This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

High-Order Convolutional Neural Network Architecture for Predicting DNA-Protein Binding Sites

QinHu Zhang, Lin Zhu, and De-Shuang Huang, *Senior Member*, *IEEE*

***Abstract*—Although Deep learning algorithms have outperformed conventional methods in predicting the sequence specificities of DNA-protein binding, they lack to consider the dependencies among nucleotides and the diverse binding lengths for different transcription factors (TFs). To address the above two limitations simultaneously, in this paper, we propose a high-order convolutional neural network architecture (HOCNN), which employs a high-order encoding method to build high-order dependencies among nucleotides, and a multi-scale convolutional layer to capture the motif features of different lengths. The experimental results on real ChIP-seq datasets show that the proposed method outperforms the state-of-the-art deep learning method (DeepBind) in the motif discovery task. In addition, we provide further insights about the importance of introducing additional convolutional kernels and the *degeneration* problem of importing high-order in the motif discovery task.**

***Index Terms*—High-order, Multi-scale convolutional layer, Transcription factor, Binding specificity**

I. Introduction

FIGURING out the interactions between regulatory proteins and DNA, especially the interactions between transcription factors (TFs) and their corresponding binding sites, is one of the primary goal in gene-regulatory process, since TFs can activate or suppress the transcription by binding to specific regions of DNA sequences, which are known as transcription factor binding sites (TFBS) . Previous researches have concluded that TFs are relatively conserved in the long-term evolution, and are inclined to bind to specific regions of DNA sequences, which are also called TFBS motifs [1-3]. Discovery of such motifs not only can help understand the expression of genes, but also identify causal disease variants and design therapeutic drugs.

However, motif discovery has always been a challenging task since it calls for accurate biophysical models to pinpoint TFBS on DNA sequences. For this requirement, some word-based algorithms [4-12] are first introduced, which model TF binding preferences with a consensus sequence, but they cannot get a nuanced representation of motif. Probabilistic-based algorithms [13-20] are then proposed, which model TF binding preferences by performing local searches for the most represented segments in the input sequences. Position Weight Matrix (PWM) [13] is a commonly-used probabilistic-based model, which models TF binding preferences with a $4 \times l$ matrix whose elements represent a probability distribution over DNA alphabet. However, these models still have considerable limitations [21], especially for handling large-scale biological data produced by some high-throughput sequence technologies (e.g., ChIP-seq [22]).

Growing evidences indicate that binding sites can be more accurately predicted by more complex techniques [23, 24]. Recently, deep learning has been successfully applied to a variety of domains and achieved some impressive performance, including computer vision [25], natural language processing (NLP) [26], motif discovery [27, 28], and others [29-34]. DeepBind [27] successfully applied deep convolutional neural network (CNN) [35], which is a variant of multilayer artificial neural network [36-38] and specialized for processing images, to modeling the sequence specificities of DNA-binding proteins, and its performance is superior to some best existing conventional methods. Zeng *et al*. [28] replicated the DeepBind model using the Caffe platform [39], which achieves nearly identical performance to the original DeepBind in the motif discovery task, and then extended it to nine different architectures by varying CNN width, depth and pooling designs. To the best of our knowledge, these deep CNNs, which try to capture local patterns by utilizing a weight-sharing strategy, is well suited to genomics [40], since convolution kernels can be thought of as motif scanners like PWMs to capture sequence features. For the motif discovery task, a genomic sequence with four nucleotides {A, C, G, T} can be transformed into an image-like input by using the one-hot encoding way. Therefore this task is analogous to a two-class image classification in the computer vision. Through the powerful classification ability of deep CNN, a target sequence can be well classified, meanwhile its PWMs can be obtained by analyzing convolution kernels.

Although existing deep learning methods have achieved remarkable performance for the motif discovery task, they still have some limitations. (1) They only considered the independent relationship among nucleotides in the binding sites,

This work was supported by the grants of the National Science Foundation of China, Nos. 61732012, 61520106006, 31571364, 61672382, U1611265, 61532008, 61472280, 61472173, 61572447, and 61772357, China Postdoctoral Science Foundation Grant, Nos. 2016M601646 and 2017M611619, and supported by "BAGUI Scholar" Program of Guangxi Province of China.

Q.H Zhang (E-mail: zhangqinhu1@qq.com), L. Zhu (E-mail: lizhonyx@163.com), and D.S. Huang (E-mail: dshuang@tongji.edu.cn) are all with the Institute of Machine Learning and Systems Biology, School of Electronics and Information Engineering, Tongji University, No. 4800 Caoan Road, Shanghai 201804, China.

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

and ignored the high-order dependencies among nucleotides in practice. Models based on PWM including deep learning, make a strong assumption that all nucleotides in the binding sites are statistically independent, and participate independently in the corresponding DNA-protein interactions. Recent studies have proved that taking into account high-order dependencies among nucleotides within TFBS not only can improve the discriminative performance, but also yield a better motif representation [41-43]. However, complex models based on high-order dependencies contain more parameters than the PWM-based ones, and are prone to overfitting [42]. Deep learning is still a good choice for alleviating this issue due to its efficiency of leveraging high amount of available data. (2) They only employed a fixed motif length to capture the binding features in the genomic sequences. For instance, DeepBind used a fixed motif length 24 to predict DNA sequence specificities for all ENCODE transcription factor data. Various TFs, however, have diverse binding lengths [44, 45]. Sela *et al*. [44] presented that a typical length of TF binding sites varies between 6 and 20 nucleotides. Actually, there exists a similar issue in other applications of CNN [46-51]. For example, in the image classification application, an image is composed of objects of multiple scales, therefore Szegedy *et al*. [46] proposed the 'Inception model' which uses a series of fixed filters (convolutional kernels) of different sizes to capture multi-scale features. In the text classification application, a sentence is composed of words of different lengths, therefore Kim *et al*. [47] employed multiple filters of varying windows size to obtain multi-scale features.

Motivated by the aforementioned observations, here we propose a high-order convolutional neural network architecture (HOCNN) for the motif discovery task, which integrates the high-order dependencies among nucleotides and multi-scale motif features into the original CNN. Firstly, HOCNN uses a high-order encoding method to transform genomic sequences into image-like inputs of non-independent, as opposed to using the simple one-hot encoding way, and the encoded sequences are feed into a multi-scale convolutional layer, which is consisted of multiple filters of different motif lengths, and the outputs (feature maps) are then concatenated as inputs of the subsequent layers. Finally, these features from the penultimate layer are passed through a softmax layer to generate a probability distribution over two labels. Experimental results on real data illustrate the performance of the proposed approach in terms of AUC metric.

The reminder of this paper is organized as follows. Firstly, we give a brief description of DeepBind [27] in section II. Then, the proposed approach is introduced in section III. Finally, we show the results of the proposed approach on real data and conclude the paper in section IV and V respectively.

II. Related Works

In recent years, CNN has made great achievements in various application scenarios, which is perhaps the reason why so many researchers want to apply it to their own domains. DeepBind [27] is the first attempt to apply deep CNN to the motif discovery task, by which the binding preferences of TFs to DNA sequence is well characterized. It employs a convolutional layer followed by a nonlinear layer, a max-pooling layer and two fully-connected layers to estimate the probability distribution of input sequences over two labels. Zeng *et al*. [28] replicated DeepBind using Caffe platform, which achieves nearly identical performance in the motif discovery task. In order to have a better knowledge of this architecture, convolution, pooling and neural network stages are detailed as follows.

A. Convolution

This stage contains a convolutional layer followed by a nonlinear layer which adopts rectified linear unit (ReLU) [52] as the nonlinear function. ReLU is a commonly nonlinear function used in deep neural networks, due to its ability of alleviating the vanishing gradient problem. Therefore, how to use the convolutional neural network to learn the motif patterns of DNA sequences? DeepBind employed the one-hot encoding way to transform a DNA sequence composed of {A, C, G, T} into an image-like input of 4 channels. For example: A, C, G, and T are encoded into four one-hot vectors [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], and [0, 0, 0, 1] respectively. Given a DNA sequence $s = (s_1, s_2, \dots, s_n)$ of length n and a fixed motif scanner length m, we firstly pad (m-1) 'zero' on both sides of the input sequence, and get the encoded input $S_{i,j}$ through the following equation:

$$S_{i,j} = \begin{cases} 1 \text{ if } s_{i-m+1} = j^{th}\text{base in } \{\text{A,C,G,T}\} \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

According to (2), put $S_{i,j}$ through the convolutional layer whose kernels can be thought of as motif scanners like PWMs, and the ReLU layer which corresponds to the rectification stage in the DeepBind, and then we can get the outputs X which are the convolution of the kernels M and the one-hot vectors S.

$$X_{i,k} = \max\left(0, \sum_{j=1}^{m}\sum_{l=1}^{4} S_{i+j,l} M_{k,j,l} + b_k\right) \tag{2}$$

where $i \in [1, n+m\text{-}1]$ and $k \in [1, d]$, d denotes the number of convolution kernels, b_k denotes the bias term.

B. Pooling

In the context of genomic sequences, diverse genomic data have their corresponding pooling strategies. TFBS prediction utilizes a max-pooling strategy which picks out the maximum over the outputs of the convolution stage. With consideration of the ZOOPS assumption that zero or one motif occurrences per sequence [53], this process can be deemed as a way of judging whether the motif scanned by the convolutional layer exists in the input sequence or not.

$$z_k = \max\left(X_{1,k}, \cdots, X_{n,k}\right) \tag{3}$$

where $k \in [1, d]$, d is the number of convolution kernels.

C. Neural Network

This stage contains two fully-connected layers in which the

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

first layer is followed by a nonlinear layer (ReLU), and the second layer is followed by a softmax layer which transforms the outputs into a probability distribution over two labels. The first layer consists of 32 neurons, and the second layer consists of 2 neurons, which corresponds to a two classification task. In addition, the dropout strategy [54] has often been used to avoid overfitting in this stage, which randomly masks the outputs of the first fully-connected layer to zero. The computation of this stage is in terms of the following equation:

$$h_j = \max\left(0,\ \sum_{k=1}^{d} W^1_{j,k} z_k + b_j\right),\ \text{ for } j \in [1,32]$$
$$f_i = \text{softmax}\left(\sum_{j=1}^{32} W^2_{i,j} h_j + b_i\right),\ \text{ for } i \in [1,2] \qquad (4)$$

where h_j denotes the outputs of the first fully-connected layer, f_i denotes a probability distribution over two labels.

III. Methods

A. High-order encoding method

DeepBind [27] employed the one-hot encoding way to transform DNA sequences into one-hot vectors, and then iteratively update the coefficients of kernel matrices (PWMs) by learning these vectors. Actually, the one-hot encoding way is based on an implicit assumption that all nucleotides in the binding sites are statistically independent, and participate independently in the corresponding DNA-protein interactions, which makes modeling of certain TFBS difficult. Recently, some related works have demonstrated that taking into account high-order dependencies among nucleotides not only can improve the discriminative performance, but also yield a better motif representation [41]. Inspired by this fact, here we propose a high-order encoding method which takes into consideration the dependencies among nucleotides. This encoding method can be implemented in terms of (5):

$$S_{i,j} = \begin{cases} 1 \ \text{ if } s_{i-m+1}\cdots s_{i-m+h} = j^{th}\text{base in}\left\{4^h \ \ h\text{-nucleotides}\right\} \\ 0 \ \text{ otherwise} \end{cases} \qquad (5)$$

where h denotes the degree of high-order, and m denotes the width of convolutional kernels and corresponds to the motif length. Actually, the one-hot encoding way is a special case of the high-order encoding method when h is equal to 1. For example, one-order encoding: it means only taking into account each independent nucleotide, and each nucleotide is encoded into a one-hot vector of size 4 (A → [1, 0, 0, 0], C → [0, 1, 0, 0], G → [0, 0, 1, 0], and T → [0, 0, 0, 1]). Two-order encoding: it means taking into account the dependencies between two adjacent nucleotides, which has 16 dinucleotides in total, and each dinucleotide is encoded into a one-hot vector of size 16 (AA → [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], ⋯ , TT → [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]). Therefore if the degree of high-order is set to h, we will get 4^h h-nucleotides in which each h-nucleotide is encoded into a one-hot vector of size 4^h.

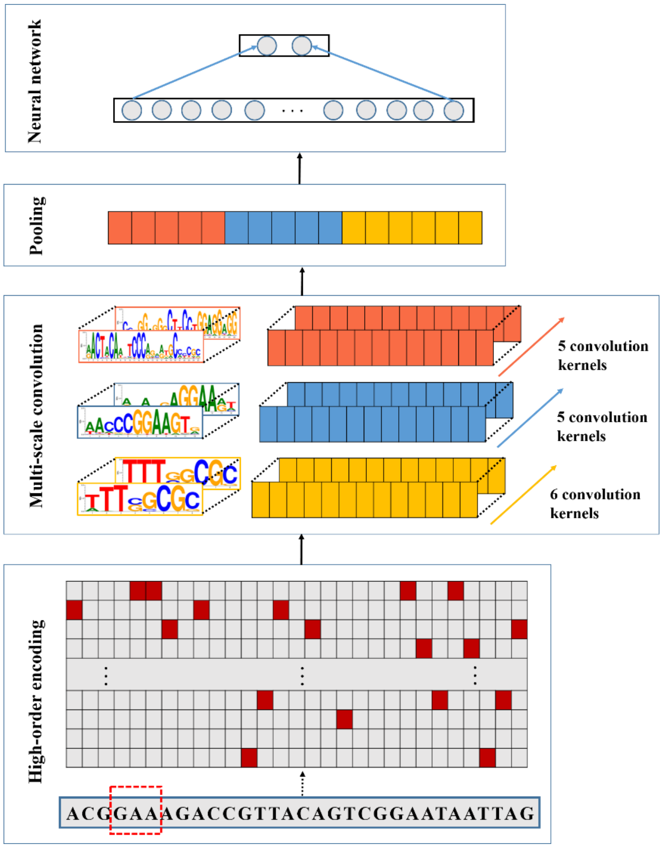

Fig. 1. A graphical illustration of the HOCNN architecture. (1) An input DNA sequence is first encoded into a 4^h-row bit matrix in which each column represents an h-nucleotide, by using the high-order encoding method. The width of the red dotted rectangle denotes the degree of high-order. (2) A multi-scale convolutional layer with ReLU activations which acts as a motif scanner takes the encoded matrix as input to produce an output matrix containing multi-scale motif features. This layer contains 16 convolutional kernels of different motif lengths (24, 12, and 8), in which the number of convolutional kernels of size 24, 12, and 8 is 5, 5, and 6 respectively. The first row denotes convolutional kernels of motif length 24, and produces the outputs labelled by red color. The second row denotes convolutional kernels of motif length 12, and produces the outputs labelled by blue color. The last row denotes convolutional kernels of motif length 8, and produces the outputs labelled by yellow color. These outputs labelled by different color are then concatenated as inputs of the next layer. (3) Max-pooling layer takes the maximum over the outputs produced by the previous layer, and produces a vector of 16 motif features. (4) In the end, these significant features are fed into a neural network to produce a probability distribution over two classes.

However, the model is getting more and more complex with the increase of the degree, since it will contain more and more parameters. So the current issue we faced is how to determine the degree of high-order. If the degree of high-order is too high, the model will be prone to overfitting. Therefore the degree of high-order is considered as a hyper-parameter in this paper, and investigated through cross-validation experiments. Considering the above situation, a few relative low degrees are tested in this paper, including one, two, three, and four-order encoding.

B. Multi-scale convolutional layer

Motif length plays an important role in accurately predicting DNA-protein binding sites, and various TFs have diverse motif lengths [44]. However, DeepBind fixed motif length to 24 when predicting TF binding sites for all ENCODE transcription factor datasets. To the best of our knowledge, models based on a fixed

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

motif length cannot accurately capture the different local patterns (motifs) of genomic sequences. To address this limitation, a multi-scale convolutional layer, which employs multiple convolutional kernels of different motif lengths, is introduced here. This layer integrates multiple motif scanners of different lengths which separately capture the different local patterns of genomic sequences, and generates multi-scale outputs which are then concatenated as inputs of the subsequent layers. In theory, the more the diversity of features, the better the performance is. However, in order to make a trade-off between the performance and computational complexity of the proposed model in this paper, three different motif lengths (8, 12, and 24) are chosen as a fixed hyper-parameter. On the basis of (2), the implementation of this convolutional layer is as follows:

$$X_{i,k} = \underset{m \in \Phi}{\text{Concatenate}} \left(\max\left(0,\ \sum_{j=1}^{m} \sum_{l=1}^{4^h} S_{i+j,l} M_{k,j,l} + b_k \right) \right) \tag{6}$$

where Φ denotes a set of motif lengths, without loss of generality, and Φ can contains any motif lengths, *m* denotes the motif length of scanners, and *h* denotes the degree of high-order.

The rest of the layers is the same as the DeepBind. Fig.1 depicts the overall architecture of the proposed method in this paper.

IV. Results

In this section, the performance of HOCNN is systematically evaluated by comparing it with DeepBind. HOCNN was implemented on the Caffe platform, which is freely available at https://github.com/turningpoint1988/HOCNN. In order to make a fair comparison in the same platform, therefore we adopted its Caffe-based implementation [28] as the baseline method, which achieves nearly identical performance in the motif discovery task. For simplicity, its Caffe-based implementation is abbreviated as DeepBind_C in this paper.

A. Data

We collected 214 public ChIP-seq datasets from ENCODE to evaluate the performance of our proposed HOCNN method, which stem from 5 different cell lines in both HaibTfbs datasets[1] and SydhTfbs datasets[2]. For each ChIP-seq dataset, 1000-5000 top ranking peaks were chosen as the foreground (positive) set in which each sequence consists of 200 bps. On the other hand, the way of generating background (negative) sequences is also crucial. It is widely recognized that the background sequences have to be selected to match the statistical properties of the foreground set [55-57], otherwise the elicited motifs could be biased [58]. To satisfy such requirements, equal numbers of background sequences were generated by matching the length, GC content and repeat fraction of the foreground set following [55].

In order to accurately assess the performance of our proposed method, here we adopted three-fold cross-validation strategy, that is, each ChIP-seq dataset is randomly partitioned into 3 sets (folds) of roughly equal size, and two of them are used as the training set while the rest is used as the test set. During training, we randomly sampled 1/8 of the training set as the validation set.

B. Evaluation metric

In this paper, AUC (the area under the receiver operating characteristic curve) was adopted to evaluate the performance of the proposed method. It is a widely used evaluation metric in machine learning and motif discovery [21, 27, 28, 58, 59], which is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one [60].

C. Hyper-parameter settings and implementation

We performed hyper-parameter search as described in [28]. For each ChIP-seq experiment, 30 hyper-parameter settings were randomly sampled from hyper-parameter space, in which three hyper-parameters are random, including dropout ratio, momentum, and delta, and the rest is fixed. In order to show the effect of the degree of high-order and the number of convolutional kernels on the experimental results, we instead used a few fixed hyper-parameters for them, such as using {16, 64, and 128} as the number of convolutional kernels, and {1, 2, 3, and 4} as the degree of high-order. In the experiments, the number of convolutional kernels of the two methods was selected from {16, 64, and 128} one by one, and DeepBind_C adopted the one-hot encoding way which is corresponding to the 1-order encoding method, while our proposed method HOCNN used the 2, 3, and 4-order encoding method in turn. DeepBind_C employed a fixed motif length 24 as the convolutional kernel size, while HOCNN used multiple motif lengths {24, 12, and 8} as the convolutional kernel size. To make a fair comparison between the two methods, we kept the equal number of convolutional kernels all the time. For example, if the number of convolutional kernels of size 24 in DeepBind_C is 16, the number in HOCNN is also 16 in which the number of convolutional kernels of size 24, 12, and 8 is 5, 5, and 6 respectively. Besides, all weights were initialized by drawing from Xavier uniform distribution *U* (-*b*, +*b*) [61], and all biases were initialized to zero. The two methods were trained using the AdaDelta algorithm [62] to minimize the binary cross entropy loss function on the training set. The learning rate, weight decay, and iterations were set to 1, 0.0005, and 6000 in each experiment, respectively. The neural network has two fully-connected layers which consist of 32 and 2 neurons respectively. All hyper-parameter settings are summarized in Table I.

The whole process contains three stages including hyper-parameter search stage, training stage and test stage. In the hyper-parameter search stage, for each ChIP-seq experiment, we first used 30 candidate hyper-parameter settings to train our

[1] http://hgdownload.cse.ucsc.edu/goldenpath/hg19/encodeDCC/wgEncodeHaibTfbs/

[2] http://hgdownload.cse.ucsc.edu/goldenpath/hg19/encodeDCC/wgEncodeSydhTfbs/

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

proposed method HOCNN on the training set with a mini-batch size of 100, and tested it on the validation set. Then, a hyper-parameter setting with the best validation accuracy which best fits the corresponding ChIP-seq dataset was obtained. In the training stage, we used the best hyper-parameter setting to train HOCNN on the whole training set involving the validation set with a mini-batch size of 300. In the test stage, the test set was tested using the final trained model, and the performance of HOCNN was evaluated using the AUC metric mentioned above.

TABLE I
HYPER-PARAMETER SETTINGS

Hyper-parameters	DeepBind_C	HOCNN	
Dropout ratio	0.75, 0.5, 0.1	0.75, 0.5, 0.1	random
Momentum in AdaDelta	0.999, 0.99, 0.9	0.999, 0.99, 0.9	random
Delta in AdaDelta	1e-4, 1e-6, 1e-8	1e-4, 1e-6, 1e-8	random
Learning rate	1.0	1.0	fixed
Weight decay	0.0005	0.0005	fixed
The degree of high-order	1	2, 3, 4	fixed
Kernel size (motif length)	24	$24 \oplus 12 \oplus 8$	fixed
No. of convolutional kernels	16, 64, 128	16(5, 5, 6), 64(20, 20, 24), 128(40, 40, 48)	fixed
No. of neurons	32/2	32/2	fixed
Iterations	6000	6000	fixed

$24 \oplus 12 \oplus 8$ denotes the concatenation of three kernel sizes.

Similarly, DeepBind_C has the same three stages.

D. *Performance comparison*

To make a comprehensive comparison of DeepBind_C and HOCNN, a series of experiments were conducted by varying the number of convolutional kernels and degree of high-order. Fig.3 displays the contrast results of the two methods using different hyper-parameter configurations. The top scatter plots show the cross-validation performance of HOCNN using 2-order encoding compared with DeepBind_C, and the average AUCs of HOCNN are almost all higher than those of DeepBind_C. The middle ones show the cross-validation performance of HOCNN using 3-order encoding, and the average AUCs of HOCNN are almost all higher than those of DeepBind_C. The bottom ones show the cross-validation performance of HOCNN using 4-order encoding, and most of the average AUCs of HOCNN are higher than those of DeepBind_C. Moreover, Fig.2 displays the distribution of average AUCs across all 214 experiments, and Table II shows the median AUCs of the two methods across all 214 experiments. We observe that the median AUCs of HOCNN using 2-order, 3-order and 4-order encoding are all higher than those of DeepBind_C when keeping equal numbers of convolutional kernels, and the maximal gain of median AUC can reach ~2%. It is evident that our proposed method could achieve higher AUC scores than DeepBind_C, and has better performance for the motif discovery task, which demonstrates the effectiveness of allowing for the high-order dependencies among nucleotides and multi-scale motif features.

In order to investigate how the number of convolutional kernels affects the two methods, the performance of the two methods using 16, 64, and 128 convolutional kernels was separately tested, and then compared with each other. Some existing works have proved that appropriately increasing the width of the convolutional layer can significantly improve the performance of methods [28, 63]. Our experimental results are consistent with the conclusion. With the increase of convolutional kernels, as shown in Fig.2 and Table II, the performance of our proposed method and DeepBind_C is both improved in terms of AUC metric. As mentioned in [28], the improvement seemed to be almost saturated when more than 128 convolutional kernels were deployed. Therefore the upper limit of kernels is set to 128 in this paper.

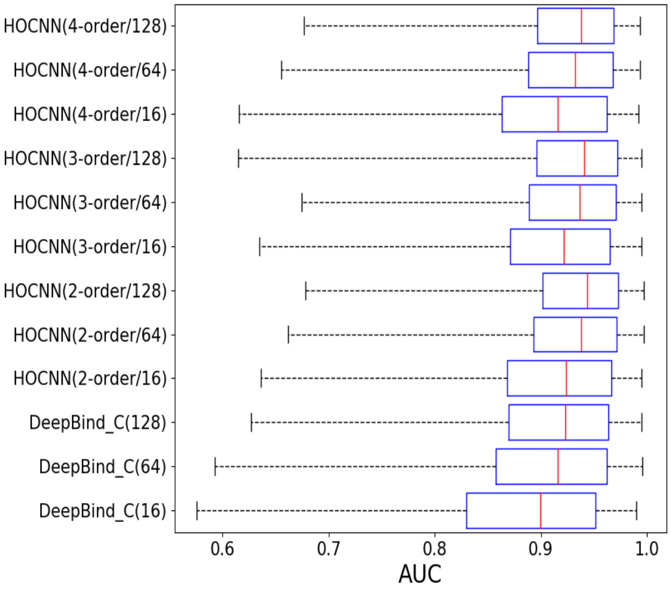

Fig. 2. The distribution of average AUCs across 214 experiments in the motif discovery task.

TABLE II
MEDIAN AUC ACROSS 214 CHIP-SEQ DATASETS

Methods / Kernels	DeepBind_C	HOCNN		
	1-order	2-order	3-order	4-order
No. = 16	0.9098	**0.9234**	0.9218	0.9159
No. = 64	0.9185	**0.9382**	0.9365	0.9322
No. = 128	0.9286	**0.9436**	0.9406	0.9379

In order to investigate how high-order affects our proposed method HOCNN, the performance of HOCNN using 2-order, 3-order and 4-order encoding was separately tested, and then compared with each other. Fig.4 shows a comparison of HOCNN using different degrees of high-order. We observe that the performance of HOCNN using 2-order encoding is better than that of using 3-order encoding, and the performance of HOCNN using 3-order encoding is better than that of using 4-order encoding, which shows that the performance of our proposed method is getting worse as the degree of higher-order increases. Table II quantitatively shows the degenerated

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

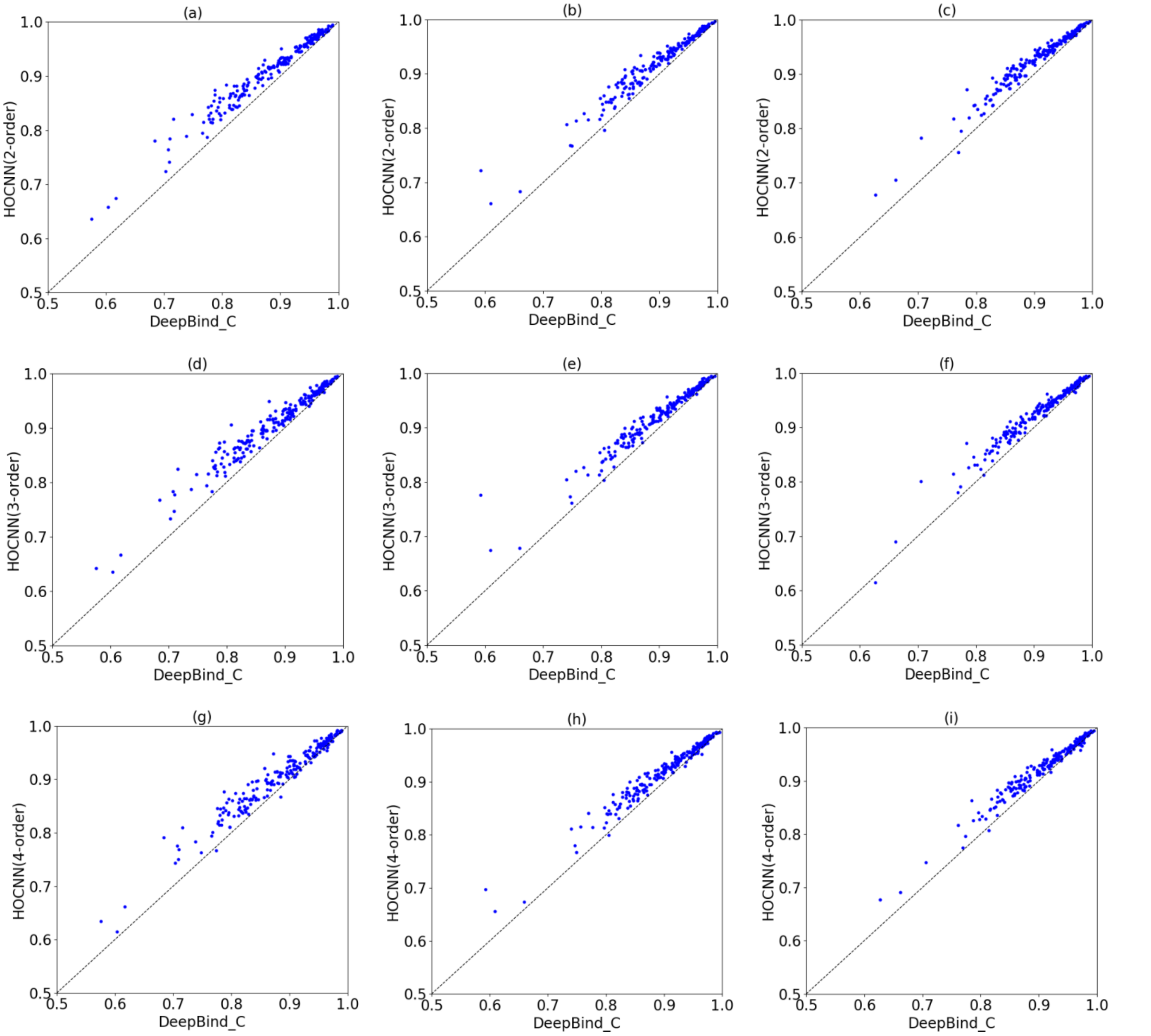

Fig. 3. Average cross-validation AUC comparison of HOCNN and DeepBind_C on 214 ChIP-seq datasets. The top scatter plots (a, b, and c) show a comparison of HOCNN using 2-order encoding and DeepBind_C. The middle ones (d, e, and f) show a comparison of HOCNN using 3-order encoding and DeepBind_C. The bottom ones (g, h, and i) show a comparison of HOCNN using 4-order encoding and DeepBind_C. From left to right, the number of convolutional kernels is 16, 64 and 128 respectively.

performance with the increase of high-order. Such phenomenon is in accord with the issue mentioned above that complex models consisting of more learnable parameters are prone to overfitting. The extra parameters of HOCNN mainly lie in the multi-scale convolutional layer in which convolutional kernels are represented by a tensor of size $4^h \times W \times C$, where 4^h denotes the height of kernels, and h denotes the degree of high-order, and W denotes the width of kernels and corresponds to the motif length, and C denotes the number of kernels. For example, if the degree h is set to 1, the convolutional layer will contain $4*W*C$ learnable parameters, and if the degree h is set to 4, the convolutional layer will contain $256*W*C$ learnable parameters. Thus it can be seen that the number of learnable parameters grows exponentially with the increase of the degree of high-order, which is the main reason why the performance of our proposed method is degenerated as the degree increases. For ChIP-seq datasets of moderate size in this paper, 2-order or 3-order encoding may be a good choice. For ChIP-seq datasets of large size, higher-order encoding may be a good attempt, such as 4-order encoding, but this is our future work.

V. Conclusions

In this paper, we propose High-order convolutional neural network (HOCNN) for the motif discovery task, which

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

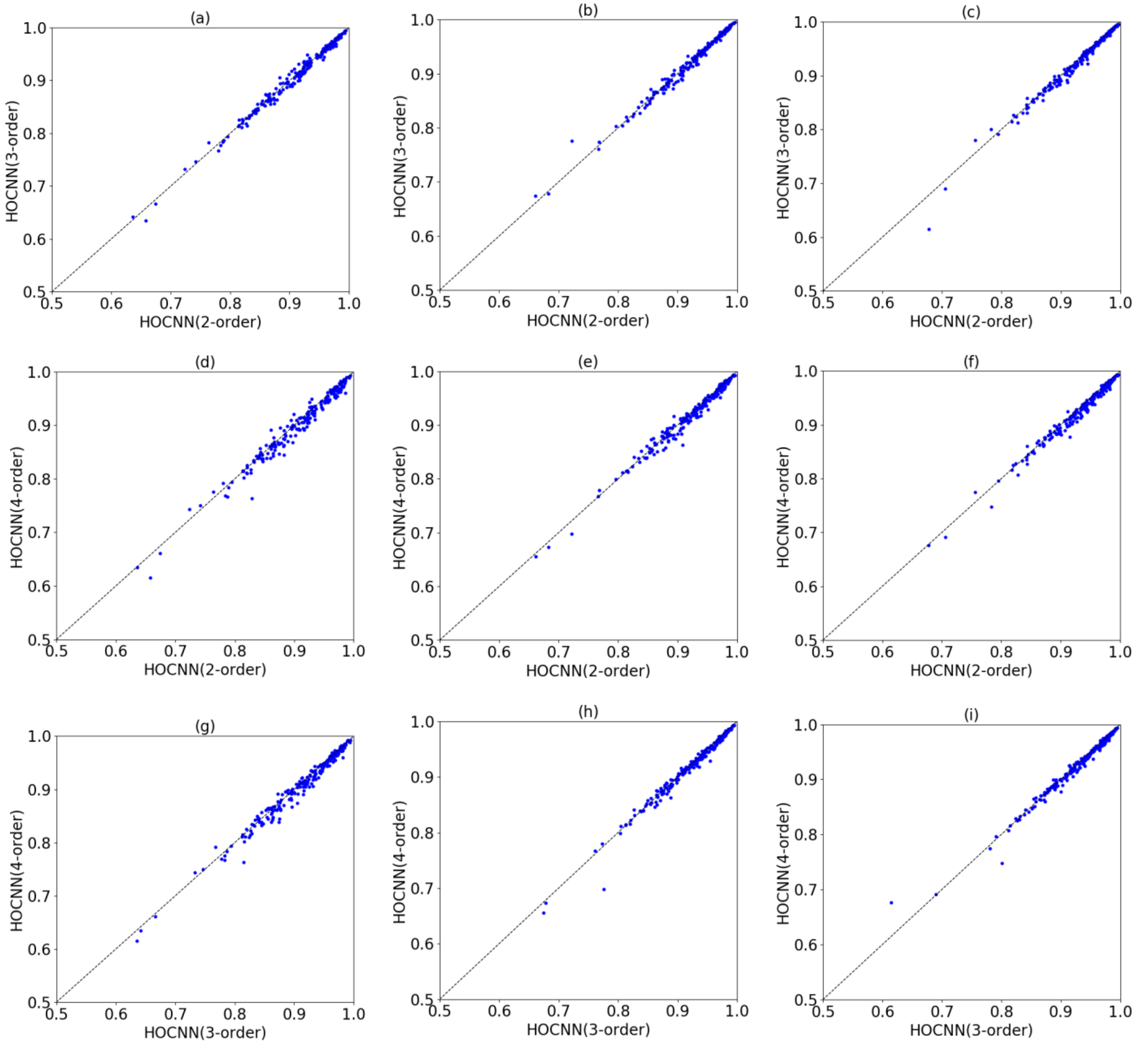

Fig. 4. The effect of high-order on the performance of HOCNN across 214 ChIP-seq datasets. The top scatter plots (a, b, and c) show a comparison of HOCNN using 2-order encoding and HOCNN using 3-order encoding. The middle ones (d, e, and f) show a comparison of HOCNN using 2-order encoding and HOCNN using 4-order encoding. The bottom ones (g, h, and i) show a comparison of HOCNN using 3-order encoding and HOCNN using 4-order encoding. From left to right, the number of convolutional kernels is 16, 64 and 128 respectively.

integrates a new encoding method for building high-order dependencies among nucleotides and a multi-scale convolutional layer for capturing multi-scale motif features. Through a series of comparative experiments, we have shown that our proposed method HOCNN outperforms the baseline DeepBind_C in the motif discovery task, which directly proves the effectiveness of allowing for the dependencies among nucleotides and multi-scale motif features. Besides, we have investigated the effect of high-order on HOCNN by varying the degree of high-order. It is shown that the performance of HOCNN is degenerated as the degree of high-order increases, since the learnable parameters of HOCNN will grow exponentially. Moreover, a general suggestion is given that a low degree could be a good choice for datasets of moderate size, and a relative high degree could be an attempt for datasets of large size.

How to effectively utilize high-order is not only an issue mentioned above, but also an interesting research direction in the future. We have proven that the high-order encoding method is superior to the conventional method (the one-hot encoding way), but it suffers the degradation problem with the increase of the degree of high-order. In other words, it implies that a higher-order encoding may further improve the performance of HOCNN. So one future work is how to reduce the number of parameters while maintaining good performance [64]. As we known, massive data could alleviate the overfitting

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

problem. So another future work is that we could collect more data to test the performance of HOCNN using higher-order encoding.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

[1] L. Elnitski, V. X. Jin, P. J. Farnham, and S. J. Jones, “Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques,” *Genome Research*, vol. 16, no. 12, pp. 1455-1464, 2006.

[2] Y. Zhao, D. Granas, and G. D. Stormo, “Inferring binding energies from selected binding sites,” *PLoS Computational Biology,* vol. 5, no. 12, pp. e1000590, 2009.

[3] B. Wang, S. Valentine, S. Raghuraman, M. Plasencia, and X. Zhang, “Prediction of peptide drift time in ion mobility-mass spectrometry,” *BMC Bioinformatics,* vol. 10, no. 7, pp. A1, 2009.

[4] D. Lee, R. Karchin, and M. A. Beer, “Discriminative prediction of mammalian enhancers from DNA sequence,” *Genome Research,* vol. 21, no. 12, pp. 2167-2180, 2011.

[5] Q. Yu, H. Huo, J. S. Vitter, J. Huan, and Y. Nekrich, “An efficient exact algorithm for the motif stem search problem over large alphabets,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 12, no. 2, pp. 384-397, 2015.

[6] H.J. Yu, and D.S. Huang, “Normalized feature vectors: a novel alignment-free sequence comparison method based on the numbers of adjacent amino acids,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 10, no. 2, pp. 457-467, 2013.

[7] K. Qu, K. Han, S. Wu, G. Wang, and L. Wei, “Identification of DNA-Binding Proteins Using Mixed Feature Representation Methods,” *Molecules,* vol. 22, no. 10, pp. 1602, 2017.

[8] L. Wei, J. Tang, and Q. Zou, “Local-DPP: An improved DNA-binding protein prediction method by exploring local evolutionary information,” *Information Sciences,* vol. 384, pp. 135-144, 2017.

[9] L. Song, D. Li, X. Zeng, Y. Wu, L. Guo, and Q. Zou, “nDNA-prot: identification of DNA-binding proteins based on unbalanced classification,” *BMC Bioinformatics,* vol. 15, no. 1, pp. 298, 2014.

[10] W. Bao, Z. Huang, C.A. Yuan, and D.S. Huang, “Pupylation sites prediction with ensemble classification model,” *International Journal of Data Mining and Bioinformatics,* vol. 18, no. 2, pp. 91-104, 2017.

[11] W. Bao, Z. Jiang, and D. Huang, “Novel human microbe-disease association prediction using network consistency projection,” *BMC Bioinformatics,* vol. 18, no. 16, pp. 543, 2017.

[12] W. Bao, Z. H. You, and D. S. Huang, “CIPPN: computational identification of protein pupylation sites by using neural network,” *Oncotarget,* 2017.

[13] G. D. Stormo, “DNA binding sites: representation and discovery,” *Bioinformatics,* vol. 16, no. 1, pp. 16-23, 2000.

[14] L. Li, Y. Liang, and R. L. Bass, “GAPWM: a genetic algorithm method for optimizing a position weight matrix,” *Bioinformatics,* vol. 23, no. 10, pp. 1188-1194, 2007.

[15] C. Linhart, Y. Halperin, and R. Shamir, “Transcription factor and microRNA motif discovery: the Amadeus platform and a compendium of metazoan target sets,” *Genome Research,* vol. 18, no. 7, pp. 1180-1189, 2008.

[16] C. H. Zheng, L. Zhang, V. T. Ng, S. C. Shiu, and D. S. Huang, “Molecular pattern discovery based on penalized matrix decomposition,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 8, no. 6, pp. 1592-603, Nov-Dec, 2011.

[17] L. Zhu, W.L. Guo, S.P. Deng, and D.S. Huang, “ChIP-PIT: enhancing the analysis of ChIP-Seq data using convex-relaxed pair-wise interaction tensor decomposition,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 13, no. 1, pp. 55-63, 2016.

[18] L. Zhu, S.P. Deng, and D.S. Huang, “A two-stage geometric method for pruning unreliable links in protein-protein networks,” *IEEE Transactions on Nanobioscience,* vol. 14, no. 5, pp. 528-534, 2015.

[19] W. Bao, D. Wang, and Y. Chen, “Classification of Protein Structure Classes on Flexible Neutral Tree,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 14, no. 5, pp. 1122-1133, 2017.

[20] W. Bao, Y. Chen, and D. Wang, “Prediction of protein structure classes with flexible neural tree,” *Bio-medical Materials and Engineering,* vol. 24, no. 6, pp. 3797-806, 2014.

[21] M. T. Weirauch, A. Cote, R. Norel, M. Annala, Y. Zhao, T. R. Riley, J. Saez-Rodriguez, T. Cokelaer, A. Vedenko, and S. Talukder, “Evaluation of methods for modeling transcription factor sequence specificity,” *Nature Biotechnology,* vol. 31, no. 2, pp. 126-134, 2013.

[22] T. S. Furey, “ChIP-seq and beyond: new and improved methodologies to detect and characterize protein-DNA interactions,” *Nature Reviews. Genetics,* vol. 13, no. 12, pp. 840, 2012.

[23] R. Nutiu, R. C. Friedman, S. Luo, I. Khrebtukova, D. Silva, R. Li, L. Zhang, G. P. Schroth, and C. B. Burge, “Direct measurement of DNA affinity landscapes on a high-throughput sequencing instrument,” *Nature Biotechnology,* vol. 29, no. 7, pp. 659-664, 2011.

[24] T. Siggers, and R. Gordân, “Protein–DNA binding: complexities and multi-protein codes,” *Nucleic Acids Research,* vol. 42, no. 4, pp. 2099-2111, 2013.

[25] Krizhevsky, Alex, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks,” *International Conference on Neural Information Processing Systems* Curran Associates Inc. pp. 1097-1105, 2012.

[26] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” *Journal of Machine Learning Research,* vol. 12, no. Aug, pp. 2493-2537, 2011.

[27] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, “Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning,” *Nature Biotechnology,* vol. 33, no. 8, pp. 831-838, 2015.

[28] H. Zeng, M. D. Edwards, G. Liu, and D. K. Gifford, “Convolutional neural network architectures for predicting DNA–protein binding,” *Bioinformatics,* vol. 32, no. 12, pp. i121-i127, 2016.

[29] V. Bevilacqua, G. Tattoli, D. Buongiorno, C. Loconsole, D. Leonardis, M. Barsotti, A. Frisoli, and M. Bergamasco, "A novel BCI-SSVEP based approach for control of walking in virtual environment using a convolutional neural network,” *International Joint Conference on Neural Networks IEEE*. pp. 4121-4128.

[30] V. Bevilacqua, A. Brunetti, M. Triggiani, D. Magaletti, M. Telegrafo, and M. Moschetta, "An optimized feed-forward artificial neural network topology to support radiologists in breast lesions classification,” pp. 1385-1392.

[31] S. P. Deng, L. Zhu, and D. S. Huang, “Mining the bladder cancer-associated genes by an integrated strategy for the construction and analysis of differential co-expression networks,” *BMC Genomics,* vol. 16, no. S3, pp. S4, 2015.

[32] S. P. Deng, and D. S. Huang, "SFAPS: an R package for structure/function analysis of protein sequences based on informational spectrum method,” *Methods*, vol. 69, no. 3, pp. 207-212, 2014.

[33] D. S. Huang, L. Zhang, K. Han, S. Deng, K. Yang, and H. Zhang, “Prediction of protein-protein interactions based on protein-protein correlation using least squares regression,” *Curr Protein Pept Sci,* vol. 15, no. 6, pp. 553-560, 2014.

[34] L. Zhu, Z. H. You, D. S. Huang, and B. Wang, “t-LSE: A Novel Robust Geometric Approach for Modeling Protein-Protein Interaction Networks,” *Plos One,* vol. 8, no. 4, pp. e58368, 2013.

[35] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE,* vol. 86, no. 11, pp. 2278-2324, 1998.

[36] D.S. Huang, “Systematic theory of neural networks for pattern recognition,” *Publishing House of Electronic Industry of China, Beijing,* vol. 201, 1996.

[37] D.S. Huang, “Radial basis probabilistic neural networks: Model and application,” *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 13, no. 07, pp. 1083-1101, 1999.

[38] D.S. Huang, and J.X. Du, “A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks,” *IEEE Transactions on Neural Networks,* vol. 19, no. 12, pp. 2099-2115, 2008.

[39] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding,” pp. 675-678.

[40] Y. Park, and M. Kellis, “Deep learning for regulatory genomics,” *Nature Biotechnology,* vol. 33, no. 8, pp. 825-826, 2015.

[41] J. Keilwagen, and J. Grau, “Varying levels of complexity in transcription factor binding motifs,” *Nucleic Acids Research,* vol. 43, no. 18, pp. e119-e119, 2015.

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2018.2819660, IEEE/ACM Transactions on Computational Biology and Bioinformatics

[42] M. Siebert, and J. Söding, “Bayesian Markov models consistently outperform PWMs at predicting motifs in nucleotide sequences,” *Nucleic Acids Research,* vol. 44, no. 13, pp. 6055-6069, 2016.

[43] R. Eggeling, T. Roos, P. Myllymäki, and I. Grosse, “Inferring intra-motif dependencies of DNA binding sites from ChIP-seq data,” *BMC Bioinformatics,* vol. 16, no. 1, pp. 375, 2015.

[44] I. Sela, and D. B. Lukatsky, “DNA sequence correlations shape nonspecific transcription factor-DNA binding affinity,” *Biophysical Journal,* vol. 101, no. 1, pp. 160-166, 2011.

[45] J. Telorac, S. V. Prykhozhij, S. Schöne, D. Meierhofer, S. Sauer, M. Thomas-Chollier, and S. H. Meijsing, “Identification and characterization of DNA sequences that prevent glucocorticoid receptor binding to nearby response elements,” *Nucleic Acids Research,* vol. 44, no. 13, pp. 6142-6156, 2016.

[46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions,” pp. 1-9.

[47] Y. Kim, “Convolutional neural networks for sentence classification,” *arXiv preprint arXiv:1408.5882,* 2014.

[48] C. H. Zheng, D. S. Huang, L. Zhang, and X. Z. Kong, “Tumor clustering using nonnegative matrix factorization with gene selection,” *IEEE Transactions on Information Technology in Biomedicine A Publication of the IEEE Engineering in Medicine & Biology Society,* vol. 13, no. 4, pp. 599-607, 2009.

[49] S. P. Deng, L. Zhu, and D. S. Huang, "Predicting hub genes associated with cervical cancer through gene co-expression networks,” *IEEE Computer Society Press*, 2016.

[50] D. S. Huang, and C. H. Zheng, “Independent component analysis-based penalized discriminant method for tumor classification using gene expression data,” *Bioinformatics,* vol. 22, no. 15, pp. 1855-62, 2006.

[51] D. S. Huang, and W. Jiang, “A general CPL-AdS methodology for fixing dynamic parameters in dual environments,” *IEEE Transactions on Systems Man & Cybernetics Part B,* vol. 42, no. 5, pp. 1489-1500, 2012.

[52] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks,” *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics,* pp. 315-323, 2011.

[53] T. L. Bailey, and C. Elkan, "The value of prior knowledge in discovering motifs with MEME,” *Ismb*, pp. 21-29, 1995.

[54] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research,* vol. 15, no. 1, pp. 1929-1958, 2014.

[55] C. Fletez-Brant, D. Lee, A. S. McCallion, and M. A. Beer, “kmer-SVM: a web server for identifying predictive regulatory sequence features in genomic data sets,” *Nucleic Acids Research*, vol. 41, no. W1, pp. W544-W556, 2013.

[56] D. Lee, D. U. Gorkin, M. Baker, B. J. Strober, A. L. Asoni, A. S. McCallion, and M. A. Beer, “A method to predict the impact of regulatory variants from DNA sequence,” *Nature Genetics,* vol. 47, no. 8, pp. 955-961, 2015.

[57] Y. Orenstein, and R. Shamir, “A comparative analysis of transcription factor binding models learned from PBM, HT-SELEX and ChIP data,” *Nucleic Acids Research,* vol. 42, no. 8, pp. e63-e63, 2014.

[58] Z. Yao, K. L. MacQuarrie, A. P. Fong, S. J. Tapscott, W. L. Ruzzo, and R. C. Gentleman, “Discriminative motif analysis of high-throughput dataset,” *Bioinformatics,* vol. 30, no. 6, pp. 775-783, 2013.

[59] C.H. Zheng, L. Zhang, T.Y. Ng, C. K. Shiu, and D.S. Huang, “Metasample-based sparse representation for tumor classification,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 8, no. 5, pp. 1273-1282, 2011.

[60] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognition Letters,* vol. 27, no. 8, pp. 861-874, 2006.

[61] X. Glorot, and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks,” *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics,* pp. 249-256, 2010.

[62] M. D. Zeiler, “ADADELTA: an adaptive learning rate method,” *arXiv preprint arXiv:1212.5701*, 2012.

[63] S. Zagoruyko, and N. Komodakis, “Wide residual networks,” *arXiv preprint arXiv:1605.07146*, 2016.

[64] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation,” *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP),* pp. 1532-1543, 2014.

Qin-Hu Zhang received his M.S. degree in communication and information system from Yunnan University, Kunming, China, in 2015. Now he is pursuing the Ph.D. degree in computer science and technology at Tongji University, China. His research interests include Bioinformatics, machine learning, and deep learning.

Lin Zhu obtained his Ph.D. degree in pattern recognition and intelligent system from University of Science and Technology of China (USTC), Hefei, China, in 2013. Now, he is an associate researcher in the college of Electronics and Information Engineering, Tongji University, China. His research interests include Bioinformatics, latent feature learning, dimensionality reduction, and large-scale learning.

De-Shuang Huang received the B.Sc., M.Sc. and Ph.D. degrees all in electronic engineering from Institute of Electronic Engineering, Hefei, China, National Defense University of Science and Technology, Changsha, China and Xidian University, Xian, China, in 1986, 1989 and 1993, respectively. During 1993-1997 period he was a postdoctoral student respectively in Beijing Institute of Technology and in National Key Laboratory of Pattern Recognition, Chinese Academy of Sciences, Beijing, China. In Sept, 2000, he joined the Institute of Intelligent Machines, Chinese Academy of Sciences as the Recipient of “Hundred Talents Program of CAS”. In September 2011, he entered into Tongji University as Chaired Professor. From Sept 2000 to Mar 2001, he worked as Research Associate in Hong Kong Polytechnic University. From Aug. to Sept. 2003, he visited the George Washington University as visiting professor, Washington DC, USA. From July to Dec 2004, he worked as the University Fellow in Hong Kong Baptist University. From March, 2005 to March, 2006, he worked as Research Fellow in Chinese University of Hong Kong. From March to July, 2006, he worked as visiting professor in Queen’s University of Belfast, UK. In 2007, 2008, 2009, he worked as visiting professor in Inha University, Korea, respectively. At present, he is the Director of Institute of Machines Learning and Systems Biology, Tongji University. Dr. Huang is currently IAPR Fellow and a senior member of the IEEE. He has published over 180 journal papers. His current research interest includes Bioinformatics, pattern recognition and machine learning.

1545-5963 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.