

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/327019925>

# Deep learning for smart agriculture: Concepts, tools, applications, and opportunities

Article in *International Journal of Agricultural and Biological Engineering* · July 2018

DOI: 10.25165/j.ijabe.20181104.4475

CITATIONS

35

READS

1,186

11 authors, including:



Kai hu

Jiangnan University

35 PUBLICATIONS 318 CITATIONS

[SEE PROFILE](#)



Yingkuan Wang

Chinese Academy of Agricultural Engineering, Ministry of Agriculture (PRC)

29 PUBLICATIONS 1,365 CITATIONS

[SEE PROFILE](#)



Min Huang

166 PUBLICATIONS 4,247 CITATIONS

[SEE PROFILE](#)



Qibing Zhu

Jiangnan University

70 PUBLICATIONS 1,225 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Editor-in-Chief, International Journal of Agricultural and Biological Engineering [View project](#)



Deep Learning and Applications [View project](#)

# Deep learning for smart agriculture: Concepts, tools, applications, and opportunities

Nanyang Zhu<sup>1,2</sup>, Xu Liu<sup>1,2</sup>, Ziqian Liu<sup>1,2</sup>, Kai Hu<sup>1,2</sup>, Yingkuan Wang<sup>3</sup>, Jinglu Tan<sup>4</sup>, Min Huang<sup>1</sup>,  
Qibing Zhu<sup>1</sup>, Xunsheng Ji<sup>1</sup>, Yongnian Jiang<sup>5</sup>, Ya Guo<sup>1,2,4\*</sup>

(1. Key Laboratory of Advanced Process Control for Light Industry, Ministry of Education, Jiangnan University, Wuxi 214122, China;

2. School of Internet of Things, Jiangnan University, Wuxi 214122, China;

3. Chinese Academy of Agricultural Engineering, Beijing 100125, China;

4. Department of Bioengineering, University of Missouri, Columbia, MO 65211, USA;

5. Jiangsu Zhongnong IoT Technology Co., LTD, Yixing 214200, China)

**Abstract:** In recent years, Deep Learning (DL), such as the algorithms of Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Generative Adversarial Networks (GAN), has been widely studied and applied in various fields including agriculture. Researchers in the fields of agriculture often use software frameworks without sufficiently examining the ideas and mechanisms of a technique. This article provides a concise summary of major DL algorithms, including concepts, limitations, implementation, training processes, and example codes, to help researchers in agriculture to gain a holistic picture of major DL techniques quickly. Research on DL applications in agriculture is summarized and analyzed, and future opportunities are discussed in this paper, which is expected to help researchers in agriculture to better understand DL algorithms and learn major DL techniques quickly, and further to facilitate data analysis, enhance related research in agriculture, and thus promote DL applications effectively.

**Keywords:** deep learning, smart agriculture, neural network, convolutional neural networks, recurrent neural networks, generative adversarial networks, artificial intelligence, image processing, pattern recognition

**DOI:** 10.25165/ijabe.20181104.4475

**Citation:** Zhu N Y, Liu X, Liu Z Q, Hu K, Wang Y K, Tan J L, et al. Deep learning for smart agriculture: Concepts, tools, applications, and opportunities. Int J Agric & Biol Eng, 2018; 11(4): 32–44.

## 1 Introduction

A standard artificial neural network (ANN) model consists of many neurons (connected processors), each producing a sequence of real-valued activations<sup>[1]</sup>. When sensors perceive environment changes, input neurons will be activated and other neurons will then get activated through weighted connections from previously active neurons. Depending on the specific problem and the neuron topology, these behaviors may require long chains of computational stages, where each of the stage transforms the aggregate activation of the network. DL is about how to accurately assign credit across many such stages<sup>[2]</sup>. Deep learning allows computational models that are composed of

multiple processing layers to represent data with multiple levels of abstraction. Great improvements of the method can be found in many research domains. The concept of BP (Back Propagation) Neural Network is the basis for many DL algorithms.

With massive enthusiasm pouring into the DL field, great improvements have been achieved in recent years. DL has drawn a lot of attention in agriculture. One of its applications in agriculture is image recognition, which has conquered a lot of obstacles that limit fast development in robotic and mechanized agro-industry and agriculture<sup>[3]</sup>. These improvements can be seen in many aspects of agriculture, such as plant disease detection, weed control, and plant counting. Researchers in agriculture may not be experienced programmers. They often directly use publicly available software frameworks for deep learning without carefully examining the learning mechanisms used. An understanding of DL algorithms can facilitate data analysis and thus enhance research in agriculture. Although various commercial software frameworks are available, there is a lack of a systematic summary of major DL algorithms<sup>[3]</sup>, including concepts, application limitations, flow charts, and example codes, which can help researchers in agriculture to learn major DL techniques quickly and use them effectively.

In order to provide a holistic picture of DL to researchers in agriculture fields and enhance modern smart agriculture development, this work summarizes BP and common DL algorithms (Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Generative Adversarial Networks (GAN)) and their applications in agriculture, with a focus on

**Received date:** 2018-06-16 **Accepted date:** 2018-07-23

**Biographies:** Nanyang Zhu, Master candidate, research interests: IoT, Email: zhunanyang08@163.com; Xu Liu, Undergraduate, research interests: IoT, Email: francislucien2017@126.com; Ziqian Liu, Undergraduate, research interests: IoT, Email: jolley1074@gmail.com; Kai Hu, PhD, research interests: cartography and geographic information engineering, Email: hukai\_wlw@jiangnan.edu.cn; Yingkuan Wang, PhD, Professor, research interests: agricultural mechanization and information, Email: wangyk@agri.gov.cn; Jinglu Tan, PhD, Professor, research interests: bioengineering, Email: tanj@missouri.edu; Min Huang, PhD, Professor, research interests: control theory and engineering, Email: huangmzqb@163.com; Qibing Zhu, PhD, Professor, research interests: mechanical engineering, Email: zhuqib@163.com; Xunsheng Ji, PhD, Professor, research interests: precision instrumentation and mechanical engineering, Email: jixunsheng@163.com; Yongnian Jiang, CEO of Jiangsu Zhongnong IoT Technology Co., LTD, Email: jxrjyn@126.com;

\*Corresponding Author: Ya Guo, PhD, Professor, research interest: bioengineering, Room C510, School of IoT Engineering, Jiangnan University, Wuxi 214122, China, Tel: +86-15706185006, Email: guoy@jiangnan.edu.cn.

applications published in the last three years. Example codes for BP, CNN, RNN, and GAN in Python are also provided.

## 2 Common deep learning algorithms

CNN, RNN, and GAN are the most commonly used DL algorithms. There are many other sub-category DL algorithms, such as VGGNet<sup>[4]</sup>, ConvNets<sup>[5]</sup>, LSTM<sup>[6,7]</sup> and DCGAN<sup>[8,9]</sup>. It is unfeasible to include all of them in one summary. They can be derived from the three common DL algorithms directly or indirectly<sup>[10]</sup>. Understanding the three common DL algorithms is very helpful for learning the sub-category DL algorithms. Therefore, the sub-category DL algorithms are not reviewed with details. The concept of backpropagation (BP) is the basis for many ANN although itself does not mean a deep network. It thus will be introduced in this section first.

### 2.1 Feedforward Neural Network and Backpropagation (BP)

An example structure of feedforward neural network based on backpropagation or BP neural network is shown in Figure 1. It is a supervised learning algorithm using error back propagation, composed of multiple layers of hidden neurons in full connection. This means that a layer of neurons is connected to an upper layer of neurons and each layer has an activation function to limit the output of amplitude of neurons by linear or nonlinear transformation of the input of the afferent neuron. Hidden neurons can learn the salient features of training data from continuous forward propagation<sup>[11]</sup>.

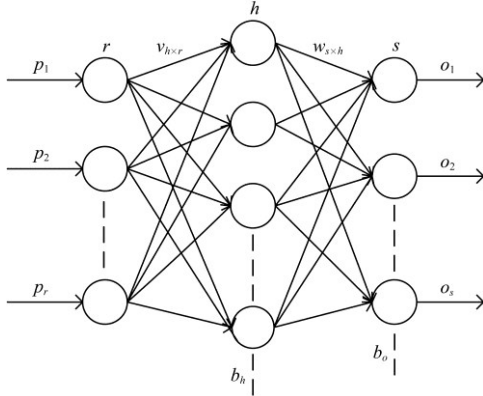


Figure 1 Example structure of a BP neural network

Assuming that there is a set of training data  $\{(p^1, d^1), (p^2, d^2), \dots, (p^r, d^r)\}$ .  $[v_{ij}]_{h \times r}$  is the weights between neuron code  $h$  and neuron code  $r$ .  $b_h$  denotes the threshold in neuron code  $h$ ,  $(d_1, d_2, \dots, d_s)$  denotes code output.  $net_i^h$  is the input of the first hidden layer.  $a_i$  is the output of the first hidden layer, and  $f_h$  is the activation function of the first hidden layer.  $net_k^o$  is the input of the output layer.  $f_o$  is the activation function of the output layer.  $o_k$  is the output of the output layer. The output of each neuron code is determined by the output of previous neuron. The major formulations for the algorithm can be listed as Equations (1)-(4).

Input layer to hidden layer input:

$$net_i^h = \sum_{j=1}^r v_{ij} p_j + b_{hj} \quad (1)$$

Hidden layer output:

$$a_i = f_h(net_i^h) = f_h\left(\sum_{j=1}^r v_{ij} p_j + b_{hj}\right) \quad (2)$$

Output layer input:

$$net_k^o = \sum_{i=1}^h w_{ki} a_i + b_{ok} \quad (3)$$

Output layer output:

$$o_k = f_o(net_k^o) = f_o\left(\sum_{i=1}^h w_{ki} a_i + b_{ok}\right) \quad (4)$$

Cost function:

For neural networks, the cost function plays an important role in optimizing the model parameters (weights and thresholds). A cost function is a measure of the error between the predicted output and the actual output for the training samples. The training process gradually reduces the value of the cost function by the gradient descent. There are two types of cost function expressions. One is a quadratic loss function<sup>[11]</sup>, and the other is a cross-entropy loss function<sup>[12]</sup>. Here, the quadratic loss function  $E$  is used as an example to demonstrate BP and it is expressed as Equation (5):

$$\begin{aligned} E &= \frac{1}{2} \sum_{k=1}^s (d_k - o_k)^2 = \frac{1}{2} \sum_{k=1}^s (d_k - f_o(net_k^o))^2 \\ &= \frac{1}{2} \sum_{k=1}^s \left[ d_k - f_o\left(\sum_{i=1}^h w_{ki} f_h\left(\sum_{j=1}^r v_{ij} p_j + b_{hj}\right) + b_{ok}\right) \right]^2 \end{aligned} \quad (5)$$

Backpropagation is based on the gradient descent of the cost function and the chain rule of differentiation<sup>[13]</sup>, whose conceptual structure is shown in Figure 2. First, there is a need to differentiate the weights of the cost function to get an error value, then backpropagate the error value. During this process, there is a need to constantly adjust and update the weights and thresholds.

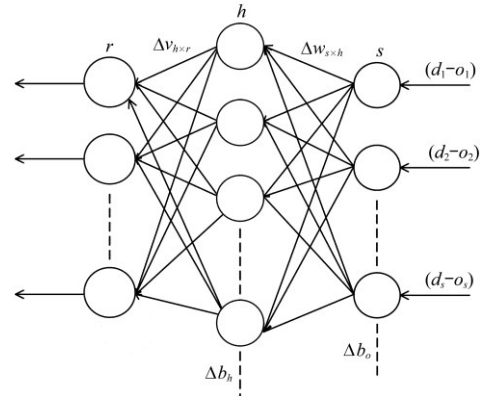


Figure 2 Concept of backpropagation

Assuming that:

$\Delta w_{ki}$  - weight adjustment (local gradient) from the output layer to the hidden layer,

$\Delta b_{ok}$  - threshold adjustment (local gradient) from the output layer to the hidden layer,

$\Delta v_{ij}$  - weight adjustment (local gradient) from the hidden layer to the input layer, and

$\Delta b_{hj}$  - threshold adjustment (local gradient) from the hidden layer to the input layer.

The major formulations of backpropagation algorithm can be listed as Equations (6)-(9).

Weight and threshold adjustment (local gradient) from the hidden layer to the output layer:

$$\Delta w_{ki} = -\eta \times \frac{\partial E}{\partial w_{ki}} = \eta \times (d_k - o_k) \times f_o'(net_k^o) \times a_i \quad (6)$$

$$\Delta b_{ok} = -\eta \times \frac{\partial E}{\partial b_{ok}} = \eta \times (d_k - o_k) \times f_o'(net_k^o) \quad (7)$$

Weight and threshold adjustment (local gradient) from the input layer to the hidden layer:

$$\Delta v_{ij} = -\eta \times \frac{\partial E}{\partial v_{ij}} = \eta \times \left\{ \left[ \sum_{k=1}^s (d_k - o_k) \times f'_o(\text{net}_k^o) \times w_{ki} \right] \times f'_h(\text{net}_i^h) \right\} \times p_j \quad (8)$$

$$\Delta b_{hj} = -\eta \times \frac{\partial E}{\partial b_{hj}} = \eta \times \left[ \sum_{k=1}^s (d_k - o_k) \times f'_o(\text{net}_k^o) \times w_{ki} \right] \times f'_h(\text{net}_i^h) \quad (9)$$

In the process of backpropagation, the weights and thresholds are adjusted by reversing the value of error<sup>[14]</sup>. A flowchart of the BP algorithm is shown in Figure 3.

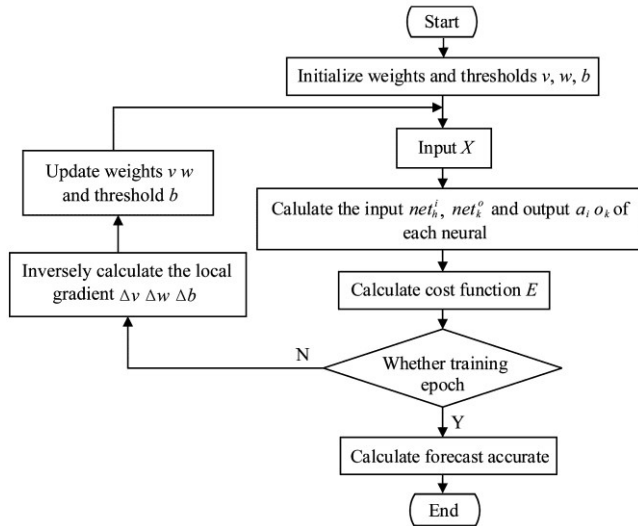


Figure 3 Flowchart of the BP programming

BP neural networks have many merits, including strong nonlinear mapping capabilities and high degree of self-learning and self-adaptability. They are widely applied in many fields such as speech analysis, image recognition, digital watermarking, and computer vision. They solve many problems that cannot be done by traditional machine learning algorithms. Its classification function is particularly suitable for applications in pattern recognition<sup>[15]</sup> and classification<sup>[16]</sup>. Nevertheless, the networks used for object recognition or natural language processing contain thousands of hidden units, which bring exponentially increasing number of parameters. BP neural networks may fall into local minima in practical applications. Thus, how to select typical samples from training data is an unsolved problem when massive amounts of complex data are available. The traditional BP neural networks are no longer sufficiently effective in some areas. Hence, deep networks such as CNN, RNN, and GAN have caught increasing attention.

## 2.2 Convolutional Neural Networks (CNN)

CNN is a deep learning algorithm composed of multiple convolutional layers, pooling layers, and fully connected layers, which has resulted in many breakthroughs in speech recognition, face recognition, natural language processing and so on<sup>[17]</sup>. The structure composed of the convolutional layers and the pooling layers is for feature extraction and the fully connected layers function as a classifier. BP neural networks mainly map features through the network to specific values, whereas convolutional networks first convert signals into features and then map the features to a specific target value<sup>[18]</sup>.

### Convolutional Layer:

Assuming that an RGB image is taken as input  $X$  and there are six convolution kernels  $W^{[19]}$ . The image size is an  $H \times W \times 3$  three-dimensional matrix. The convolution kernel size associated

with each convolutional layer is  $h \times w \times 3$ . The threshold of convolutional layer is  $b$ . The size of new image feature  $X$  after convolution calculation is  $H_{new} \times W_{new}$ , whose convolutional layer computation process is shown in Figure 4. For each convolutional layer, the number of convolution kernels determines the number of output feature maps or the number of inputs in the pooling layer. To identify the edge information of an image, the method of zero-padding in which zero is added to the border of the input vector is applied and the size of zero-padding is  $P$ . It is also vital for convolution calculation to set a suitable stride with size  $S$ <sup>[20]</sup> during the process of calculating convolution in the convolutional layer. The major formulations are listed as Equations (10)-(12).

$$X_{new} = X \times W = \sum_{k=1}^3 (X_k \times W_k) + b \quad (10)$$

$$H_{new} = \frac{H - h + 2 \times P}{S} + 1 \quad (11)$$

$$W_{new} = \frac{W - w + 2 \times P}{S} + 1 \quad (12)$$

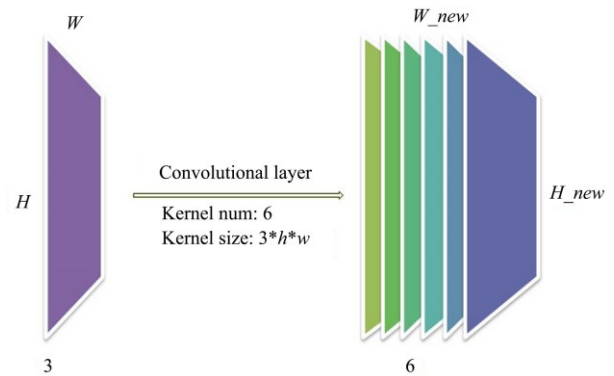


Figure 4 Convolutional layer computation process

### Pooling Layer:

The function of the pooling layer is to compress the input tensor and reduce the size of the input data, which means changing each  $n \times n$  submatrix of the input image into one specific element value. There are two common pooling methods: maximum pooling and mean pooling. Maximum pooling takes the maximum of the corresponding  $n \times n$  area as the pooled element value and mean pooling takes the average value of the corresponding  $n \times n$  area as the pooled element value. The input volume for the pooling layer is  $W \times H \times D$ , the output volume for the pooling layer is  $W_{pooling} \times H_{pooling} \times D_{pooling}$ <sup>[20]</sup>. The pooling layer computation process is shown in Figure 5. Besides, the pooling layer requires two hyper parameters: the spatial extent  $f$  and the stride size  $s$ . The major formulations are listed as Equations (13)-(15):

$$W_{pooling} = (W - f) / s + 1 \quad (13)$$

$$H_{pooling} = (H - f) / s + 1 \quad (14)$$

$$D_{pooling} = D \quad (15)$$

A flowchart of the CNN algorithm is shown in Figure 6.

CNN simulates the behavior of human brain processing of signals and combines feature extraction in image processing with BP neural networks. CNN has fewer parameters than deep networks because of its local perception mechanism and parameter sharing mechanism which can reduce parameters. CNN can handle high-dimensional arrays, especially for image classification. Many data modalities are in the form of multiple arrays. CNN has

unique advantages in natural language processing<sup>[21]</sup>, face recognition<sup>[22]</sup>, and image processing because of its special structures shared by local weights. Its layout is closer to the actual biological neural network. Nevertheless, CNN needs a lot of training samples.

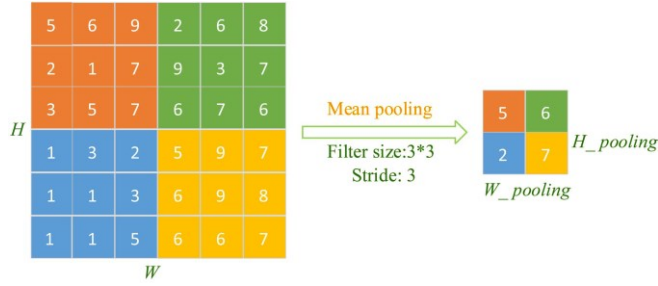


Figure 5 Pooling layer computation process

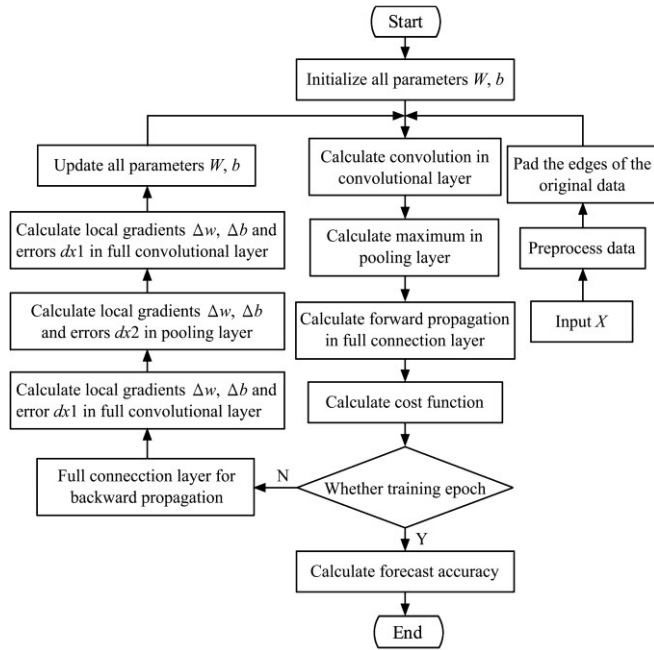


Figure 6 Flowchart of CNN programming

### 2.3 Recurrent Neural Networks (RNN)

RNN can mine temporal information and semantic information and has achieved breakthroughs in time series analysis, speech recognition and language modeling. RNN is a variant of ANN, which means that the current input of the network is related to the output of the previous moment. The specific manifestation is that the network will remember the previous information and apply it to the current network output calculation; in other words, RNN can be treated as BP neural network of which the output will be used as the input of the next network<sup>[19]</sup>.

As Figure 7 shows,  $x^t$  represents the input of training data at the time  $t$ ,  $h^t$  represents the hidden state at the time  $t$ , which is determined by the current input  $x^t$  and the previous hidden layer state  $h^{t-1}$ .  $o^t$  denotes the output of hidden layer at the time  $t$ .  $l$  is the error at the current time  $t$ , which is determined by the  $o^t$  and the true output of the training data  $y^t$ .  $u$ ,  $w$ ,  $v$  are the weights of recurrent neural network and are shared across the recurrent neural networks.  $b_1$  and  $b_2$  are the thresholds of recurrent neural network and are shared across the recurrent neural networks<sup>[23]</sup>.  $f$  is the activation function of the hidden layer, whose network structure propagation process is shown in Figure 7. The specific formulations are listed as Equations (16)-(18).

Hidden state value at time  $t$ :

$$h^t = f(u \times x^t + w \times h^{t-1} + b_1) \quad (16)$$

The predicted output of recurrent neural network at time  $t$ :

$$o^t = f(v \times h^t + b_2) \quad (17)$$

Recurrent neural network error at time  $t$ :

$$l = o^t - y^t \quad (18)$$

A flowchart for RNN programming is shown in Figure 8.

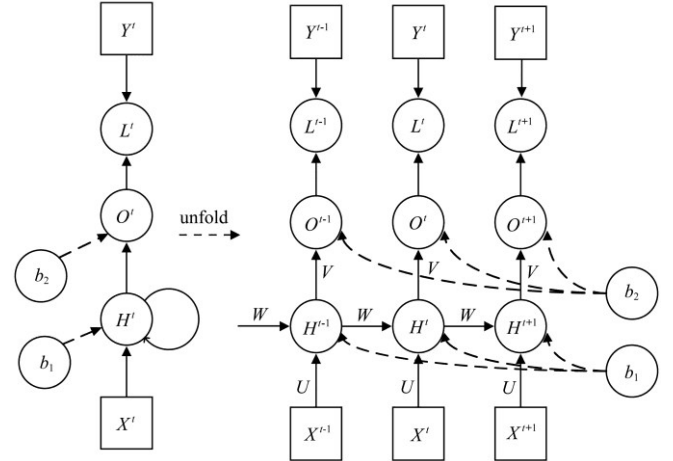


Figure 7 Recurrent neural network propagation process

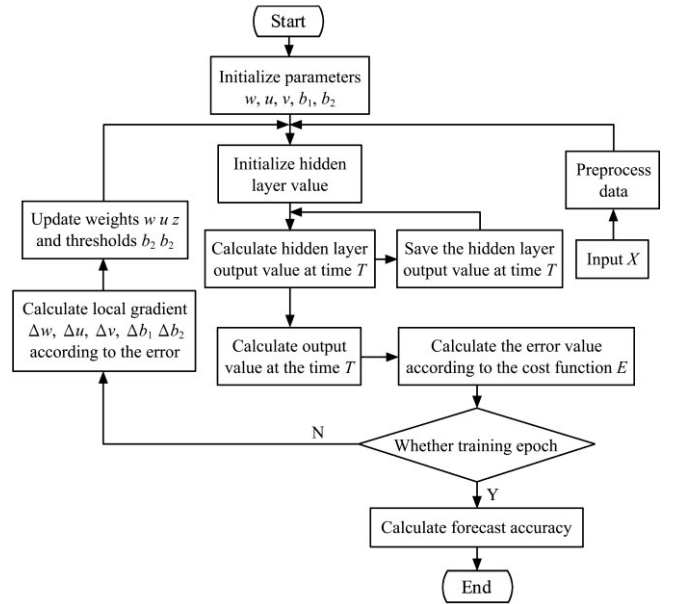


Figure 8 Flowchart of RNN programming

Although RNN solves the problems of time series theoretically, it is difficult to solve the problems of long time series due to the length of information varies in practical applications, which can cause gradients to disappear or to explode. Long Short-Term Memory (LSTM) network<sup>[24]</sup> is an improvement of the recurrent neural network, which is mainly designed to solve time series problems with long intervals and long delays. LSTM depends on the structure of some “doors” to selectively affect the state of the moment in the recurrent neural network. LSTM network has been used in speech recognition<sup>[25]</sup>, machine translation<sup>[26]</sup> and other fields.

### 2.4 Generative Adversarial Networks (GAN)

The ingenuity of GAN lies in its design. Technically, it is a combination of existing algorithms of BP. GAN is a way that a set of noise is used to learn the distribution of the real data and to generate new data. The structure of this network consists of two models, a generation model which is to capture the distribution of



the real data, and a discrimination model which is similar to a binary classifier<sup>[27]</sup>.

Assuming that the generation model is a deep neural network which generates a new vector called fake data  $G(z)$  and the discrimination model is a fully connected network which obtains a probability value  $D(z)$  reflecting real data set.  $x$  is a real data set<sup>[28]</sup>,  $y$  is the output of the distribution model,  $E_d$  is the loss function of discrimination model,  $E_g$  is the loss function of generation model,  $E$  is the loss function of the entire network, whose network structure is shown in Figure 9 and the major formulations for the loss function are listed as Equations (19)-(24).

Discrimination model loss function  $E_d$ :

$$E_d = -((1-y)\log(1-D(G(z))) + y\log D(x)) \quad (19)$$

Generation model loss function  $E_g$ :

$$E_g = (1-y)\log(1-D(G(z)))(2 \times D(G(z)) - 1) \quad (20)$$

Total network loss function  $E$ :

$$E = (1-y)\log(1-D(G(z)))(2 \times D(G(z)) - 1) \quad (21)$$

After the loss functions are obtained, the network is optimized by us  $V(D, G)$ , the optimization function:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \quad (22)$$

The expressions above can be divided into two functions: discrimination model optimization function and generation model optimization function.

Discrimination model optimization function:

$$\max_D V(D, G) = E_{x \sim p_{data}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \quad (23)$$

Generation model optimization function:

$$\min_G V(D, G) = E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \quad (24)$$

According to the above, it can be observed that the optimization of the discrimination model and the optimization of the generation model are independent of each other.

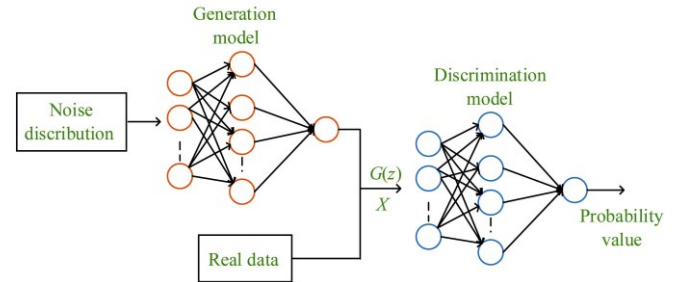


Figure 9 Generative adversarial networks structure

The power of generative adversarial networks (GAN) lies in the ability to automatically learn the distribution of real sample data. Deep convolutional generative adversarial networks (DCGAN) resulting from the rise of convolutional neural network is widely used in image processing such as image restoration from split image, dynamic scene generation<sup>[29]</sup>, image generation<sup>[30]</sup>, and resolution enhancement<sup>[31]</sup>. In addition, DCGAN plays an important role in face detection and recognition<sup>[32]</sup>. However, it is difficult to train a GAN due to entail synchronization of generation model and discrimination model, and there is room for further development of GAN applications.

A flowchart for GAN programming is shown in Figure 10.

A summary of BP, CNN, RNN, and GAN is shown in Table 1.

Codes of BP, CNN, RNN, and GAN in Python are provided in the supplemental files.

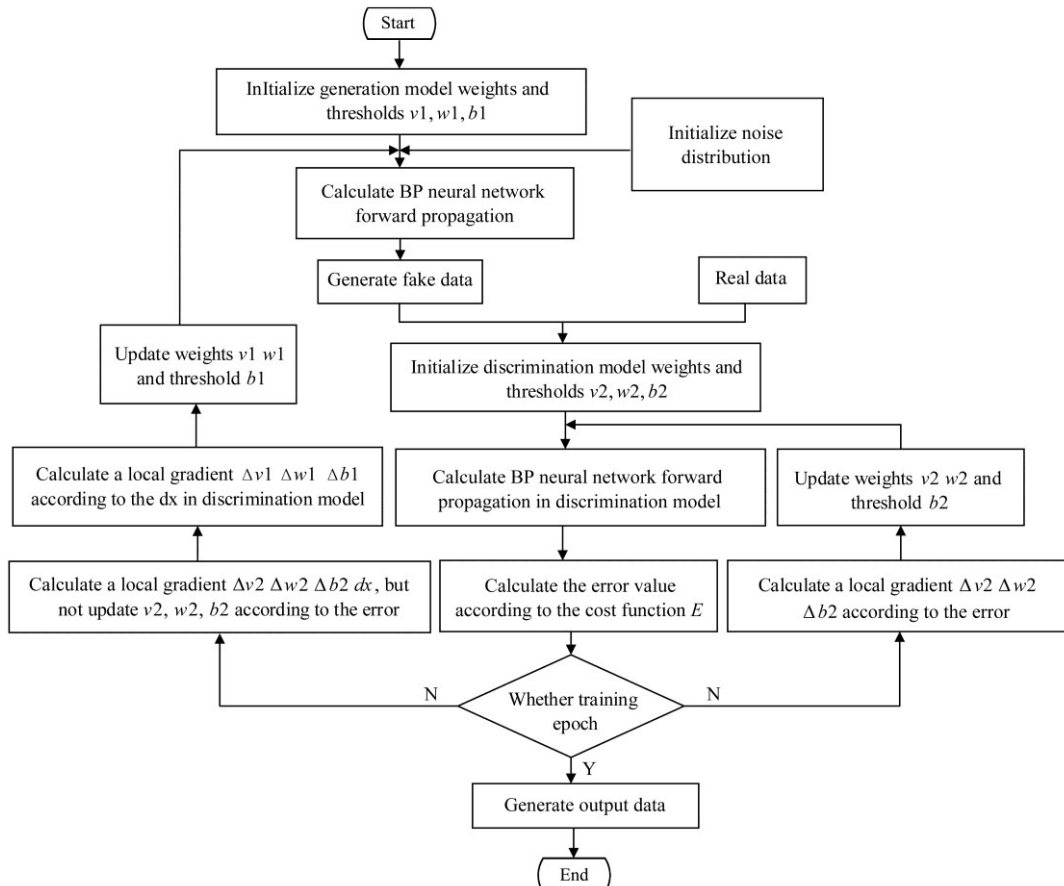


Figure 10 Flowchart of GAN programming

**Table 1 Summary of BP, CNN, RNN, and GAN**

Type	Example references	Variants	Network structure	Applications
BP	Rumelhart <sup>[33]</sup>	RBF GRNN	Input layer Output layer Hidden layer	Data fitting Pattern recognition Classification
CNN	LeCun <sup>[34]</sup> Krizhevsky <sup>[35]</sup>	LeNet, AlexNet VggNet	Input layer Convolution layer Pooling layer Full connected layer	Image processing Speech signal Natural Language Processing
RNN	Mikolov <sup>[36]</sup> Sundermeyer <sup>[37]</sup>	LSTM	Input layer Hidden layer Output layer	Time series analysis Emotion analysis Natural Language Processing
GAN	Goodfellow <sup>[28]</sup>	DCGAN	Discrimination model Generation model	Image generation Video generation

### 3 Deep learning frameworks

TensorFlow and Caffe are two commonly-used DL frameworks, which allow users to use DL without significant programming. The most popular TensorFlow and Caffe frameworks are introduced below briefly.

#### 3.1 TensorFlow

TensorFlow is an open source computing framework of Google that supports deep learning algorithms, including CNN, RNN, GAN and other variants, which can be used on Linux, Windows, and Mac platforms. TensorFlow has some advantages including high flexibility, true portability, multi-language support, rich algorithm library, and excellent documentation. TensorFlow provides a very rich set of deep learning application programming interfaces (API) including basic vector matrix calculations, optimization algorithms, convolutional neural networks, recurrent neural networks, and visual aids. TensorFlow uses dataflow graphs to represent computation, shared state, and the operations that mutate state. It maps the nodes of a dataflow graph across multiple computational devices, such as multi-core CPUs, general-purpose GPUs, and custom-designed ASICs, which are known as Tensor Processing Units (TPUs)<sup>[38]</sup>.

A deep learning model, typically a multi-layer neural network, is composed of several computational layers that process data in a hierarchical fashion. Each layer takes an input and produces an output, often computed as a non-linear function or a weighted linear function of a weighted linear combination of the input values. It is particularly popular that convolutional layers apply a local function or filter to all subsets of the layer's input, such as portions of an image<sup>[39]</sup>.

#### 3.2 Caffe

Convolution Architecture For Feature Extraction (Caffe) is the first deep learning framework that has been widely used in industry as an open source. The Caffe model and the corresponding optimizing methods are given as texts instead of codes. Caffe gives the definition of the model, optimal settings, and pre-training weights. Caffe handles massive data with high speed. Besides, Caffe can be modular which easily extended to new tasks. Users can define their own models using the types of neuron layers provided by Caffe.

Caffe provides multimedia scientists and practitioners with a clean and modifiable framework for state-of-the-art deep learning algorithms and a collection of reference models. The framework is a BSD-licensed C++ library with Python and MATLAB bindings for training and deploying general-purpose convolutional neural networks and other deep learning models efficiently on varied architectures<sup>[40]</sup>.

A comparison between TensorFlow and Caffe can be found in Table 2.

**Table 2 Comparison between Tensorflow and Caffe**

Traits	TensorFlow	Caffe
Support language	C++, Python	C++, Python, MATLAB
Support system	Linux, Mac OS X, Android, IOS	Linux, Mac OS X, Windows
Data input format	Data, ImageData	Feeding, Data from a file, Preloaded data
Support model	CNN, RNN, GAN...	CNN
Loss function	Cross-entropy loss function Mean squared error	Contrative loss, Hingle loss, Softmax loss

### 4 Recent applications of DL in smart agriculture

Recent applications of CNN, RNN, and GAN in smart agriculture are summarized in this section.

#### 4.1 CNN applications in smart agriculture

CNN has strong capability in image processing, which makes it widely used in agriculture research. Generally speaking, most applications of DL in agriculture can be categorized as plant or crop classification, which is vital for pest control, robotic harvesting, yield prediction, disaster monitoring etc.

Plant disease detection is time-consuming when it is done manually. Fortunately, with the development of artificial intelligence, plant disease detection can be accomplished through image processing. Plant disease recognition models are mostly based on leaf image classification and pattern recognition<sup>[41]</sup>. A novel DL framework developed by the Berkley Vision and Learning Centre was used to build a plant disease detection model. The model is able to recognize 13 different types of plant diseases out of healthy leaves, with the ability to distinguish plant leaves from their surroundings<sup>[42]</sup>. In another research of using DL in detection of plant diseases, the overall accuracy may reach 95.8% after 100 training iterations and may be improved to 96.3% after further fine-tuning. The results are actually better than manual detection<sup>[43]</sup>. All these proved that DL has very impressive performance in detecting plant diseases.

As many countries across the world have been developing initiatives to build national agriculture monitoring network systems, plant classification and weed identification are particularly important because of the implications for automating agriculture. Since image recognition can be applied to detect plenty of features of plants, CNN has been extensively used to detect weeds or classify plants<sup>[44-48]</sup>. In 2017, a new approach that combined CNN and K-means feature learning was proposed for weed identification and control. Manual design features in weed identification may cause unstable identification results and weak generalization ability in feature extraction. Thus, the application of DL and K-means pre-training resulted in an accuracy of identification of 92.89%<sup>[44]</sup>.

One of the pre-trained CNN architecture that is widely used for plant classification is AlexNet. Experimental results based on AlexNet from the Istanbul Technical University in 2017 suggest that the CNN architecture outperforms machine learning algorithms that are based on hand-crafted features for the discrimination of phenological stages<sup>[47]</sup>. In another study, self-organizing Kohonen maps (SOMs) were used for optical images segmentation and subsequent restoration of missing data in a time-series of satellite imagery. Supervised classification with CNN was performed. This method added a post-processing step that included several filtering algorithms based on the available

information and geospatial analysis. An accuracy of 85% was achieved for classification of major crops (wheat, maize, sunflower, soybean, and sugar beet)<sup>[42]</sup>. However, there are challenges that have slowed down the application of CNN for plant classification. For example, each pixel of the space borne SAR (synthetic aperture radar) imagery is characterized by backscatter phase and intensity in multiple polarizations. Both data sources have multitemporal nature and different spatial resolutions<sup>[47]</sup>. Information fusion is thus important in the future to make DL more applicable in this area.

Fruit counting is important for yield prediction and robotic harvesting. The traditional manual counting or mobile camera counting cannot provide satisfactory results and are time-consuming. Because of changes in occlusion and illumination, pre-processing such images is challenging. Regular DL methods have difficulty in solving these problems. A blob detection method has proven to be useful<sup>[48]</sup>, which was proposed to accompany a fully convolutional network (FCN). The first step of the method is to collect human-generated labels from a set of fruit images. Then, a blob detection FCN was trained to perform image segmentation. After that, a count convolutional network was trained to take the segmented image and output an intermediate estimate of the fruit count. The final step of the work was to train a linear regression equation to map intermediate fruit count estimates to final counts using human-generated labels as the ground truth. This approach using DL with blob detection improved not only the accuracy but also the efficiency of counting.

Land classification usually involves classification of large areas of land and is important to such purposes as land use and land cover (LULC), disaster risk assessment, agriculture, and food security<sup>[49]</sup>. For classification and area estimation in the remote sensing and other agriculture imagery, DL techniques has been applied. The general idea of this DL approach is to fuse or integrate data acquired by multiple heterogeneous sources by using machine learning techniques and emerging big data and geo-information technologies to provide data processing and visualization capabilities. The methodology can be divided into four steps: noise filtration and data clustering, classifying land cover, map post-processing with filtering, and geospatial analysis. In addition to satellites, unmanned aerial vehicles (UAV) are now widely used to investigate various resources<sup>[50]</sup> based on deep CNN (DCNN) and transfer learning (DTCLE). A feature extraction method based on DCNN was used to extract cultivated land information by introducing a transfer learning mechanism. Finally, cultivated land information extraction results were completed by the DTCLE and e-Cognition for cultivated land information extraction (ECLE). The overall precision of DTCLE and ECLE were both around 90%, but in terms of integrity and continuity, DTCLE outperformed ECLE. This instance is an extension of DL that can be applied in agriculture. The utilization of UAVs permits acquisition of high-quality images.

Item detection is a fast-growing domain in DL. In agriculture fields, obstacle detection is also important for farmers, especially when highly autonomous machines have been increasingly used. In order to operate these machines safely without supervision, they must perform automatic real-time risk detection with high reliability<sup>[51]</sup>. An image classification method with the AlexNet and DCNN were utilized to enhance performances. The accuracy reached 99.9% in row crops and 90.8% in grass mowing, which is much better than traditional methods<sup>[52]</sup>.

Information from satellite is very precious and important for making sustainable land use planning for minimizing CO<sub>2</sub> emission, maximizing economic returns, and minimizing land degradation. The challenge with using information is to interpret the images collected. Translating satellite images by using convolutional neural networks (CNN) and genetic algorithms has become a useful strategy for decision making, especially for precision agriculture and agroindustry. The data can be used to classify plant types in a land area using CNN. Land types and other data can be added to the grid form. A grid form model was evaluated to assess objectives and a genetic algorithm was used produce an optimal solution<sup>[53]</sup>. Flower grading was also done by using a similar concept<sup>[54]</sup>.

CNN can also be used in weather forecasting<sup>[55]</sup>, which is key to agriculture. Crop yield prediction before harvest is crucial to farmers, consumers, and the government in their efforts to design strategies for selling, purchasing, market intervention, and food shortage relief. CNN has also been used to predict yield in agriculture<sup>[56]</sup>. It can be used for studying not only crops but also animals. For example, CNN has been extensively used to classify animal behaviors<sup>[57,58]</sup>.

## 4.2 RNN applications in smart agriculture

RNN is very useful to process time series data and has been used in many agricultural areas, such as land cover classification, phenotype recognition, crop yield estimation, leaf area index estimation, weather prediction, soil moisture estimation, animal research, and event date estimation.

Land cover classification (LCC) is considered as a vital and challenging task in agriculture, and the key point is to recognize what class a typical piece of land is in. In the past, a lot of applications are based on mono-temporal observations and ignoring time-series effects in some problems. For instance, vegetation changes its spatial appearance periodically, which can confuse the mono-temporal approaches. Meanwhile, mono-temporal approaches might be influenced by some biases, like weather. Therefore, deep sequence models have been applied, and a widely-used variant RNN model is the LSTM. In an experiment led by Rußwurm et al.<sup>[59]</sup>, the LSTM network outperformed all mono-temporal models (CNN and SVM) as well as the standard RNN. Ienco et al.<sup>[60]</sup> combined the LSTM units with other machine learning models (SVM, RF) and compared them with mono-temporal approaches. They concluded that SVM with LSTM units worked the best. In addition, RNN models are used not only in recognition of land cover classes but also detection of the changes of land. Lyu et al.<sup>[61]</sup> established a network called REFEREE (learning a transferable change rule from RNN for change detection), which consisted of two groups of picture input (same region, different timestamp), several LSTM units, and a graphical output showing regions experiencing changes. By applying RNN models, REFEREE could learn a stable and rational change rule, in both binary change cases and multi-class change cases.

Plant phenotyping has become a hot topic, yet a challenging one, because of the increasing need of precision agriculture. Briefly, plant phenotyping means to recognize the kind of a plant through its appearance or traits. Most machine learning approaches have relied on individual static observations, which cause incorrect recognition of similar plants in typical periods. A new deep learning structure for plant phenotype recognition was created by Namin et al.<sup>[62]</sup>, in which CNN was combined with LSTM units. According to their structure, CNN was to extract



features and its output was fed into an LSTM unit in order to build a sequence model. Experiment results indicated that the sequence model improved accuracy significantly, compared with a previous pure CNN model, from 76.8% to 93%.

Beyond CNN, RNN has also been used for crop yield estimation, which uses time series data to reduce biasing. Minh et al.<sup>[63]</sup> trained two RNN-based classifiers, an LSTM network and a Gated Recurrent Unit (GRU) network, and applied them in the task of mapping winter vegetation quality coverage, along with mono-temporal models. Their results indicated that the GRU model outperformed all other models, with an accuracy of 99.05% for a 5-fold cross validation dataset.

The leaf area index (LAI) is a key attribute of many agricultural models. Accurate LAI and its dynamics are widely used for estimations of environment, vegetation status and carbon cycle etc. Traditional LAI estimation methods fall into two categories, empirical methods and physical methods. The LAI results may suffer from spatial or temporal discontinuities, which limit their applications in climate simulation and weaken their robustness. To solve the problem, an RNN-based model named NARX (Nonlinear Autoregressive model process with eXogenous input) was applied. The inputs of the NARX model were previous prediction values and the current and previous values of an exogenous input signal. The inputs were fed to an RNN and the prediction  $Y$  were a part of inputs in the next time step. This model took not only independent inputs into consideration but also the output of the model in the past, making it more powerful and. Chai et al.<sup>[64]</sup> established a model based on the NARX model called NARXNN to estimate time-series LAI. They trained the model on several datasets and made indirect and direct validation, both of which suggested that NARXNN is a promising tool for time-series LAI estimation. In application, Chen et al.<sup>[65]</sup> applied a NARX model to predict the LAI of rubber.

RNN is useful for time series and thus has been used in weather prediction. In Biswas et al.<sup>[66]</sup>, three models were compared for weather prediction: an RNN-based model named NARXnet, a case-based reasoning model (CBR), and a segmented CBR model. The input of NARXnet was the weather attributes of  $a$  days before the target day and  $b$  predictions and targets. This structure means that NARXnet could not only learn from historical data but also from the previous predictions. The NARXnet got an accuracy of 93.95%, outperforming the other two models significantly. In addition, Zaytar et al.<sup>[67]</sup> established a new LSTM model to predict 24 and 72 hours weather attributes of a city: temperature, humidity, and wind speed. Compared with normal RNNs, their structure consisted of an input layer, two stacked LSTM layers connected with a dense layer, an activation layer and a repeat layer. They used hourly attributes data of 15 years to train the model and got competitive results compared with other traditional methods. These suggest that deep RNN-based methods are a competitive alternative for weather forecasting.

Soil moisture (SM) is a vital hydrological attribute for precision agriculture, meteorology, and climate change. However, SM in farmlands is a function of many factors and can vary extremely with time and space, causing difficulty in precise estimation. Neural networks are applied to this task naturally because they can estimate complex functions and time-series input. Lu et al.<sup>[68]</sup> tested a simplified NARX model whose input was only the current features and the prediction it had given in the last time step. They compared the predictions with the soil moisture data given by Japan Aerospace Exploration Agency (JAXA), the Land

Surface Parameter Model (LPRM), and the Global Land Data Assimilation System (GLDAS). The direct validation indicated that their model remained stable and competitive in both frozen and unfrozen seasons. In addition, Tzeng et al.<sup>[69]</sup> used a typical NARX model to estimate the dynamics of soil moisture. They used a NARX model (DLNN as they called) to predict soil moisture on an hourly basis and compared the predictions with ground measurements. The experiments showed that the model was a promising tool for the task.

RNN has also been used in studying animals in both the macroscopical and the microcosmic scales. With the development of deep learning, RNN-based models have proven to be very competitive. In South Africa, the movement of elephant herds hurt the endangered species of vegetation. Palangpour et al.<sup>[70]</sup> trained an RNN model combined with the particle swarm optimization (PSO) algorithm to predict the locations of elephant herds. The results indicated that the RNN model could provide predictions with a low level of errors. In another research, Demmers et al.<sup>[71]</sup> applied a first-order RNN for estimation of pig growth. The result showed that the first-order RNN worked well to predict pig growth.

RNN can also be used for event date estimation<sup>[72]</sup> and many other purposes. As we have seen, sequence models have been increasingly applied in agriculture, even though the main deep learning methods used in agricultural tasks are still the CNN-based models. If we look through the applications of sequence models, it is not hard to conclude that where sequence models can play an important role are usually tasks that involve a long time period or require stability in the long term. The main mechanism of improvements resulting from sequence models is to overcome bias that occurs at typical time points or locations.

### 4.3 GAN applications in smart agriculture

GAN is a new kind of neural network but has been considered a very useful method in many fields, especially in image processing. GAN has often been used to enrich datasets. It has not been applied to agriculture widely.

Ledig et al.<sup>[73]</sup> used GAN to solve feature loss caused by down sampling. If a picture is compressed, some features can be lost or become inaccurate and there is a need to recover photo-realistic textures from it. To do that, they introduced a perceptual loss function made up of an adversarial loss and a content loss. Compared with widely-used pixel-wise MSE loss, the content loss function they used was motivated by perceptual similarity. After being trained with 350 K images, their model could recover highly compressed images and outperformed some state-of-the-art models at that time. This work is so fundamental that it can be used in almost every project containing image processing, particularly in agriculture fields where many applications are based on remote sensing images.

In another study, Barth et al.<sup>[74]</sup> attempted to overcome the gap between large quantity of data deep learning models require and the shortage of manually annotated datasets. They used a GAN-based model, called unsupervised cycle generative adversarial network, to optimize the realism of synthetic agricultural images. In their work, 10500 synthetic, 50 empirically annotated, and 225 unlabeled empirical images were used to train their model and they hypothesized that the similarity between synthetic images and empirical images can be improved qualitatively to improve the translation of features. The results showed that the synthetic images were translated well on local features such as color, illumination scattering and texture while global feature translation

was not so good.

#### 4.4 A Meta-Analysis of DL applications in smart agriculture

Many pure DL research articles are published in the ACM proceedings. In contrast, applications of DL in agriculture are more often published in research journals. Among the published papers, the well-known databases of Science Citation Index (SCI) and Social Science Citation Index (SSCI) collect the most representative and qualified papers. Thus, a meta-data analysis was performed using the bibliographic datasets, which contain the metadata information, like journals, authors, publication year, citations, and institutions. The metadata analysis, also called as bibliometric analysis, belongs to the field of scientometrics. The analysis has the advantages of data-driven characteristics, and being objective, complete and repeatable<sup>[75]</sup>.

The analysis involved the following steps: collecting datasets from the bibliographic databases with explicit search strategies; analysis of yearly output, topics, and related disciplines; and displaying of an overall picture of DL-agriculture research.

The search strategy is as follows:

(TS=(deep learning) AND SU=agriculture) AND language: (English) AND type: (Article)

Time span: 1985-2018 Index: SCI-EXPANDED, SSCI where, TS stands for “topic”, SU stands for “research area”.

Forty-seven records were obtained by applying the search term in the core collection of Web of Science database. “SU=agriculture” was used to confine the research areas in the field of “agriculture”. The analysis results were displayed as follows.

##### Publication Yearly Output

The publication counts, to some degree, reveal the research intensity in a field. The yearly output of publications is shown in Figure 11. Also, the Total Local Citation Scores (TLCS) and Total Global Citation Scores (TGCS) are shown in the figure. The primary vertical axis is set for the “Recs” and “TLCS”, and the secondary vertical axis is set for “TGCS”. TLCS stands for the field recognition and TGCS stands for the recognition without field constraint. These values are computed with the bibliometric software, HistCite<sup>[76]</sup>.

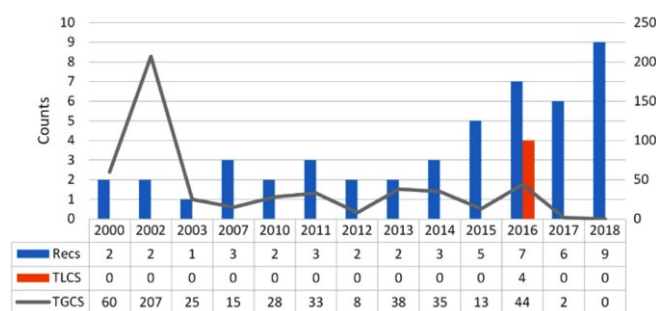


Figure 11 Publication yearly output (Recs stands for article records, TLCS stands for the Total Local Citation Scores, TGCS stands for the Total Global Citation Scores)

From the figure, it can be observed that the overall publication trend is upward from the Recs counts. However, from the number changes of the TLCS, one can tell that most of the TLCS are zeros, meaning that DL-agriculture application articles are not citing each other. This suggests that applications of DL in agriculture fields are highly divergent and scattered in very different research purposes. It also implies that more challenges and chances remain in the research direction. A notable point in the TGCS is that in 2002, the paper “*Changing systems for supporting farmers’*

*decisions: problems, paradigms, and prospects*”<sup>[77]</sup> drew wide attentions from all disciplines. This paper may be regarded as a pioneering article advocating machine learning for agriculture applications. As there are citation windows, the papers need time to accrue citations. The papers in 2016 are impressive for having acquired 44 global citations (TGCS=44). DL-agriculture papers in 2016 discussed topics including plant identifications<sup>[78]</sup>, and pest detections<sup>[79]</sup>, etc.

To have a close look at the research topics, the following co-word analysis was conducted, which was based on the keyword co-occurrence relations. The illustration was generated by the bibliometric software, CiteSpace<sup>[80]</sup>, as shown in Figure 12.

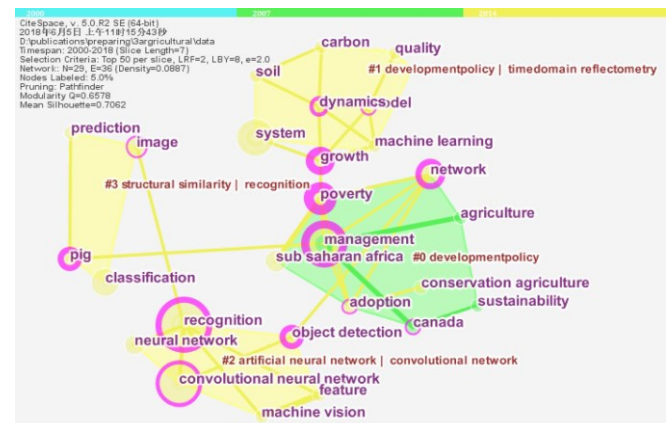


Figure 12 Co-word visualization for the research articles

In Figure 12, the node size is proportional to the keyword frequencies. The different clusters are generated and tagged with polygons. The topics for each cluster were generated with Latent Dirichlet Allocation (LDA)<sup>[81]</sup>, as shown by the red-colored words. The first and the second cluster (cluster #0 and cluster #1) focus on the topic of development policy. The third and fourth cluster (cluster #2 and cluster #3) focus on recognition, using CNN and structure similarity theory, respectively.

The revealed topics provide a landscape of the domain expert knowledge at a micro level. How the subject of agriculture in a macro-level is affected by the research fever is still not clear. Journals can reveal the macro status of research trends to some level. Relevant journals were investigated and the results are listed in Table 3.

Table 3 Top ten high-impact journals that publish DL-agriculture papers

Index	Journal	Recs	TLCS	TGCS
1	Agricultural Systems	3	0	253
2	Biology and Fertility of Soils	1	0	36
3	Vadose Zone Journal	1	0	25
4	Agricultural Economics	1	0	24
5	Computers and Electronics in Agriculture	10	3	23
6	Agricultural Sciences in China	1	0	16
7	Biosystems Engineering	2	1	15
8	Canadian Journal of Soil Science	1	0	15
9	Agriculture and Human Values	1	0	13
10	Crop & Pasture Science	1	0	13

The journals are listed by the descending order of TGCS index, higher ranking meaning higher all-discipline impacts. The number provides a direct and simple indicator for the journal impacts on the DL-agriculture research. However, the distribution on how the research combines different disciplines is still not clear.

The following overlay visualization based on dual-mapping was thus performed, as shown in Figure 13.

The dual-mapping consists of two mappings based on datasets of large-scale journals from all disciplines. The whole science mappings consist of the citing journal cluster on the left side and the cited journal cluster on the right. The citing cluster contain 10 330 all-discipline journals and the cited cluster contain 10 253 all-discipline journals<sup>[82]</sup>. Each point in the figure stands for a

journal and the large-scale network clustering method generated these clusters, called “Blondel clusters”<sup>[83]</sup>. Thus, the citing trajectories over different disciplines can be visually depicted on this base map. One can tell that plant science, chemistry, and economics are also the citation sources besides the computer science discipline. This is in line with intuitive understanding. More research energy may have the potential to link the above mentioned disciplines in the near future.

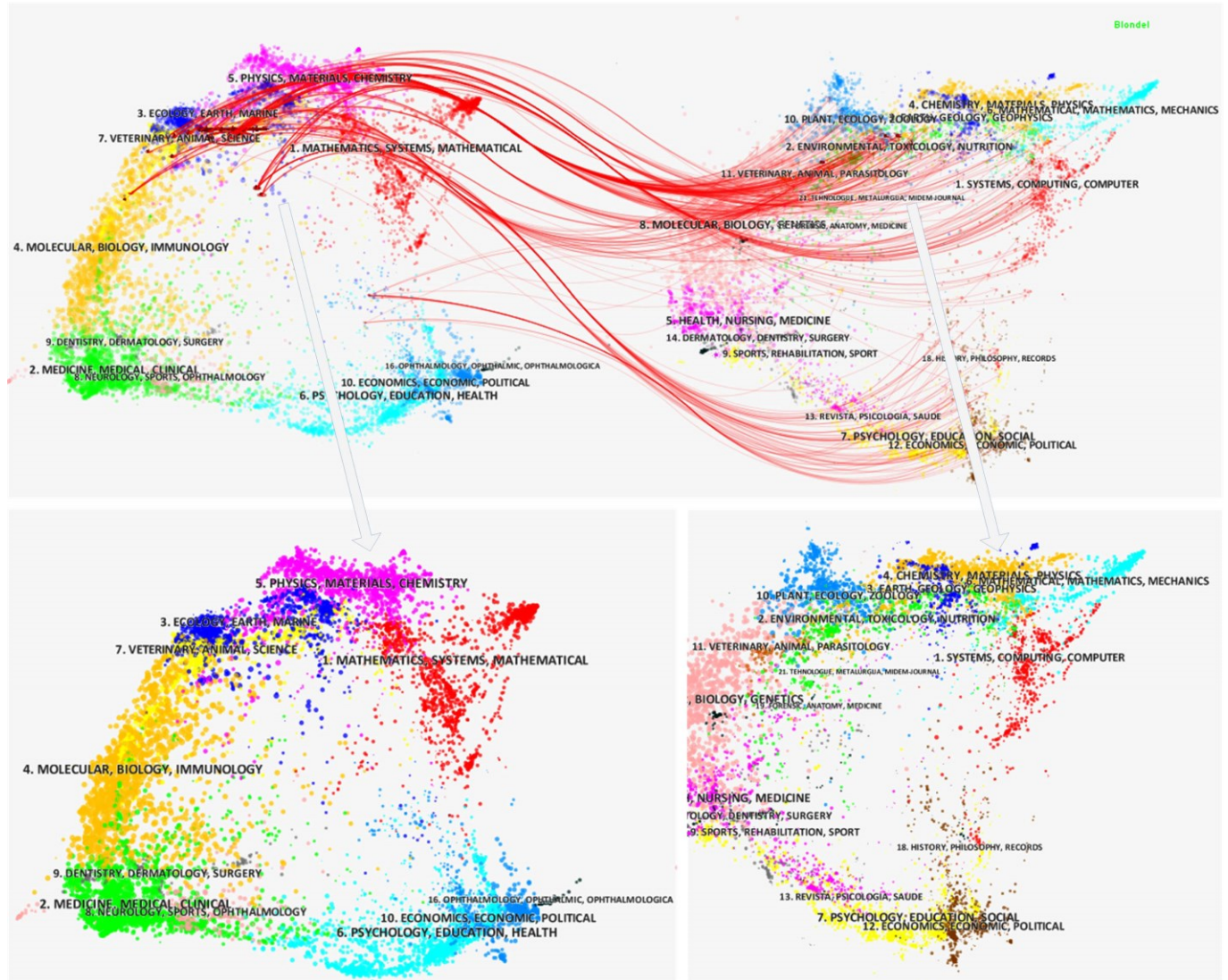


Figure 13 Overlay visualization of DL-agriculture research on the whole science dual-mapping

## 5 Discussion

Most of the recent advances in agriculture fields made by researchers are closely connected to production and every other part of agriculture for the purposes of improving productivity of crops, reducing and preparing for the plant diseases, boosting mechanized and automated modern agriculture and agro-industry. DL is usually used for image recognition or data classification, which can be summarized to include four steps: data collection and data preprocessing, neural network training, model testing, and final result analysis.

For the first step, the combination of DL with other advanced technologies, such as unmanned aerial vehicles (UAVs), radar, and Internet of Things, can provide high quality datasets of images and other forms. These data greatly enhance applications of DL in agriculture and improve the accuracy of resulting tools<sup>[84]</sup>. For the

second step, new training algorithms and methods can enhance the accuracy, especially for those applications that require high precision. K-means feature learning, blob detection, FCNN, AlexNet etc. play an increasingly important role. Model testing with new data is always an important third step. In the final step, the results are interpreted and analyzed.

Either from historical research statistics or the recent research results, it can be shown that DL can be well applied in agriculture to solve various problems that have concerned farmers and scientists for a long time. With the help of novel techniques and new theories, these new approaches outperform the previous methods in many ways. Take image recognition for instance, previously, scholars may collect images and use DL to do classification by analyzing pixels in RGB images<sup>[85]</sup>. The result of this method can be satisfactory under certain optimal conditions. Because of illumination changes, displacement of targets by winds,



camera jitter, zoom changes, unexpected changes in camera parameters; inconsistent recognition and classification occur<sup>[41]</sup>. Therefore, color analysis can be unreliable since it usually relies on the color distribution in an image and there are many conditions where the temporal consistency of this feature is violated<sup>[36]</sup>. For plant color application, methods based on vein and pattern recognition have been proposed to reduce the uncertainty for analysis of color in an image<sup>[86]</sup>, because vein is also a robust and significant feature of the plants and usually vein morphological patterns are a good leaf fingerprint. The correlation between vein characteristics and some properties of the leaf seldom change when and after pre-processing the images.

Because there are many applications in agriculture not equipped with recent new techniques, there is still a lot of room for expanding DL in agriculture research. Although some of the results gained accuracy around or higher than 95%, robustness and reliability are still challenges. The promise of the application of DL in agriculture can be foreseen. Moreover, it is very possible that future development of DL in agriculture will be based on not only a single theory or method but a combination of multiple methods. The metadata analysis shows that applications of DL in agriculture are upward from the Recs counts. However, they are highly divergent and scattered in very different research purposes. It implies that more challenges and opportunities remain in the area and more efforts are needed.

## 6 Conclusions

In this summary, the concepts, tools, limitations, algorithms of DL are summarized. Applications of DL in agriculture are reviewed. It can be observed that DL has been widely used in different areas of agriculture, such as plant disease detection, plant classification and weed identification, fruit counting, land classification, obstacle detection, image translation, weather forecasting, yield prediction, and animal behavior classification. While DL research is gaining momentum in general, research of DL for agriculture is very divergent according to the metadata analysis. There are many opportunities for DL applications in smart agriculture, such as:

(1) Agriculture information processing. Monitoring the status of plants and animals is vital to agriculture production. Some status variables of plants and animals cannot be measured directly. Under this case, DP can be used to determine unmeasured information from measured one because different variables of plant status may have some dependent relationships. In agriculture production, plants or animals interact with environmental factors. It is difficult to build a pure mechanism-based-model structure to describe the relationship between plants (or animals) and environmental factors. As a data-driven-model structure, DP can be directly used to build the relationship of plant (or animal) factors, environmental factors and plant (or animal) growth status. This will facilitate agriculture information processing.

(2) Agriculture production system optimal control. Control strategies in agriculture production system often rely on farmer experience or experts knowledge, which do not consider plant (animal) physiological status or real time demand. This unavoidably makes the strategies not optimal. An advantage of DP is to model complex systems without heavily relying on the knowledge of mechanisms. By using DP to model agriculture production systems, optimal control strategy development thus becomes possible, which makes use real time measurement of plant (animal) physiological status and historical data.

(3) Smart agriculture machinery equipments. Agriculture production involves numerous kinds of tasks. These tasks are often labor consuming and the working environment is very challenging. Using DP to mimic human behavior and driven agriculture machinery equipments has prospective future in many areas of agriculture, such as seeding, management, harvesting, and post-harvest processing. For example, a robot that can be used to harvest apples is very useful. It must be intelligent enough to position apples and pick apples with a high efficiency. Under natural condition, background light, tree branches and leaves have strong interference to computer vision signal. The routes of robot arms reach targets should also be optimized. For all these issues, DP techniques may be very useful.

(4) Agricultural economic system management. Agriculture yield itself is not enough for agriculture. There are many more factors should be considered such as the prices and the quality of agriculture products. It is very meaningful to predict agriculture product prices. However, prices are related to many variables. Under this case, DP can be used to model price changes with different variables. There are complex relationships between agriculture product quality and nutrition, human health, and economy. DP can be used to model the complex relationship and enhance agricultural economic system management.

## Acknowledgements

This project is partially supported by National Natural Science Foundation of China (No. 31771680), Fundamental Research Funds for the Central Universities of China (No: JUSRP51730A), the Modern Agriculture Funds of Jiangsu Province (No. BE2015310), the Modern Agriculture Funds of Jiangsu Province (Vegetable) (No. SXGC[2017]210), the New Agricultural Engineering of Jiangsu Province (No. SXGC[2016]106), the 111 Project (B1208), and the Research Funds for New Faculty of Jiangnan University.

## [References]

- [1] Schalkoff R J. Artificial Neural Networks. Vol. 1. New York: McGraw-Hill, 1997.
- [2] Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*, 2015; 61: 85–117.
- [3] Kamilaris A, Prenafeta-Boldú F X. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 2018; 147: 70–90.
- [4] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *Cornell University Library*, Available at: <https://arxiv.org/abs/1409.1556>.
- [5] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. *Advances in Neural Information Processing Systems*, 2014; 1-4: 568–576.
- [6] Fan Y, Qian Y, Xie F, Soong F K. TTS synthesis with bidirectional LSTM based recurrent neural networks. *Proc. Interspeech*, 2014; pp.1964–1968.
- [7] Alex G, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 2005; 18(5-6): 602–610.
- [8] Alec R, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. *Cornell University Library*, Available at: <https://arxiv.org/abs/1511.06434>.
- [9] Suárez P L, Sappa A D, Vintimilla B X. Infrared image colorization based on a triplet dcgan architecture. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.
- [10] Jin J Q, Fu K, Zhang C H. Traffic sign recognition with hinge loss trained convolutional neural networks. *IEEE Transactions on Intelligent Transportation Systems*, 2014; 15(5): 1991–2000.
- [11] Haykin S S. *Neural Networks and Learning Machines*. Pearson Schweiz Ag, 2008.

- [12] Alpaydin E. Neural Networks and Deep Learning, 2016; 224p.
- [13] Hecht-Nielsen R C. Neural Networks for Perception. Academic Press, Inc., 1992; pp.65-93. Available at: <https://doi.org/10.1016/B978-0-12-741252-8.50002-9>
- [14] Goh A T C. Back-propagation neural networks for modeling complex systems. Artificial Intelligence in Engineering, 1995; 9(3): 143–151.
- [15] Li J C, Wei H G. BP neural network used in recognition algorithm for star pattern. AOPC 2017: Optical Sensing and Imaging Technology and Applications. Vol. 10462. International Society for Optics and Photonics, 2017.
- [16] Xia X L, Han R Y. Research on classifications of mobile networking device users based on BP neural networks. 2nd International Conference on Mechatronics and Information Technology (ICMIT 2017), 2017; pp.104–109.
- [17] Razavian A S, Azizpour H, Sullivan J, Carlsson S. CNN features off-the-shelf: an astounding baseline for recognition. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2014.
- [18] Vincent P, Larochelle H, Bengio Y, Manzagol P A. Extracting and composing robust features with denoising autoencoders. In Proc. 25th Int. Conf. on Machine Learning, 2008; pp.1096–1103.
- [19] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015; 521(7553): 436.
- [20] Bouvrie J. Notes on convolutional neural networks. Cambridge: MIT, 2006.
- [21] Yin W P, Kann K, Yu M, Schütze H. Comparative study of CNN and RNN for natural language processing. Cornell University Library, 2017. Available at: <https://arxiv.org/abs/1702.01923>.
- [22] Parkhi O M, Vedaldi A, Zisserman A. Deep Face Recognition. BMVC, 2015; 1(3): 1–12.
- [23] Chung J Y, Gulchre C, Cho K, Bengio Y. Gated feedback recurrent neural networks. International Conference on Machine Learning, Lille, France, 2015.
- [24] Sak H, Senior A W. Processing acoustic sequences using long short-term memory (LSTM) neural networks that include recurrent projection layers. U.S. Patent No. 9,620,108. 11 Apr. 2017.
- [25] Han S, Kang J L, Mao H Z, Hu Y M, Li X, Li Y B, et al. ESE: Efficient speech recognition engine with sparse LSTM on FPGA. Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2017.
- [26] Britz D, Goldie A, Luong M T, Le Q. Massive exploration of neural machine translation architectures. ACL, 2017. Available at: <https://arxiv.org/pdf/1703.03906.pdf>.
- [27] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. ICLR, 2016. Available at: <https://arxiv.org/abs/1511.06434>.
- [28] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. Proceedings of Advances in Neural Information Processing Systems, 2014.
- [29] Schoner G, Kelso J A. Dynamic pattern generation in behavioral and neural systems. Science, 1988; 239(4847): 1513–1520.
- [30] Denton E L, Chintala S, Fergus R. Deep generative image models using a variational laplacian pyramid of adversarial networks. Proceedings of Advances in Neural Information Processing Systems, 2015.
- [31] Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceeding of IEEE Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, 2017. Available at: <https://arxiv.org/abs/1609.04802>.
- [32] Tran L, Yin X, Liu X M. Disentangled representation learning gan for pose-invariant face recognition. In Proceeding of IEEE Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, 2017. Available at: [http://cvlab.cse.msu.edu/pdfs/Tran\\_Yin\\_Liu\\_CVPR2017.pdf](http://cvlab.cse.msu.edu/pdfs/Tran_Yin_Liu_CVPR2017.pdf)
- [33] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors. Nature, 1986; 323(6088): 533–536.
- [34] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998; 86(11): 2278–2324.
- [35] Krizhevsky, Alex, Ilya Sutskever, Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. Proceedings of Advances in Neural Information Processing Systems, 2012.
- [36] Mikolov T, Karafiát M, Burget L, Černocký J H, Khudanpur S. Recurrent neural network based language model. Eleventh Annual Conference of the International Speech Communication Association, Chiba, Japan, 2010.
- [37] Sundermeyer M, Schlüter R, Ney H. LSTM neural networks for language modeling. Thirteenth Annual Conference of the International Speech Communication Association, Portland, Oregon, USA, 2012.
- [38] Abadi M, Barham P, Chen J M, Chen Z F, Davis A, Dean J, et al. TensorFlow: A system for large-scale machine learning. Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), Savannah, GA, USA, 2016.
- [39] Rampasek L, Goldenberg A. Tensorflow: Biology's gateway to deep learning? Cell Systems, 2016; 2(1): 12–14.
- [40] Jia Y Q, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, et al. Caffe: Convolutional architecture for fast feature embedding. Proceedings of the 22nd ACM international conference on Multimedia. 2014. Available at: <https://arxiv.org/pdf/1408.5093.pdf>.
- [41] Yalcin H, Razavi S. Plant classification using convolutional neural networks. Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics), IEEE, 2016.
- [42] Sladojevic S, Arsenovic M, Anderla A, Culibrk D, Stefanovic D. Deep neural networks based recognition of plant diseases by leaf image classification. Computational Intelligence and Neuroscience, 2016; Article ID 3289801, 11p.
- [43] Hanson A, Joel M G, Joy A, Francis J. Plant leaf disease detection using deep learning and convolutional neural network. International Journal of Engineering Science, 2017; 5324.
- [44] Tang J L, Wang D, Zhang Z G, He L J, Xin J, Xu Y. Weed identification based on K-means feature learning combined with convolutional neural network. Computers and Electronics in Agriculture, 2017; 135: 63–70.
- [45] Srbinovska M, Gavrovski C, Dimcev V, Krkoleva A, Borozan V. Environmental parameters monitoring in precision agriculture using wireless sensor networks, Journal of Cleaner Production, 2015; 88: 297–307.
- [46] Ge Z Y, McCool C, Sanderson C, Corke P. Content specific feature learning for fine-grained plant classification. CLEF (Working Notes), 2015. Available at: <http://ceur-ws.org/Vol-1391/39-CR.pdf>.
- [47] Yalcin H. Plant phenology recognition using deep learning: 6th International Conference on Deep-Pheno. Agro-Geoinformatics, IEEE, 2017.
- [48] Chen S W, Shivakumar S S, Dcunha S, Das J, Okon E, Qu C, et al. Counting apples and oranges with deep learning: A data-driven approach. IEEE Robotics and Automation Letters, 2017; 2(2): 781–788.
- [49] Kussul N, Shelestov A, Lavrenyuk M, Butko I, Skakun S. Deep learning approach for large scale land cover mapping based on remote sensing data fusion. Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 2016. doi: 10.1109/IGARSS.2016.7729043.
- [50] Lu H, Fu X, Liu C, Li L G, He Y X, Li N W. Cultivated land information extraction in UAV imagery based on deep convolutional neural network and transfer learning. Journal of Mountain Science, 2017; 14(4): 731–741.
- [51] Christiansen P, Hansen M, Steen K, Karstoft H, Jørgensen R. Advanced sensor platform for human detection and protection in autonomous farming. In Precision Agriculture'15; Wageningen Academic Publishers: Wageningen, the Netherlands, 2015; pp.1330–1334.
- [52] Steen K A, Christiansen P, Karstoft H, Jørgensen R N. Using deep learning to challenge safety standard for highly autonomous machines in agriculture. Journal of Imaging, 2016; 2(1): 6.
- [53] Arkeman Y, Buono A, Hermadi I. Satellite image processing for precision agriculture and agroindustry using convolutional neural network and genetic algorithm. IOP Conference Series: Earth and Environmental Science 54 (2017) 012102. doi: 10.1088/1755-1315/54/1/012102.
- [54] Sun Y, Zhu L, Wang G, Zhao F. Multi-input convolutional neural network for flower grading. Journal of Electrical and Computer Engineering, 2017; Article ID 9240407, 8p. <https://doi.org/10.1155/2017/9240407>.
- [55] Salman A G, Kanigoro B, Heryadi Y. Weather forecasting using deep learning techniques. Advanced International Conference on Computer Science and Information Systems (ICACSIS), IEEE, 2015.
- [56] You J X, Li X C, Low M, Lobell D, Ermon S. Deep Gaussian process for crop yield prediction based on remote sensing data. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), pp.4559–4665.
- [57] Kasfi K T, Hellicar A, Rahman A. Convolutional neural network for time series cattle behaviour classification. Proceedings of the Workshop on Time Series Analytics and Applications. ACM, 2016.

- [58] Rahman A, Smith D, Hills J, Bishop-Hurley G, Henry D, Rawnsley R. A comparison of autoencoder and statistical features for cattle behaviour classification. In IEEE World Congress on Computational Intelligence (IEEE WCCI), 2016; pp.1–7.
- [59] Rußwurm M, Körner M. Multi-temporal land cover classification with long shortterm memory neural networks. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-1/W1, 2017; pp.551–558.
- [60] Ienco D, Gaetano R, Dupaquier C, Maurel P. Land cover classification via multi-temporal spatial data by recurrent Neural Networks IEEE Geoscience & Remote Sensing Letters, 2017; 99: 1–5.
- [61] Lyu H, Lu H, Mou L. Learning a transferable change rule from a recurrent neural network for land cover change detection. Remote Sensing, 2016; 8(6): 506.
- [62] Namin S T, Esmailzadeh M, Najafi M, Brown T B, Borevitz J O. Deep phenotyping: Deep learning for temporal phenotype/genotype classification. bioRxiv, 2017. doi: <https://doi.org/10.1101/134205>.
- [63] Minh D H T, Ienco D, Gaetano R, Lalande N, Ndikumana E. Deep recurrent neural networks for winter vegetation quality mapping via multitemporal SAR Sentinel-1. IEEE Geoscience & Remote Sensing Letters, 2017; 99: 1–5.
- [64] Chai L N, Qu Y H, Zhang L X, Liang S L, Wang J D. Estimating time-series leaf area index based on recurrent nonlinear autoregressive neural networks with exogenous inputs International Journal of Remote Sensing, 2012; 33 (18): 5712–5731.
- [65] Chen B Q, Wu Z X, Wang J K, Dong J W, Guan L M, Chen J M, et al. Spatio-temporal prediction of leaf area index of rubber plantation using HJ-1A/1B CCD images and recurrent neural network. ISPRS Journal of Photogrammetry and Remote Sensing, 2015; 102: 148–160.
- [66] Biswas S K, Sinha N, Purkayastha B, Marbaniang L. Weather prediction by recurrent neural network dynamics. International Journal of Intelligent Engineering Informatics, 2014; 2(2-3): 166–180.
- [67] Zaytar M A, El Amrani C. Sequence to sequence weather forecasting with long short-term memory recurrent neural networks. International Journal of Computer Applications, 2016; 143(11): 7–11.
- [68] Lu Z, Chai L N, Liu S M, Cui H Z, Zhang Y H, Jiang L M, et al. Estimating time series soil moisture by applying recurrent nonlinear autoregressive neural networks to passive microwave data over the Heihe River Basin, China. Remote Sensing, 2017; 9(6): 574.
- [69] Tzeng Y C, Fan K T, Lin C Y, Lee Y J, Chen K S. Estimation of soil moisture dynamics using a recurrent dynamic learning neural network. Geoscience and Remote Sensing Symposium, 2012; 88(8): 1251–1253.
- [70] Palangpour P, Venayagamoorthy G K, Duffy K. Recurrent neural network based predictions of elephant migration in a South African game reserve. International Joint Conference on Neural Networks, 2006; pp.4084–4088.
- [71] Demmers T G M, Gauss S, Wathes C M, Cao Y, Parsons D J. Simultaneous monitoring and control of pig growth and ammonia emissions. IX International Livestock Environment Symposium, 2012; C-1323.
- [72] Wahlberg F, Wilkinson T, Brun A. Historical manuscript production date estimation using deep convolutional neural networks. 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), IEEE, 2016.
- [73] Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, et al. Photo-realistic single image super-resolution using a generative adversarial network. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, USA, 2017; pp.105–114.
- [74] Barth R, IJsselmuiden J M M, Hemming J, van Henten E J. Optimising realism of synthetic agricultural images using cycle generative adversarial networks. Proceedings of the IEEE IROS workshop on Agricultural Robotics/Kounalakis, Tsampikos, van Evert, Frits, Ball, David Michael, Kootstra, Gert, Nalpantidis, Lazaros, Wageningen: Wageningen University & Research, 2017; pp.18–22. <http://library.wur.nl/WebQuery/wurpubs/533105>
- [75] Hu K, Qi K, Guan Q, Wu C, Yu J, Qing Y, et al. A scientometric visualization analysis for night-time light remote sensing research from 1991 to 2016. Remote Sensing, 2017; 9(8): 802. doi: 10.3390/rs9080802.
- [76] Garfield E. From the science of science to scientometrics visualizing the history of science with HistCite software. Journal of Informetrics, 2009; 3(3): 173–179.
- [77] McCown R L. Changing systems for supporting farmers' decisions: problems, paradigms, and prospects. Agricultural Systems, 2002; 74(1): 179–220. doi: [https://doi.org/10.1016/S0308-521X\(02\)00026-4](https://doi.org/10.1016/S0308-521X(02)00026-4).
- [78] Grinblat G L, Uzal L C, Larese M G, Granitto P M. Deep learning for plant identification using vein morphological patterns. Computers and Electronics in Agriculture, 2016; 127: 418–424. doi: 10.1016/j.compag.2016.07.003.
- [79] Ding W, Taylor G. Automatic moth detection from trap images for pest management. Computers and Electronics in Agriculture, 2016; 123: 17–28. doi: <https://doi.org/10.1016/j.compag.2016.02.003>.
- [80] Chen C. CiteSpace II: Detecting and visualizing emerging trends and transient patterns in scientific literature. Journal of the American Society for Information Science and Technology, 2006; 57(3): 359–77.
- [81] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation. Journal of Machine Learning Research, 2003; 3(1): 993–1022.
- [82] Chen C, Leydesdorff L. Patterns of connections and movements in dual-map overlays: A new method of publication portfolio analysis. Journal of the Association for Information Science and Technology, 2013; 65(2): 334–51. doi: 10.1002/asi.22968.
- [83] Blondel V D, Guillaume J-L, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory And Experiment, 2008; 2008(10): P10008. doi: 10.1088/1742-5468/2008/10/P10008.
- [84] Gwan J, Moon H H, Tae J, Syed K, Hassan I, Dang M, et al. Deep convolutional neural network for classifying Fusarium wilt of radish from unmanned aerial vehicles. Journal of Applied Remote Sensing, 2017; 11(4). doi: 10.1117/1.JRS.11.042621.
- [85] Dyrmann M, Midtby H S, Mortensen A K. Pixel-wise classification of weeds and crops in images by using a fully convolutional neural network. Proceedings of the International Conference on Agricultural Engineering, Aarhus, Denmark, 2016.
- [86] Grinblat G L, Uzal L C, Larese M G, Granitto P M. Deep learning for plant identification using vein morphological patterns. Computers and Electronics in Agriculture, 2016; 127: 418–424.