{x}))$. A regularization function $\Omega(h)$ can also be added to the loss function to force a more sparse solution. The regularization function can involve penalty terms for model complexity, model prior information, penalizing based on derivatives, or penalties based on some other criteria such as supervised classification results, etc. (reference §14.2 of¹¹).

A Denoising AE (DAE) is an AE designed to remove noise from a signal or an image. Chen et al. developed an efficient DAE, which marginalizes the noise and has a computationally efficient closed form solution.³⁵ To provide robustness, the system is trained using additive Gaussian noise or binary masking noise (force some percentage of inputs to zero). Many RS applications utilize an AE for denoising. Figure 1(a) shows an example of a AE. The diabolo shape results in dimensionality reduction.

2.3.2 Convolutional Neural Network (CNN)

A CNN is a network that is loosely inspired by the human visual cortex. A typical CNN is composed of multiple dual-layers of convolutional masks followed by pooling, and these layers are then usually followed by either fully-connected or partially-connected layers, which perform classification or class probability estimation. Some CNNs also utilize data normalization layers. The convolution masks have coefficients that are learned by the CNN. A CNN that analyzes grayscale imagery will employ 2D convolution masks, while a CNN using Red-Green-Blue (RGB) imagery will use 3D masks. Through training, these masks learn to extract features directly from the data, in stark contrast to traditional machine learning approaches, which utilize "hand-crafted" features. The pooling layers are non-linear operators (usually maximum operators), which allows the CNN to learn non-linear features, which greatly increases its learning capabilities. Figure 1(b) shows an example CNN, where the input is a HSI, and there are two convolution and pooling layers, followed by two fully connected (FC) layers.

The number of convolution masks, the size of the masks, and the pooling functions are all parameters of the CNN. The masks at the first layers of the CNN typically learn basic features, and as one traverses the depths of the network, the features become more complex and are built-up hierarchically. Normalization layers provide regularization and can aid in training. The fully-connected layers (or partially-connected layers) are usually near the end of the CNN network, and

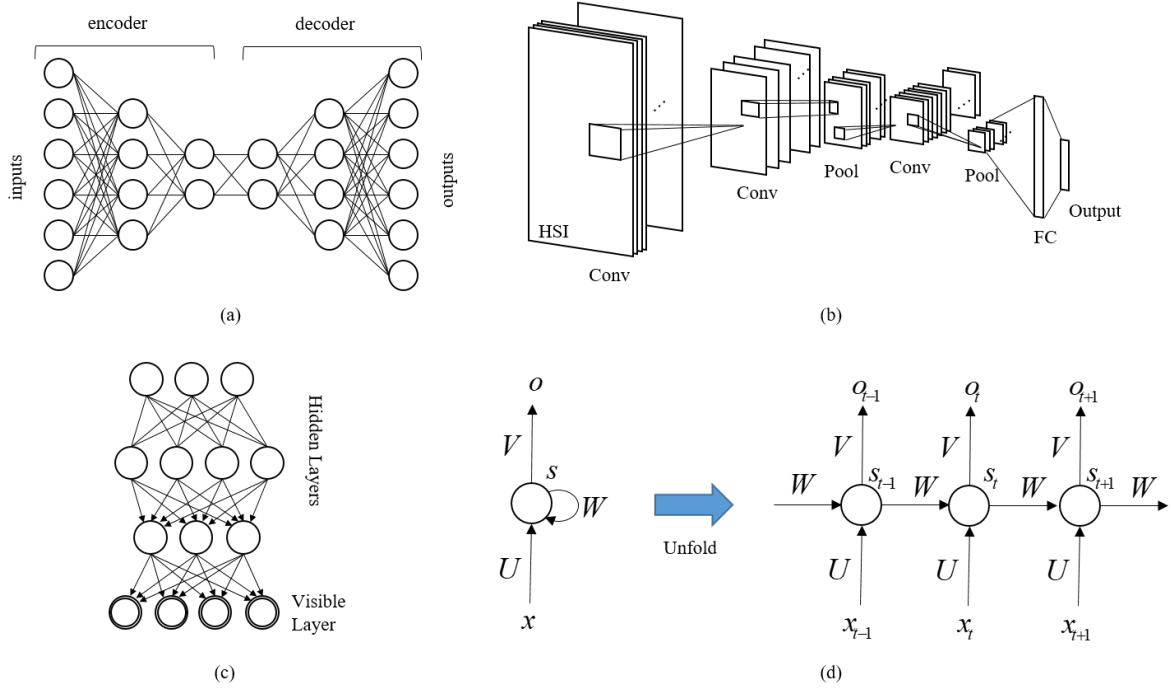


Fig 1 Block diagrams of DL architectures. (a) AE. (b) CNN. (c) DBN. (d) RNN.

allow complex non-linear functions to be learned from the hierarchical outputs of the previous layers. These final layers typically output class labels or estimates of the probabilities of the class label.

CNNs have dominated in many perceptual tasks. Following Ujjwalkarn³⁶, the image recognition community has shown keen interest in CNNs. Starting in the 1990s, LeNet was developed by LeCun et al.³⁷, and was designed for reading zip codes. It generated great interest in the image processing community. In 2012, Krizhevsky et al.³⁸ introduced AlexNet, a deep CNN. It won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012 by a significant margin. In 2013, Zeiler and Fergus³⁹ created ZFNet, which was AlexNet with tweaked parameters and won ILSVRC. Szegedy et al.⁴⁰ won ILSVRC with GoogLeNet in 2014, which used a much smaller number of parameters (4 million) than AlexNet (60 million). In 2015, ResNets were developed by He et al.⁴¹, which allowed CNNs to have very deep networks. In 2016, Huang et al.⁴² published DenseNet, where each layer is directly connected to every other layer in a feedforward fashion. This architecture also eliminates the vanishing-gradient problem, allowing very deep networks. The examples above are only a few examples of CNNs.

2.3.3 Deep Belief Network (DBN)

A DBN is a type (generative) of Probabilistic Graphical Model (PGM), the marriage of probability and graph theory. Specifically, a DBN is a “deep” (large) Directed Acyclic Graph (DAG). A number of well-known algorithms exist for exact and approximate inference (infer the states of unobserved (hidden) variables) and learning (learn the interactions between variables) in PGMs. A DBN can also be thought of as a type of deep NN. In⁴³, Hinton showed that a DBN can be viewed and trained (in a greedy manner) as a stack of simple unsupervised networks, namely

Restricted Boltzmann Machines (RBMs), or generative AEs. To date, CNNs have demonstrated better performance on various benchmark CV data sets. However, in theory DBNs are arguably superior. CNNs possess generally a lot more “constraints”. The DBN versus CNN topic is likely subject to change as better algorithms are proposed for DBN learning. Figure 1(c) depicts a DBN, which is made up of RBM layers and a visible layer.

2.3.4 Recurrent Neural Network (RNN)

The RNN is a network where connections form directed cycles. The RNN is primarily used for analyzing non-stationary processes such as speech and time-series analysis. The RNN has memory, so the RNN has persistence, which the AE and CNN don’t possess. A RNN can be unrolled and analyzed as a series of interconnected networks that process time-series data. A major breakthrough for RNNs was the seminal work of Hochreiter and Schmidhuber⁴⁴, the long short-term memory (LSTM) unit, which allows information to be written to a cell, output from the cell, and stored in the cell. The LSTM allows information to flow and helps counteract the vanishing/exploding gradient problems in very deep networks. Figure 1(d) shows a RNN and its unfolded version.

2.3.5 Deconvolutional Neural Network (DeconvNet)

CNNs are often used for classification only. However, a wealth of questions exist beyond classification, e.g., what are our filters really learning, how transferable are these filters, what filters are the most active in a given image, where is a filter the most active at in a given image (or images), or more holistically, where in the image is our object(s) of interest (soft or hard segmentation). To this end, researchers have recently explored deconvolutional NN (DeconvNet)^{39,45–47}. Whereas CNNs use pooling, which helps us filter noisy activations and address affine transformations, a DeconvNet uses unpooling—the “inverse” of pooling. Unpooling makes use of “switch variables”, which help us place activation in layer l back to its original pooled location in layer $l - 1$. Unpooling results in an enlarged, be it sparse, activation map that is fed to deconvolution filters (that are either learned or derived from the CNN filters). In⁴⁷, the Visual Geometry Group (VGG) developed the VGG 16-layer CNN, thus no deconvolution, with its last classification layer removed was used relative to computer vision on non-remotely sensed data. The resultant DeconvNet is twice as large as the VGG CNN. The first part of the network is the VGG CNN and the second part is an architecturally reversed copy of the VGG CNN with pooling replaced by unpooling. The entire network was trained and used for semantic image segmentation. In a different work, Zeiler et al. showed that a DeconvNet can be used to visualize a single CNN filter at any layer or a combination of CNN filters can be visualized for one or more images^{39,45}. The point is, relevant DeconvNet research exists in the CV literature.

Two high-level comments are worth noting. First, DeconvNets have been used by some to help rationalize their architecture and operation selections in the context of a visual odyssey of its impact on the filters relative to one another, e.g., existence of a single dominant feature versus a diverse set of features. In many cases its not a rationalization of the final network performance per se, but instead a DeconvNet is a helpful tool that aids them in exploring the vast sea of choices in designing the network. Second, whereas DeconvNet can be used in many cases for segmentation, they do not always produce the segmentation that we might desire. Meaning, if the CNN learned parts, not the full object, then activation of those parts, or a subset thereof, may not equate to the whole and those parts might also be spatially separated in the image. The later makes it challenging

to construct a high quality full object segmentation, or segmentation's if there are more than one instance of that object in an image. DeconvNets are basically very recent and have not (yet) been widely adopted by the RS community.

2.4 *DL Meets the Real World*

It is important to understand the different “factors” related to the rise and success of DL. This section discusses these factors: GPUs, DL NN expressiveness, big data, and tools.

2.4.1 *GPUs*

GPUs are hardware devices that are optimized for fast parallel processing. GPUs enable DL by offloading computations from the computer's main processor (which is basically optimized for serial tasks) and efficiently performing the matrix-based computations at the heart of many DL algorithms. The DL community can leverage the personal computer gaming industry, which demands relatively inexpensive and powerful GPUs. A major driver of the research interest in CNNs is the Imagenet contest, which has over one million training images and 1,000 classes⁴⁸. DNNs are inherently parallel, utilize matrix operations, and use a large number of floating point operations per second. GPUs are a match because they have the same characteristics⁴⁹. GPU speedups have been measured at 8.5 to 9⁴⁹ and even higher depending on the GPU and the code being optimized. The CNN convolution, pooling and activation calculation operations are readily portable to GPUs.

2.4.2 *DL NN Expressiveness*

Cybenko⁵⁰ proved that MLPs are universal function approximators. Specifically, Cybenko showed that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on compact subsets of \mathbb{R}^n , with respect to relatively minimalistic assumptions regarding the activation function. However, Cybenko's proof is an existence theorem, meaning it tells us a solution exists, but it does not tell us how to design or learn such a network. The point is, NNs have an intriguing mathematical foundation that make them attractive with respect to machine learning. Furthermore, in a theoretical work, Telgarsky⁵¹ has shown that for NN with Rectified Linear Units (ReLU) (1) functions with few oscillations poorly approximate functions with many oscillations, and (2) functions computed by NN with few (many) layers have few (many) oscillations. Basically, a deep network allows decision functions with high oscillations. This gives evidence to show why DL performs well in classification tasks, and that shallower networks have limitations with highly oscillatory functions. Sharir et al.⁵² showed that having overlapping local receptive fields and more broadly denser connectivity gives an exponential increase in the expressive capacity of the NN. Liang et al.⁵³ showed that shallow networks require exponentially more neurons than a deep network to achieve the level of accuracy for function approximation.

2.4.3 *Big Data*

Every day, approximately 350 million images are uploaded to Facebook⁴⁹, Wal-Mart collects approximately 2.5 petabytes of data per day⁴⁹, and National Aeronautics and Space Administration (NASA) alone is actively streaming 1.73 gigabytes of spacecraft borne observation data for active missions alone.⁵⁴ IBM reports that 2.5 quintillion bytes of data are now generated every day, which means that “90% of the data in the world today has been created in the last two years alone”⁵⁵.

The point is, an unprecedented amount of (varying quality) data exists due to technologies like remote sensing, smart phones, inexpensive data storage, etc. In times past, researchers used tens to hundreds, maybe thousands of data training samples, but nothing on the order of magnitude as today. In areas like CV, high data volume and variety have been at the heart of advancements in performance. Meaning, reported results are a reflection of advances in both data and machine learning.

To date, a number of approaches have been explored relative to large scale deep networks (e.g., hundreds of layers) and Big Data (e.g., high volume of data). For example, in⁵⁶ Raina et al. put forth CPU and GPU ideas to accelerate DBNs and sparse coding. They reported a 5 to 15-fold speed up for networks with 100 million plus parameters versus previous works that used only a few million parameters at best. On the other hand, CNNs typically use back propagation and they can be implemented either by pulling or pushing⁵⁷. Furthermore, ideas like circular buffers⁵⁸ and multi GPU based CNN architectures, e.g., Krizhevsky³⁸, have been put forth. Outside of hardware speedups, operators like ReLUs have been shown to run sever times faster than other common nonlinear functions. In⁵⁹, Deng et al. put forth a Deep Stacking Network (DSN) that consists of specialized NNs (called modules), each of which have a single hidden layer. Hutchinson et al. put forth Tensor-DSN is an efficient and parallel extension of DSNs for CPU clusters⁶⁰. Furthermore, DistBelief is a library for distributed training and learning of deep networks with large models (billions of parameters) and massive sized data sets⁶¹. DistBelief makes use of machine clusters to manage the data and parallelism via methods like multi-threading, message passing, synchronization and machine-to-machine communication. DistBelief uses different optimization methods, namely SGD and Sandblaster⁶². Last, but not least, there are network architectures such as highway networks, residual networks and dense nets⁶³⁻⁶⁷. For example, highway networks are based on LSTM recurrent networks and they allow for the efficient training of deep networks with hundreds of layers based on gradient descent⁶⁴⁻⁶⁶.

2.4.4 Tools

Tools are also a large factor in DL research and development. Wan et al. observe that DL is at the intersection of NNs, graphical modeling, optimization, pattern recognition and signal processing⁵, which means there is a fairly high background level required for this area. Good DL tools allow researchers and students to try some basic architectures and create new ones more efficiently.

Table 2 lists some popular DL toolkits and links to the code. Herein, we review some of the DL tools, and the tool analysis below are based on our experiences with these tools. We thank our graduate students for providing detailed feedback on these tools.

AlexNet³⁸ was a revolutionary paper that re-introduced the world to the results that DL can offer. AlexNet utilizes ReLU because it is several times faster to evaluate than the hyperbolic tangent. AlexNet revealed the importance of pre-processing by incorporating some data augmentation techniques and was able to combat overfitting by using max pooling and dropout layers.

Caffe⁶⁸ was the first widely used deep learning toolkit. Caffe is C++ based and can be compiled on various devices, and offers command line, Python, and Matlab interfaces. There are many useful examples provided. The cons of Caffe are that it is relatively hard to install, due to lack of documentation and not being developed by an organized company. For those interested in something other than image processing, (e.g. image classification, image segmentation), it is not really suitable for other areas, such as audio signal processing.

TensorFlow⁶⁹ is arguably the most popular DL tool available. It's pros are that TensorFlow (1) is relatively easy to install both with CPU and GPU version on Ubuntu (The GPU version needs CUDA and cuDNN to be installed ahead of time, which is a little complicated); (2) has most of the state-of-the-art models implemented, and while some original implementation are not implemented in TensorFlow, but it is relatively easy to find a re-implementation in TensorFlow; (3) has very good documentation and regular updates; (4) supports both Python and C++ interfaces; and (5) is relatively easy to expand to other areas besides image processing, as long as you understand the tensor processing. One con of TensorFlow is that it is really restricted to Linux applications, as the windows version is barely usable.

MatConvNet⁷⁰ is a convenient tool, with unique abstract implementations for those very comfortable with using Matlab. It offers many popular trained CNNs, and the data sets used to train them. It is fairly easy to install. Once the GPU setup is ready (installation of drivers + CUDA support), training with the GPU is very simple. It also offers Windows support. The cons are (1) there is a substantially smaller online community compared to TensorFlow and Caffe, (2) code documentation is not very detailed and in general does not have good online tutorials besides the manual. Lack of getting started help besides a very simple example, and (3) GPU setup can be quite tedious. For Windows, Visual Studio is required, due to restrictions on Matlab and its mex setup, as well as Nvidia drivers and CUDA support. On Linux, one has much more freedom, but must be willing to adapt to manual installations of Nvidia drivers, CUDA-support, and more.

2.5 DL in other domains

DL has been utilized in other areas than RS, namely human behavior analysis⁷⁶⁻⁷⁹, speech recognition⁸⁰⁻⁸², stereo vision⁸³, robotics⁸⁴, signal-to-text⁸⁵⁻⁸⁹, physics^{90,91}, cancer detection⁹²⁻⁹⁴, time-series analysis⁹⁵⁻⁹⁷, image synthesis⁹⁸⁻¹⁰⁴, stock market analysis¹⁰⁵, and security applications¹⁰⁶. These diverse set of applications show the power of DL.

3 DL approaches in RS

There are many RS tasks that utilize RS data, including automated target detection, pansharpening, land cover and land use classification, time series analysis, change detection, etc. Many of these tasks utilize shape analysis, object recognition, dimensionality reduction, image enhancement, and other techniques, which are all amenable to DL approaches. Table 3 groups DL papers reviewed in this paper into these basic categories. From the table, it can be seen that there is a large diversity of applications, indicating that RS researchers have seen value in using DL methods in many different areas. Several representative papers are reviewed and discussed.

Due to the large number of recent RS papers, we can't review all of the papers utilizing DL or FL in RS applications. Instead, herein we focus on several papers in different areas of interest that offer creative solutions to problems encountered in DL and FL and should also have a wide interest to the readers. We do provide a summary of all of the DL in RS papers we reviewed online at <http://www.cs-chan.com/source/FADL/>.

3.1 Classification

Classification is the task of labeling pixels (or regions in an image) into one of several classes. The DL methods outlined below utilize many forms of DL to learn features from the data itself and

perform classification at state-of-the-art levels. The following discusses classification in HSI, 3D, satellite imagery, traffic sign detection and Synthetic Aperture Radar (SAR).

HSI: HSI data classification is of major importance to RS applications, so many of the DL results we reviewed were on HSI classification. HSI processing has many challenges, including high data dimensionality and usually low numbers of training samples. Chen et al.³¹⁴ propose an DBN-based HSI classification framework. The input data is converted to a 1D vector and processed via a DBN with three RBM layers, and the class labels are output from a two-layer logistic regression NN. A spatial classifier using Principal Component Analysis (PCA) on the spectral dimension followed by 1D flattening of a 3D box, a three-level DBN and two level logistic regression classifier. A third architecture uses a combinations of the 1D spectrum and the spatial classifier architecture. He et al.¹⁵¹ developed a DBN for HSI classification that does not require stochastic gradient descent (SGD) training. Nonlinear layers in the DBN allow for the nonlinear nature of HSI data and a logistic regression classifier is used to classify the outputs of the DBN layers. A parametric depth study showed depth of nine layers produced the best results of depths of 1 to 15, and after a depth of nine, no improvement resulted by adding more layers.

Some of the HSI DL approaches utilize both spectral and spatial information. Ma et al.¹⁶⁹ created a HSI spatial updated deep AE which integrates spatial information. Small training sets are mitigated by a collaborative, representation-based classifier and salt-and-pepper noise is mitigated by a graph-cut-based spatial regularization. Their method is more efficient than comparable kernel-based methods, and the collaborative representation-based classification makes their system relatively robust to small training sets. Yang et al.¹⁸¹ use a two-channel CNN to jointly learn spectral and spatial features. Transfer learning is used when the number of training samples is limited, where low-level and mid-level features are transferred from other scenes. The network has a spectral CNN and a spatial CNN, and the results are combined in three FC layers. A softmax classifier produces the final class labels. Pan et al.¹⁷⁵ proposed the so called rolling guidance filter and vertex component analysis network (R-VCANet), which also attempts to solve the common problem of lack of HSI training data. The network combines spectral and spatial information. The rolling guidance filter is an edge-preserving filter used to remove noise and small details from imagery. The VCANet is a combination of vertex component analysis³¹⁵, which is used to extract pure endmembers, and PCANet³¹⁶. A parameter analysis of the number of training samples, rolling times, and the number and size of the convolution kernels. The system performs well even when the training ratio is only 4%. Lee et al.¹⁵⁸ designed a contextual deep fully convolutional DL network with fourteen layers that jointly exploits spatial and HSI spectral features. Variable size convolutional features are utilized to create a spectral-spatial feature map. A novel feature of the architecture is the initial layers uses both $[3 \times 3 \times B]$ convolutional masks to learn spatial features, and $[1 \times 1 \times B]$ for spectral features, where B is the number of spectral bands. The system is trained with a very small number of training samples (200/class).

3D: In 3D analysis, there are several interesting DL approaches. Chen et al.³¹⁷ used a 3D CNN-based feature extraction model with regularization to extract effective spectral-spatial features from HSI. L_2 regularization and dropout are used to help prevent overfitting. In addition, a virtual enhanced method imputes training samples. Three different CNN architectures are examined: (1) a 1D using only spectral information, consisting of convolution, pooling, convolution, pooling, stacking and logistic regression; (2) a 2D CNN with spatial features, with 2D convolution, pooling, 2D convolution, pooling, stacking, and logistic regression; (3) 3D convolution (2D for spatial and

third dimension is spectral); the organization is same as 2D case except with 3D convolution. The 3D CNN achieves near-perfect classification on the data sets.

Chen et al.¹⁹¹ propose a novel 3D CNN to extract the spectral-spatial features of HSI data, a deep 2D CNN to extract the elevation features of Light Detection and Ranging (LiDAR) data, and then a FC DNN to fuse the 2D and 3D CNN outputs. The HSI data are processed via two layers of 3D convolution followed by pooling. The LiDAR elevation data are processed via two layers of 2D convolution followed by pooling. The results are stacked and processed by a FC layer followed by a logistic regression layer.

Cheng et al.²²⁶ developed a rotation-invariant CNN (RICNN), which is trained by optimizing a objective function with a regularization constraint that explicitly enforces the training feature representations before and after rotating to be mapped close to each other. New training samples are imputed by rotating the original samples by k rotation angles. The system is based on AlexNet,³⁸ which has five convolutional layers followed by three FC layers. The AlexNet architecture is modified by adding a rotation-invariant layer that used the output of AlexNet's FC7 layer, and replacing the 1000-way softmax classification layer with a $(C + 1)$ -layer softmax classifier layer. AlexNet is pretrained, then fine tuned using the small number of HSI training samples. Haque et al.¹⁰⁹ developed a attention-based human body detector that leverages 4D spatio-temporal signatures and detects humans in the dark (depth images with no RGB content). Their DL system extracts voxels then encodes data using a CNN, followed by a LSTM. An action network gives the class label and a location network selects the next glimpse location. The process repeats at the next time step.

Traffic Sign Recognition: In the area of traffic sign recognition, a nice result came from Cire-san et al.¹¹⁹, who created a biologically plausible DNN is based on the feline visual cortex. The network is composed of multiple columns of DNNs, coded for parallel GPU speedup. The output of the columns is averaged. It outperforms humans by a factor of two in traffic sign recognition.

Satellite Imagery: In the area of satellite imagery analysis, Zhang et al.¹⁸⁶ propose a gradient-boosting random convolutional network (GBRCN) to classify very high resolution (VHR) satellite imagery. In GBRCN, a sum of functions (called boosts) are optimized. A modified multi-class softmax function is used for optimization, making the optimization task easier. SGD is used for optimization. Proposed future work was to utilize a variant of this method on HSI. Zhong et al.¹⁹⁰ use efficient small CNN kernels and a deep architecture to learn hierarchical spatial relationships in satellite imagery. A softmax classifier output class labels based on the CNN DL outputs. The CPU handles preprocessing (data splitting and normalization), while the GPU performs convolution, ReLU and pooling operations, and the the CPU handles dropout and softmax classification. Networks with one to three convolution layers are analyzed, with receptive fields of 10×10 to 1000×1000 . SGD is used for optimization. A hyper-parameter analysis of the learning rate, momentum, training-to-test ratio, and number of kernels in the first convolutional layer were also performed.

SAR: In the area of SAR processing, De et al.²⁸⁸ use DL to classify urban areas, even when rotated. Rotated urban target exhibit different scattering mechanisms, and the network learns the α and γ parameters from the HH, VV and HV bands (H=Horizontal, V-Vertical polarization). Bentes et al.¹²⁴ use a constant false alarm rate (CFAR) processor on SAR data followed by N AEs. The final layer associates the learned features with class labels. Geng et al.¹⁴⁹ used a eight-layer network with a convolutional layer to extract texture features from SAR imagery, a scale transformation layer to aggregate neighbor features, four Stacked AE (SAE) layers for feature

optimization and classification, and a two-layer post processor. Gray level co-occurrence matrix and Gabor features are also extracted, and average pooling is used in layer two to mitigate noise.

3.2 Transfer Learning

Transfer learning utilizes training in one image (or domain) to enable better results in another image (or domain). If the learning crosses domains, then it may be possible to utilize lower to mid-level features learned from one domain in the other domain.

Marmanis et al.²⁵⁹ attacked the common problem in RS of limited training data by utilizing transfer learning across domains. They utilized a CNN pretrained on the ImageNet dataset, and extracted an initial set of representations from orthoimagery. These representations are then transferred to a CNN classifier. This paper developed a novel cross-domain feature fusion system. Their system has seven convolution layers followed by two long MLP layers, three convolution layers, two more large MLP layers, and finally a softmax classifier. They extract feature from the last layer, since the work of Donahue et al.³¹⁸ showed that most of the discriminative information is contained in the deeper layers. In addition, they take features from the large ($1 \times 1 \times 4096$) MLP, which is a very long vector output, and transform it into a 2D array followed by a large convolution (91×91) mask layer. This is done because the large feature vector is a computational bottleneck, while the 2D data can very effectively be processed via a second CNN. This approach will work if the second CNN can learn (disentangle) the information in the 2D representation through its layers. This approach is very unique and it raises some interesting questions about alternate DL architectures. This approach was also successful because the features learned by the original CNN were effective in the new image domain.

Penatti et al.²¹⁹ asked if deep features generalize from everyday objects to remote sensing and aerial scene domains? A CNN was trained for recognizing everyday objects using ImageNet. The CNNs analyzed performed well, in areas well outside of their training. In a similar vein, Salberg¹²¹ use CNNs pretrained on ImageNet to detect seal pups in aerial RS imagery. A linear SVM was used for classification. The system was able to detect seals with high accuracy.

3.3 3D Processing and Depth Estimation

Cadena et al.¹⁰⁷ utilized multi-modal AEs for RGB imagery, depth images, and semantic labels. Through the AE, the system learns a shared representation of the distinct inputs. The AEs first denoise the given inputs. Depth information is processed as inverse depth (so sky can be handled). Three different architectures are investigated. Their system was able to make a sparse depth map more dense by fusing RGB data.

Feng et al.¹⁰⁸ developed a content-based 3D shape retrieval system. The system uses a low-cost 3D sensor (e.g. Kinect or Realsense) and a database of 3D objects. An ensemble of AEs learns compressed representations of the 3D objects, and the AE act as probabilistic models which output a likelihood score. A domain adaptation layer uses weakly supervised learning to learn cross-domain representations (noisy imagery and 3D computer aided design (CAD)). The system uses the AE encoded objects to reconstruct the objects, and then additional layers rank the outputs based on similarity scores. Segaghat et al.¹¹⁴ use a 3D voxel net that predicts the object pose as well as its class label, since 3D objects can appear very differently based on their poses. The results were tested on LiDAR data, CAD models, and RGB plus depth (RGBD) imagery. Finally, Zelener et al.³¹⁹ labels missing 3D LiDAR points to enable the CNN to have higher accuracy. A

major contribution of this method is creating normalized patches of low-level features from the 3D LiDAR point cloud. The LiDAR data is divided into multiple scan lines, and positive and negative samples. Patches are randomly selected for training. A sliding block scheme is used to classify the entire image.

3.4 Segmentation

Segmentation means to process imagery and divide it into regions (segments) based on the content. Basaeed et al.²⁶⁹ use a committee of CNNs that perform multi-scale analysis on each band to estimate region boundary confidence maps, which are then inter-fused to produce an overall confidence map. A morphological scheme integrates these maps into a hierarchical segmentation map for the satellite imagery.

Couprie et al.²⁵⁴ utilized a multi-scale CNN to learn features directly from RGBD imagery. The image RGB channels and the depth image are transformed through a Laplacian pyramid approach, where each scale is fed to a 3-stage convolutional network that create feature maps. The feature maps of all scales are concatenated (the coarser-scale feature maps are upsampled to match the size of the finest-scale map). A parallel segmentation of the image into superpixels is computed to exploit the natural contours of the image. The final labeling is obtained by the aggregation of the classifier predictions into the superpixels.

In his Master's thesis, Kaiser²⁵⁷ (1) generated new ground truth datasets for three different cities consisting of VHR aerial images with ground sampling distance on the order of centimeters and corresponding pixel-wise object labels, (2) developed FC networks (FCNs) were used to perform pixel-dense semantic segmentation, (3) created two modifications of the FCN architecture were found that gave performance improvements, and (4) utilized transfer learning was shown using FCN model was trained on huge and diverse ground truth data of the three cities, which achieved good semantic segmentations of areas not used for training.

Långkvist et al.¹⁵⁷ applied a CNN to orthorectified multispectral imagery (MSI) and a digital surface model of a small city for a full, fast and accurate per-pixel classification. The predicted low-level pixel classes are then used to improve the high-level segmentation. Various design choices of the CNN architecture are evaluated and analyzed.

3.5 Object Detection and tracking

Object detection and tracking is important in many RS applications. It requires understanding at a higher level than just at the pixel-level. Tracking then takes the process one step further and estimates the location of the object over time.

Diao et al.²²⁸ propose a pixel-wise DBN for object recognition. A sparse RBM is trained in an unsupervised manner. Several layers of RBM are stacked to generate a DBN. For fine-tuning, a supervised layer is attached to the top of the DBN and the network is trained using BP with a sparse penalty constraint. Ondruska et al.²³⁴ used RNN to track multiple objects from 2D laser data. This system uses no hand-coded plant or sensor models (these are required in Kalman filters). Their system uses an end-to-end RNN approach that maps raw sensor data to a hidden sensor space. The system then predicts the unoccluded state from the sensor space data. The system learns directly from the data and does not require a plant or sensor model.

Schwegmann et al.²⁷³ use a very deep Highway Network for ship discrimination in SAR imagery, and a three-class SAR dataset is also provided. Deep networks of 2, 20, 50 and 100 layers

were tested, and the 20 layer network had the best performance. Tang et al.²⁷⁴ utilized a hybrid approach in both feature extraction and machine learning. For feature extraction, the Discrete Wavelet Transform (DWT) LL, LH, HL and HH (L=Low Frequency, H = High Frequency) features from the JPEG2000 CDF9/7 encoder were utilized. The LL features were inputs to a Stacked DAE (SDAE). The high frequency DWT subbands LH, HL and HH are inputs to a second SDAE. Thus the hand-coded wavelets provide features, while the two SDAEs learn features from the wavelet data. After initial segmentation, the segmentation area, major-to-minor axis ratio and compactness, which are classical machine learning features, are also used to reduce false positives. The training data are normalized to zero mean and unity variance, and the wavelet features are normalized to the $[0, 1]$ range. The training batches have different class mixtures, and 20% of inputs are dropped to the SDAEs and there is a 50% dropout in the hidden units. The extreme learning machine is used to fuse the low-frequency and high-frequency subbands. An online-sequential extreme learning machine, which is a feedforward shallow NN, is used for classification.

Two of the most interesting results were developed to handle incomplete training data, and how object detectors emerge from CNN scene classifiers. Mnih et al.²⁴⁶ developed two robust loss functions to deal with incomplete training labeling and misregistration (location of object in map) is inaccurate. A NN is used to model pixel distributions (assuming they are independent). Optimization is performed using expectation maximization. Zhou et al.²³³ show that object detectors emerge from CNNs trained to perform scene classification. They demonstrated that the same CNN can perform both scene recognition and object localization in a single forward pass, without having to explicitly learn the notion of objects. Images had their edges removed such that each edge removal produces the smallest change to the classification discriminant function. This process is repeated until the image is misclassified. The final product of that analysis is a set of simplified images which still have high classification accuracies. For instance, in bedroom scenes, 87% of these contained a bed. To estimate the empirical receptive field (RF), the images were replicated and random 11×11 occluded patches were overlaid. Each occluded image is input to the trained DL network and the activation function changes are observed; a large discrepancy indicates the patch was important to the classification task. From this analysis, a discrepancy map is built for each image. As the layers get deeper in the network, the RF size gradually increases and the activation regions are semantically meaningful. Finally, the objects that emerging in one specific layer indicated that the network was learning object categories (dogs, humans, etc.) This work indicates there is still extensive research to be performed in this area.

3.6 Super-resolution

Super-resolution analysis attempts to infer sub-pixel information from the data. Dong et al.²⁷⁷ utilized a DL network that learns a mapping between the low and high-resolution images. The CNN takes the low-resolution (LR) image as input and outputs the high-resolution (HR) image. In this method, all layers of the DL system are jointly optimized. In a typical super-resolution pipeline with sparse dictionary learning, image patches are densely sampled from the image and encoded in a sparse dictionary. The DL system does not explicitly learn the sparse dictionaries or manifolds for modeling the image patches. The proposed system provides better results than traditional methods and has a fast on-line implementation. The results improve when more data is available or when deeper networks are utilized.

3.7 Weather Forecasting

Weather forecasting attempts to use physical laws combined with atmospheric measurements to predict weather patterns, precipitation, etc. The weather affects virtually every person on the planet, so it is natural that there are several RS papers utilizing DL to improve weather forecasting. DL ability to learn from data and understand highly-nonlinear behavior shows much promise in this area of RS.

Chen et al.¹⁹⁵ utilize DBNs for drought prediction. A three-step process (1) computes the Standardized Precipitation Index (SPI), which is effectively a probability of precipitation, (2) normalizes the SPI, and (3) determines the optimal network architecture (number of hidden layers) experimentally. Firth³¹¹ introduced a Differential Integration Time Step network composed of a traditional NN and a weighted summation layer to produce weather predictions. The NN computes the derivatives of the inputs. These elemental building blocks are used to model the various equations that govern weather. Using time series data, forecast convolutions feed time derivative networks which perform time integration. The output images are then fed back to the inputs at the next time step. The recurrent deep network can be unrolled. The network is trained using backpropagation. A pipelined, parallel version is also developed for efficient computation. The model outperformed standard models. The model is efficient and works on a regional level, versus previous models which are constrained to local levels.

Kovordanyi et al.³¹² utilized NNs in cyclone track forecasting. The system uses a multi-layer NN designed to mimic portions of the human visual system to analyze National Oceanic and Atmospheric Administration's Advanced Very High Resolution Radiometer (NOAA AVHRR) imagery. At the first network level, shape recognition focuses on narrow spatial regions, e.g. detecting small cloud segments. Regions in the image can be processed in parallel using a matrix feature detector architecture. Rotational variations, which are paramount in cyclone analysis, are incorporated into the architecture. Later stages combine previous activations to learn more complex and larger structures from the imagery. The output at the end of the processing system is a directional estimator of cyclone motion. The simulation tool Leabra++ (<http://ccnbook.colorado.edu/>) was used. This tool is designed for simulating brain-like artificial NNs (ANNs). There are a total of five layers in the system: an input layer, three processing layers, and an output layer. During training, images were divided into smaller blocks and rotated, shifted, and enlarged. During training, the network was first given inputs and allowed to settle to steady state. Weak activations were suppressed, with at most k nodes were allowed to stay active. Then the inputs and correct outputs were presented to the network and the weights are all zeroed. The learned weights are a combination of the two schemes. Conditional PCA and contrastive Hebbian learning were used to train the network. The system was very effective if the Cyclone's center varied about 6% or less of the original image size, and less effective if there was more variation.

Shi et al.¹⁹⁸ extended the FC LSTM (FC-LSTM) network that they call ConvLSTM, which has convolutional structures in the input-to-state and state-to-state transitions. The application is precipitation nowcasting, which takes weather data and predicts immediate future precipitation. ConvLSTM used 3D tensors whose last two dimensions are spatial to encode spatial data into the system. An encoding LSTM compresses the input sequence into a latent tensor, while the forecasting LSTM provides the predictions.

3.8 Automated object and target detection and identification

Automated object and automated target detection and identification is an important RS task for military applications, border security, intrusion detection, advanced driver assistance systems, etc. Both automated target detection and identification are hard tasks, because usually there are very few training samples for the target (but almost all samples of the training data are available as non-target), and often there are large variations in aspect angles, lighting, etc.

Ghazi et al.²³⁸ used DL to identify plants in photographs using transfer parameter optimization. The main contributions of this work are (1) a state-of-the-art plant detection transfer learning system, and (2) an extensive study of fine-tuning, iteration size, batch size and data augmentation (rotation, translation, reflection, and scaling). It was found that transfer learning (and fine tuning) provided better results than training from scratch. Also, if training from scratch, smaller networks performed better, probably due to smaller training data. The authors suggest using smaller networks in these cases. Performance was also directly related to the network depth. By varying the iteration sizes, it is seen that the validation accuracies rise quickly initially, then grow slowly. The networks studied are all resilient to overfitting. The batch sizes were varied, and larger batch sizes resulted in higher performance at the expense of longer training times. Data augmentation also had a significant effect on performance. The number of iterations had the most effect on the output, followed by the number of patches, and the batch size had the least significant effect. There were significant differences in training times of the systems. Li et al.¹²² used DL for anomaly detection. In this work, a reference image with pixel pairs (a pair of samples from the same class, and a pair from different classes) is required. By using transfer learning, the system is utilized on another image from the same sensor. Using vicinal pixels, the algorithm recognizes central pixels as anomalies. A 16-level network contains layers of convolution followed by ReLUs. A fully-connected layer then provides output labels.

3.9 Image Enhancement

Image enhancement includes many areas such as pansharpening, denoising, image registration, etc. Image enhancement is often performed prior to feature extraction or other image processing steps. Huang et al.²³⁶ utilize a modified sparse denoising AE (SPDAE), denoted MSDA, which uses the SPDAE to represent the relationship between the HR image patches as clean data to the lower spatial resolution, high spectral resolution MSI image as corrupted data. The reconstruction error drives the cost function and layer-by-layer training is utilized. Quan et al.²⁰⁶ use DL for SAR image registration, which is in general a harder problem than RGB image registration due to high speckle noise. The RBM learns features useful for image registration, and the random sample consensus (RANSAC) algorithm is run multiple times to reduce outlier points.

Wei et al.²⁰⁴ applied a five-layer DL network to perform image quality improvement. In their approach, degraded images are modeled as downsampled images that are degraded by a blurring function and additive noise. Instead of trying to estimate the inverse function, a DL network performs feature extraction at layer 1, then the second layer learns a matrix of kernels and biases to perform non-linear operations to layer 1 outputs. Layers 3 and 4 repeat the operations of layers 1 and 2. Finally, an output layer reconstructs the enhanced imagery. They demonstrated results with non-uniform haze removal and random amounts of Gaussian noise. Zhang et al.²⁰⁵ applied DL to enhance thermal imagery, based on first compensating for the camera transfer function (small-scale and large-scale nonlinearities), and then super-resolution target signature enhancement via

DL. Patches are extracted from low-resolution imagery, and the DL learns feature maps from this imagery. A nonlinear mapping of these feature maps to a HR image are then learned. SGD is utilized to train the network.

3.10 Change Detection

Change detection is the process of utilizing two registered RS images taken at different times and detecting the changes, which can be due to natural phenomenon such as drought or flooding, or due to man-made phenomenon, such as adding a new road or tearing down an old building. We note that there is a paucity of DL research into change detection. Pacifici et al.¹³⁷ used DL for change detection in VHR satellite imagery. The DL system exploits the multispectral and multitemporal nature of the imagery. Saturation is avoided by normalizing data to $[-1, 1]$ range. To mitigate illumination changes, band ratios such as blue/green are utilized. These images are classified according to (1) man-made surfaces, (2) green vegetation, (3) bare soil and dry vegetation, and (4) water. Each image undergoes a classification and a multitemporal operator creates a change mask. The two classification maps and the change mask are fused using an AND operator.

3.11 Semantic Labeling

Semantic labeling attempts to label scenes or objects semantically, such as “there is a truck next to the tree”. Sherrah et al.²⁶³ utilized the recent development of FC NNs (FC-CNNs), which were developed by Long et al.³²⁰ The FC-CNN is applied to remote sensed VHR imagery. In their network, there is no downsampling. The system labels images semantically pixel-by-pixel. Xie et al.²⁹³ used transfer learning to avoid training issues due to scarce training data, transfer learning is utilized. A FC CNN trains in daytime imagery and predicts nighttime lights. The system also can infer poverty data from the night lights, as well as delineating man-made structures such as roads, buildings and farmlands. The CNN was trained on ImageNet and uses the NOAA nighttime remote sensing satellite imagery. Poverty data was derived from a living standards measurement survey in Uganda. Mini-batch gradient descent with momentum, random mirroring for data augmentation, and 50% dropout was used to help avoid overfitting. The transfer learning approach gave higher performance in accuracy, F1 scores, precision and area under the curve.

3.12 Dimensionality reduction

HSI are inherently highly dimensional, and often contain highly correlated data. Dimensionality reduction can significantly improve results in HSI processing. Ran et al.¹⁹² split the spectrum into groups based on correlation, then apply m CNNs in parallel, one for each band group. The CNN output are concatenated and then classified via a two-layer FC-CNN. Zabalza et al.¹⁹³ used segmented SAEs are utilized for dimensionality reduction. The spectral data are segmented into k regions, each of which has a SAE to reduce dimensionality. Then the features are concatenated into a reduced profile vector. The segmented regions are determine by using the correlation matrix of the spectrum. In Ball et al.³²¹, it was shown that band selection is task and data dependent, and often better results can be found by fusing similarity measures versus using correlation, so both of these methods could be improved using similar approaches. Dimensionality reduction is an important processing step in many classification algorithms^{322,323}, pixel unmixing^{30,31,324–328}, etc.

4 Unsolved challenges and opportunities for DL in RS

DL applied to RS has many challenges and open issues. Table 4 gives some representative DL and FL survey papers and discusses their main content. Based on these reviews, and the reviews of many survey papers in RS, we have identified the following major open issues in DL in RS. Herein, we focus on unsolved challenges and opportunities as it relates to (i) inadequate data sets (4.1), (ii) human-understandable solutions for modelling physical phenomena (4.2), (iii) Big Data (4.3), (iv) non-traditional heterogeneous data sources (4.4), (v) DL architectures and learning algorithms for spectral, spatial and temporal data (4.5), (vi) transfer learning (4.6), (vii) an improved theoretical understanding of DL systems (4.7), (viii) high barriers to entry (4.8), and (ix) training and optimizing the DL (4.9).

4.1 Inadequate data sets

Open Question #1a: How can DL systems work well with limited datasets?

There are two main issues with most current RS data sets. Table 5 provides a summary of the more common open-source datasets for the DL papers utilizing HSI data. Many of these papers utilized custom datasets, and these are not reported. Table 5 shows that the most commonly used datasets were Indian Pines, Pavia University, Pavia City Center, and Salinas.

A very detailed online table (too large to put in this paper) is provided which lists each paper cited in Table 3. For each paper, a summary of the contributions is given, the datasets utilized are listed, and the papers are categorized in areas (e.g. HSI/MSI, SAR, 3D, etc.). The interested reader can find this at <http://www.cs-chan.com/source/FADL/>.

While these are all good datasets, the accuracies from many of the DL papers are nearly saturated. This is shown clearly in Table 6. Table 6 shows results for the HSI DL papers against the commonly-used datasets Indian Pines, Kennedy Space Center, Pavia City Center, Pavia University, Salinas, and the Washington DC Mall. First, OA results must be taken with a grain of salt, since (1) the number of training samples per class can differ for each paper, (2) the number of testing samples can also differ, (3) classes with few relative training samples can even have 0% overall accuracy, and if there is a large number of test samples of the other classes, the final overall accuracies can still be high. Nevertheless, it is clear from examination of the table that the Indian Pines, Pavia City Center, Pavia University and Salinas datasets are basically saturated.

In general, it is good to compare new methods to commonly-used datasets, new and challenging datasets are required. Cheng et al.^{1,346} point out that many existing RS datasets lack image variation, diversity, and have a small number of classes. Datasets are also saturating with accuracy. They created a large-scale benchmark dataset, "NWPU-RESISC45", which attempts to address all of these issues, and made it available to the RS community. The RS community can also benefit from a common practice in the CV community: publishing both datasets and algorithms online, allowing for more comparisons. A typical RS paper may only test their algorithm on two or three images and against only a few other methods. In the CV community, papers usually compare against a large amount of other methods and with many datasets, which may provide more insight about the proposed solution and how it compares to previous work.

An extensive table of all the datasets utilized in the papers reviewed for this survey paper is made available online, because it is too large to include in this paper. The table is available at <http://www.cs-chan.com/source/FADL/>. The table lists the dataset name, briefly

describes the datasets, provides a URL (if one is available), and a reference. Hopefully, this table will assist researchers starting work and looking for publicly available datasets.

Open Question #1b: How can DL systems work well with limited training data?

The second issue is that most RS data has a very small amount of training data available. Ironically, in the CV community, DL has an insatiable hunger for larger and larger data sets (millions or tens of millions of training images), while in the RS field, there is also a large amount of imagery, however, there is usually only a small amount with labeled training samples. RS training data is expensive, error-prone, and usually requires some expert interpretation, which is typically expensive (in terms of time, effort involved, and money) and often requires large amounts of field work and many hours or days post processing the data. Many DL systems, especially those with large numbers of parameters, require large amounts of training data, or else they can easily overtrain and not generalize well. This problem has also plagued other shallow systems as well, such as SVMs.

Approaches used to mitigate small training samples are (1) transfer learning, where one trains on other imagery to obtain low-level to mid-level features which can still be used, or on other images from the same sensor - Transfer learning is discussed in section 4.6 below; (2) data augmentation, including affine transformations, rotations, small patch removal, etc.; (3) using ancillary data, such as data from other sensor modalities (e.g. LiDAR, digital elevation models (DEMs), etc.); and (4) unsupervised training, where training labels are not required, e.g. AEs and SAEs. SAEs that have a diabolo shape will force the AE network to learn a lower-dimensional representation.

Ma et al.¹⁶⁹ utilized a DAE and employed a collaborative representation-based classification, where each test sample can be linearly represented by the training samples in the same class with the minimum residual. In classification, features of each sample are approximated with a linear combination of features of all training sample within each class, and the label can be derived according to the class which best approximates the test features. Interested reader is referred to references 46–48 in¹⁶⁹ for more information on collaborative representation. Tao et al.³⁴⁹ utilized a Stacked Sparse AE (SSAE) that was shown to be very generalizable and performed well in cases when there were limited training samples. Ghamasi et al.²⁰⁷ use Darwinian particle swarm optimization in conjunction with CNNs to select an optimal band set for classifying HSI data. By reducing the input dimensionality, fewer training samples are required. Yang et al.¹⁸¹ utilized dual CNNs and transfer learning to improve performance. In this method, the lower and middle layers can be trained on other scenes, and train the top layers on the limited training samples. Ma et al.¹⁷⁰ imposed a relative distance prior on the SAE DL network to deal with training instabilities. This approach extends the SAE by adding the new distance prior term and corresponding SGD optimization. LeCun reviews a number of unsupervised learning algorithms using AEs, which can possibly aid when training data is minimal³⁵⁰. Pal³⁵¹ reviews kernel methods in RS and argues that SVMs are a good choice when there are a small number of training samples. Petersson et al.³³⁵ suggest using SAEs to handle small training samples in HSI processing.

4.2 Human-understandable solutions for modelling physical phenomena

Open Question #2a: How can DL improve model-based RS?

Many RS applications depend on models (e.g. a model of crop output given rain, fertilizer and soil nitrogen content, and time of year), many of which are very complicated and often highly nonlinear. Model outputs can be very inaccurate if the models don't adequately capture the true input data and properly handle the intricate inter-relationships between input variables.

Abdel-Rahman et al.³⁵² pointed out that more accurate estimation of nitrogen content and water availability can aid biophysical parameter estimation for improving plant yield models. Ali et al.³⁵³ examine biomass estimation, which is a nonlinear and highly complex problem. The retrieval problem is ill-posed and the electromagnetic response is the complex result of many contributions. The data pixels are usually mixed, making this a hard problem. ANNs and support vector regression have shown good results. They anticipate that DL models can provide good results. Both Adam et al.³⁵⁴ and Ozesmi et al.³⁵⁵ agree that there is need for improvement in wetland vegetation mapping. Wetland species are hard to detect and identify compared to terrestrial plants. Hyperspectral sensing with narrow bandwidths in frequency can aid. Pixel unmixing is important since canopy spectra are similar and combine with underlying hydrologic regime and atmospheric vapor. Vegetation spectra highly correlated among species, making separation difficult. Dorigo et al.³⁵⁶ analyzed inversion-based models for plant analysis, which is inherently an ill-posed and hard task. They found that using ANN inversion techniques have shown good results. DL may be able to help improve results. Canopy reflections are governed by large number of canopy elements interacting and by external factors. Since DL networks can learn very complex non-linear systems, it seems like there is much room for improvement in applying DL models. DBNs or other DL systems seem like a natural fit for these types of problems.

Kuenzer et al.³⁵⁷ and Wang et al.³⁵⁸ assess biodiversity modeling. Biodiversity occurs at all levels from molecular to individual animals, to ecosystem, to global. This requires a large variety of sensors and analysis at multiple scales. However, a main challenge is low temporal resolution. There needs to be a focus beyond just pixel-level processing, and utilizing spatial patterns and objects. DL systems have been shown to learn hierarchical features, with smaller scale features learned at the beginning of the network, and more complex and abstract features learned in the deeper portions.

Open Question #2b: What tools and techniques are required to “understand” how the DL works?

It is also worth mentioning that many of these applications involve biological and scientific end-users, who will definitely want to understand how the DL systems work. For instance, a linear model that models some biological process is easily understood - both the mathematical model and the statistics resulting from estimating the model parameters are well understood by scientists and biologists. However, a DL system can be so large and complex as to defy analysis. We note that this is not specific to RS, but a general problem in the broader DL community.

The DL system is seen by many researchers, especially scientists and RS end-users, as a black box that is hard to understand what is happening “under the hood”. Egmont-Peterson et al.³³¹ and Fassnacht et al.³⁵⁹ both state that disadvantages of NNs are understanding what they are actually doing, which can be difficult to understand. In many RS applications, just making a decision is not enough; people need to understand how reliable the decision is and how the system arrived at that decision. Ali et al.³⁶⁰ also echo this view in their review paper on improving biomass estimation. Visualization tools which show the convolutional filters, learning rates, and tools with deconvolution capabilities to localize the convolutional firings are all helpful^{39,361–364}. Visualization of what the DL is actually learning is an open area of research. Tools and techniques capable of visualizing what the network is learning and measures of how robust the network is (estimating how well it may generalize) would be of great benefit to the RS community (and the general DL community).

4.3 Big Data

Open Question #3: What happens when DL meets Big Data?

As already discussed in Section 2.4.3, a number of mathematics, algorithms and hardware have been put forth to date relative to large scale DL networks and DL in Big Data. However, this challenge is not close to being solved. Most approaches to date have focused on Big Data challenges in RGB or RGBD data for tasks like face and object detection or speech. With respect to remote sensing, we have many of the same problems as CV, but there are unique challenges related to different sensors and data. First, we can break Big Data into its different so-called “parts”, e.g., volume, variety and velocity. With respect to DBNs, CNNs, AEs, etc., we are primarily concerned with creating new robust and distributed mathematics, algorithms and hardware that can ingest massive streams of large, missing, noisy data from different sources, such as sensors, humans and machines. This means being able to combine image stills, video, audio, text, etc., with symbolic and semantic variation. Furthermore, we require real-time evaluation and possibly online learning. As Big Data in DL is a large topic, we restrict our focus herein to factors that are unique to remote sensing.

The first factor that we focus on is high spatial and more-so spectral dimensionality. Traditional DLs operate on relatively small grayscale or RGB imagery. However, SAR imagery has challenges due to noise, and MSI and HSI can have from four to hundreds to possibly thousands of channels. As Arel et al.⁷ pointed out, a very difficult question is how well DL architectures scale with dimensionality. To date, preliminary research has tried to combat dimensionality by applying dimensionality reduction or feature selection prior to DL, e.g., Benediktsson et al.³⁶⁵ reference different band selection, grouping, feature extraction and subspace identification in HSI remote sensing.

Ironically, most RS areas suffer from a lack of training data. Whereas they may have massive amounts of temporal and spatial data, there may not be the seasonal variations, times of day, object variation (e.g., plants, crops, etc.), and other factors that ultimately lead to sufficient variety needed to train a DL model. For example, most online hyperspectral data sets have little-to-no variety and it is questionable about what they are, and really can at that, learn. In stark contrast, most DL systems in CV use very large training sets, e.g., millions or billions of faces in different illuminations, poses, inner class variations, etc. Unless the remote sensing DL applies a method like transfer learning, DL in RS often have very limited training data. For example, in¹⁷⁰ Ma et al. tried to address this challenge by developing a new prior to deal with the instability of parameter estimation for HSI classification with small training samples. The SAE is modified by adding the relative distance prior in the fine-tuning process to cluster the samples with the same label and separate the ones with different labels. Instead of minimizing classification error, this network enforces intra-class compactness and attempts to increase inter-class discrepancy.

4.4 Non-traditional heterogeneous data sources

Open Question #4a: How can DL work with non-traditional data sources?

Non-traditional data sources, such as twitter, YouTube, etc. offer data that can be useful to RS. These methods will probably never replace RS, but usually offer benefits to augment RS data, or provide quality real-time data before RS methods, which usually take longer, can provide RS-based data.

Fohringer et al.³⁶⁶ utilized information extracted from social media photos to enhance RS data for flood assessments. They found one major challenge was filtering posts to a manageable amount of relevant ones to further assess. The data from Twitter and Flickr proved useful for flood depth estimation prior to RS-based methods, which typically take 24-28 hours. Frias-Martinez et al.³⁶⁷ take advantage of large amounts of geolocated content in social media by analyzing Twitter tweets as a complimentary source of data for urban land-use planning. Data from Manhattan (New York, USA), London (UK), and Madrid (Spain) was analyzed using a self-organizing map³⁶⁸ followed by a Voronoi tessellation. Middleton et al.³⁶⁹ match geolocated tweets and created real-time crisis maps via statistical analysis, which are compared to the US National Geospatial Agency post-event impact assessments. A major issue is that only about 1% of tweets contain geolocation data. The tweets usually follow a pattern of a small number of first-hand reports and many re-tweets and comments. High-precision results were obtained. Singh et al.³⁷⁰ aggregate user's social interest about any particular theme from any particular location into so called "social pixels", which are amenable to media processing techniques (e.g., segmentation, convolution), which allow semantic information to be derived. They also developed a declarative operator set to allow queries to visualize, characterize, and analyze social media data. Their approach would be a promising front-end to any social media analysis system. In the survey paper of Sui and Goodchild³⁷¹, the convergence of Geographic Information System (GIS) and social media are examined. They observed that GIS has moved from software helping a user at a desk to a means of communicating earth surface data to the masses (e.g. OpenStreetMap, Google Maps, etc.).

In all of the above mentioned methods, DL can play a significant role of parsing data, analyzing data and estimating results from the data. It seems that social media is not going away, and data from social media can often be used to augment RS data in many applications. Thus the question is what novel work awaits the researcher in the area of using DL to combine non-traditional data sources with RS?

Open Question #4b: How does DL ingest heterogeneous data?

Fusion can take place at numerous so-called "levels", including signal, feature, algorithm and decision. For example, Signal In Signal Out (SISO) is where multiple signals are used to produce a signal out. For \Re -valued signal data, a common example is the trivial concatenation of their underlying vectorial data, i.e., $X = \{\hat{x}_1, \dots, \hat{x}_N\}$ becomes $[\hat{x}_1 \hat{x}_2 \dots \hat{x}_N]$ of length $|\hat{x}_1| + \dots + |\hat{x}_N|$. Feature In Feature Out (FIFO), which is often related to if not the same as SISO, is where multiple features are combined, e.g., a HOG and a Local Binary Pattern (LBP), and the result is a new feature. One example is Multiple Kernel Learning (MKL), e.g., ℓ - p norm genetic algorithm MKL (GAMKLp)³⁷². Typically the input is N \Re -valued Cartesian spaces and the result is a Cartesian space. Most often, one engages in MKL to search for space in which pattern obey some property, e.g., they are nicely linearly separable and a machine learning tool like a SVM can be employed. On the other hand, Decision In Decision Out (DIDO), e.g., the Choquet integral (ChI), is often used for the fusion of input from *decision makers*, e.g., human experts, algorithms, classifiers, etc.³⁷³ Technically speaking, a CNN is typically a Signal In Decision Out (SIDO) or Feature In Decision Out (FIDO) system. Internally, the *feature learning* part of the CNN is a SIFO or FIFO and the classifier is a FIDO. To date, most DL approaches have "fused" via (1) concatenation of \Re -valued input data (SISO or FIFO) relative to a single DL, (2) each source has its own DL, minus classification, that is later combined into a single DL, or (3) multiple DLs are used, one for each source, and their results are once again concatenated and subjected to the classifier (either a MLP, SVM or other classifier).

Herein, we highlight the challenges of syntactic and semantic fusion. Most DL approaches to date syntactically have addressed how N things, which are typically homogeneous mathematically, can be ingested by a DL. However, the more difficult challenge is semantically how should these sources be combined, what is a proper architecture, what is learned (can we understand it) and why should we trust the solution. This is of particular importance to numerous challenges in remote sensing that require a physically meaningful/grounded solution, e.g., model-based approaches. The most typical example of fusion in remote sensing is the combining of data from two (or more) sensors. Whereas there may be semantic variation but little-to-no semantic variation, e.g., both are possibly \mathbb{R} -valued vector data, the reality is most sensors record objective evidence about our universe. However, if human information (e.g., linguistic or text) is involved or algorithmic outputs (e.g., binary decisions, labels/symbols, probabilities, etc.), fusion becomes increasingly more difficult syntactically and semantically. Many theoretical (mathematical and philosophical) investigations, which are beyond the scope of this work, have concerned themselves with how to meaningfully combine objective vs. subjective data/information, qualitative vs. quantitative data/information, or evidences with beliefs other numerous other flavors information. It is a naive and dangerous belief that one can simply just “cram” data/information into a DL and get a meaningful and useful result. *How is fusion occurring? Where is it occurring?* Fusion is further compounded if one is using uncertain information, e.g., probabilistic, possibilities, or other interval or distribution-based input. The point is, heterogeneous, be it mathematical representation, associated uncertainty, etc., is a real and serious challenge and if the DL community wishes to fuse multiple inputs or sources (humans, sensors and algorithms) then DL must theoretically rise to the occasion to ensure that the architectures and what is subsequently being learned is useful and meaningful. Example, and really preliminary at that, associated DL works to date include fusing hyperspectral with LiDAR³⁷⁴ (two sensors yielding objective data) and text with imagery or video³⁷⁵ (thus high-level human information sensor data), to name a few. The point is, the question remains, how can/does DL fuse data/information arising from one or more sources?

4.5 DL architectures and learning algorithms for spectral, spatial and temporal data

DL Open Question #5: What architectural extensions will DL systems require in order to tackle complicated RS problems?

Current DL architectures, components (e.g. convolution), and optimization techniques may not be adequate to solve complex RS problems. In many cases, researchers have developed novel network architectures, new layer structures with their associated SGD or BP equations for training, or new combinations of multiple DL networks. This problem is also an open issue in the broader CV community. This question is at the heart of DL research. Other questions related to the open issues are:

- *What architecture should be used?*
- *How deep should a DL system be, and what architectural elements will allow it to work at that depth?*
- *What architectural extensions (new components) are required to solve this problem?*
- *What training methods are required to solve this problem?*

We examine several general areas where DL systems have evolved to handle RS data: (i) multi-sensor processing, (ii) utilizing multiple DL systems, (iii) Rotation and displacement-invariant DL

systems, (iv) new DL architectures, (v) SAR, (vi) Ocean and atmospheric processing, (vii) 3D processing, (viii) spectral-spatial processing, and (ix) multi-temporal analysis. Furthermore, we examine some specific RS applications noted in several RS survey papers as areas that DL can benefit: (a) Oil spill detection, (b) pedestrian detection, (c) urban structure detection, (d) pixel unmixing, and (e) road extraction. This is by no means an exhaustive list, but meant to highlight some of the important areas.

Multi-Sensor Processing: Chen et al.¹⁹¹ utilize two deep networks, one analyzing HSI pixel neighbors (spatial data), and the other LiDAR data. The outputs are stacked and a FC and logistic regression lay provides outputs. Huang et al.²³⁶ use a modified sparse DAE (MSDAE) to train the relationship between HR and LR image patches. The stacked MSDAE (S-MSDAE) are used to pretrain a DNN. The HR MSI image is then reconstructed from the observed LR MSI image using the trained DNN.

Multi-DL system: In certain problems, multiple DL systems can provide significant benefit. Chen et al.²⁹⁸ utilize parallel DNNs with no cross-connections to both speed up processing and provide good results in vehicle detection from satellite imagery. Ciresan et al.¹¹⁹ utilize multiple parallel DNNs that are averaged for image classification. Firth et al.³¹¹ use 186 RNNs to perform accurate weather prediction. Hou et al.¹⁵² use RBMs to train from polarimetric SAR data and a three-layer DBN is used for classification. Kira et al.³⁷⁶ used stereo-imaging for robotic human detection, utilizing a CNN which was trained on appearance and stereo disparity-based features, and a second CNN, which is used for long-range detection. Marmanis et al.²⁶⁰ utilized an ensemble of CNNs to segment VHR aerial imagery using a FCN to perform pixel-based classification. They trained multiple networks with different initializations and average the ensemble results. The authors also found errors in the dataset, Vaihingen³⁷⁷.

Rotation- and displacement-invariant systems: Some RS problems require systems that are rotation and displacement-invariant. CNNs have some robustness to translation, but not in general to rotations. Cheng et al.²²⁶ incorporated a rotation-invariant layer into a DL CNN architecture to detect objects in satellite imagery. Du et al.¹²⁷ developed a displacement- and rotation-insensitive deep CNN for SAR Automated Target Recognition (ATR) processing that is trained by augmented dataset and specialized training procedure.

Novel DL architectures: Some problems in RS require novel DL architectures. Dong et al.²⁷⁷ use a CNN that takes the LR image and outputs the HR image. He et al.¹⁵¹ proposed a deep stacking network for HSI classification that utilizes nonlinear activations in the hidden layers and does not require SGD for training. Kotschieder et al.¹⁵⁶ developed deep neural decision forests, which uses a stochastic and differentiable decision tree model that steers the representation learning usually conducted in the initial layers of a deep CNN. Lee et al.¹⁵⁸ analyze HSI by applying multiple local 3D convolutional filters of different sizes jointly exploiting spatial and spectral features, followed by a fully-convolutional layers to predict pixel classes. Zhang et al.¹⁸⁶ propose GBRCN to classify VHR satellite imagery. Ouyang et al.²⁰² developed a probabilistic parts-detector based model to robustly handle human detection with occlusions are large deformations utilizing a discriminative RBM to learn the visibility relationship among overlapping parts. The RBM has three layers that handle different size parts. Their results can possibly be improved by adding additional rotation invariance.

Novel DL SAR architectures: SAR imagery has unique challenges due to noise and the grainy nature of the images. Geng et al.¹⁴⁹ developed a deep convolutional AE, which is a combination of a CNN, AE, classification, and post-processing layers to classify high-resolution SAR images.

Hou et al.¹⁵² developed a polarimetric SAR DBN. Filters are extracted from the RBMs and a final three-layer DBN performs classification. Liu et al.²¹⁰ utilize a Deep Sparse Filtering Network to classify terrain using polarimetric SAR data. The proposed network is based on sparse filtering³⁷⁸, and the proposed network performs a minimization on the output L_1 norm to enforce sparsity. Qin et al.¹⁷⁸ performed object-oriented classification of polarimetric SAR data using a RBM and built an adaptive boosting framework (AdaBoost³⁷⁹) vice a stacked DBN in order to handle small training data. They also put forth the RBM-AdaBoost algorithm. Schwegmann et al.²⁷³ utilized a very deep Highway Network configuration as a ship discrimination stage for SAR ship detection. They also presented a three-class SAR dataset that allows for more meaningful analysis of ship discrimination performances. Zhou et al.³⁸⁰ proposed a three-class change detection approach for multitemporal SAR images using a RBM. These images either have increases or decreases in the backscattering values for changes, so the proposed approach classifies the changed areas into the positive and negative change classes, or no change if none is detected.

Oceanic and atmospheric studies: Oceanic and atmospheric studies present unique challenges to DL systems that require novel developments. Ducournau et al.²⁷⁸ developed a CNN architecture, which analyzes sea surface temperature fields and provides a significant gain in terms of peak signal-to-noise ratio compared to classical downscaling techniques. Shi et al.¹⁹⁸ extended the FC-LSTM network that they call ConvLSTM, which has convolutional structures in the input-to-state and state-to-state transitions for precipitation nowcasting.

3D Processing: Guan et al.²³⁹ use voxel-based filtering removes ground points from LiDAR data, then a DL architecture generates high-level features from the trees 3D geometric structure. Haque et al.¹⁰⁹ utilize both of CNN and RNN to process 4D spatio-temporal signatures to identify humans in the dark.

Spectral-spatial HSI processing: HSI processing can be improved by fusion of spectral and spatial information. Ma et al.¹⁶⁹ propose a spatial updated deep AE which adds a similarity regularization term to the energy function to enforce spectral similarity. The regularization terms is a cosine similarity term (basically the spectral angle mapper) between the edges of a graph, which the nodes are samples, which enforces keeping the sample correlations. Ran et al.¹⁹² classify HSI data by learning multiple CNN-based submodels for each correlated set of bands, while in parallel a conventional CNN learns spatial-spectral characteristics. The models are combined at the end. Li et al.¹⁶² incorporated vicinal pixel information by combining the center pixel and vicinal pixels, and utilizing a voting strategy to classify the pixels.

Multi-temporal analysis: Multi-temporal analysis is a subset of RS analysis that has its own challenges. Data revisit rates are often long, and ground-truth data is even more expensive as multiple imagery sets have to be analyzed, and images must be co-registered for most applications.

Jianya et al.²⁵ review multi-temporal analysis, and observe that it is hard, the changes are often non-linear, and changes occur on different timescales (seasons, weeks, years, etc.). The process from ground objects to images is not reversible, and image change to earth change is a very difficult task. Hybrid method involving classification, object analysis, physical modeling, and time series analysis can all potentially benefit from DL approaches. Arel et al.⁷ ask if DL frameworks can understand trends over short, medium and long times? This is an open question for RNNs.

Change detection is an important subset of multi-temporal analysis. Hussain et al.²⁴ state that change detection can benefit from texture analysis, accurate classifications, and ability to detect anomalies. DL has huge potential to address these issues, but it is recognized that DL algorithms are not common in image processing software in this field and large training sets and large training

times may also be required. In cases of non-normal distributions, ANNs have shown superior results to other statistical methods. They also recognize that DL-based change detection can go beyond traditional pixel-based change detection methods. Tewkesbury et al.³⁸¹ observe that change detection can occur at the pixel, kernel (group of pixels), image-object, multi-temporal image-object (created by segmenting over time series), vector-polygon, and hybrid. While the pixel level is suitable for many applications, hybrid approaches can yield better results in many cases. Approaches to change detection can utilize DL to (1) co-register images and (2) detect changes at hierarchical (e.g. more than just pixel levels).

Some selected specific applications that can benefit from DL analysis: This section discusses some selected applications where DL can benefit the results. This is by no means an exhaustive list, and many other areas can potentially benefit from DL approaches. In oil spill detection, Brekke et al.³⁸² point out that training data is scarce. Oil spills are very rare, which usually means oil spill detection approaches are anomaly detectors. Physical proximity, slick shape, and texture play important roles. SAR imagery is very useful, but there are look-alike phenomena that cause false positives. Algal information fusion from optical sensors and probability models can aid detection. Current algorithms are not reliable, and DL has great promise in this area.

In the area of pedestrian detection, Dollar et al.²¹ discuss that many images with pedestrians have only a small number of pixels. Robust detectors must handle occlusions. Motion features can achieve very high performance, but few have utilized them. Context (ground plane) approaches are needed, especially at lower resolutions. More datasets are needed, especially with occlusions. Again, DL can provide significant results in this area.

For urban structure analysis, Mayer et al.³⁸³ report that scale-space analysis is required due to different scales of urban structures. Local contexts can be utilized in the analysis. Analyzing parts (dormers, windows, etc) can improve results. Sensor fusion can aid results. Object variability is not treated sufficiently (e.g. highly non-planar roofs). The DL system's ability to learn hierarchical components and learn parts makes it a good candidate for improving results in this area.

In pixel unmixing, Shi et al.³⁴ and Somers et al.³⁸⁴ review papers both point out that whether an unmixing system uses a spectral library or extracts endmembers spectra from the imagery, the accuracy highly depends on the selection of appropriate endmembers. Adding information from a spatial neighborhood can enhance the unmixing results. DL methods such as CNNs or other tailored systems can potentially inherently combine spectral and spatial information. DL systems utilizing denoising, iterative unmixing, feature selection, spectral weighting, and spectral transformations can benefit unmixing.

Finally in the area of road extraction, Wang et al.³³⁷ point out that roads can have large variability, are often curves, and can change size. In bad weather, roads can be very hard to identify. Object shadows, occlusions, etc. can cause the road segmentation to miss sections. Multiple models and multiple features can improve results. The natural ability of DL to learn complicated hierarchical features from data makes them a good candidate for this application area also.

4.6 Transfer Learning

Open Question #6: How can DL in RS successfully utilize transfer learning?

In general, we note that transfer learning is also an open question in DL in general, not just in DL related to remote sensing. Section 4.9 discusses transfer learning in the broader context of the entire field of DL, which this section discusses transfer learning in a RS context.

According to Tuia et al.³⁸⁵ and Pan et al.²⁸, transfer learning seeks to learn from one area to another in one of four ways: instance-transfer, feature-representation transfer, parameter transfer, and relational-knowledge transfer. Typically in remote sensing, when changing sensors or changing to a different part of a large image or other imagery collected at different times, the transfer fails. Remote sensing systems need to be robust, but doesn't necessarily require near-perfect knowledge. Transfer between HSI images where the number and types of endmembers are different has very few studies. Ghazi et al.²³⁸ suggest that two options for transfer learning are to (1) utilize pre-trained network and learn new features in the imagery to be analyzed, or (2) fine-tune the weights of the pre-trained network using the imagery to be analyzed. The choice depends on the size and similarity of the training and testing datasets.

There are many open questions about transfer learning in HSI RS:

- *How does HSI transfer work when the number and type of endmembers are different?*
- *How can DL systems transfer low-level to mid-level features from other domains into RS?*
- *How can DL transfer learning be made robust to imagery collected at different times and under different atmospheric conditions?*

Although in general these open questions remain, we do note that the following papers have successfully utilized transfer learning in RS applications: Yang et al.¹⁸¹ trained on other remote sensing imagery and transferred low-level to mid-level features to other imagery. Othman et al.²¹⁸ utilized transfer learning by training on the ILSVRC-12 challenge data set, which has 1.2 million 224×224 RGB images belonging to 1,000 classes. The trained system was applied to the UC Merced Land Use³⁸⁶ and Banja-Luka³⁸⁷ datasets. Iftene et al.¹⁵⁴ applied a pretrained CaffeNet and GoogLeNet models on the ImageNet dataset, and then applying the results to the VHR imagery denoted the WHU-RS dataset.^{388,389} Xie et al.²⁹³ trained a CNN on night-time imagery and used it in a poverty mapping. Ghazi et al.²³⁸ and Lee et al.³⁹⁰ used a pre-trained networks AlexNet, GoogLeNet and VGGNet on the LifeCLEF 2015 plant task dataset³⁹¹ and MalayaKew dataset³⁹² for plant identification. Alexandre²²³ used four independent CNNs, one for each channel of RGBD, instead of using a single CNN receiving the four input channels. The four independent CNNs are then trained in a sequence by using the weights of a trained CNN as starting point to train the other CNNs that will process the remaining channels. Ding et al.³⁹³ utilized transfer learning for automatic target recognition from mid-wave infrared (MWIR) to longwave IR (LWIR). Li et al.¹²² used transfer learning by utilizing pixel-pairs based on reference data with labeled sampled using Airborne Visible / Infrared Imaging Spectrometer (AVIRIS) hyperspectral data.

4.7 An improved theoretical understanding of DL systems

DL Open Question #7: What new developments will allow researchers to better understand DL systems theoretically?

The CV and NN image processing communities understand BP and SGD, but until recently, researchers struggled to train deep networks. One issue has been identified as vanishing or exploding gradients^{394,395}. Using normalized initialization and normalization layers can help alleviate this problem. Using special architectures, such as deep residual learning⁴¹ or highway networks³⁹⁶ feed data into the deeper layers, thus allowing very deep networks to be trained. FC networks³²⁰ have achieved success in pixel-based semantic segmentation tasks, and are another alternative to going deep. Sokolić et al.³⁹⁷ determined that the spectral norm of the NN's Jacobian matrix in

the neighbourhood of the training samples must be bounded in order for the network to generalize well. All of these methods deal with a central problem in training very deep NNs: The gradients must not vanish, explode, or become too uncorrelated, or else learning is severely hindered.

The DL field needs practical (and theoretical) methods to go deep, and ways to train efficiently with good generalization capabilities. Many DL RS systems will probably require new components, and these networks with the new components need to be analyzed to see if the methods above (or new methods not yet invented) will enable efficient and robust network training. Egmont-Peterson et al.³³¹ point out that DL training is sensitive to the initial training samples, and it is a well-known problem in SGD and BP of potentially reaching a local minimum solution but not being at the global minimum. In the past, seminal papers such as Hinton's³⁹⁸ which allow efficient training of the network, allow researchers to break past a previously difficult barrier. What new algorithmic and theoretical developments will spur the next large surge in DL?

4.8 *High barriers to entry*

DL Open Question #8: How to best handle high entry barriers to DL?

Most DL papers assume that the reader is familiar with DL concepts, backpropagation, etc. This is in reality a steep learning curve that takes a long time to master. Good tutorials and online training can aid students and practitioners who are willing to learn. Implementing BP or SGD on a large DL system is a difficult task, and simple errors can be hard to determine. Furthermore, BP can fail in large networks, so alternate architectures such as highway nets are usually required.

Many DL systems have a large number of parameters to learn, and often require large amounts of training data. Computers with GPUs and GPU-capable DL programs can greatly benefit by offloading computations onto the GPUs. However, multi-GPU systems are expensive, and students often use laptops that cannot be equipped with a GPU. Some DL systems run under Microsoft Windows, while others run under variants of Linux (e.g. Ubuntu or Red Hat). Furthermore, DL systems are programmed in a variety of languages, including Matlab, C, C++, Lua, Python, etc. Thus practitioners and researchers have a potentially steep learning curve to create custom DL solutions.

Finally, the large variety of data types in remote sensing, including RGB imagery, RGBD imagery, MSI, HSI, SAR, LiDAR, stereo imagery, tweets, GPS data, etc., all of which may require different architectures of DL systems. Often, many of the tasks in the RS community require components that are not part of a standard DL library tool. A good understanding of DL systems and programming is required to integrate these components into off-the-shelf DL systems.

4.9 *Training*

Open Question #9: How to train and optimize the DL system?

Training a DL system can be difficult. Large systems can have millions of parameters. There are many methods that DL researchers use to effectively train systems. These methods are discussed below.

Data imputation: Data imputation³⁹⁸ is important in RS, since there are often a small number of training samples. In imagery, image patches can be extracted and stretched with affine transformations, rotated, and made lighter or darker (scaling). Also, patches can be zeroed (removed) from

training data to help the DL be more robust to occlusions. Data can also be augmented by simulations. Another method that can be useful in some circumstances is domain transfer, discussed below (transfer learning).

Pre-training: Erhan et al.³⁹⁹ performed a detailed study trying to answer the questions “*How does unsupervised pre-training work?*” and “*Why does unsupervised pre-training help DL?*”. Their empirical analysis shows that unsupervised pre-training guides the learning towards attraction basins of minima that support better generalization and pre-training also acts as a regularizer. Furthermore, early training example have a large impact on the overall DL performance. Of course, these are experimental results, and results on other datasets or using other DL methods can yield different results. Many DL systems utilize pre-training followed by fine-tuning.

Transfer Learning: Transfer learning is also discussed in section 4.6. Transfer learning attempts to transfer learned features (which can also be thought of as DL layer activations or outputs) from one image to another, from one part of an image to another part, or from one sensor to another. This is a particularly thorny issue in RS, due to variations in atmosphere, lighting conditions, etc. Pan et al.²⁸ point out that typically in remote sensing, when changing sensors or changing to a different part of a large image or other imagery collected at different times, the transfer fails. Remote sensing systems need to be robust, but they don’t necessarily require near-perfect knowledge. Also, transfer between images where the number and types of endmembers are different has very few studies. Zhang et al.³ also cite transfer learning as an open issue in DL in general, and not just in RS.

Regularization: Regularization is defined by Goodfellow et al.¹¹ as “any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.” There are many forms of regularizer - parameter size penalty terms (such as the L_2 or L_1 norm, and other regularizers that enforce sparse solutions; diagonal loading of a matrix so the matrix inverse (which is required for some algorithms) is better conditioned; Dropout and early stopping (both are described below); adding noise to weights or inputs; semi-supervised learning, which usually means that some function that has a very similar representation to examples from the same class is learned by the NN; Bagging (combining multiple models); and adversarial training, where a weighted sum of the sample and an adversarial sample is used to boost performance. The interested reader is referred to chapter 7 of¹¹ for further information. An interesting DL example in RS is, Mei et al.¹⁷², who utilized a Parametric Rectified Linear unit (PReLU)⁴⁰⁰, which can help improve model fitting without adding computational cost and with little overfitting risk.

Early stopping: Early stopping is a method where the training validation error is monitored and previous coefficient values are recorded. Once the training level reaches a stopping criteria, then the coefficients are used. Early stopping helps to mitigate overtraining. It also acts as a regularizer, constraining the parameter space to be close to the initial configuration³⁹⁹.

Dropout: Dropout usually uses some number of randomly selected links (or a probability that a link will be dropped)⁴⁰¹. As the network is trained, these links are zeroed, basically stopping data from flowing from the shallower to deeper layers in the DL system. Dropout basically allows a bagging-like effect, but instead of the individual networks being independent, they share values, but the mixing occurs at the dropout layer, and the individual subnetworks share parameters¹¹.

Batch Normalization: Batch normalization was developed by Ioffe et al.⁴⁰² Batch normalization breaks the data into small batches and then normalizes the data to be zero mean and unity variance. Batch normalization can also be added internally as layers in the network. Batch normalization reduces the so-called *internal covariate shift* problem for each training mini-batch.

Applying batch normalization had the following benefits: (1) allowed a higher learning rate; (2) the DL network was not as sensitive to initialization; (3) dropout was not required to mitigate overfitting; (4) The L_2 weight regularization could be reduced, increasing accuracy. Adding batch normalization increases the two extra parameters per activation.

Optimization: Optimization of DL networks is a major area of study in DL. It is nontrivial to train a DL network, much less squeeze out high performance on both the training and testing datasets. SGD is a training method that uses small batches of training data to generate an estimate of the gradients. Li et al.⁴⁰³ argues the SGD is not inherently parallel, and often requires training many models and choosing the one that performs best on the validation set. They also show that no one method works best in all cases. They found that optimization performance varies per problem. A nice review paper for many gradient descent algorithms is provided by Ruder⁴⁰⁴. According to Ruder, complications for gradient descent algorithms include:

- *How to choose a proper learning rate?*
- *How to properly adjust learning-rate schedules for optimal performance?*
- *How to adjust learning rates independently for each parameter?*
- *How to avoid getting trapped in local minima and saddle points when one dimension slopes up and one down (the gradients can get very small and training halts)*

Various gradient descent methods such as Adagrad⁴⁰⁵, which adapts the learning rate to the parameters, AdaDelta⁴⁰⁶, which uses a fixed size window of past data, and Adam⁴⁰⁷, which also has both mean and variance terms for the gradient descent, can be utilized for training. Another recent approach seeks to optimize the learning rate from the data is described in Schaul et al.⁴⁰⁸. Finally, Sokolić et al.³⁹⁷ concluded experimentally that for a DNN to generalize well, the spectral norm of the NN's Jacobian matrix in the neighbourhood of the training samples must be bounded. They furthermore show that the generalization error can be bounded independent of the DL network's depth or width, provided that the Jacobian spectral norm is bounded. They also analyze residual networks, weight normalized networks, CNN's with batch normalization and Jacobian regularization, and residual networks with Jacobian regularization. The interested reader is referred to chapter 8 of¹¹ for further information.

Data Propagation: Both highway networks⁴⁰⁹ and residual networks⁴¹⁰ are methods that take data from one layer and incorporate it, either directly (highway networks) or as a difference (residual networks) into deeper layers. These methods both allow very deep networks to be trained, at the expense of some additional components. Balduzzi et al.⁴¹¹ examined networks and determined that there is a so-called "shattered gradient" problem in DNN, which is manifested by the gradient correlation decaying exponentially with depth and thus gradients resemble white noise. A "looks linear" initialization is developed that prevents the gradient shattering. This method appears not to require skip connections (highway networks, residual networks).

5 Conclusions

In this letter, we have performed a thorough review and analyzed 207 RS papers that utilize FL and DL, as well as 57 survey papers in DL and RS. We provide researches with a clustered set of 12 areas where DL RS papers have been applied. We examine why DL is popular and what is enabling DL. We examined many DL tools and provided opinions about the tools pros and cons. We critically looked at the DL RS field and identified nine general areas with unsolved challenges

and opportunities, specifically enumerated 11 difficult and thought-provoking open questions in this area. We reviewed current DL research in CV and discussed recent methods that could be utilized in DL in RS. We provide a table of DL survey papers covering DL in RS and feature learning in RS.

Disclosures

The authors declare no conflict of interest.

Acknowledgments

The authors wish to thank graduate students Vivi Wei, Julie White and Charlie Veal for their valuable inputs related to DL tools.

References

- 1 G. Cheng, J. Han, and X. Lu, “Remote Sensing Image Scene Classification: Benchmark and State of the Art,” *arXiv:1703.00121 [cs.CV]* (2017). DOI:10.1109/JPROC.2017.2675998.
- 2 L. Deng, “A tutorial survey of architectures, algorithms, and applications for deep learning,” *APSIPA Transactions on Signal and Information Processing* **3** (e2)(January), 1–29 (2014).
- 3 L. Zhang, L. Zhang, and V. Kumar, “Deep learning for Remote Sensing Data,” *IEEE Geoscience and Remote Sensing Magazine* **4**(June), 22–40 (2016). DOI:10.1155/2016/7954154.
- 4 H. Wang and B. Raj, “A survey: Time travel in deep learning space: An introduction to deep learning models and how deep learning models evolved from the initial ideas,” *arXiv preprint arXiv:1510.04781 abs/1510.04781* (2015).
- 5 J. Wan, D. Wang, S. C. H. Hoi, *et al.*, “Deep learning for content-based image retrieval: A comprehensive study,” in *Proceedings of the 22nd ACM international conference on Multimedia*, 157–166 (2014).
- 6 J. Ba and R. Caruana, “Do deep nets really need to be deep?,” in *Advances in neural information processing systems*, 2654–2662 (2014).
- 7 I. Arel, D. C. Rose, and T. P. Karnowski, “Deep Machine Learning - A New Frontier in Artificial Intelligence Research,” *IEEE Computational Intelligence Magazine* **5**(4), 13–18 (2010). DOI:10.1109/MCI.2010.938364.
- 8 W. Liu, Z. Wang, X. Liu, *et al.*, “A survey of deep neural network architectures and their applications,” *Neurocomputing* **234**, 11–26 (2016). DOI:10.1016/j.neucom.2016.12.038.
- 9 D. Yu and L. Deng, “Deep Learning and Its Applications to Signal and Information Processing [Exploratory DSP],” *Signal Processing Magazine, IEEE* **28**(1), 145–154 (2011). DOI:10.1109/MSP.2010.939038.
- 10 J. Schmidhuber, “Deep Learning in neural networks: An overview,” *Neural Networks* **61**, 85–117 (2015). DOI:10.1016/j.neunet.2014.09.003.
- 11 I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, MIT Press (2016).
- 12 L. Arnold, S. Rebecchi, S. Chevallier, *et al.*, “An introduction to deep learning,” in *2012 11th International Conference on Information Science, Signal Processing and their Applications, ISSPA 2012*, (April), 1438–1439 (2012).
- 13 Y. Bengio, A. C. Courville, and P. Vincent, “Unsupervised feature learning and deep learning: A review and new perspectives,” *CoRR, abs/1206.5538* **1** (2012).

- 14 X.-w. Chen and X. Len, "Big Data Deep Learning: Challenges and Perspectives," *IEEE Access* **2**, 514–525 (2014). DOI:10.1109/ACCESS.2014.2325029.
- 15 L. Deng and D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends® in Signal Processing* **7**(3-4), 197–387 (2013).
- 16 M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, *et al.*, "Deep learning applications and challenges in big data analytics," *Journal of Big Data* **2**(1), 1–21 (2015).
- 17 J. M. Bioucas-dias, A. Plaza, G. Camps-valls, *et al.*, "Hyperspectral Remote Sensing Data Analysis and Future Challenges," *IEEE Geoscience and Remote Sensing Magazine* (June), 6–36 (2013).
- 18 G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *Geoscience and Remote Sensing, IEEE Transactions on* **43**(6), 1351–1362 (2005).
- 19 G. Camps-Valls, D. Tuia, L. Bruzzone, *et al.*, "Advances in hyperspectral image classification: Earth monitoring with statistical learning methods," *IEEE Signal Processing Magazine* **31**(1), 45–54 (2013). DOI:10.1109/MSP.2013.2279179.
- 20 H. Deborah, N. Richard, and J. Y. Hardeberg, "A comprehensive evaluation of spectral distance functions and metrics for hyperspectral image processing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **8**(6), 3224–3234 (2015).
- 21 P. Dollar, C. Wojek, B. Schiele, *et al.*, "Pedestrian detection: An evaluation of the state of the art," *IEEE transactions on pattern analysis and machine intelligence* **34**(4), 743–761 (2012). DOI:10.1109/TPAMI.2011.155.
- 22 P. Du, J. Xia, W. Zhang, *et al.*, "Multiple classifier system for remote sensing image classification: A review," *Sensors* **12**(4), 4764–4792 (2012).
- 23 M. Fauvel, Y. Tarabalka, J. A. Benediktsson, *et al.*, "Advances in spectral-spatial classification of hyperspectral images," *Proceedings of the IEEE* **101**(3), 652–675 (2013).
- 24 M. Hussain, D. Chen, A. Cheng, *et al.*, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *ISPRS Journal of Photogrammetry and Remote Sensing* **80**, 91–106 (2013). DOI:10.1016/j.isprsjprs.2013.03.006.
- 25 G. Jianya, S. Haigang, M. Guorui, *et al.*, "A review of multi-temporal remote sensing data change detection algorithms," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **37**(B7), 757–762 (2008).
- 26 D. J. Lary, A. H. Alavi, A. H. Gandomi, *et al.*, "Machine learning in geosciences and remote sensing," *Geoscience Frontiers* **7**(1), 3–10 (2016).
- 27 D. Lunga, S. Prasad, M. M. Crawford, *et al.*, "Manifold-learning-based feature extraction for classification of hyperspectral data: A review of advances in manifold learning," *IEEE Signal Processing Magazine* **31**(1), 55–66 (2014).
- 28 S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering* **22**(10), 1345–1359 (2010). DOI:10.1109/TKDE.2009.191.
- 29 A. Plaza, P. Martinez, R. Pérez, *et al.*, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE transactions on geoscience and remote sensing* **42**(3), 650–663 (2004). DOI:10.1109/TGRS.2003.820314.
- 30 N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE signal processing magazine* **19**(1), 44–57 (2002).

- 31 N. Keshava, "A survey of spectral unmixing algorithms," *Lincoln Laboratory Journal* **14**(1), 55–78 (2003).
- 32 M. Parente and A. Plaza, "Survey of geometric and statistical unmixing algorithms for hyperspectral images," in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop on*, 1–4, IEEE (2010).
- 33 A. Plaza, G. Martín, J. Plaza, *et al.*, "Recent developments in endmember extraction and spectral unmixing," in *Optical Remote Sensing*, S. Prasad, L. M. Bruce, and J. Chanussot, Eds., 235–267, Springer (2011). DOI:10.1007/978-3-642-14212-3_12.
- 34 C. Shi and L. Wang, "Incorporating spatial information in spectral unmixing: A review," *Remote Sensing of Environment* **149**, 70–87 (2014). DOI:10.1016/j.rse.2014.03.034.
- 35 M. Chen, Z. Xu, K. Weinberger, *et al.*, "Marginalized denoising autoencoders for domain adaptation," *arXiv preprint arXiv:1206.4683* (2012).
- 36 ujjwalkarn, "An Intuitive Explanation of Convolutional Neural Networks." <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/> (2016).
- 37 Y. LeCun, L. Bottou, Y. Bengio, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE* **86**(11), 2278–2324 (1998).
- 38 A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 1097–1105 (2012).
- 39 M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*, 818–833, Springer (2014).
- 40 C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9 (2015).
- 41 K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778 (2016).
- 42 G. Huang, Z. Liu, K. Q. Weinberger, *et al.*, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993* (2016).
- 43 G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science* **313**(5786), 504 – 507 (2006).
- 44 S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation* **9**(8), 1735–1780 (1997).
- 45 M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *2011 International Conference on Computer Vision*, 2018–2025 (2011).
- 46 M. D. Zeiler, D. Krishnan, G. W. Taylor, *et al.*, "Deconvolutional networks," in *In CVPR*, (2010).
- 47 H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," *CoRR* **abs/1505.04366** (2015).
- 48 O. Russakovsky, J. Deng, H. Su, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision* **115**(3), 211–252 (2015).
- 49 L. Brown, "Deep Learning with GPUs." www.nvidia.com/content/events/geoInt2015/LBrown_DL.pdf (2015).

- 50 G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems* **2**(4), 303–314 (1989).
- 51 M. Telgarsky, "Benefits of depth in neural networks," *arXiv preprint arXiv:1602.04485* (2016).
- 52 O. Sharir and A. Shashua, "On the Expressive Power of Overlapping Operations of Deep Networks," *arXiv preprint arXiv:1703.02065* (2017).
- 53 S. Liang and R. Srikant, "Why deep neural networks for function approximation?," *arXiv preprint arXiv:1610.04161* (2016).
- 54 Y. Ma, H. Wu, L. Wang, *et al.*, "Remote sensing big data computing: Challenges and opportunities," *Future Generation Computer Systems* **51**, 47 – 60 (2015). Special Section: A Note on New Trends in Data-Aware Scheduling and Resource Provisioning in Modern HPC Systems.
- 55 M. Chi, A. Plaza, J. A. Benediktsson, *et al.*, "Big data for remote sensing: Challenges and opportunities," *Proceedings of the IEEE* **104**, 2207–2219 (2016).
- 56 R. Raina, A. Madhavan, and A. Y. Ng, "Large-scale deep unsupervised learning using graphics processors," in *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09*, 873–880, ACM, (New York, NY, USA) (2009).
- 57 D. C. Cireşan, U. Meier, J. Masci, *et al.*, "Flexible, high performance convolutional neural networks for image classification," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two, IJCAI'11*, 1237–1242, AAAI Press (2011).
- 58 D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Proceedings of the 20th International Conference on Artificial Neural Networks: Part III, ICANN'10*, 92–101, Springer-Verlag, (Berlin, Heidelberg) (2010).
- 59 L. Deng, D. Yu, and J. Platt, "Scalable stacking and learning for building deep architectures," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2133–2136 (2012).
- 60 B. Hutchinson, L. Deng, and D. Yu, "Tensor deep stacking networks," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**, 1944–1957 (2013).
- 61 J. Dean, G. S. Corrado, R. Monga, *et al.*, "Large scale distributed deep networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems, NIPS'12*, 1223–1231, Curran Associates Inc., (USA) (2012).
- 62 J. Dean, G. Corrado, R. Monga, *et al.*, "Large scale distributed deep networks," in *Advances in neural information processing systems*, 1223–1231 (2012).
- 63 K. He, X. Zhang, S. Ren, *et al.*, "Deep residual learning for image recognition," *CoRR* **abs/1512.03385** (2015).
- 64 R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *CoRR* **abs/1505.00387** (2015).
- 65 K. Greff, R. K. Srivastava, and J. Schmidhuber, "Highway and residual networks learn unrolled iterative estimation," *CoRR* **abs/1612.07771** (2016).
- 66 J. G. Zilly, R. K. Srivastava, J. Koutník, *et al.*, "Recurrent highway networks," *CoRR* **abs/1607.03474** (2016).

- 67 R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," *CoRR abs/1507.06228* (2015).
- 68 Y. Jia, E. Shelhamer, J. Donahue, *et al.*, "Caffe: Convolutional Architecture for Fast Feature Embedding," in *ACM International Conference on Multimedia*, 675–678 (2014).
- 69 M. Abadi, P. Barham, J. Chen, *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*. Savannah, Georgia, USA, (2016).
- 70 A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM international conference on Multimedia*, 689–692, ACM (2015).
- 71 A. Handa, M. Bloesch, V. Patraucean, *et al.*, "Gvnn: Neural Network Library for Geometric Computer Vision," in *Computer Vision-ECCV 2016 Workshop*, 67–82, Springer International (2016).
- 72 F. Chollet, "Keras," (2015). <https://github.com/fchollet/keras>.
- 73 T. Chen, M. Li, Y. Li, *et al.*, "MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems," *eprint arXiv:1512.01274*, 1–6 (2015).
- 74 R. Al-Rfou, G. Alain, A. Almahairi, *et al.*, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv preprint arXiv:1605.02688* (2016).
- 75 "Deep Learning with Torch: The 60-minute Blitz." [https://github.com/soumith/cvpr2015/blob/master/Deep Learning with Torch.ipynb](https://github.com/soumith/cvpr2015/blob/master/Deep%20Learning%20with%20Torch.ipynb).
- 76 M. Baccouche, F. Mamalet, C. Wolf, *et al.*, "Sequential deep learning for human action recognition," in *International Workshop on Human Behavior Understanding*, 29–39, Springer (2011).
- 77 Z. Shou, J. Chan, A. Zareian, *et al.*, "Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos," *arXiv preprint arXiv:1703.01515* (2017).
- 78 D. Tran, L. Bourdev, R. Fergus, *et al.*, "Learning spatiotemporal features with 3d convolutional networks," in *Computer Vision (ICCV), 2015 IEEE International Conference on*, 4489–4497 (2015).
- 79 L. Wang, Y. Xiong, Z. Wang, *et al.*, "Temporal segment networks: towards good practices for deep action recognition," in *European Conference on Computer Vision*, 20–36 (2016).
- 80 G. E. Dahl, D. Yu, L. Deng, *et al.*, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing* **20**(1), 30–42 (2012).
- 81 G. Hinton, L. Deng, D. Yu, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine* **29**(6), 82–97 (2012).
- 82 A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 IEEE international conference on*, 6645–6649, IEEE (2013).
- 83 W. Luo, A. G. Schwing, and R. Urtasun, "Efficient deep learning for stereo matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5695–5703 (2016).

- 84 S. Levine, P. Pastor, A. Krizhevsky, *et al.*, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” *arXiv preprint arXiv:1603.02199* (2016).
- 85 S. Venugopalan, M. Rohrbach, J. Donahue, *et al.*, “Sequence to sequence-video to text,” in *Proceedings of the IEEE International Conference on Computer Vision*, 4534–4542 (2015).
- 86 R. Collobert, “Deep learning for efficient discriminative parsing,” in *AISTATS*, **15**, 224–232 (2011).
- 87 S. Venugopalan, H. Xu, J. Donahue, *et al.*, “Translating videos to natural language using deep recurrent neural networks,” *arXiv preprint arXiv:1412.4729* (2014).
- 88 Y. H. Tan and C. S. Chan, “phi- lstm: A phrase-based hierarchical LSTM model for image captioning,” in *13th Asian Conference on Computer Vision (ACCV)*, 101–117 (2016).
- 89 A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3128–3137 (2015).
- 90 P. Baldi, P. Sadowski, and D. Whiteson, “Searching for exotic particles in high-energy physics with deep learning,” *Nature communications* **5** (2014).
- 91 J. Wu, I. Yildirim, J. J. Lim, *et al.*, “Galileo: Perceiving physical object properties by integrating a physics engine with deep learning,” in *Advances in neural information processing systems*, 127–135 (2015).
- 92 A. A. Cruz-Roa, J. E. A. Ovalle, A. Madabhushi, *et al.*, “A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 403–410, Springer (2013).
- 93 R. Fakoor, F. Ladhak, A. Nazi, *et al.*, “Using deep learning to enhance cancer diagnosis and classification,” in *Proceedings of the International Conference on Machine Learning*, (2013).
- 94 K. Sirinukunwattana, S. E. A. Raza, Y.-W. Tsang, *et al.*, “Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images,” *IEEE transactions on medical imaging* **35**(5), 1196–1206 (2016).
- 95 M. Längkvist, L. Karlsson, and A. Loutfi, “A review of unsupervised feature learning and deep learning for time-series modeling,” *Pattern Recognition Letters* **42**, 11–24 (2014).
- 96 S. Sarkar, K. G. Lore, S. Sarkar, *et al.*, “Early detection of combustion instability from hi-speed flame images via deep learning and symbolic time series analysis,” in *Annual Conference of The Prognostics and Health Management*, (2015).
- 97 T. Kuremoto, S. Kimura, K. Kobayashi, *et al.*, “Time series forecasting using a deep belief network with restricted boltzmann machines,” *Neurocomputing* **137**, 47–56 (2014).
- 98 A. Dosovitskiy, J. Tobias Springenberg, and T. Brox, “Learning to generate chairs with convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1538–1546 (2015).
- 99 J. Zhao, M. Mathieu, and Y. LeCun, “Energy-based generative adversarial network,” *arXiv preprint arXiv:1609.03126* (2016).
- 100 S. E. Reed, Z. Akata, S. Mohan, *et al.*, “Learning what and where to draw,” in *Advances in Neural Information Processing Systems*, 217–225 (2016).

- 101 D. Berthelot, T. Schumm, and L. Metz, “Began: Boundary equilibrium generative adversarial networks,” *arXiv preprint arXiv:1703.10717* (2017).
- 102 W. R. Tan, C. S. Chan, H. Aguirre, *et al.*, “Artgan: Artwork synthesis with conditional categorical gans,” *arXiv preprint arXiv:1702.03410* (2017).
- 103 K. Gregor, I. Danihelka, A. Graves, *et al.*, “Draw: A recurrent neural network for image generation,” *arXiv preprint arXiv:1502.04623* (2015).
- 104 A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434* (2015).
- 105 X. Ding, Y. Zhang, T. Liu, *et al.*, “Deep learning for event-driven stock prediction,” in *IJCAI*, 2327–2333 (2015).
- 106 Z. Yuan, Y. Lu, Z. Wang, *et al.*, “Droid-sec: deep learning in android malware detection,” in *ACM SIGCOMM Computer Communication Review*, **44**(4), 371–372, ACM (2014).
- 107 C. Cadena, A. Dick, and I. D. Reid, “Multi-modal Auto-Encoders as Joint Estimators for Robotics Scene Understanding,” in *Proceedings of Robotics: Science and Systems Conference (RSS)*, (2016).
- 108 J. Feng, Y. Wang, and S.-F. Chang, “3D Shape Retrieval Using a Single Depth Image from Low-cost Sensors,” in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1–9 (2016).
- 109 A. Haque, A. Alahi, and L. Fei-fei, “Recurrent Attention Models for Depth-Based Person Identification,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1229–1238 (2016).
- 110 V. Hegde Stanford and R. Zadeh Stanford, “FusionNet: 3D Object Classification Using Multiple Data Representations,” *arXiv preprint arXiv:1607.05695* (2016).
- 111 J. Huang and S. You, “Point Cloud Labeling using 3D Convolutional Neural Network,” in *International Conference on Pattern Recognition*, (December) (2016).
- 112 W. Kehl, F. Milletari, F. Tombari, *et al.*, “Deep Learning of Local RGB-D Patches for 3D Object Detection and 6D Pose Estimation,” in *European Conference on Computer Vision*, 205–220 (2016).
- 113 C. Li, A. Reiter, and G. D. Hager, “Beyond Spatial Pooling: Fine-Grained Representation Learning in Multiple Domains,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* **2**(1), 4913–4922 (2015).
- 114 N. Sedaghat, M. Zolfaghari, and T. Brox, “Orientation-boosted Voxel Nets for 3D Object Recognition,” *arXiv preprint arXiv:1604.03351* (2016).
- 115 Z. Xie, K. Xu, W. Shan, *et al.*, “Projective Feature Learning for 3D Shapes with Multi-View Depth Images,” in *Computer Graphics Forum*, **34**(7), 1–11, Wiley Online Library (2015).
- 116 C. Chen, “DeepDriving : Learning Affordance for Direct Perception in Autonomous Driving,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2722–2730 (2015).
- 117 X. Chen, H. Ma, J. Wan, *et al.*, “Multi-View 3D Object Detection Network for Autonomous Driving,” *arXiv preprint arXiv:1611.07759* (2016).
- 118 A. Chigorin and A. Konushin, “A system for large-scale automatic traffic sign recognition and mapping,” *CMRT13–City Models, Roads and Traffic* **2013**, 13–17 (2013).

- 119 D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *2012 IEEE Conf. on Computer Vision and Pattern Recog. (CVPR)*, (February), 3642–3649 (2012).
- 120 Y. Zeng, X. Xu, Y. Fang, *et al.*, "Traffic sign recognition using extreme learning classifier with deep convolutional features," in *The 2015 international conference on intelligence science and big data engineering (IScIDE 2015), Suzhou, China*, (2015).
- 121 A.-B. Salberg, "Detection of seals in remote sensing images using features extracted from deep convolutional neural networks," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, (0373), 1893–1896 (2015).
- 122 W. Li, G. Wu, and Q. Du, "Transferred Deep Learning for Anomaly Detection in Hyperspectral Imagery," *IEEE Geoscience and Remote Sensing Letters* **PP**(99), 1–5 (2017).
- 123 J. Becker, T. C. Havens, A. Pinar, *et al.*, "Deep belief networks for false alarm rejection in forward-looking ground-penetrating radar," in *SPIE Defense+ Security*, 94540W–94540W, International Society for Optics and Photonics (2015).
- 124 C. Bentes, D. Velotto, and S. Lehner, "Target classification in oceanographic SAR images with deep neural networks: Architecture and initial results," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 3703–3706, IEEE (2015).
- 125 L. E. Besaw, "Detecting buried explosive hazards with handheld GPR and deep learning," in *SPIE Defense+ Security*, 98230N–98230N, International Society for Optics and Photonics (2016).
- 126 L. E. Besaw and P. J. Stimac, "Deep learning algorithms for detecting explosive hazards in ground penetrating radar data," in *SPIE Defense+ Security*, 90720Y–90720Y, International Society for Optics and Photonics (2014).
- 127 K. Du, Y. Deng, R. Wang, *et al.*, "SAR ATR based on displacement-and rotation-insensitive CNN," *Remote Sensing Letters* **7**(9), 895–904 (2016). DOI:10.1080/2150704X.2016.1196837.
- 128 S. Chen and H. Wang, "SAR target recognition based on deep learning," in *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*, 541–547, IEEE (2014). DOI:10.1109/dsaa.2014.7058124.
- 129 D. A. E. Morgan, "Deep convolutional neural networks for ATR from SAR imagery," in *SPIE Defense+ Security*, 94750F–94750F, International Society for Optics and Photonics (2015). DOI:10.1117/12.2176558.
- 130 J. C. Ni and Y. L. Xu, "SAR automatic target recognition based on a visual cortical system," in *Image and Signal Processing (CISP), 2013 6th International Congress on*, **2**, 778–782, IEEE (2013). DOI:10.1109/cisp.2013.6745270.
- 131 Z. Sun, L. Xue, and Y. Xu, "Recognition of sar target based on multilayer auto-encoder and snn," *International Journal of Innovative Computing, Information and Control* **9**(11), 4331–4341 (2013).
- 132 H. Wang, S. Chen, F. Xu, *et al.*, "Application of deep-learning algorithms to MSTAR data," in *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, 3743–3745, IEEE (2015). DOI:10.1109/igarss.2015.7326637.
- 133 L. Zhang, L. Zhang, D. Tao, *et al.*, "A multifeature tensor for remote-sensing target recognition," *IEEE Geoscience and Remote Sensing Letters* **8**(2), 374–378 (2011). DOI:10.1109/LGRS.2010.2077272.

- 134 L. Zhang, Z. Shi, and J. Wu, "A hierarchical oil tank detector with deep surrounding features for high-resolution optical satellite imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **8**(10), 4895–4909 (2015).
- 135 P. F. Alcantarilla, S. Stent, G. Ros, *et al.*, "Streetview change detection with deconvolutional networks," in *Robotics: Science and Systems Conference (RSS)*, (2016).
- 136 M. Gong, Z. Zhou, and J. Ma, "Change Detection in Synthetic aperture Radar Images Based on Deep Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems* **27**(4), 2141–51 (2016).
- 137 F. Pacifici, F. Del Frate, C. Solimini, *et al.*, "An Innovative Neural-Net Method to Detect Temporal Changes in High-Resolution Optical Satellite Imagery," *IEEE Transactions on Geoscience and Remote Sensing* **45**(9), 2940–2952 (2007).
- 138 S. Stent, "Detecting Change for Multi-View, Long-Term Surface Inspection," in *Proceedings of the 2015 British Machine Vision Conference (BCMV)*, 1–12 (2015).
- 139 J. Zhao, M. Gong, J. Liu, *et al.*, "Deep learning to classify difference image for image change detection," in *Neural Networks (IJCNN), 2014 International Joint Conference on*, 411–417, IEEE (2014).
- 140 S. Basu, S. Ganguly, S. Mukhopadhyay, *et al.*, "DeepSat - A Learning framework for Satellite Imagery," in *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 1–22 (2015).
- 141 Y. Bazi, N. Alajlan, F. Melgani, *et al.*, "Differential Evolution Extreme Learning Machine for the Classification of Hyperspectral Images," *IEEE Geoscience and Remote Sensing Letters* **11**(6), 1066–1070 (2014).
- 142 J. Cao, Z. Chen, and B. Wang, "Deep Convolutional Networks With Superpixel Segmentation for Hyperspectral Image Classification," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 3310–3313 (2016).
- 143 J. Cao, Z. Chen, and B. Wang, "Graph-Based Deep Convolutional Networks With Superpixel Segmentation for Hyperspectral Image Classification," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 3310–3313 (2016).
- 144 C. Chen, W. Li, H. Su, *et al.*, "Spectral-spatial classification of hyperspectral image based on kernel extreme learning machine," *Remote Sensing* **6**(6), 5795–5814 (2014).
- 145 G. Cheng, C. Ma, P. Zhou, *et al.*, "Scene Classification of High Resolution Remote Sensing Images Using Convolutional Neural Networks," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 767–770 (2016).
- 146 F. Del Frate, F. Pacifici, G. Schiavon, *et al.*, "Use of neural networks for automatic classification from high-resolution images," *IEEE Transactions on Geoscience and Remote Sensing* **45**(4), 800–809 (2007).
- 147 Z. Fang, W. Li, and Q. Du, "Using CNN-based high-level features for remote sensing scene classification," in *IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 2610–2613 (2016).
- 148 Q. Fu, X. Yu, X. Wei, *et al.*, "Semi-supervised classification of hyperspectral imagery based on stacked autoencoders," in *Eighth International Conference on Digital Image Processing (ICDIP 2016)*, 100332B–100332B, International Society for Optics and Photonics (2016).

- 149 J. Geng, J. Fan, H. Wang, *et al.*, “High-Resolution SAR Image Classification via Deep Convolutional Autoencoders,” *IEEE Geoscience and Remote Sensing Letters* **12**(11), 2351–2355 (2015).
- 150 X. Han, Y. Zhong, B. Zhao, *et al.*, “Scene classification based on a hierarchical convolutional sparse auto-encoder for high spatial resolution imagery,” *International Journal of Remote Sensing* **38**(2), 514–536 (2017).
- 151 M. He, X. Li, Y. Zhang, *et al.*, “Hyperspectral image classification based on deep stacking network,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 3286–3289 (2016).
- 152 B. Hou, X. Luo, S. Wang, *et al.*, “Polarimetric Sar Images Classification Using Deep Belief Networks with Learning Features,” in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, (2), 2366–2369 (2015).
- 153 W. Hu, Y. Huang, L. Wei, *et al.*, “Deep convolutional neural networks for hyperspectral image classification,” *Journal of Sensors* **2015**, 1–12 (2015).
- 154 M. Iftene, Q. Liu, and Y. Wang, “Very high resolution images classification by fine tuning deep convolutional neural networks,” in *Eighth International Conference on Digital Image Processing (ICDIP 2016)*, 100332D–100332D, International Society for Optics and Photonics (2016).
- 155 P. Jia, M. Zhang, Wenbo Yu, *et al.*, “Convolutional Neural Network Based Classification for Hyperspectral Data,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 5075–5078 (2016).
- 156 P. Kotschieder, M. Fiterau, A. Criminisi, *et al.*, “Deep neural decision forests,” in *Proceedings of the IEEE International Conference on Computer Vision*, 1467–1475 (2015).
- 157 M. Långkvist, A. Kiselev, M. Alirezaie, *et al.*, “Classification and segmentation of satellite orthoimagery using convolutional neural networks,” *Remote Sensing* **8**(4), 1–21 (2016).
- 158 H. Lee and H. Kwon, “Contextual Deep CNN Based Hyperspectral Classification,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, **1604.03519**, 2–4 (2016).
- 159 J. Li, “Active learning for hyperspectral image classification with a stacked autoencoders based neural network,” in *2016 IEEE International Conference on Image Processing (ICIP)*, 1062–1065 (2016).
- 160 J. Li, L. Bruzzone, and S. Liu, “Deep feature representation for hyperspectral image classification,” in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 4951–4954 (2015).
- 161 T. Li, J. Zhang, and Y. Zhang, “Classification of hyperspectral image based on deep belief networks,” in *2014 IEEE International Conference on Image Processing (ICIP)*, 5132–5136 (2014).
- 162 W. Li, G. Wu, F. Zhang, *et al.*, “Hyperspectral image classification using deep pixel-pair features,” *IEEE Transactions on Geoscience and Remote Sensing* **55**(2), 844–853 (2017).
- 163 Y. Li, W. Xie, and H. Li, “Hyperspectral image reconstruction by deep convolutional neural network for classification,” *Pattern Recognition* **63**(August 2016), 371–383 (2016).
- 164 Z. Lin, Y. Chen, X. Zhao, *et al.*, “Spectral-spatial classification of hyperspectral image using autoencoders,” in *Information, Communications and Signal Processing (ICICS) 2013 9th International Conference on*, 1–5, IEEE (2013).

- 165 Z. Lin, Y. Chen, X. Zhao, *et al.*, “Spectral-spatial classification of hyperspectral image using autoencoders,” in *2013 9th International Conference on Information, Communications & Signal Processing*, (61301206), 1–5 (2013).
- 166 Y. Liu, G. Cao, Q. Sun, *et al.*, “Hyperspectral classification via deep networks and superpixel segmentation,” *International Journal of Remote Sensing* **36**(13), 3459–3482 (2015).
- 167 Y. Liu, P. Lasang, M. Siegel, *et al.*, “Hyperspectral Classification via Learnt Features,” in *International Conference on Image Processing (ICIP)*, 1–5 (2015).
- 168 P. Liu, H. Zhang, and K. B. Eom, “Active Deep Learning for Classification of Hyperspectral Images,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **10**(2), 712–724 (2016).
- 169 X. Ma, H. Wang, and J. Geng, “Spectral-spatial classification of hyperspectral image based on deep auto-encoder,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **9**(9), 4073–4085 (2016).
- 170 X. Ma, H. Wang, J. Geng, *et al.*, “Hyperspectral image classification with small training set by deep network and relative distance prior,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 3282–3285, IEEE (2016).
- 171 X. Mei, Y. Ma, F. Fan, *et al.*, “Infrared ultraspectral signature classification based on a restricted Boltzmann machine with sparse and prior constraints,” *International Journal of Remote Sensing* **36**(18), 4724–4747 (2015).
- 172 S. Mei, J. Ji, Q. Bi, *et al.*, “Integrating spectral and spatial information into deep convolutional Neural Networks for hyperspectral classification,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 5067–5070 (2016).
- 173 A. Merentitis and C. Debes, “Automatic Fusion and Classification Using Random Forests and Features Extracted with Deep Learning,” in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2943–2946 (2015).
- 174 K. Nogueira, O. A. B. Penatti, and J. A. dos Santos, “Towards better exploiting convolutional neural networks for remote sensing scene classification,” *Pattern Recognition* **61**, 539–556 (2017).
- 175 B. Pan, Z. Shi, and X. Xu, “R-VCANet: A New Deep-Learning-Based Hyperspectral Image Classification Method,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **PP**(99), 1–12 (2017).
- 176 M. Papadomanolaki, M. Vakalopoulou, S. Zagoruyko, *et al.*, “Benchmarking Deep Learning Frameworks for the Classification of Very High Resolution Satellite Multispectral Data,” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 83–88 (2016).
- 177 S. Piramanayagam, W. Schwartzkopf, F. W. Koehler, *et al.*, “Classification of remote sensed images using random forests and deep learning framework,” in *SPIE Remote Sensing*, 100040L–100040L, International Society for Optics and Photonics (2016).
- 178 F. Qin, J. Guo, and W. Sun, “Object-oriented ensemble classification for polarimetric SAR Imagery using restricted Boltzmann machines,” *Remote Sensing Letters* **8**(3), 204–213 (2017).
- 179 S. Rajan, J. Ghosh, and M. M. Crawford, “An Active Learning Approach to Hyperspectral Data Classification,” *IEEE Transactions on Geoscience and Remote Sensing* **46**(4), 1231–1242 (2008).

- 180 Z. Wang, N. M. Nasrabadi, and T. S. Huang, "Semisupervised hyperspectral classification using task-driven dictionary learning with laplacian regularization," *IEEE Transactions on Geoscience and Remote Sensing* **53**(3), 1161–1173 (2015).
- 181 J. Yang, Y. Zhao, J. Cheung, *et al.*, "Hyperspectral Image Classification Using Two-Channel Deep Convolutional Neural Network," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 5079–5082 (2016).
- 182 S. Yu, S. Jia, and C. Xu, "Convolutional neural networks for hyperspectral image classification," *Neurocomputing* **219**, 88–98 (2017).
- 183 J. Yue, S. Mao, and M. Li, "A deep learning framework for hyperspectral image classification using spatial pyramid pooling," *Remote Sensing Letters* **7**(9), 875–884 (2016).
- 184 J. Yue, W. Zhao, S. Mao, *et al.*, "Spectral-spatial classification of hyperspectral images using deep convolutional neural networks," *Remote Sensing Letters* **6**(6), 468–477 (2015).
- 185 A. Zeggada and F. Melgani, "Multilabel classification of UAV images with Convolutional Neural Networks," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 5083–5086, IEEE (2016).
- 186 F. Zhang, B. Du, and L. Zhang, "Scene classification via a gradient boosting random convolutional network framework," *IEEE Transactions on Geoscience and Remote Sensing* **54**(3), 1793–1802 (2016).
- 187 H. Zhang, Y. Li, Y. Zhang, *et al.*, "Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network," *Remote Sensing Letters* **8**(5), 438–447 (2017).
- 188 W. Zhao and S. Du, "Learning multiscale and deep representations for classifying remotely sensed imagery," *ISPRS Journal of Photogrammetry and Remote Sensing* **113**(March), 155–165 (2016).
- 189 Y. Zhong, F. Fei, and L. Zhang, "Large patch convolutional neural networks for the scene classification of high spatial resolution imagery," *Journal of Applied Remote Sensing* **10**(2), 25006 (2016).
- 190 Y. Zhong, F. Fei, Y. Liu, *et al.*, "SatCNN: satellite image dataset classification using agile convolutional neural networks," *Remote Sensing Letters* **8**(2), 136–145 (2017).
- 191 Y. Chen, C. Li, P. Ghamisi, *et al.*, "Deep Fusion Of Hyperspectral And Lidar Data For Thematic Classification," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 3591–3594 (2016).
- 192 L. Ran, Y. Zhang, W. Wei, *et al.*, "Bands Sensitive Convolutional Network for Hyperspectral Image Classification," in *Proceedings of the International Conference on Internet Multimedia Computing and Service*, 268–272, ACM (2016).
- 193 J. Zabalza, J. Ren, J. Zheng, *et al.*, "Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging," *Neurocomputing* **185**, 1–10 (2016).
- 194 Y. Liu and L. Wu, "Geological Disaster Recognition on Optical Remote Sensing Images Using Deep Learning," *Procedia Computer Science* **91**, 566–575 (2016).
- 195 J. Chen, Q. Jin, and J. Chao, "Design of deep belief networks for short-term prediction of drought index using data in the Huaihe river basin," *Mathematical Problems in Engineering* **2012** (2012).

- 196 P. Landschützer, N. Gruber, D. C. E. Bakker, *et al.*, “A neural network-based estimate of the seasonal to inter-annual variability of the Atlantic Ocean carbon sink,” *Biogeosciences* **10**(11), 7793–7815 (2013).
- 197 M. Shi, F. Xie, Y. Zi, *et al.*, “Cloud detection of remote sensing images by deep learning,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 701–704, IEEE (2016).
- 198 X. Shi, Z. Chen, H. Wang, *et al.*, “Convolutional LSTM network: A machine learning approach for precipitation nowcasting,” *Advances in Neural Information Processing Systems* **28**, 802–810 (2015).
- 199 S. Lee, H. Zhang, and D. J. Crandall, “Predicting Geo-informative Attributes in Large-scale Image Collections using Convolutional Neural Networks,” in *2015 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 550–557 (2015).
- 200 W. Kehl, F. Milletari, F. Tombari, *et al.*, “Deep Learning of Local RGB-D Patches for 3D Object Detection and 6D Pose Estimation,” in *European Conference on Computer Vision*, 205–220 (2016).
- 201 Y. Kim and T. Moon, “Human Detection and Activity Classification Based on Micro-Doppler Signatures Using Deep Convolutional Neural Networks,” *IEEE Geoscience and Remote Sensing Letters* **13**(1), 1–5 (2015).
- 202 W. Ouyang and X. Wang, “A discriminative deep model for pedestrian detection with occlusion handling,” in *2012 IEEE Conf. on Computer Vision and Pattern Recog. (CVPR)*, 3258–3265, IEEE (2012).
- 203 D. Tomè, F. Monti, L. Baroffio, *et al.*, “Deep convolutional neural networks for pedestrian detection,” *Signal Processing: Image Communication* **47**, 482–489 (2016).
- 204 Y. Wei, Q. Yuan, H. Shen, *et al.*, “A Universal Remote Sensing Image Quality Improvement Method with Deep Learning,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 6950–6953 (2016).
- 205 H. Zhang, P. Casaseca-de-la Higuera, C. Luo, *et al.*, “Systematic infrared image quality improvement using deep learning based techniques,” in *SPIE Remote Sensing*, 100080P–100080P, International Society for Optics and Photonics (2016).
- 206 D. Quan, S. Wang, M. Ning, *et al.*, “Using Deep Neural Networks for Synthetic Aperture Radar Image Registration,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 2799–2802 (2016).
- 207 P. Ghamisi, Y. Chen, and X. X. Zhu, “A Self-Improving Convolution Neural Network for the Classification of Hyperspectral Data,” *IEEE Geoscience and Remote Sensing Letters* **13**(10), 1–5 (2016).
- 208 N. Kussul, A. Shelestov, M. Lavreniuk, *et al.*, “Deep learning approach for large scale land cover mapping based on remote sensing data fusion,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 198–201, IEEE (2016).
- 209 W. Li, H. Fu, L. Yu, *et al.*, “Stacked Autoencoder-based deep learning for remote-sensing image classification: a case study of African land-cover mapping,” *International Journal of Remote Sensing* **37**(23), 5632–5646 (2016).
- 210 H. Liu, Q. Min, C. Sun, *et al.*, “Terrain Classification With Polarimetric Sar Based on Deep Sparse Filtering Network,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 64–67 (2016).

- 211 K. Makantasis, K. Karantza, A. Doukakis, *et al.*, “Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks,” *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* , 4959–4962 (2015).
- 212 M. Castelluccio, G. Poggi, C. Sansone, *et al.*, “Land Use Classification in Remote Sensing Images by Convolutional Neural Networks,” *arXiv preprint arXiv:1508.00092* , 1–11 (2015).
- 213 G. Cheng, J. Han, L. Guo, *et al.*, “Effective and Efficient Midlevel Visual Elements-Oriented Land-Use Classification using VHR Remote Sensing Images,” *IEEE Transactions on Geoscience and Remote Sensing* **53**(8), 4238–4249 (2015).
- 214 F. P. S. Luus, B. P. Salmon, F. Van Den Bergh, *et al.*, “Multiview Deep Learning for Land-Use Classification,” *IEEE Geoscience and Remote Sensing Letters* **12**(12), 2448–2452 (2015).
- 215 Q. Lv, Y. Dou, X. Niu, *et al.*, “Urban land use and land cover classification using remotely sensed SAR data through deep belief networks,” *Journal of Sensors* **2015** (2015).
- 216 X. Ma, H. Wang, and J. Wang, “Semisupervised classification for hyperspectral image based on multi-decision labeling and deep feature learning,” *ISPRS Journal of Photogrammetry and Remote Sensing* **120**, 99–107 (2016).
- 217 M. E. Midhun, S. R. Nair, V. T. N. Prabhakar, *et al.*, “Deep Model for Classification of Hyperspectral image using Restricted Boltzmann Machine,” in *Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing - ICONIAAC '14*, 1–7 (2014).
- 218 E. Othman, Y. Bazi, N. Alajlan, *et al.*, “Using convolutional features and a sparse autoencoder for land-use scene classification,” *International Journal of Remote Sensing* **37**(10), 1977–1995 (2016).
- 219 A. B. Penatti, K. Nogueira, and J. A. Santos, “Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains ?,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 44–51 (2015).
- 220 A. Romero, C. Gatta, and G. Camps-Valls, “Unsupervised deep feature extraction for remote sensing image classification,” *IEEE Transactions on Geoscience and Remote Sensing* **54**(3), 1349–1362 (2016).
- 221 Y. Sun, J. Li, W. Wang, *et al.*, “Active learning based autoencoder for hyperspectral imagery classification,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 469–472 (2016).
- 222 N. K. Uba, *Land Use and Land Cover Classification Using Deep Learning Techniques*. PhD thesis, Arizona State University (2016).
- 223 L. Alexandre, “3D Object Recognition using Convolutional Neural Networks with Transfer Learning between Input Channels,” in *13th International Conference on Intelligent Autonomous Systems*, 889–898, Springer International (2016).
- 224 X. Chen, S. Xiang, C. L. Liu, *et al.*, “Aircraft detection by deep belief nets,” in *Proceedings - 2nd IAPR Asian Conference on Pattern Recognition, ACPR 2013*, 54–58 (2013).
- 225 X. Chen and Y. Zhu, “3D Object Proposals for Accurate Object Class Detection,” *Advances in Neural Information Processing Systems* , 424–432 (2015).

- 226 G. Cheng, P. Zhou, and J. Han, "Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images," *IEEE Transactions on Geoscience and Remote Sensing* **54**(12), 7405–7415 (2016).
- 227 M. Dahmane, S. Foucher, M. Beaulieu, *et al.*, "Object Detection in Pleiades Images using Deep Features," in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 1552–1555 (2016).
- 228 W. Diao, X. Sun, F. Dou, *et al.*, "Object recognition in remote sensing images using sparse deep belief networks," *Remote Sensing Letters* **6**(10), 745–754 (2015).
- 229 G. Georgakis, M. A. Reza, A. Mousavian, *et al.*, "Multiview RGB-D Dataset for Object Instance Detection," in *2016 IEEE Fourth International Conference on 3D Vision (3DV)*, 426–434 (2016).
- 230 D. Maturana and S. Scherer, "3D Convolutional Neural Networks for Landing Zone Detection from LiDAR," *International Conference on Robotics and Automation* (Figure 1), 3471–3478 (2015).
- 231 C. Wang and K. Siddiqi, "Differential geometry boosts convolutional neural networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 51–58 (2016).
- 232 Q. Wu, W. Diao, F. Dou, *et al.*, "Shape-based object extraction in high-resolution remote-sensing images using deep Boltzmann machine," *International Journal of Remote Sensing* **37**(24), 6012–6022 (2016).
- 233 B. Zhou, A. Khosla, A. Lapedriza, *et al.*, "Object Detectors Emerge in Deep Scene CNNs." <http://hdl.handle.net/1721.1/96942> (2015).
- 234 P. Ondruska and I. Posner, "Deep Tracking: Seeing Beyond Seeing Using Recurrent Neural Networks," in *Proceedings of the 30th Conference on Artificial Intelligence (AAAI 2016)*, 3361–3367 (2016).
- 235 G. Masi, D. Cozzolino, L. Verdoliva, *et al.*, "Pansharpening by Convolutional Neural Networks," *Remote Sensing* **8**(7), 594 (2016).
- 236 W. Huang, L. Xiao, Z. Wei, *et al.*, "A new pan-sharpening method with deep neural networks," *IEEE Geoscience and Remote Sensing Letters* **12**(5), 1037–1041 (2015).
- 237 L. Palafox, A. Alvarez, and C. Hamilton, "Automated Detection of Impact Craters and Volcanic Rootless Cones in Mars Satellite Imagery Using Convolutional Neural Networks and Support Vector Machines," in *46th Lunar and Planetary Science Conference*, 1–2 (2015).
- 238 M. M. Ghazi, B. Yanikoglu, and E. Aptoula, "Plant Identification Using Deep Neural Networks via Optimization of Transfer Learning Parameters," *Neurocomputing* (2017).
- 239 H. Guan, Y. Yu, Z. Ji, *et al.*, "Deep learning-based tree classification using mobile LiDAR data," *Remote Sensing Letters* **6**(11), 864–873 (2015).
- 240 P. K. Goel, S. O. Prasher, R. M. Patel, *et al.*, "Classification of hyperspectral data by decision trees and artificial neural networks to identify weed stress and nitrogen status of corn," *Computers and Electronics in Agriculture* **39**(2), 67–93 (2003).
- 241 K. Kuwata and R. Shibasaki, "Estimating crop yields with deep learning and remotely sensed data," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 858–861 (2015).

- 242 J. Rebetez, H. F. Satizábal, M. Mota, *et al.*, “Augmenting a convolutional neural network with local histograms-A case study in crop classification from high-resolution UAV imagery,” in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 515–520 (2016).
- 243 S. Sladojevic, M. Arsenovic, A. Anderla, *et al.*, “Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification,” *Computational Intelligence and Neuroscience* **2016**, 1–11 (2016).
- 244 D. Levi, N. Garnett, and E. Fetaya, “StixelNet: a deep convolutional network for obstacle detection and road segmentation,” in *BMCV*, (2015).
- 245 P. Li, Y. Zang, C. Wang, *et al.*, “Road network extraction via deep learning and line integral convolution,” in *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International*, 1599–1602, IEEE (2016).
- 246 V. Mnih and G. Hinton, “Learning to Label Aerial Images from Noisy Data,” in *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 567–574 (2012).
- 247 J. Wang, J. Song, M. Chen, *et al.*, “Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine,” *International Journal of Remote Sensing* **36**(12), 3144–3169 (2015).
- 248 Y. Yu, J. Li, H. Guan, *et al.*, “Automated Extraction of 3D Trees from Mobile LiDAR Point Clouds,” *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **XL-5**(June), 629–632 (2014).
- 249 Y. Yu, H. Guan, and Z. Ji, “Automated Extraction of Urban Road Manhole Covers Using Mobile Laser Scanning Data,” *IEEE Transactions on Intelligent Transportation Systems* **16**(4), 1–12 (2015).
- 250 Z. Zhong, J. Li, W. Cui, *et al.*, “Fully convolutional networks for building and road extraction: preliminary results,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 1591–1594 (2016).
- 251 R. Hadsell, P. Sermanet, J. Ben, *et al.*, “Learning long-range vision for autonomous off-road driving,” *Journal of Field Robotics* **26**(2), 120–144 (2009).
- 252 L. Mou and X. X. Zhu, “Spatiotemporal scene interpretation of space videos via deep neural network and tracklet analysis,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 1823–1826, IEEE (2016).
- 253 Y. Yuan, L. Mou, and X. Lu, “Scene Recognition by Manifold Regularized Deep Learning Architecture,” *IEEE Transactions on Neural Networks and Learning Systems* **26**(10), 2222–2233 (2015).
- 254 C. Couprie, C. Farabet, L. Najman, *et al.*, “Indoor Semantic Segmentation using depth information,” *arXiv preprint arXiv:1301.3572*, 1–8 (2013).
- 255 Y. Gong, Y. Jia, T. Leung, *et al.*, “Deep Convolutional Ranking for Multilabel Image Annotation,” *arXiv preprint arXiv:1312.4894*, 1–9 (2013).
- 256 M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, “Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 1–9 (2016).
- 257 P. Kaiser, “Learning City Structures from Online Maps,” Master’s thesis, ETH Zurich (2016).

- 258 A. Lagrange, B. L. Saux, A. Beaup, *et al.*, “Benchmarking classification of earth-observation data: from learning explicit features to convolutional networks,” in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 4173–4176 (2015).
- 259 D. Marmanis, M. Datcu, T. Esch, *et al.*, “Deep learning earth observation classification using ImageNet pretrained networks,” *IEEE Geoscience and Remote Sensing Letters* **13**(1), 105–109 (2016).
- 260 D. Marmanis, J. D. Wegner, S. Galliani, *et al.*, “Semantic Segmentation of Aerial Images with an Ensemble of CNNs,” in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2016, **3**, 473–480 (2016).
- 261 S. Paisitkriangkrai, J. Sherrah, P. Janney, *et al.*, “Effective semantic pixel labelling with convolutional networks and Conditional Random Fields,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, **2015-Oct**, 36–43 (2015).
- 262 B. Qu, X. Li, D. Tao, *et al.*, “Deep Semantic Understanding of High Resolution Remote Sensing Image,” in *International Conference on Computer, Information and Telecommunication Systems (CITS)*, 1–5 (2016).
- 263 J. Sherrah, “Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery,” *arXiv preprint arXiv:1606.02585* (2016).
- 264 C. Vaduva, I. Gavat, and M. Datcu, “Deep learning in very high resolution remote sensing image information mining communication concept,” in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, 2506–2510, IEEE (2012).
- 265 M. Volpi and D. Tuia, “Dense semantic labeling of sub-decimeter resolution images with convolutional neural networks,” *IEEE Transactions on Geoscience and Remote Sensing* , 1–13 (2016).
- 266 D. Zhang, J. Han, C. Li, *et al.*, “Co-saliency detection via looking deep and wide,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **07-12-June**, 2994–3002 (2015).
- 267 F. I. Alam, J. Zhou, A. W.-C. Liew, *et al.*, “CRF learning with CNN features for hyperspectral image segmentation,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 6890–6893 (2016).
- 268 N. Audebert, B. L. Saux, and S. Lefèvre, “How Useful is Region-based Classification of Remote Sensing Images in a Deep Learning Framework?,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 5091–5094 (2016).
- 269 E. Basaeed, H. Bhaskar, P. Hill, *et al.*, “A supervised hierarchical segmentation of remote-sensing images using a committee of multi-scale convolutional neural networks,” *International Journal of Remote Sensing* **37**(7), 1671–1691 (2016).
- 270 E. Basaeed, H. Bhaskar, and M. Al-Mualla, “Supervised remote sensing image segmentation using boosted convolutional neural networks,” *Knowledge-Based Systems* **99**, 19–27 (2016).
- 271 S. Pal, S. Chowdhury, and S. K. Ghosh, “DCAP: A deep convolution architecture for prediction of urban growth,” in *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International*, 1812–1815, IEEE (2016).
- 272 J. Wang, Q. Qin, Z. Li, *et al.*, “Deep hierarchical representation and segmentation of high resolution remote sensing images,” in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 4320–4323 (2015).

- 273 C. P. Schwegmann, W. Kleynhans, B. P. Salmon, *et al.*, “Very deep learning for ship discrimination in Synthetic Aperture Radar imagery,” in *2016 IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, 104–107, IEEE (2016).
- 274 J. Tang, C. Deng, G.-b. Huang, *et al.*, “Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine,” *IEEE Transactions on Geoscience and Remote Sensing* **53**(3), 1174–1185 (2015).
- 275 R. Zhang, J. Yaoa, K. Zhanga, *et al.*, “S-CNN-Based Ship Detection From High-Resolution Remote Sensing Images,” in *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLI-B7*, 423–430 (2016).
- 276 Z. Cui, H. Chang, S. Shan, *et al.*, “Deep network cascade for image super-resolution,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **8693 LNCS**(PART 5), 49–64 (2014).
- 277 C. Dong, C. C. Loy, K. He, *et al.*, “Image super-resolution using deep convolutional networks,” *IEEE transactions on pattern analysis and machine intelligence* **38**(2), 295–307 (2016).
- 278 A. Ducournau and R. Fablet, “Deep Learning for Ocean Remote Sensing : An Application of Convolutional Neural Networks for Super-Resolution on Satellite-Derived SST Data,” in *9th Workshop on Pattern Recognition in Remote Sensing*, (October) (2016).
- 279 L. Liebel and M. Körner, “Single-Image Super Resolution for Multispectral Remote Sensing Data Using Convolutional Neural Networks,” in *XXIII ISPRS Congress proceedings, XXIII*(July), 883–890 (2016).
- 280 W. Huang, G. Song, H. Hong, *et al.*, “Deep architecture for traffic flow prediction: Deep belief networks with multitask learning,” *IEEE Transactions on Intelligent Transportation Systems* **15**(5), 2191–2201 (2014).
- 281 Y. Lv, Y. Duan, W. Kang, *et al.*, “Traffic Flow Prediction With Big Data: A Deep Learning Approach,” *IEEE Transactions on Intelligent Transportation Systems* **16**(2), 865–873 (2015).
- 282 M. Elawady, *Sparse Coral Classification Using Deep Convolutional Neural Networks*. PhD thesis, University of Burgundy, University of Girona, Heriot-Watt University (2014).
- 283 H. Qin, X. Li, J. Liang, *et al.*, “DeepFish: Accurate underwater live fish recognition with a deep architecture,” *Neurocomputing* **187**, 49–58 (2016).
- 284 H. Qin, X. Li, Z. Yang, *et al.*, “When Underwater Imagery Analysis Meets Deep Learning : a Solution at the Age of Big Visual Data,” in *2011 7th International Symposium on Image and Signal Processing and Analysis (ISPA)*, 259–264 (2015).
- 285 D. P. Williams, “Underwater Target Classification in Synthetic Aperture Sonar Imagery Using Deep Convolutional Neural Networks,” in *Proc. 23rd International Conference on Pattern Recognition (ICPR)*, (2016).
- 286 F. Alidoost and H. Arefi, “Knowledge Based 3D Building Model Recognition Using Convolutional Neural Networks From Lidar and Aerial Imageries,” *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **XLI-B3**(July), 833–840 (2016).
- 287 C.-A. Brust, S. Sickert, M. Simon, *et al.*, “Efficient Convolutional Patch Networks for Scene Understanding,” in *CVPR Workshop*, 1–9 (2015).

- 288 S. De and A. Bhattacharya, "Urban classification using PolSAR data and deep learning," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 353–356, IEEE (2015).
- 289 Z. Huang, G. Cheng, H. Wang, *et al.*, "Building extraction from multi-source remote sensing images via deep deconvolution neural networks," in *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International*, 1835–1838, IEEE (2016).
- 290 D. Marmanis, F. Adam, M. Datcu, *et al.*, "Deep Neural Networks for Above-Ground Detection in Very High Spatial Resolution Digital Elevation Models," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* **II-3/W4**(March), 103–110 (2015).
- 291 S. Saito and Y. Aoki, "Building and road detection from large aerial imagery," in *SPIE/IS&T Electronic Imaging*, 94050K–94050K, International Society for Optics and Photonics (2015).
- 292 M. Vakalopoulou, K. Karantza, N. Komodakis, *et al.*, "Building detection in very high resolution multispectral data with deep learning features," *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* , 1873–1876 (2015).
- 293 M. Xie, N. Jean, M. Burke, *et al.*, "Transfer learning from deep features for remote sensing and poverty mapping," *arXiv preprint arXiv:1510.00098* , 16 (2015).
- 294 W. Yao, P. Poleswki, and P. Krzystek, "Classification of urban aerial data based on pixel labelling with deep convolutional neural networks and logistic regression," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* **41**(July), 405–410 (2016).
- 295 Q. Zhang, Y. Wang, Q. Liu, *et al.*, "CNN based suburban building detection using monocular high resolution Google Earth images," in *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International*, 661–664, IEEE (2016).
- 296 Z. Zhang, Y. Wang, Q. Liu, *et al.*, "A CNN based functional zone classification method for aerial images," in *Geoscience and Remote Sensing Symposium (IGARSS), 2016 IEEE International*, 5449–5452, IEEE (2016).
- 297 L. Cao, Q. Jiang, M. Cheng, *et al.*, "Robust vehicle detection by combining deep features with exemplar classification," *Neurocomputing* **215**, 225–231 (2016).
- 298 X. Chen, S. Xiang, C.-L. Liu, *et al.*, "Vehicle detection in satellite images by hybrid deep convolutional neural networks," *IEEE Geoscience and remote sensing letters* **11**(10), 1797–1801 (2014).
- 299 K. Goyal and D. Kaur, "A Novel Vehicle Classification Model for Urban Traffic Surveillance Using the Deep Neural Network Model," *International Journal of Education and Management Engineering* **6**(1), 18–31 (2016).
- 300 A. Hu, H. Li, F. Zhang, *et al.*, "Deep Boltzmann Machines based vehicle recognition," in *The 26th Chinese Control and Decision Conference (2014 CCDC)*, 3033–3038 (2014).
- 301 J. Huang and S. You, "Vehicle detection in urban point clouds with orthogonal-view convolutional neural network," in *2016 IEEE International Conference on Image Processing (ICIP)*, (2), 2593–2597 (2016).
- 302 B. Huval, T. Wang, S. Tandon, *et al.*, "An Empirical Evaluation of Deep Learning on Highway Driving," *arXiv preprint arXiv:1504.01716* , 1–7 (2015).

- 303 Q. Jiang, L. Cao, M. Cheng, *et al.*, “Deep neural networks-based vehicle detection in satellite images,” in *Bioelectronics and Bioinformatics (ISBB), 2015 International Symposium on*, 184–187, IEEE (2015).
- 304 G. V. Konoplich, E. O. Putin, and A. A. Filchenkov, “Application of deep learning to the problem of vehicle detection in UAV images,” in *Soft Computing and Measurements (SCM), 2016 XIX IEEE International Conference on*, 4–6, IEEE (2016).
- 305 A. Krishnan and J. Larsson, *Vehicle Detection and Road Scene Segmentation using Deep Learning*. PhD thesis, Chalmers University of Technology (2016).
- 306 S. Lange, F. Ulbrich, and D. Goehring, “Online vehicle detection using deep neural networks and lidar based preselected image patches,” *IEEE Intelligent Vehicles Symposium, Proceedings* **2016-Augus(Iv)**, 954–959 (2016).
- 307 B. Li, “3D Fully Convolutional Network for Vehicle Detection in Point Cloud,” *arXiv preprint arXiv:1611.08069* (2016).
- 308 H. Wang, Y. Cai, and L. Chen, “A vehicle detection algorithm based on deep belief network,” *The scientific world journal* **2014** (2014).
- 309 J.-G. Wang, L. Zhou, Y. Pan, *et al.*, “Appearance-based Brake-Lights recognition using deep learning and vehicle detection,” in *Intelligent Vehicles Symposium (IV), 2016 IEEE, (Iv)*, 815–820, IEEE (2016).
- 310 H. Wang, Y. Cai, X. Chen, *et al.*, “Night-Time Vehicle Sensing in Far Infrared Image with Deep Learning,” *Journal of Sensors* **2016** (2015).
- 311 R. J. Firth, *A Novel Recurrent Convolutional Neural Network for Ocean and Weather Forecasting*. PhD thesis, Louisiana State University (2016).
- 312 R. Kovordányi and C. Roy, “Cyclone track forecasting based on satellite images using artificial neural networks,” *ISPRS Journal of Photogrammetry and Remote Sensing* **64**(6), 513–521 (2009).
- 313 C. Yang and J. Guo, “Improved cloud phase retrieval approaches for China’s FY-3A/VIRR multi-channel data using Artificial Neural Networks,” *Optik-International Journal for Light and Electron Optics* **127**(4), 1797–1803 (2016).
- 314 Y. Chen, X. Zhao, and X. Jia, “Spectral Spatial Classification of Hyperspectral Data Based on Deep Belief Network,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **8**(6), 2381–2392 (2015).
- 315 J. M. P. Nascimento and J. M. B. Dias, “Vertex component analysis: A fast algorithm to unmix hyperspectral data,” *IEEE transactions on Geoscience and Remote Sensing* **43**(4), 898–910 (2005).
- 316 T. H. Chan, K. Jia, S. Gao, *et al.*, “PCANet: A Simple Deep Learning Baseline for Image Classification?,” *IEEE Transactions on Image Processing* **24**(12), 5017–5032 (2015).
- 317 Y. Chen, H. Jiang, C. Li, *et al.*, “Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks,” *IEEE Transactions on Geoscience and Remote Sensing* **54**(10), 6232–6251 (2016).
- 318 J. Donahue, Y. Jia, O. Vinyals, *et al.*, “DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition,” *Icml* **32**, 647–655 (2014).
- 319 A. Zelener and I. Stamos, “CNN-based Object Segmentation in Urban LIDAR With Missing Points,” in *3D Vision (3DV), 2016 Fourth International Conference on*, 417–425, IEEE (2016).

- 320 J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440 (2015).
- 321 J. E. Ball, D. T. Anderson, and S. Samiappan, "Hyperspectral band selection based on the aggregation of proximity measures for automated target detection," in *SPIE Defense+ Security*, 908814–908814, International Society for Optics and Photonics (2014).
- 322 J. E. Ball and L. M. Bruce, "Level Set Hyperspectral Image Classification Using Best Band Analysis," *IEEE Transactions on Geoscience and Remote Sensing* **45**(10), 3022–3027 (2007). DOI:10.1109/TGRS.2007.905629.
- 323 J. E. Ball, L. M. Bruce, T. West, *et al.*, "Level set hyperspectral image segmentation using spectral information divergence-based best band selection," in *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*, 4053–4056, IEEE (2007). DOI:10.1109/IGARSS.2007.4423739.
- 324 D. T. Anderson and A. Zare, "Spectral unmixing cluster validity index for multiple sets of endmembers," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **5**(4), 1282–1295 (2012).
- 325 M. E. Winter, "Comparison of approaches for determining end-members in hyperspectral data," in *Aerospace Conference Proceedings, 2000 IEEE*, **3**, 305–313, IEEE (2000).
- 326 A. S. Charles, B. A. Olshausen, and C. J. Rozell, "Learning sparse codes for hyperspectral imagery," *IEEE Journal of Selected Topics in Signal Processing* **5**(5), 963–978 (2011).
- 327 A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction of hyperspectral images," in *Proc. 6th Workshop Hyperspectral Image Signal Process. Evol. Remote Sens.(WHISPERS)*, (2014).
- 328 J. E. Ball, L. M. Bruce, and N. H. Younan, "Hyperspectral pixel unmixing via spectral band selection and DC-insensitive singular value decomposition," *IEEE Geoscience and Remote Sensing Letters* **4**(3), 382–386 (2007).
- 329 P. M. Atkinson and A. Tatnall, "Introduction neural networks in remote sensing," *International Journal of remote sensing* **18**(4), 699–709 (1997). DOI:10.1080/014311697218700.
- 330 G. Cavallaro, M. Riedel, M. Richerzhagen, *et al.*, "On understanding big data impacts in remotely sensed image classification using support vector machine methods," *IEEE journal of selected topics in applied earth observations and remote sensing* **8**(10), 4634–4646 (2015). DOI:10.1109/JSTARS.2015.2458855.
- 331 M. Egmont-Petersen, D. DeRidder, and H. Handels, "Image Processing with neural networks - a review," *Pattern Recognition* **35**, 2279–2301 (2002). DOI:10.1016/S0031-3203(01)00178-9.
- 332 G. Huang, G.-B. Huang, S. Song, *et al.*, "Trends in extreme learning machines: A review," *Neural Networks* **61**, 32–48 (2015). DOI:10.1016/j.neunet.2014.10.001.
- 333 X. Jia, B.-C. Kuo, and M. M. Crawford, "Feature mining for hyperspectral image classification," *Proceedings of the IEEE* **101**(3), 676–697 (2013). DOI:10.1109/JPROC.2012.2229082.
- 334 D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," *International journal of Remote sensing* **28**(5), 823–870 (2007). DOI:10.1080/01431160600746456.

- 335 H. Petersson, D. Gustafsson, and D. Bergstrom, "Hyperspectral image analysis using deep learning—A review," in *Image Processing Theory Tools and Applications (IPTA), 2016 6th International Conference on*, 1–6, IEEE (2016). DOI:10.1109/ipta.2016.7820963.
- 336 A. Plaza, J. A. Benediktsson, J. W. Boardman, *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote sensing of environment* **113**, S110–S122 (2009). DOI:10.1016/j.rse.2007.07.028.
- 337 W. Wang, N. Yang, Y. Zhang, *et al.*, "A review of road extraction from remote sensing images," *Journal of Traffic and Transportation Engineering (English Edition)* **3**(3), 271–282 (2016).
- 338 "2013 IEEE GRSS Data Fusion Contest." <http://www.grss-ieee.org/community/technical-committees/data-fusion/>.
- 339 "2015 IEEE GRSS Data Fusion Contest." <http://www.grss-ieee.org/community/technical-committees/data-fusion/>.
- 340 "2016 IEEE GRSS Data Fusion Contest." <http://www.grss-ieee.org/community/technical-committees/data-fusion/>.
- 341 "Indian Pines Dataset." <http://dynamo.ecn.purdue.edu/biehl/MultiSpec/>.
- 342 "Kennedy Space Center." http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes#Kennedy_Space_Center_.28KSC.29.
- 343 "Pavia Dataset." http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes#Pavia_Centre_and_University.
- 344 "Salinas Dataset." http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes#Salinas.
- 345 "Washington DC Mall." <https://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html>.
- 346 G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing* **117**, 11–28 (2016). DOI:10.1016/j.isprsjprs.2016.03.014.
- 347 Y. Chen, Z. Lin, X. Zhao, *et al.*, "Deep Learning-Based Classification of Hyperspectral Data," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **7**(6), 2094–2107 (2014).
- 348 V. Slavkovikj, S. Verstocket, W. De Neve, *et al.*, "Hyperspectral image classification with convolutional neural networks," in *Proceedings of the 23rd ACM international conference on Multimedia*, 1159–1162, ACM (2015).
- 349 C. Tao, H. Pan, Y. Li, *et al.*, "Unsupervised Spectral-Spatial Feature Learning With Stacked Sparse Autoencoder for Hyperspectral Imagery Classification," *IEEE Geoscience and Remote Sensing Letters* **12**(12), 2438–2442 (2015).
- 350 Y. LeCun, "Learning invariant feature hierarchies," in *Computer vision—ECCV 2012. Workshops and demonstrations*, 496–505, Springer (2012).
- 351 M. Pal, "Kernel methods in remote sensing: a review," *ISH Journal of Hydraulic Engineering* **15**(sup1), 194–215 (2009).
- 352 E. M. Abdel-Rahman and F. B. Ahmed, "The application of remote sensing techniques to sugarcane (*Saccharum spp. hybrid*) production: a review of the literature," *International Journal of Remote Sensing* **29**(13), 3753–3767 (2008).

- 353 I. Ali, F. Greifeneder, J. Stamenkovic, *et al.*, “Review of machine learning approaches for biomass and soil moisture retrievals from remote sensing data,” *Remote Sensing* **7**(12), 16398–16421 (2015).
- 354 E. Adam, O. Mutanga, and D. Rugege, “Multispectral and hyperspectral remote sensing for identification and mapping of wetland vegetation: A review,” *Wetlands Ecology and Management* **18**(3), 281–296 (2010).
- 355 S. L. Ozesmi and M. E. Bauer, “Satellite remote sensing of wetlands,” *Wetlands ecology and management* **10**(5), 381–402 (2002). DOI:10.1023/A:1020908432489.
- 356 W. A. Dorigo, R. Zurita-Milla, A. J. W. de Wit, *et al.*, “A review on reflective remote sensing and data assimilation techniques for enhanced agroecosystem modeling,” *International journal of applied earth observation and geoinformation* **9**(2), 165–193 (2007). DOI:10.1016/j.jag.2006.05.003.
- 357 C. Kuenzer, M. Ottinger, M. Wegmann, *et al.*, “Earth observation satellite sensors for biodiversity monitoring: potentials and bottlenecks,” *International Journal of Remote Sensing* **35**(18), 6599–6647 (2014). DOI:10.1080/01431161.2014.964349.
- 358 K. Wang, S. E. Franklin, X. Guo, *et al.*, “Remote sensing of ecology, biodiversity and conservation: a review from the perspective of remote sensing specialists,” *Sensors* **10**(11), 9647–9667 (2010). DOI:10.3390/s101109647.
- 359 F. E. Fassnacht, H. Latifi, K. Stereńczak, *et al.*, “Review of studies on tree species classification from remotely sensed data,” *Remote Sensing of Environment* **186**, 64–87 (2016). DOI:10.1016/j.rse.2016.08.013.
- 360 I. Ali, F. Greifeneder, J. Stamenkovic, *et al.*, “Review of machine learning approaches for biomass and soil moisture retrievals from remote sensing data,” *Remote Sensing* **7**(12), 16398–16421 (2015). DOI:10.3390/rs71215841.
- 361 A. Mahendran and A. Vedaldi, “Understanding deep image representations by inverting them,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5188–5196 (2015).
- 362 J. Yosinski, J. Clune, A. Nguyen, *et al.*, “Understanding neural networks through deep visualization,” *arXiv preprint arXiv:1506.06579* (2015).
- 363 K. Simonyan, A. Vedaldi, and A. Zisserman, “Deep inside convolutional networks: Visualising image classification models and saliency maps,” *arXiv preprint arXiv:1312.6034* (2013).
- 364 D. Erhan, Y. Bengio, A. Courville, *et al.*, “Visualizing higher-layer features of a deep network,” *University of Montreal* **1341**, 3 (2009).
- 365 J. A. Benediktsson, J. Chanussot, and W. W. Moon, “Very High-Resolution Remote Sensing : Challenges and Opportunities,” *Proceedings of the IEEE* **100**(6), 1907–1910 (2012). DOI:10.1109/JPROC.2012.2190811.
- 366 J. Fohringer, D. Dransch, H. Kreibich, *et al.*, “Social media as an information source for rapid flood inundation mapping,” *Natural Hazards and Earth System Sciences* **15**(12), 2725–2738 (2015).
- 367 V. Frias-Martinez and E. Frias-Martinez, “Spectral clustering for sensing urban land use using Twitter activity,” *Engineering Applications of Artificial Intelligence* **35**, 237–245 (2014).
- 368 T. Kohonen, “The self-organizing map,” *Neurocomputing* **21**(1), 1–6 (1998).

- 369 S. E. Middleton, L. Middleton, and S. Modafferi, "Real-time crisis mapping of natural disasters using social media," *IEEE Intelligent Systems* **29**(2), 9–17 (2014).
- 370 V. K. Singh, "Social Pixels : Genesis and Evaluation," in *Proceedings of the 18th ACM international conference on Multimedia (ACMM)*, 481–490 (2010).
- 371 D. Sui and M. Goodchild, "The convergence of GIS and social media: challenges for GIScience," *International Journal of Geographical Information Science* **25**(11), 1737–1748 (2011).
- 372 A. J. Pinar, J. Rice, L. Hu, *et al.*, "Efficient multiple kernel classification using feature and decision level fusion," *IEEE Transactions on Fuzzy Systems* **PP**(99), 1–1 (2016).
- 373 D. T. Anderson, T. C. Havens, C. Wagner, *et al.*, "Extension of the fuzzy integral for general fuzzy set-valued information," *IEEE Transactions on Fuzzy Systems* **22**, 1625–1639 (2014).
- 374 P. Ghamisi, B. Hfle, and X. X. Zhu, "Hyperspectral and lidar data fusion using extinction profiles and deep convolutional neural network," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **PP**(99), 1–14 (2016).
- 375 J. Ngiam, A. Khosla, M. Kim, *et al.*, "Multimodal deep learning," in *ICML*, L. Getoor and T. Scheffer, Eds., 689–696, Omnipress (2011).
- 376 Z. Kira, R. Hadsell, G. Salgian, *et al.*, "Long-Range Pedestrian Detection using stereo and a cascade of convolutional network classifiers," in *IEEE International Conference on Intelligent Robots and Systems*, 2396–2403 (2012).
- 377 F. Rottensteiner, G. Sohn, J. Jung, *et al.*, "The ISPRS benchmark on urban object classification and 3D building reconstruction," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci* **1**, 293–298 (2012).
- 378 J. Ngiam, Z. Chen, S. A. Bhaskar, *et al.*, "Sparse filtering," in *Advances in neural information processing systems*, 1125–1133 (2011).
- 379 Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *European conference on computational learning theory*, 23–37, Springer (1995).
- 380 Q. Zhao, M. Gong, H. Li, *et al.*, "Three-Class Change Detection in Synthetic Aperture Radar Images Based on Deep Belief Network," in *Bio-Inspired Computing-Theories and Applications*, 696–705, Springer (2015).
- 381 A. P. Tewkesbury, A. J. Comber, N. J. Tate, *et al.*, "A critical synthesis of remotely sensed optical image change detection techniques," *Remote Sensing of Environment* **160**, 1–14 (2015).
- 382 C. Brekke and A. H. S. Solberg, "Oil spill detection by satellite remote sensing," *Remote sensing of environment* **95**(1), 1–13 (2005). [DOI:10.1016/j.rse.2004.11.015].
- 383 H. Mayer, "Automatic object extraction from aerial imagery a survey focusing on buildings," *Computer vision and image understanding* **74**(2), 138–149 (1999). DOI:10.1006/cviu.1999.0750.
- 384 B. Somers, G. P. Asner, L. Tits, *et al.*, "Endmember variability in spectral mixture analysis: A review," *Remote Sensing of Environment* **115**(7), 1603–1616 (2011). DOI:10.1016/j.rse.2011.03.003.
- 385 D. Tuia, C. Persello, and L. Bruzzone, "Domain Adaptation for the Classification of Remote Sensing Data: An Overview of Recent Advances," *IEEE Geoscience and Remote Sensing Magazine* **4**(2), 41–57 (2016). DOI:10.1109/MGRS.2016.2548504.

- 386 Y. Yang and S. Newsam, “Bag-of-visual-words and spatial extensions for land-use classification,” in *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, 270–279, ACM (2010).
- 387 V. Risojević, S. Momić, and Z. Babić, “Gabor descriptors for aerial image classification,” in *International Conference on Adaptive and Natural Computing Algorithms*, 51–60, Springer (2011).
- 388 K. Chatfield, K. Simonyan, A. Vedaldi, *et al.*, “Return of the devil in the details: Delving deep into convolutional nets,” *arXiv preprint arXiv:1405.3531* (2014).
- 389 G. Sheng, W. Yang, T. Xu, *et al.*, “High-resolution satellite scene classification using a sparse coding based multiple feature combination,” *International journal of remote sensing* **33**(8), 2395–2412 (2012).
- 390 S. H. Lee, C. S. Chan, S. J. Mayo, *et al.*, “How deep learning extracts and learns leaf features for plant classification,” *Pattern Recognition* **71**, 1 – 13 (2017).
- 391 A. Joly, H. Goëau, H. Glotin, *et al.*, “LifeCLEF 2016: multimedia life species identification challenges,” in *International Conference of the Cross-Language Evaluation Forum for European Languages*, 286–310, Springer (2016).
- 392 S. H. Lee, C. S. Chan, P. Wilkin, *et al.*, “Deep-plant: Plant identification with convolutional neural networks,” in *2015 IEEE International Conference on Image Processing (ICIP)*, 452–456 (2015).
- 393 Z. Ding, N. Nasrabadi, and Y. Fu, “Deep transfer learning for automatic target classification: MWIR to LWIR,” in *SPIE Defense+ Security*, 984408, International Society for Optics and Photonics (2016).
- 394 Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE transactions on neural networks* **5**(2), 157–166 (1994).
- 395 X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Aistats*, **9**, 249–256 (2010).
- 396 R. K. Srivastava, K. Greff, and J. Schmidhuber, “Training very deep networks,” in *Advances in neural information processing systems*, 2377–2385 (2015).
- 397 J. Sokolic, R. Giryes, G. Sapiro, *et al.*, “Robust large margin deep neural networks,” *arXiv preprint arXiv:1605.08254* (2016).
- 398 G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural computation* **18**(7), 1527–1554 (2006).
- 399 D. Erhan, A. Courville, and P. Vincent, “Why Does Unsupervised Pre-training Help Deep Learning ?,” *Journal of Machine Learning Research* **11**, 625–660 (2010). DOI:10.1145/1756006.1756025.
- 400 K. He, X. Zhang, S. Ren, *et al.*, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 1026–1034 (2015).
- 401 N. Srivastava, G. E. Hinton, A. Krizhevsky, *et al.*, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” *Journal of Machine Learning Research (JMLR)* **15**(1), 1929–1958 (2014).
- 402 S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167* (2015).

- 403 Q. V. Le, A. Coates, B. Prochnow, *et al.*, “On Optimization Methods for Deep Learning,” in *Proceedings of The 28th International Conference on Machine Learning (ICML)*, 265–272 (2011).
- 404 S. Ruder, “An overview of gradient descent optimization algorithms,” *arXiv preprint arXiv:1609.04747* (2016).
- 405 J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” *Journal of Machine Learning Research* **12**(Jul), 2121–2159 (2011).
- 406 M. D. Zeiler, “Adadelta: an adaptive learning rate method,” *arXiv preprint arXiv:1212.5701* (2012).
- 407 D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980* (2014).
- 408 T. Schaul, S. Zhang, and Y. LeCun, “No more pesky learning rates.,” *ICML (3)* **28**, 343–351 (2013).
- 409 R. K. Srivastava, K. Greff, and J. Schmidhuber, “Highway networks,” *arXiv preprint arXiv:1505.00387* (2015).
- 410 K. He, X. Zhang, S. Ren, *et al.*, “Deep residual learning for image recognition,” *arXiv preprint arXiv:1512.03385* (2015).
- 411 D. Balduzzi, B. McWilliams, and T. Butler-Yeoman, “Neural taylor approximations: Convergence and exploration in rectifier networks,” *arXiv preprint arXiv:1611.02345* (2016).

John E. Ball is an Assistant professor of Electrical and Computer Engineering at Mississippi State University (MSU), USA. He received the Ph.D. degree in in Electrical Engineering from Mississippi State University in 2007, with a certificate in remote sensing. He is a co-director of the Sensor Analysis and Intelligence Laboratory (SAIL) at MSU and the director of the Simrall Radar Laboratory. He is the author of 45 journal and conference papers, and 22 technical tutorials, white papers, and technical reports, and has written one book chapter. He received best research paper of the year from the Veterinary and Comparative Orthopaedics and Traumatology in 2016 and technical paper of the year award from the Georgia Tech Research Institute in 2012. His current research interests include deep learning, remote sensing, remote sensing, machine learning, digital signal and image processing, and radar systems. Dr. Ball is an associate editor for the SPIE Journal of Applied Remote Sensing.

Derek T. Anderson received the Ph.D. in electrical and computer engineering (ECE) in 2010 from the University of Missouri, Columbia, MO, USA. He is currently an Associate Professor and the Robert D. Guyton Chair in ECE at Mississippi State University (MSU), USA, an Intermittent Faculty Member with the Naval Research Laboratory, co-director of the Sensor Analysis and Intelligence Laboratory (SAIL) at MSU and an Associate Editor for the IEEE Transactions on Fuzzy Systems. His research interests include new frontiers in data/information fusion for pattern recognition and automated decision making in signal/image understanding and computer vision with an emphasis on uncertainty and heterogeneity. Prof. Anderson’s primary research contributions to date include multi-source (sensor, algorithm and human) fusion, Choquet integrals (extensions, embeddings, learning), signal/image feature learning, multi-kernel learning, cluster validation, hyperspectral image understanding and linguistic summarization of video. He has published 100+

(journal, conference and book chapter) articles, he is the program co-chair of FUZZ-IEEE 2019, he co-authored the 2013 best student paper in Automatic Target Recognition at SPIE, he received the best overall paper award at the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) 2012, and he received the 2008 FUZZ-IEEE best student paper award.

Chee Seng Chan received the Ph.D. degree from the University of Portsmouth, Hampshire, U.K., in 2008. He is currently a Senior Lecturer with the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia. His current research interests include computer vision and fuzzy qualitative reasoning, with an emphasis on image and video understanding. Dr. Chan was a recipient of the Institution of Engineering and Technology (Malaysia) Young Engineer Award in 2010, the Hitachi Research Fellowship in 2013, and the Young Scientist Network-Academy of Sciences Malaysia in 2015. He is the Founding Chair of the IEEE Computational Intelligence Society, Malaysia Chapter, and the Founder of Malaysian Image Analysis and Machine Intelligence Association. He is a Chartered Engineer of the Institution of Engineering and Technology, U.K.

List of Figures

- 1 Block diagrams of DL architectures. (a) AE. (b) CNN. (c) DBN. (d) RNN.

List of Tables

- 1 Acronym list.
- 2 Some popular DL tools.
- 3 DL paper subject areas in remote sensing.
- 4 Representative DL and FL Survey papers.
- 5 HSI Dataset Usage.
- 6 HSI Overall Accuracy Results in percent. IP = Indian Pines, KSC = Kennedy Space Center, PaCC = Pavia City Center, Pau = Pavia University, Sal = Salinas, DCM = Washington DC Mall. Results higher than 99% are in bold.

Table 2 Some popular DL tools.

Tool & Citation	Tool Summary and Website
AlexNet ³⁸	A large-scale CNN with a non-saturating, neurons and a very efficient GPU parallel implementation of the convolution operation to make training faster. Website: http://code.google.com/p/cuda-convnet/
Caffe ⁶⁸	C++ library with Python and Matlab interfaces. Website: http://caffe.berkeleyvision.org/
cuda-convnet2 ³⁸	The DL tool cuda-convnet2 is a fast C++/CUDA CNN implementation, and can also model any directed acyclic graphs. Training is performed using back-propagation. Offers faster training on Kepler-generation GPUs and multi-GPU training support. Website: https://code.google.com/p/cuda-convnet2/
gvnn ⁷¹	The DL package gvnn is a NN library in Torch aimed towards bridging the gap between classic geometric computer vision and DL. This DL package is used for recognition, end-to-end visual odometry, depth estimation, etc. Website: https://github.com/ankurhanda/gvnn
Keras ⁷²	Keras is a high-level Python NN library capable of running on top of either TensorFlow or Theano and was developed with a focus on enabling fast experimentation. Keras (1) allows for easy and fast prototyping, (2) supports both convolutional networks and recurrent networks, (3) supports arbitrary connectivity schemes, and (4) runs seamlessly on CPUs and GPUs. Website: https://keras.io/ and https://github.com/fchollet/keras
MatConvNet ⁷⁰	A Matlab toolbox implementing CNNs with many pre-trained CNNs for image classification, segmentation, etc. Website: http://www.vlfeat.org/matconvnet/
MXNet ⁷³	MXNet is a DL library. Features include declarative symbolic expression with imperative tensor computation and differentiation to derive gradients. MXNet runs on mobile devices to distributed GPU clusters. Website: https://github.com/dmlc/mxnet/
TensorFlow ⁶⁹	An open source software library for tensor data flow graph computation. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile devices. Website: https://www.tensorflow.org/
Theano ⁷⁴	A Python library that allows you to define, optimize, and efficiently evaluate mathematical expressions involving multi-dimensional arrays. Theano features (1) tight integration with NumPy, (2) transparent use of a GPU, (3) efficient symbolic differentiation, and (4) dynamic C code generation. Website: http://deeplearning.net/software/theano
Torch ⁷⁵	Torch is an embeddable scientific computing framework with GPU optimizations, which uses the LuaJIT scripting language and a C/CUDA implementation. Torch includes (1) optimized linear algebra and numeric routines, (2) neural network and energy-based models, and (3) GPU support. Website: http://torch.ch/

Table 3 DL paper subject areas in remote sensing.

Area	References	Area	References
3D (depth and shape) analysis	107–115	Advanced driver assistance systems	116–120
Animal detection	121	Anomaly detection	122
Automated Target Recognition	123–134	Change detection	135–139
Classification	140–190	Data fusion	191
Dimensionality reduction	192, 193	Disaster analysis/assessment	194
Environment and water analysis	195–198	Geo-information extraction	199
Human detection	200–203	Image denoising/enhancement	204, 205
Image Registration	206	Land cover classification	207–211
Land use/classification	212–222	Object recognition and detection	223–233
Object tracking	234, 235	Pansharpening	236
Planetary studies	237	Plant and agricultural analysis	238–243
Road segmentation/extraction	244–250	Scene understanding	251–253
Semantic segmentation/annotation	254–266	Segmentation	267–272
Ship classification/detection	273–275	Super-resolution	276–279
Traffic flow analysis	280, 281	Underwater detection	282–285
Urban/building	286–296	Vehicle detection/recognition	297–310
Weather forecasting	311–313		

Table 4 Representative DL and FL Survey papers.

Ref.	Paper Contents
7	A survey paper on DL. Covers CNNs, DBNs, etc.
329	Brief intro to neural networks in remote sensing.
13	Overview of unsupervised feature learning and deep learning. Provides overview of probabilistic models (undirected graphical, RBM, AE, SAE, DAE, contractive autoencoders, manifold learning, difficulty in training deep networks, handling high-dimensional inputs, evaluating performance, etc.)
330	Examines big-data impacts on SVM machine learning.
1	Covers about 170 publications in the area of scene classification and discusses limitations of datasets and problems associated with high-resolution imagery. They discuss limitations of handcrafted features such as texture descriptors, GIST, SIFT, HOG.
2	A good overview of architectures, algorithms, and applications for DL. Three important reasons for DL success are (1) GPU units, (2) recent advances in DL research. In addition, we note that (3) would be success of DL in many image processing challenges. DL is at the intersection of machine learning, Neural Networks, optimization, graphical modeling, pattern recognition, probability theory and signal processing. They discuss generative, discriminative, and hybrid deep architectures. They show there is vast room to improve the current optimization techniques in DL.
331	Overview of NN in image processing.
332	Discusses trends in extreme learning machines, which are linear, single hidden layer feedforward neural networks. ELMs are comparable or better than SVMs in generalization ability. In some cases, ELMs have comparable performance to DL approaches. They generally have high generalization capability, are universal approximators, don't require iterative learning, and have a unified learning theory.
333	Provides overview of feature reduction in remote sensing imagery.
8	A survey of deep neural networks, including the AE, the CNN, and applications.
334	Survey of image classification methods in remote sensing.
335	Short survey of DL in hyperspectral remote sensing. In particular, in one study, there was a definite sweet spot shown in the DL depth.
336	Overview of shallow HSI processing.
29	Overview of shallow endmember extraction algorithms.
10	An in-depth historical overview of DL.
4	History of DL.
337	A review of road extraction from remote sensing imagery.
9	A review of DL in signal and image processing. Comparisons are made to shallow learning, and DL advantages are given.
3	Provides a general framework for DL in remote sensing. Covers four RS perspectives: (1) image processing, (2) pixel-based classification, (3) target recognition, and (4) scene understanding.

Table 5 HSI Dataset Usage.

Dataset and Reference	Number of uses
IEEE GRSS 2013 Data Fusion Contest ³³⁸	4
IEEE GRSS 2015 Data Fusion Contest ³³⁹	1
IEEE GRSS 2016 Data Fusion Contest ³⁴⁰	2
Indian Pines ³⁴¹	27
Kennedy Space Center ³⁴²	8
Pavia City Center ³⁴³	13
Pavia University ³⁴³	19
Salinas ³⁴⁴	11
Washington DC Mall ³⁴⁵	2

Table 6 HSI Overall Accuracy Results in percent. IP = Indian Pines, KSC = Kennedy Space Center, PaCC = Pavia City Center, Pau = Pavia University, Sal = Salinas, DCM = Washington DC Mall. Results higher than 99% are in bold.

Ref	IP	KSC	PaCC	PaU	Sal	DCM
267	93.4					
141	98.0	98.0		98.4		95.4
142	97.6					
143	97.6					
347		98.8	98.5			
314	96.0		99.1			
148				94.3		
191	89.6			87.1		
151		96.6				
153	90.2			92.6	92.6	
155		84.2				
158	92.1			94.0		
159				99.9		
160	96.3					
162	94.3			96.5	94.8	
163	97.6			99.4	98.8	
164		96.0	85.6			
167	91.9		99.8	96.7	95.5	
168			94.0	93.5		
216	86.5			82.6		
169	99.2		99.9			
170			96.0			83.8
211	98.9		99.9		99.5	
172	95.7			99.6	97.4	
173	96.8					
217	79.3					
175	97.9	97.9		96.8		
179		80.5				
192	93.1			95.6		
348	96.6					
221	73.0	89.0				
180	93.1			90.4	99.4	
181	95.6					
183				67.9	85.2	
184			95.2			
193	82.1		97.4			
187	99.7			98.8		
188			99.7	96.8		