Effective identification of sequence patterns via a new convolutional model with adaptively learned kernels

Jing-Yi Li[1,#], Shen Jin[1,2,#], Xin-Ming Tu[1], Yang Ding[1,3,*], and Ge Gao[1,*]

[1] Biomedical Pioneering Innovation Center (BIOPIC) & Beijing Advanced Innovation Center for Genomics (ICG), Center for Bioinformatics (CBI), and State Key Laboratory of Protein and Plant Gene Research at School of Life Sciences, Peking University, Beijing, 100871, China

[2] Computational Biology Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, 15213

[3] Beijing Institute of Radiation Medicine, Beijing, 100850, China

[#] These authors contributed equally to this work.

* To whom correspondence should be addressed. Email: gaog@mail.cbi.pku.edu.cn, dingy@mail.cbi.pku.edu.cn

Abstract

Motif identification is among the most classical and essential computational tasks for bioinformatics and genomics. Here we propose a novel convolution-based model, Variable CNN (vCNN), for effective motif identification in high-throughput omics data based on dynamic learning of kernel length. Multiple empirical evaluations well demonstrate vCNN's superior performance in not only identification performance but also hyperparameter robustness. All source code and data are freely available on GitHub (https://github.com/gao-lab/vCNN) for academic usage.

Introduction

Recurring sequence motifs (Achar and Sætrom, 2015; Kulakovskiy and Makeev, 2013) have been well demonstrated to exert or regulate important biological functions, such as protein binding (Stormo, 2015), transcription initiation (Kadonaga, 2012), alternative splicing (Blencowe, 2000), subcellular localization (Zhang, et al., 2014), translation control (Zucchelli, et al., 2015), and microRNA targeting (Thomson and Dinger, 2016). Effectively and efficiently identifying these motifs in massive omics data is a critical first step for follow-up investigations.

Various computational tools have been developed to identify sequence motifs via word-based and profile-based models (Das and Dai, 2007; Lihu and Holban, 2015; Liu, et al., 2018; Tran and Huang, 2014; Zambelli, et al., 2013). Word-based tools start with a fixed-length and conservative segment and then perform a global search and comparison on each nucleotide; such tools include DREME (Bailey, 2011), Fmotif (Jia, et al., 2014), RSAT peak motifs (Thomas-Chollier, et al., 2012), SIOMICS (Ding, et al., 2015; Ding, et al., 2014), and Discover (Maaskola and Rajewsky, 2014). While these tools can theoretically approach the globally optimal solution, they suffer from high computational complexity when applied to data with complex motifs or large-scale datasets (Das and Dai, 2007). Profile-based tools attempt to find representative motifs by heuristically fine-tuning a series of possible motifs, either generated from a subset of input data or randomly chosen (Ikebata and Yoshida, 2015; Kulakovskiy, et al., 2010; Sharov and Ko, 2009) (Bailey, et al., 2006; Machanick and Bailey, 2011), leading to a faster (but probably sub-optimal) motif calling.

Several convolutional neural network (CNN)-based tools have been proposed recently as a more scalable approach for identifying motifs. Alipanahi et al. developed DeepBind to identify protein binding motifs from large-scale ChIP-Seq datasets (Alipanahi, et al., 2015) by treating each convolutional kernel as an individual motif scanner and discriminating motif-containing sequences from others based on the

output of all kernels. Most, if not all, subsequent CNN-based models have followed DeepBind's kernel-as-motif-scanner strategy, and these models have been used to handle input datasets of enormous volumes in various settings (Angermueller, et al., 2017; Kelley, et al., 2018; Wang, et al., 2018; Zhang, et al., 2018; Zhou, et al., 2018). However, the inherent fixed-kernel design of CNNs also hinders effective identification (Han, et al., 2018; Yin and Schütze, 2016; Zhang, et al., 2018) for *bona fide* sequence patterns, which are usually of various lengths and unknown *a priori* and often function combinatorically (Lambert, et al., 2018; Reiter, et al., 2017).

Here, we propose a novel neural network architecture called Variable CNN (vCNN), which learns the kernel length directly from the data. Evaluations based on both simulations and real-world datasets show that vCNN outperforms canonical CNNs not only accuracy of motif identification but also in model (hyper)parameter robustness, making it an ideal option for the *ab initio* discovery of motifs from high-throughput datasets. All source codes are publicly available on GitHub (https://github.com/gao-lab/vCNN).

Methods

Design and implementation

vCNN (Fig. 1A) is designed to be equipped with a trainable "mask" to adaptively tune the effective length of the kernel during training. Specifically, the "mask" for the z-th kernel w^z, $f^*(w_m^z) := f^*((w_m^{z,0}, w_m^{z,1}))$, is a matrix with the same shape (i.e., $L_k \times 4$, where L_k is the kernel length) as this kernel (Fig. 1B), and the mask boundaries are parameterized by two scalars, $w_m^{z,0}$ and $w_m^{z,1}$:

$$f^*\left(\left(w_m^{z,0}, w_m^{z,1}\right)\right)[i,j] := \frac{1}{1+e^{-(i-1)+w_m^{z,0}}} + \frac{1}{1+e^{(i-1)-w_m^{z,1}}} - 1 \quad (5)$$

As the above equation implies, when i falls outside the boundaries (i.e., $i < w_m^{z,0}$ or $i > w_m^{z,1}$), the value of $f^*\left(\left(w_m^{z,0}, w_m^{z,1}\right)\right)[i,j]$ diminishes to zero; the subsequent masking operation $f^*(w_m^z) \circ w^z$ (where $\circ$ is the Hadamard product) (Fig. 1C) will then effectively replace all kernel elements outside the boundaries with zero, producing a masked kernel that can be used as an ordinary kernel in a classical convolutional layer (Fig. 1D).

To make $w_m^{z,0}$ and $w_m^{z,1}$ converge faster during training, we combine the binary cross-entropy (BCE) loss with a sum of masked Shannon losses (MSLs) across all kernel masks. Below, we provide a detailed explanation of how this works. We have the following total loss $L(D)$:

$$\begin{aligned} L(D) &= \mathrm{BCE} + \lambda \cdot \textstyle\sum_z MSL(w^z, w_m^z) \\ &= \textstyle\sum_{c=1}^{N} \left(y_c \, log \, \hat{y}_c + (1-y_c) \, log \left(1-\hat{y}_c\right)\right) + \lambda \cdot \sum (f^*(w_m^z) \cdot H(P_z) - \\ &\qquad threshold) \quad (6) \end{aligned}$$

where $H(P_z) := \sum_{i=1}^{L_k}(-\sum_{j=1}^{4}((P_z)[i,j]) log_2((P_z)[i,j])$ is the sum of the Shannon entropy across all nucleotide positions of P_z, the position weight matrix (PWM) learned by the z-th kernel. For precision, we set $P_z = P(w^z, b = 2)$ following the exact kernel-to-PWM transformation specified by Ding et al. (Ding, et al., 2018).

One can then immediately deduce the formula for updating the left boundary $w_m^{z,0}[\mathrm{t}-1]$ at time step *t* via gradient descent with learning rate *r* (*r*>0):

$$w_m^{z,0}[t] = w_m^{z,0}[t-1] - r \cdot \frac{\partial L(D)}{\partial w_m^{z,0}[t-1]}$$

$$= w_m^{z,0}[t-1] - r \cdot \frac{\partial BCE}{\partial w_m^{z,0}[t-1]} - r \cdot \frac{\partial MSL(w^z, w_m^z[t-1])}{\partial w_m^{z,0}[t-1]}$$

$$= w_m^{z,0}[t-1] - r \cdot \frac{\partial BCE}{\partial w_m^{z,0}[t-1]} + r \sum_{i=0}^{L_k-1} \frac{e^{-(i-1)+w_m^{z,0}[t-1]}}{(1+e^{-(i-1)+w_m^{z,0}[t-1]})^2} 2(H\big(P(w^z,2)\big)_i - threshold)$$

Similarly, for the right boundary $w_m^{z,1}[t-1]$, we have:

$$w_m^{z,1}[t] = w_m^{z,1}[t-1] - r \cdot \frac{\partial L(D)}{\partial w_m^{z,1}[t-1]}$$

$$= w_m^{z,1}[t-1] - r \cdot \frac{\partial BCE}{\partial w_m^{z,1}[t-1]} - r \sum_{i=0}^{L_k-1} \frac{e^{(i-1)-w_m^{z,1}[t-1]}}{(1+e^{(i-1)-w_m^{z,1}[t-1]})^2} 2(H\big(P(w^z,2)\big)_i - threshold)$$

The Shannon entropies of all kernel positions far from the mask boundaries ($w_m^{z,0}, w_m^{z,1}$) contribute little to the final derivatives. Therefore, if most boundary-flanking kernel positions have a low Shannon entropy(i.e., a high information content; defined by $H\big(P(w^z,2)\big)_i - threshold < 0$), then the masked Shannon loss (MSL) will help push the boundaries outwards (i.e., $\frac{\partial MSL(w^z, w_m^z[t-1])}{\partial w_m^{z,0}[t-1]} < 0$ and $\frac{\partial SL(w^z, w_m^z[t-1])}{\partial w_m^{z,1}[t-1]} > 0$) during gradient descent, just as if the current mask is too narrow to span all informative positions. Likewise, if most boundary-flanking kernel positions have a high Shannon entropy, or a low information content (i.e. a low information content; defined by $H\big(P(w^z,2)\big)_i - threshold > 0$), then the MSL will help push the boundaries inwards (i.e., $\frac{\partial MSL(w^z, w_m^z[t-1])}{\partial w_m^{z,0}[t-1]} > 0$ and $\frac{\partial SL(w^z, w_m^z[t-1])}{\partial w_m^{z,1}[t-1]} < 0$), just as if the current mask is too broad to exclude some noisy, noninformative positions. We empirically set the threshold to 1.2, the value of Shannon entropy when one of Prob(A), Prob(C), Prob(G) and Prob(T) is around 0.74 and the rest three are identical to each other.

To speed up computation in practice, we approximate the sum by ignoring most small, outside-mask kernel positions and retaining only those terms with

or .

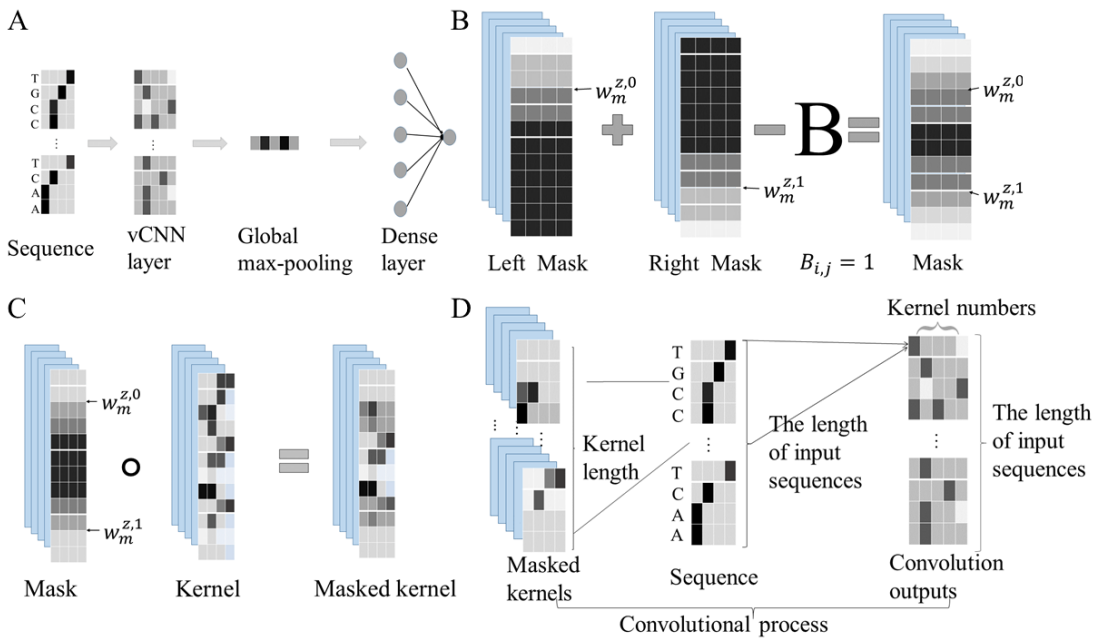


Fig. 1. The design of vCNN. (A) shows the structure diagram of our vCNN-based model, which consists of a vCNN layer, a global max-pooling layer and a multilayer perceptron. (B-D) show the design of the vCNN layer. The original kernel is first "masked" by a mask matrix (B) using the Hadamard product (C), and then it is treated as an ordinary kernel to convolve the input sequence (D).

Benchmark

In each iteration of benchmarking a certain case, we (1) simulated a particular pair of training and validation datasets (see below for more details); (2) used the training dataset to train first a vCNN-based model with certain hyperparameter settings and then a classical CNN-based model with a model structure identical to that of the vCNN-based model, except that the kernel length of the convolutional layer cannot be dynamically adjusted; and (3) obtained the AUC values of these two models on the validation dataset. Iterating over all possible hyperparameter settings yielded a series of AUC values for both the vCNN-based model and the CNN-based model for the case at hand.

Each positive sequence in the training and validation datasets is a random sequence with a "signal sequence" inserted at a random location, and each negative sequence is a fully random sequence. The "signal sequence" is a sequence fragment generated from one of the motifs for the current case (Table 1). For the cases of 2, 4, 6 and 8 motifs, new motifs were introduced incrementally (i.e., all motifs in "2 motifs" were also included in "4 motifs", all motifs in "4 motifs" were also included in "6 motifs", and all motifs in "6 motifs" were also included in "8 motifs").

Dataset name	2 motifs	4 motifs	6 motifs	8 motifs
Motif composition	MA0234.1 (length=6), MA0963.1 (length=8)	MA0234.1 (length=6), MA0963.1 (length=8), MA0626.1 (length=10), MA0667.1 (length=10)	MA0234.1 (length=6), MA0963.1 (length=8), MA0626.1 (length=10), MA0667.1 (length=10), MA1146.1 (length=15), MA1147.1 (length=15)	MA0234.1 (length=6), MA0963.1 (length=8), MA0626.1 (length=10), MA0667.1 (length=10), MA1146.1 (length=15), MA1147.1 (length=15), MA0009.2 (length=16), MA0470.1 (length=11)

Dataset name	TwoDiffMotif1	TwoDiffMotif2	TwoDiffMotif3
Motif composition	MA0138.2 (length=21), MA0157.2 (length=8)	MA1046.1 (length=9), MA1148.1 (length=18)	MA0326.1 (length=8), MA0556.1 (length=15)

Table 1. Motifs used to generate each dataset. All motifs were derived from JASPAR (Khan, et al., 2017).

To further demonstrate the performance of vCNN in the real world, we downloaded 690 ENCODE ChIP-Seq-based training and test datasets representing the DNA binding profile of various transcription factors and other DNA-binding proteins from http://cnn.csail.mit.edu/motif_discovery/. For direct comparison between vCNN and the CNNs of DeepBind and Zeng et al.'s model (Zeng, et al., 2016), vCNN-based models were implemented by replacing the convolutional layer of each model

considered for comparison with a vCNN layer only (with exactly the same hyperparameters for the number of kernels and the initial kernel length). The remaining details of the training procedure followed those of the corresponding CNN models (Alipanahi, et al., 2015; Zeng, et al., 2016).

We followed the protocol proposed by Zhang et al. (Zhang, et al., 2015) to assess the accuracy of the extracted representative motifs based on the ENCODE CTCF ChIP-Seq datasets (ENCODE Project Consortium, 2012). In brief, for each motif discovery tool and each ChIP-Seq dataset, we located the motif-containing sequence fragments for candidate motifs discovered by this tool, checked whether they overlapped with the ChIP-Seq peaks, and reported the ratio of overlapping fragments to all fragments as the accuracy of the tool on this dataset (see Supplementary Fig. 2 for more details).

Results

vCNN-based models effectively identify motifs

We first present direct comparisons between vCNN-based and CNN-based models based on multiple simulated datasets (see the Methods for more details). It is clear that a vCNN-based model performs better when more motifs are introduced (2 motifs to 8 motifs in Fig. 2A) and with increased heterogeneity of the motif length (2 motifs v.s. TwoDiffMotif1/2/3 in Fig. 2A); notably, this superior performance of vCNN-based models applies to most datasets (rather than being a biased observation due to extremely large performance differences on only a few datasets) because we computed the AUC difference for each dataset separately, suggesting that vCNN's superiority over the classical CNN approach is independent of specific datasets. In addition, the vCNN-based models show a smaller mean standard error of the AUC (5.98E-03) than the CNN-based models (5.87E-02) among different parameter initializations, suggesting that vCNN also shows better robustness (Fig. 2B).

A

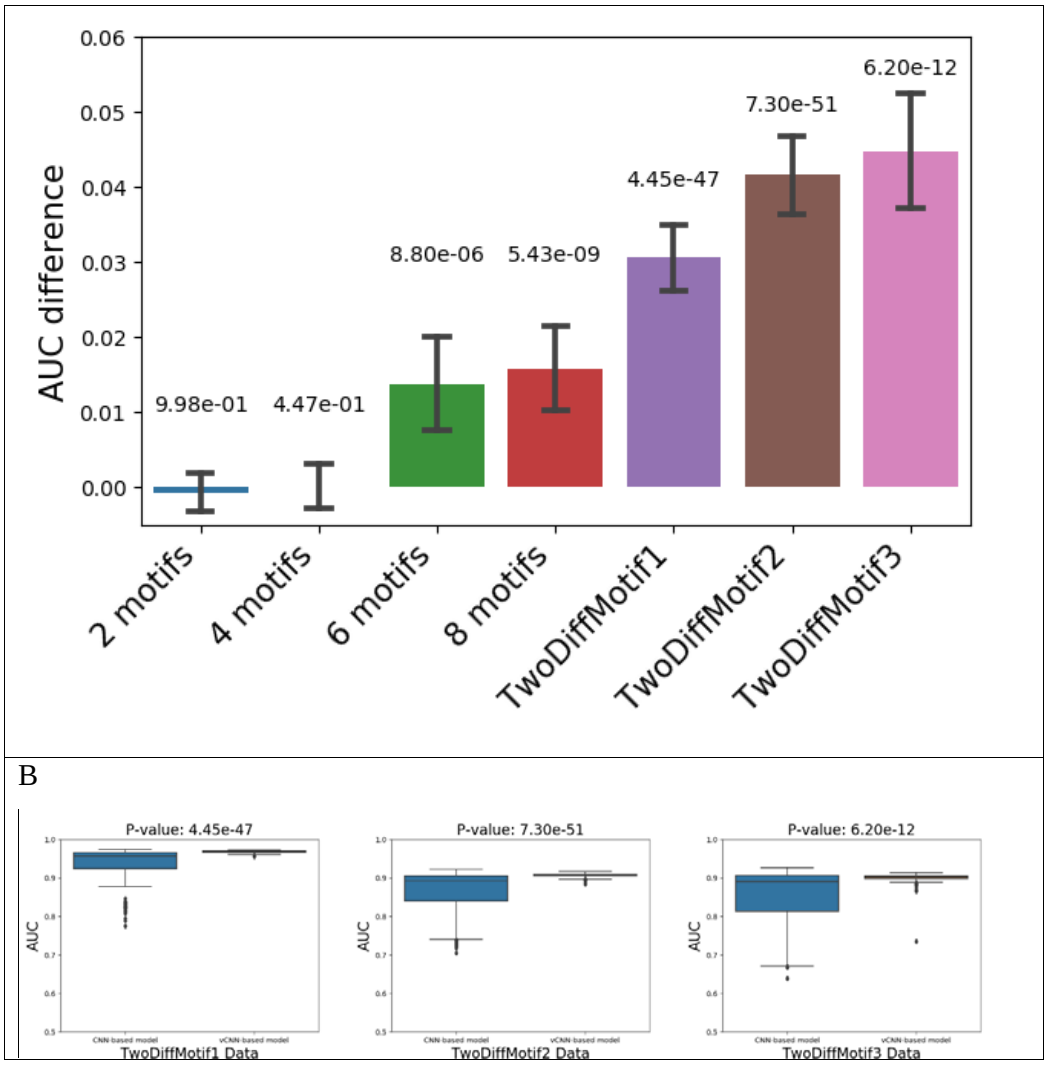


Fig. 2. vCNN-based models outperform classical CNN-based models for motifs of different lengths. The distribution of the vCNN-based models' AUC minus the CNN-based models' AUC is shown in (A), and the value above each bar of the plot is the p-value of the Wilcoxon rank sum test (the null hypothesis is that the AUC of the CNN-based model is equal to or higher than that of the vCNN-based model). Three cases with two JASPAR motifs with much larger length differences than the 2 motifs case are shown in (B), along with the p-values of the Wilcoxon rank sum test with the null hypothesis that the AUC of the CNN-based model is equal to or larger than that of the vCNN-based model. The kernel lengths were drawn from the set {6, 8, 10, 12, 14, 16, 18, 20}, the numbers of kernels were drawn from the set {64, 96, 128}, and each model structure was tested on 16 random seeds.

These findings further compel us to suspect that vCNN-based models will perform better than CNN-based models on real-world cases with combinatorial regulation. To test this suspicion, we compared a vCNN-based model with the DeepBind model (Alipanahi, et al., 2015) on 690 ENCODE ChIP-Seq datasets. The vCNN-based model showed a significantly improved performance (Fig. 3; Wilcoxon rank sum test, p =1.4e-13, single-tailed, with mean AUC 0.894 vs. 0.8298); notably, the AUC improved when switching from the CNN-based model to the vCNN-based model on 680 datasets, and on 73 datasets, this improvement is from less than 0.6 to above 0.8 (Fig. 3, points in red square). We also compared a vCNN-based model with the improved network implemented by Zeng (Zeng, et al., 2016). The results show that the vCNN-based model achieves overall better performance than the optimized CNN-based model (Wilcoxon rank sum test, p = 0.017, single-tailed; see Supplementary Fig. 3 for more details).

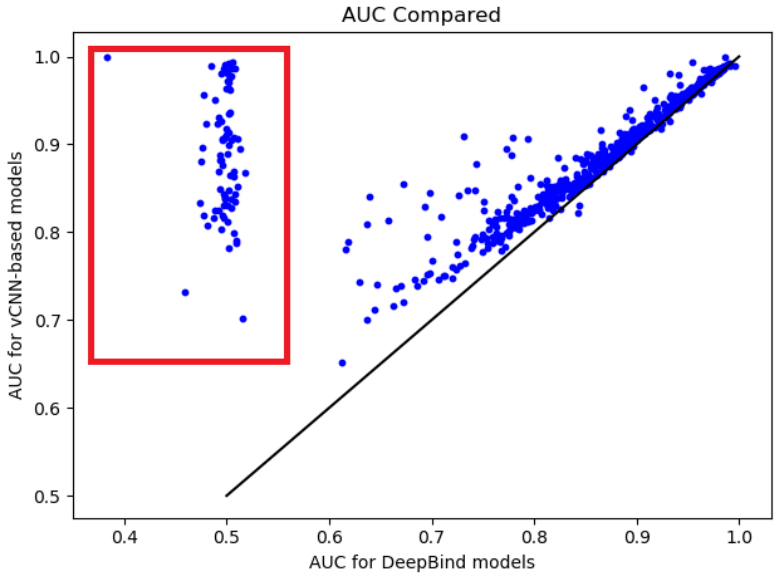


Fig. 3. vCNN-based models outperform DeepBind models (Alipanahi, et al., 2015) on 690 real-world ENCODE ChIP-Seq datasets. The points on the black line represent equal AUC values for both models. The points highlighted by the square in the figure represent datasets for which the vCNN-based model improved the AUC from less than 0.6 for the DeepBind model to above 0.8. The hyperparameter space is as follows: the kernel lengths were drawn from the set {10, 17, 24}, the numbers of kernels were drawn from the set {96, 128}, and each model structure was tested on 8 random seeds.

vCNN-based models discover motifs from real-world sequences more accurately and faster

We further evaluated vCNN-based models' performance for *ab initio* motif discovery. In brief, for a particular trained vCNN-based model, we first selected kernels with dense layer weights higher than a predetermined baseline (defined as mean minus standard deviation of all dense layer weights) and then extracted and aligned the kernels' corresponding segments to compute the representative PWM (Fig. 4A). We then compared the accuracy of the recovery of ChIP-Seq peaks (see the Methods for details) by the vCNN-based motif discovery and other motif discovery tools across all these ChIP-Seq (ENCODE Project Consortium, 2012) datasets.

As shown in Fig. 4B, the vCNN-based models outperform DREME (Bailey, 2011) on all datasets, CisFinder (Sharov and Ko, 2009) on 95 datasets and MEME-ChIP (Machanick and Bailey, 2011) on 87 datasets (out of 100 CTCF datasets within the Chip-Seq cohort); this finding also holds when each motif was considered separately (Fig. 4B), indicating that the superior performance of vCNN holds for most of these datasets. In addition, vCNN runs much faster on large datasets (46,726 sequences; see Fig. 4C).

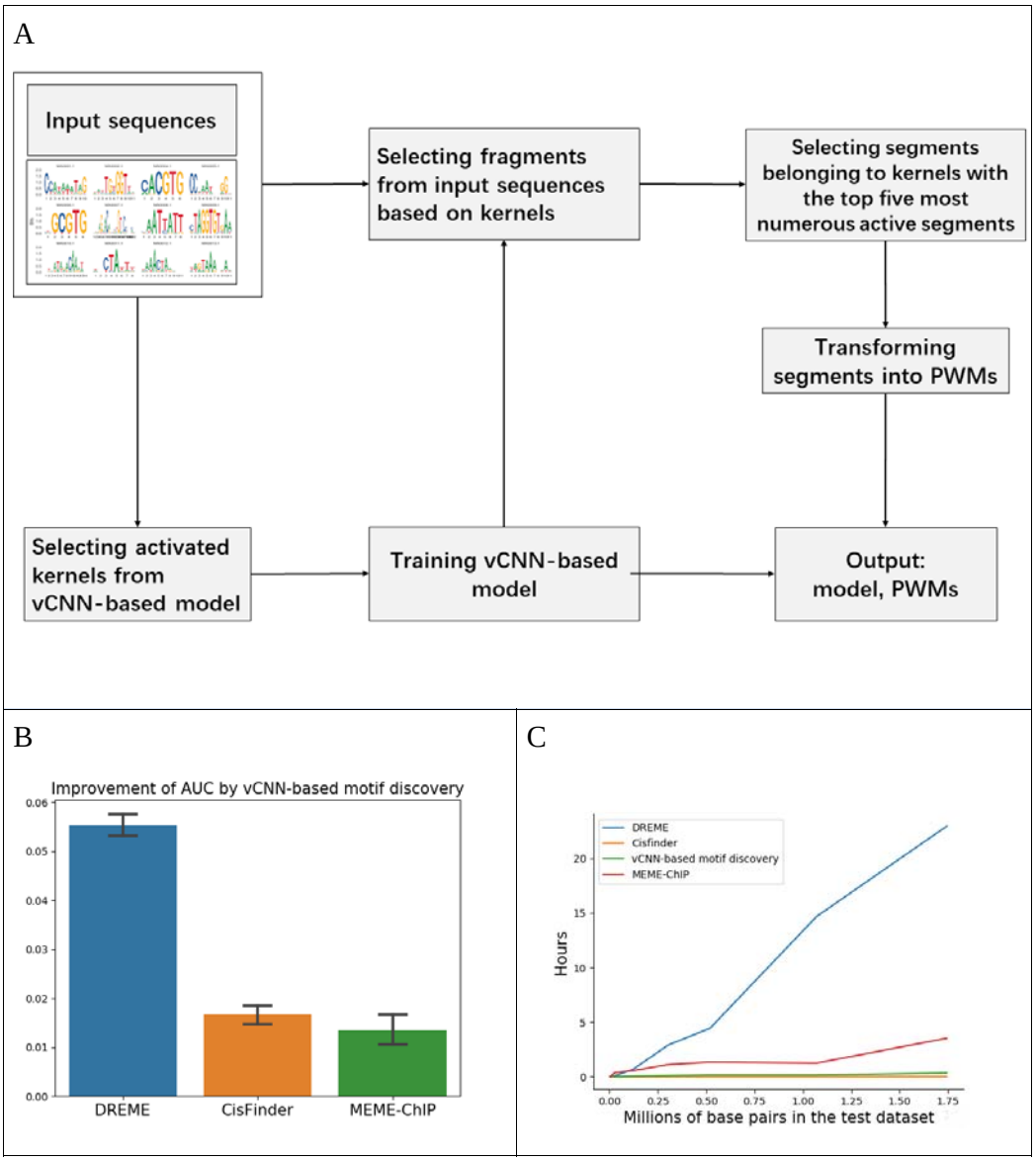


Fig. 4. vCNN helps discover motifs more accurately and faster. (A) shows the process of calling a representative motif from a trained vCNN model. (B) shows the difference in accuracy, defined as the accuracy of vCNN-based motif discovery minus the accuracy of the motif discovery algorithm shown on the x-axis on the same dataset. MEME (Bailey, et al., 2006) failed to complete within a reasonable time (~50% datasets remained unfinished even after running for 1.5 weeks with 2,000 cores, amounting to 504,000 CPU hours) for these datasets, and its results are thus not listed here. (C) shows the time cost of each motif discovery algorithm as a function of millions of base pairs in the test dataset.

Discussion

Kernel length affects the performance of convolution-based models

While such an effect has been suspected for a long time, no previous study has systematically investigated whether (and how) kernel length affects convolution-based models' performance, especially when the underlying signals are of mixed lengths (Lambert, et al., 2018; Reiter, et al., 2017). In computer vision, researchers have empirically noticed that different kernel lengths in CNNs lead to differences in performance; for example, Han (Han, et al., 2018) reported that when a CNN is applied for facial action unit recognition (FAUR), changes in the CNN's kernel size will affect the performance of the model. Moreover, Han's results show that for FAUR, the optimal kernel size is different on different datasets, and there is no overall tendency for either a large or small kernel size to be preferred. Thus, Inception model (Szegedy, et al., 2015) tries to combine multiple kernels with different sizes for boosting global performance for various computation vision tasks (Ioffe and Szegedy, 2015; Szegedy, et al., 2016).

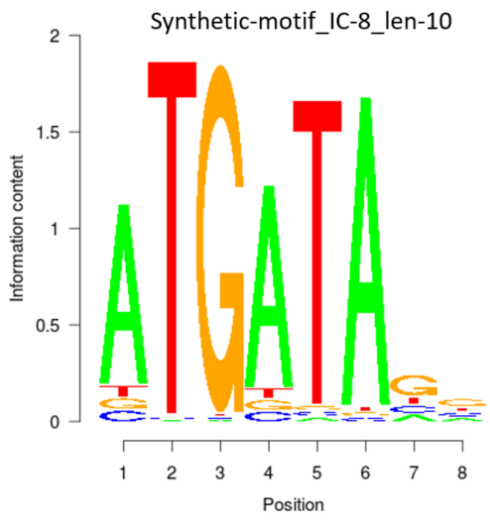
In addition to empirical assessments, we can further theoretically model the relationship between kernel length (L_k) and model performance using a probability-based scoring function (Fig. 5A; see the Supplementary Notes for the full mathematical treatment). Basically, for each combination of (1) a real motif $\mathcal{M}$, (2) the proportion (P_{ideal}) of the kernel contributed by this real motif (where the kernel is defined as $P_{ideal} * \mathcal{M} + (1 - P_{ideal}) * R$, with R being a random matrix representing noise), and (3) L_k , this scoring function computes the expected probability across all possible kernels that the kernel's convolution with an arbitrary $\mathcal{M}$-containing sequence will take its maximal value at the position at which the motif is inserted. A high score for a certain L_k indicates that kernels of this length can easily

distinguish $\mathcal{M}$-containing sequences from other sequences, thus leading to good performance of the final CNN model. The results of applying this scoring function to various cases (Fig. 5B-G) clearly demonstrate that the scoring function helps to quantify the "goodness" of a particular L_k for identifying a given motif under a particular P_{ideal}.

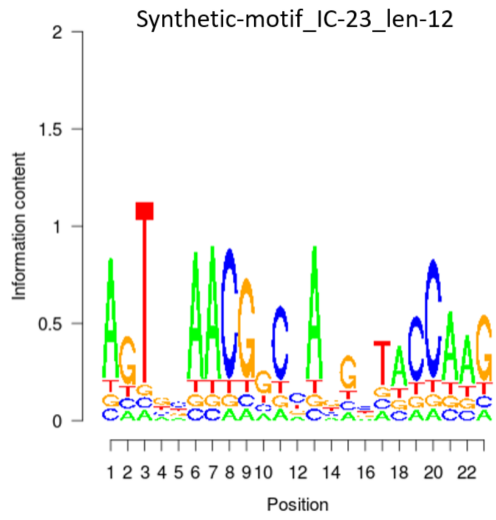
A

$$Score(L_k|P_{ideal}, \mathcal{M}) \coloneqq E_K(P_{real}) = E_K(Prob(X_s > X_n|K, \mathcal{M}))$$

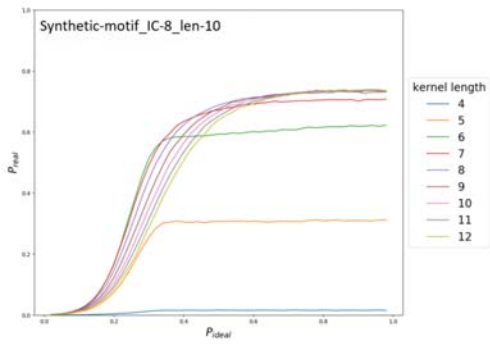
B



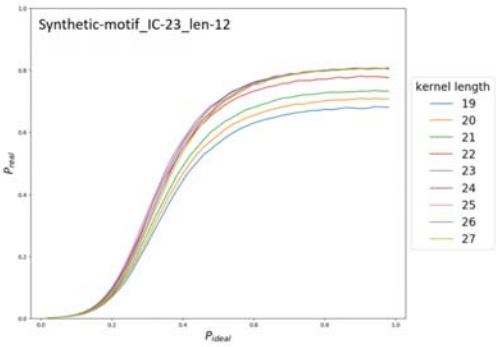
C



D



E



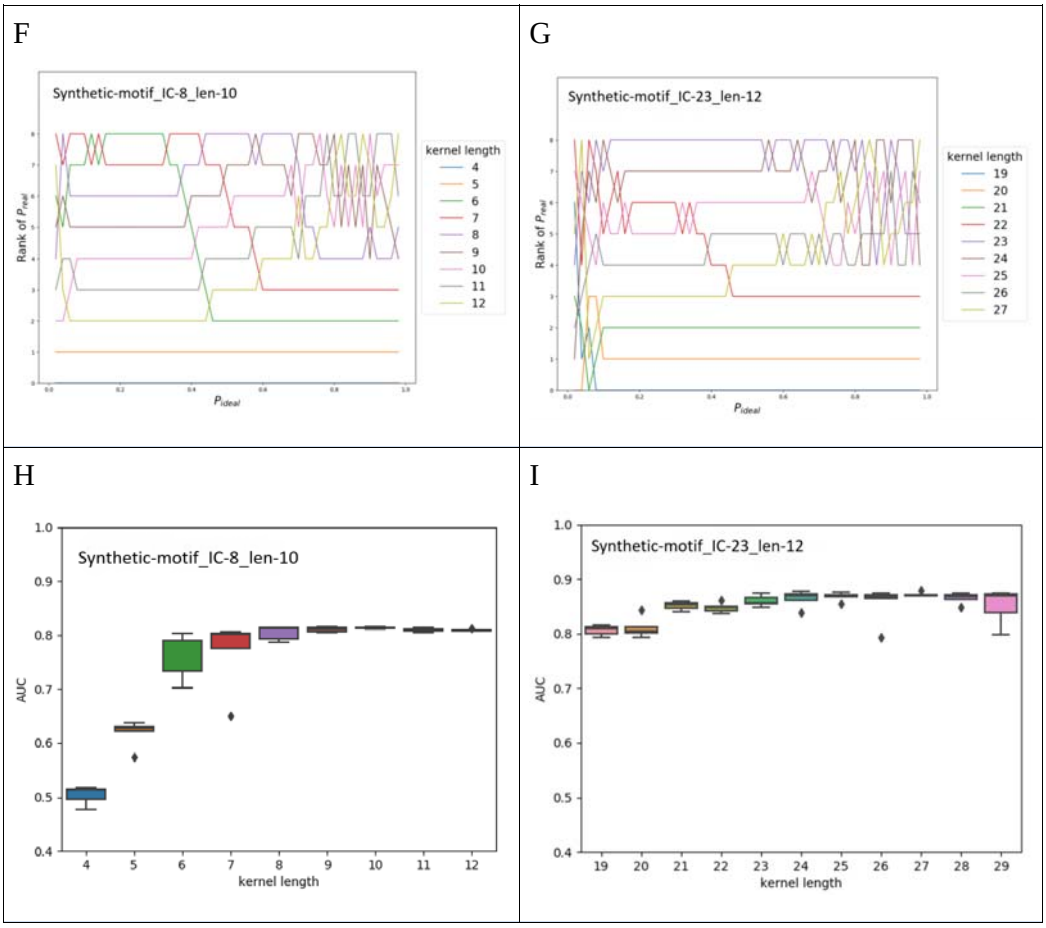


Fig. 5. Theoretical modeling of the kernel in terms of the underlying real motif helps to evaluate the "goodness" of a particular kernel length for identifying the real motif. (A-C) We have developed a scoring function (A) for calculating the kernel length and have tested it on two example motifs, one of which (B) is shorter and more conserved than the other (C). (D-I) The kernel length does affect the model performance, although in a rather complicated way even in such a simplistic setting: for the first, shorter motif, the kernel length with the largest score depends on P_{ideal} (D and F), while for the second, longer motif, the kernel length with the largest score is 23 (i.e., the motif length) for most P_{ideal} values if we ignore the differences arising from numerical error (E and G). Similarly, we observe a complicated relationship between the kernel length and the average AUC value of a CNN model in practice (H and I).

vCNN enables the implementation of a model with a dynamic kernel length

The high scalability of CNN-based models is critical for massive omics data. However,
the inherent fixed-kernel design of canonical CNNs hinders effective identification (Han, et al., 2018; Yin and Schütze, 2016; Zhang, et al., 2018) for *bona fide* sequence patterns, which are usually of various lengths and unknown *a priori* and often function combinatorically (Lambert, et al., 2018; Reiter, et al., 2017).

Inspired by the theoretical model described above, we designed and implemented a novel convolutional model, vCNN, which adaptively tunes the kernel lengths at run time without losing scalability (Supplementary Fig. 1). In addition to the performance improvement it offers, vCNN's capability of run-time length tuning could drastically mitigate the time cost relative to the time required for canonical CNN hyperparameter optimization.

To facilitate its application in various fields, we have implemented vCNN as a new type of convolutional layer in Keras (https://github.com/gao-lab/vCNN) so that users can easily replace existing convolutional layers with vCNN layers (and optionally initialize them with the pretrained kernels of those convolutional layers) to test whether the resulting model performs better.

Further work

We note that the current theoretical analysis reported above relies on prior knowledge of the real motif $\mathcal{M}$, which may not be feasible to obtain for real-world datasets. A possible workaround is to estimate the empirical null distribution of P_{real} over a set of randomly initialized PWMs and kernels and then derive the expectation of P_{real} over the given dataset.

Moreover, although current motifs in CNN-based models are automatically represented by PWMs, this might be an oversimplified representation of the genuine motifs, which may allow insertions and deletions within motifs (e.g., the HMM motifs from Pfam (El-Gebali, et al., 2018) and Rfam (Kalvari, et al., 2017)). While RNN models are expected to be able to learn such motifs (Liu, 2017; Liza and Grzes, 2019; Min, et al., 2019; Vazhayil and KP, 2018), the interpretation of such models is still challenging.

Funding

This work was supported by funds from the National Key Research and Development Program (2016YFC0901603), the China 863 Program (2015AA020108), as well as the State Key Laboratory of Protein and Plant Gene Research and the Beijing Advanced Innovation Center for Genomics (ICG) at Peking University. The research of G.G. was supported in part by the National Program for the Support of Top-notch Young Professionals.

Acknowledgments

The authors would like to thank Drs. Cheng Li, Letian Tao, Minghua Deng, Zemin Zhang, Jian Lu and Liping Wei at Peking University for their helpful comments and suggestions during the study. The analysis was supported by the High-performance Computing Platform of Peking University, and we thank Dr. Chun Fan and Yin-Ping Ma for their assistance during the analysis.

References

Achar, A. and Sætrom, P. RNA motif discovery: a computational overview. *Biology direct* 2015;10(1):61.

Alipanahi, B., *et al.* Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature biotechnology* 2015;33(8):831.

Angermueller, C., *et al.* DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. *Genome biology* 2017;18(1):67.

Bailey, T.L. DREME: motif discovery in transcription factor ChIP-seq data. *Bioinformatics* 2011;27(12):1653-1659.

Bailey, T.L., *et al.* MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic acids research* 2006;34(suppl_2):W369-W373.

Blencowe, B.J. Exonic splicing enhancers: mechanism of action, diversity and role in human genetic diseases. *Trends in biochemical sciences* 2000;25(3):106-110.

Das, M.K. and Dai, H.-K. A survey of DNA motif finding algorithms. In, *BMC bioinformatics*. Springer; 2007. p. S21.

Ding, J., *et al.* Systematic discovery of cofactor motifs from ChIP-seq data by SIOMICS. *Methods* 2015;79:47-51.

Ding, J., Hu, H. and Li, X. SIOMICS: a novel approach for systematic identification of motifs in ChIP-seq data. *Nucleic acids research* 2014;42(5):e35-e35.

Ding, Y., *et al.* An exact transformation of convolutional kernels enables accurate identification of sequence motifs. *bioRxiv* 2018:163220.

El-Gebali, S., *et al.* The Pfam protein families database in 2019. *Nucleic Acids Research* 2018;47(D1):D427-D432.

ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* 2012;489(7414):57.

Han, S., *et al.* Optimizing filter size in convolutional neural networks for facial action unit recognition. In, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018. p. 5070-5078.

Ikebata, H. and Yoshida, R. Repulsive parallel MCMC algorithm for discovering diverse motifs from large sequence sets. *Bioinformatics* 2015;31(10):1561-1568.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* 2015.

Jia, C., *et al.* A new exhaustive method and strategy for finding motifs in ChIP-enriched regions. *PLoS One* 2014;9(1).

Kadonaga, J.T. Perspectives on the RNA polymerase II core promoter. *Wiley Interdisciplinary Reviews: Developmental Biology* 2012;1(1):40-51.

Kalvari, I., *et al.* Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic Acids Research* 2017;46(D1):D335-D342.

Kelley, D.R., *et al.* Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome research* 2018;28(5):739-750.

Khan, A., *et al.* JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic acids research* 2017;46(D1):D260-D266.
Kulakovskiy, I.V., *et al.* Deep and wide digging for binding motifs in ChIP-Seq data. *Bioinformatics* 2010;26(20):2622-2623.
Kulakovskiy, I.V. and Makeev, V.J. DNA sequence motif: a jack of all trades for ChIP-Seq data. In, *Advances in protein chemistry and structural biology*. Elsevier; 2013. p. 135-171.
Lambert, S.A., *et al.* The human transcription factors. 2018;172(4):650-665.
Lihu, A. and Holban, Ş. A review of ensemble methods for de novo motif discovery in ChIP-Seq data. *Briefings in bioinformatics* 2015;16(6):964-973.
Liu, B., *et al.* An algorithmic perspective of de novo cis-regulatory motif finding based on ChIP-seq data. *Briefings in bioinformatics* 2018;19(5):1069-1081.
Liu, X. Deep recurrent neural network for protein function prediction from sequence. *arXiv preprint arXiv:1701.08318* 2017.
Liza, F.F. and Grzes, M. Relating RNN layers with the spectral WFA ranks in sequence modelling. 2019.
Maaskola, J. and Rajewsky, N. Binding site discovery from nucleic acid sequences by discriminative learning of hidden Markov models. *Nucleic acids research* 2014;42(21):12995-13011.
Machanick, P. and Bailey, T.L. MEME-ChIP: motif analysis of large DNA datasets. *Bioinformatics* 2011;27(12):1696-1697.
Min, S., *et al.* Pre-Training of Deep Bidirectional Protein Sequence Representations with Structural Information. *arXiv preprint arXiv:1912.05625* 2019.
Poplin, R., *et al.* Creating a universal SNP and small indel variant caller with deep neural networks. *bioRxiv* 2018:092890.
Reiter, F., *et al.* Combinatorial function of transcription factors and cofactors. 2017;43:73-81.
Sharov, A.A. and Ko, M.S. Exhaustive search for over-represented DNA sequence motifs with CisFinder. *DNA research* 2009;16(5):261-273.
Stormo, G.D. DNA motif databases and their uses. *Current protocols in bioinformatics* 2015;51(1):2.15. 11-12.15. 16.
Szegedy, C., *et al.* Going deeper with convolutions. In, *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. p. 1-9.
Szegedy, C., *et al.* Rethinking the inception architecture for computer vision. In, *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. p. 2818-2826.
Thomas-Chollier, M., *et al.* RSAT peak-motifs: motif analysis in full-size ChIP-seq datasets. *Nucleic acids research* 2012;40(4):e31-e31.
Thomson, D.W. and Dinger, M.E. Endogenous microRNA sponges: evidence and controversy. *Nature Reviews Genetics* 2016;17(5):272.
Tran, N.T.L. and Huang, C.-H. A survey of motif finding Web tools for detecting binding site motifs in ChIP-Seq data. *Biology direct* 2014;9(1):4.
Vazhayil, A. and KP, S. DeepProteomics: Protein family classification using Shallow and Deep Networks. *arXiv preprint arXiv:1809.04461* 2018.
Wang, M., *et al.* DeFine: deep convolutional neural networks accurately quantify intensities of transcription factor-DNA binding and facilitate evaluation of functional non-coding variants. *Nucleic acids research* 2018;46(11):e69-e69.
Yin, W. and Schütze, H.J.a.p.a. Multichannel variable-size convolution for sentence classification. 2016.

Zambelli, F., Pesole, G. and Pavesi, G. Motif discovery and transcription factor binding sites before and after the next-generation sequencing era. *Briefings in bioinformatics* 2013;14(2):225-237.
Zeng, H., *et al*. Convolutional neural network architectures for predicting DNA–protein binding. *Bioinformatics* 2016;32(12):i121-i127.
Zhang, B., *et al*. A novel RNA motif mediates the strict nuclear localization of a long noncoding RNA. *Molecular and cellular biology* 2014;34(12):2318-2329.
Zhang, J., Peng, W. and Wang, L. LeNup: learning nucleosome positioning from DNA sequences with improved convolutional neural networks. *Bioinformatics* 2018;34(10):1705-1712.
Zhang, Q., *et al*. High-order convolutional neural network architecture for predicting DNA-protein binding sites. 2018;16(4):1184-1192.
Zhou, J., *et al.* Deep learning sequence-based ab initio prediction of variant effects on expression and disease risk. *Nature genetics* 2018;50(8):1171.
Zucchelli, S., *et al.* SINEUPs: A new class of natural and synthetic antisense long non-coding RNAs that activate translation. *RNA biology* 2015;12(8):771-779.