Revealing Fine Structures of the Retinal Receptive Field by Deep-Learning Networks

Qi Yan, Yajing Zheng, Shanshan Jia, Yichen Zhang, Zhaofei Yu, *Member, IEEE*,
Feng Chen, *Member, IEEE*, Yonghong Tian, *Senior Member, IEEE*,
Tiejun Huang, *Senior Member, IEEE*, and Jian K. Liu

Abstract—Deep convolutional neural networks (CNNs) have demonstrated impressive performance on many visual tasks. Recently, they became useful models for the visual system in neuroscience. However, it is still not clear what is learned by CNNs in terms of neuronal circuits. When a deep CNN with many layers is used for the visual system, it is not easy to compare the structure components of CNNs with possible neuroscience underpinnings due to highly complex circuits from the retina to the higher visual cortex. Here, we address this issue by focusing on single retinal ganglion cells with biophysical models and recording data from animals. By training CNNs with white noise images to predict neuronal responses, we found that fine structures of the retinal receptive field can be revealed. Specifically, convolutional filters learned are resembling biological components of the retinal circuit. This suggests that a CNN learning from one single retinal cell reveals a minimal neural network carried out in this cell. Furthermore, when CNNs learned from different cells are transferred between cells, there is a diversity of transfer learning performance, which indicates that CNNs are cell specific. Moreover, when CNNs are transferred between different types of input images, here white noise versus natural images, transfer learning shows a good performance, which implies that CNNs indeed capture the full computational ability of a single retinal cell for different inputs. Taken together, these results suggest that CNNs could be used to reveal structure components of neuronal circuits, and provide a powerful model for neural system identification.

Index Terms—Convolutional neural network (CNN), deep learning, natural images, parameter pruning, receptive field, retina, transfer learning, visual coding.

I. INTRODUCTION

DEEP convolutional neural networks (CNNs) have been a powerful model for numerous tasks related to system identification in recent years [1]. By training a CNN with a large set of target images, it can achieve the human-level performance for visual object recognition. However, it is still a challenge for understanding the relationship between computation and the underlying network structure of components learned within CNNs [2], [3]. Thus, visualizing, interpreting, and understanding CNNs are not trivial [4].

Inspired by neuroscience studies [5], a typical CNN consists of a hierarchical structure of layers [6], where one of the most important properties for each convolutional (conv) layer is that one can use a conv filter as a feature detector to extract useful information from input images [7], [61]. Therefore, after learning, conv filters are meaningful. The features captured by these filters can be represented in the original natural images [4]. Often, one typical feature shares some similarities with certain image patches of natural images from the training set. These similarities are obtained by using a very large set of specific images with reasonable labels. The benefit of this is that features are relatively universal for one category of objects, which is good for recognition. However, it also causes the difficulty of visualization or interpretation due to the complex nature of natural images, that is, the complex statistical structures of natural images [8]. As a result, the filters learned in CNNs are often not obvious to be interpreted [9].

On the other hand, researchers begin to adapt CNNs for studying the central questions of neuroscience [10], [11]. For example, CNNs have been used to model the ventral visual pathway that has been suggested as a route for visual object recognition starting from the retina to the visual cortex and reaching the inferior temporal (IT) cortex [11]–[14]. The prediction of neuronal responses, in this case, has surprisingly good performance. However, the final output of this CNN model is representing dense computations conducted in

Qi Yan is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: q-yan15@mails.tsinghua.edu.cn).

Yajing Zheng, Shanshan Jia, Yichen Zhang, Zhaofei Yu, Yonghong Tian, and Tiejun Huang are with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: zyj061@pku.edu.cn; jssll00@126.com; zhangyichen2622@outlook.com; yuzf12@pku.edu.cn; yhtian@pku.edu.cn; tjhuang@pku.edu.cn).

Feng Chen is with the Department of Automation, Beijing Innovation Center for Future Chip, LSBDPA Beijing Key Laboratory, Tsinghua University, Beijing 100084, China (e-mail: chenfeng@tsinghua.edu.cn).

Jian K. Liu is with the Centre for Systems Neuroscience, Department of Neuroscience, Psychology and Behaviour, University of Leicester, Leicester LE1 7HA, U.K. (e-mail: jian.liu@leicester.ac.uk).

many layers, which may or may not be relevant to the biological underpinnings of information processing in the brain. Understanding these network components of CNNs is difficult given that the IT cortex part is sitting at a higher level of our visual system [11].

In principle, CNN models can also be applied to early sensory systems where the organization of underlying neuronal circuitry is relatively clear and simple. Thus, one expects knowledge of these neuronal circuits could provide useful and important validation for CNNs. Indeed, a few studies applied CNNs and their variations to earlier visual system, such as the retina [15]–[18], V1 [19]–[25], and V2 [26]. Most of these studies are driven by the goal that the better performance of neural response can be achieved by using either feedforward and recurrent neural networks (or both). These new approaches increase the complexity level of system identification, compared to conventional linear/nonlinear models [27]–[29]. Some of these studies also try to look into the details of network components after learning to see if and how they are comparable to the biological structure of neuronal networks [18], [23], [25].

The retina, compared to other earlier visual systems, has a relatively simple neuronal circuit with three layers of neurons as photoreceptors, bipolar cells (BCs), and ganglion cells, together with inhibitory horizontal and amacrine cells in between as illustrated in Fig. 1(a). The retinal ganglion cells (RGCs), as the only output neurons of the retina, send visual information via the optic tracts and the thalamus to cortical areas for higher cognition. Each RGC receives input from a number of excitatory BCs as a driving force to generate spikes, which traditionally is modeled by a biophysical model with a number of filters and nonlinearities as in Fig. 1(b) [27], [30], [31]. Thus, it serves as a typical model for both deciphering the structure of neuronal circuits [32]–[37] and testing novel methods for neuronal coding [27], [28], [38]. In this article, we use the retina to show what could be learned by CNNs. Unlike these previous studies focusing on a large population of RGCs [18], [23], here we take a different viewpoint by modeling single RGC with CNN as illustrated in Fig. 1(c). In this way, one can use the single cell to reveal the details of the receptive field of visual neurons [25].

Our aim is to study what kind of possible biological structure components in the retina can be learned by CNNs, and how one can use CNNs for understanding the computations carried out by single retinal cells. These questions concern the research focus of understanding, visualizing, and interpreting the CNNs components out of its black box.

Like those typical existing biophysical RGC models based on interpretable biophysical proprieties of the receptive filed measured experimentally [28], [38], we found our CNN model is also interpretable. By using a minimal biophysical model of RGC, we found the conv filters learned in CNNs are essentially the bipolar subunit components of the RGC model. Furthermore, we applied CNNs to analyze biological RGC data recorded in the salamander. The conv filters are resembling the receptive fields of BCs that sit in the previous layer of RGC and pool their computations to a downstream single RGC. Such a fine structure picture of the retinal receptive field revealed by CNNs suggests that conv filters learned are resembling biological components of the retinal circuit. Thus, a CNN learning from one single retinal cell reveals a minimal neural network carried out in this cell.

In addition, different RGCs were trained to obtain their corresponding CNN models, then we transferred these CNNs between these RGCs to test the ability of transfer learning. There is a diversity of transfer learning performance. In general, CNN models have a good performance in transfer learning. However, the best performance is still the one trained with its own original data, which implies that the CNN model is cell specific with a set of filters inherited from the target RGC. Furthermore, when CNNs are applied to a different input domain, here white noise images versus natural images, they can capture the meaningful responses of the RGCs in terms of the proper number of spikes and right spike timings of spikes, even in the cases where there is no spike for some specific images. This implies that CNNs indeed captures the full computational ability of one cell for different inputs.

Some preliminary results of this article were presented in a NIPS workshop short communication [39].

II. METHODS

A. Biophysical RGC Model

A biophysical RGC model as in Fig. 1(b) was modeled as a typical subunit model used previously [27], [40]. The model cell has four subunits with a spatial filter of the size 2×2 pixels, similar to a conv filter of CNNs but only sitting at a specific spatial location, and a temporal filter to take into account of temporal dynamics. Each subunit convolves the incoming stimulus image and then applies a nonlinearity of threshold-linear rectification. The output subunit signals are then polled together by the RGC. The polled signal is applied with a threshold-linear output nonlinearity with a positive threshold at unity to make spiking sparse. Thus, with this model, a given sequence of stimulus consisted of white noise images with the size as 8×8 pixels can generate a train of spikes.

B. Biological RGC Data

A public dataset of RGCs recorded in salamander as described in [17], [30], and [41] was used for CNNs modeling. Briefly, a population of RGC spiking activities was obtained by multielectrode array recordings as in [29]. The retinas were optically stimulated with spatiotemporal white noise images, temporally updated at a rate of 30 Hz and spatially arranged in a checkerboard layout with stimulus pixels of 30×30 μm. The recording time is about 4 h so that there are enough stimulus images and spikes for training a CNN model. A dataset of 300 natural images was also used as the stimulus for studying natural image responses [30], [41].

C. CNN RGC Model

We used naive CNN models containing a different number of convolution layers and a single dense layer as illustrated in Fig. 1(c). CNNs were implemented with Keras using Tensorflow as backend. The code is available at https://sites.google.com/site/jiankliu.
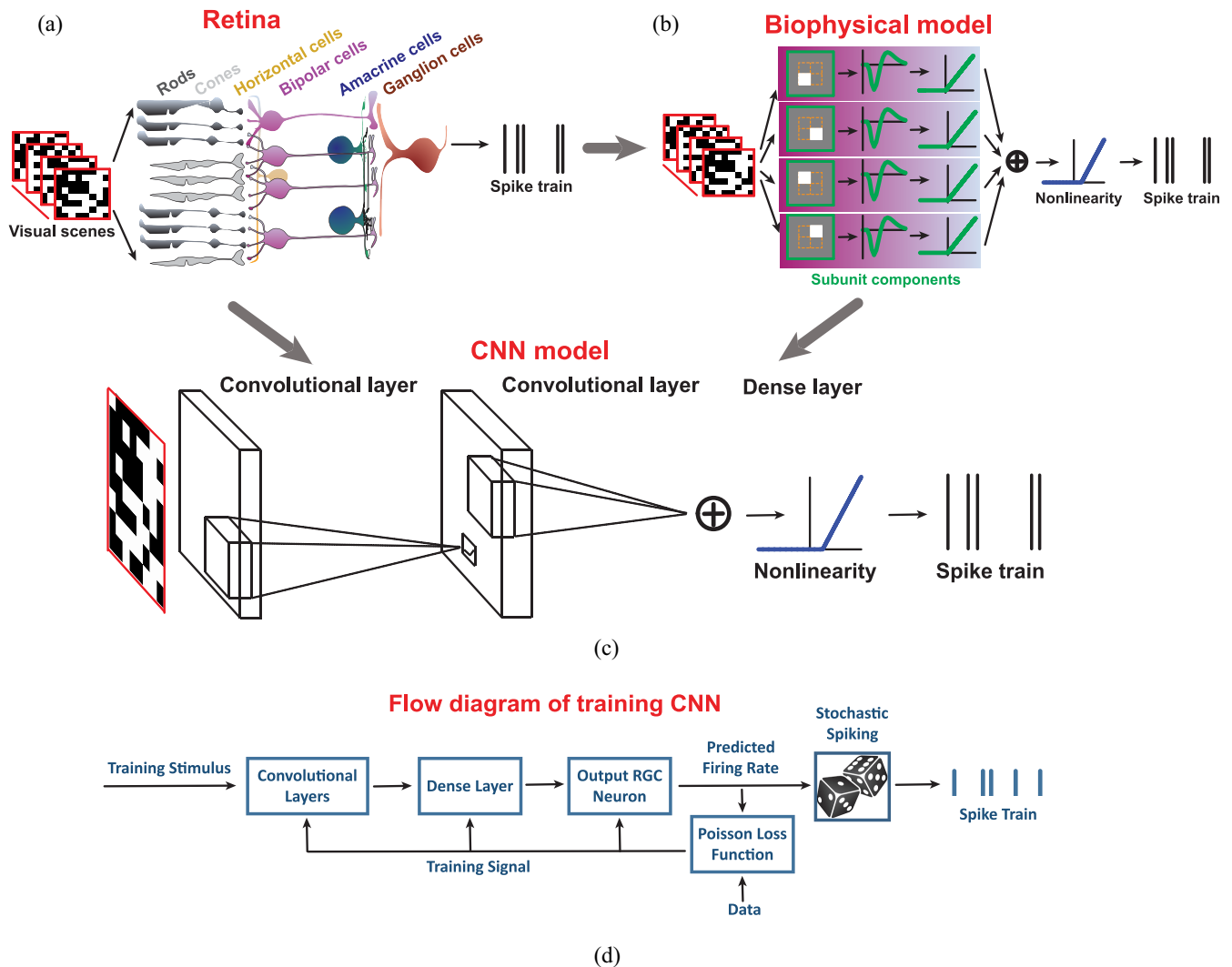
Fig. 1. Illustration of the biophysical model and the CNN model to study retinal computing. (a) Retinal circuit computes its output as a sequence of spikes for each RGC when visual scenes are received by the eyes. (b) Illustration of the RGC model structure used in the current article. Simplified neuronal circuit of a single RGC can be represented by a biophysical model that consists of a bank of subunit linear filters and nonlinearities. Note that there are four subunits playing the role of conv filters. (c) Illustration of the CNN model structure. CNN is used to train the same set of stimulus images to predict the spikes of all images for both biological RGC data and biophysical model data. (d) Flow diagram of training a CNN model to predict the RGC responses.

To model biophysical RGC model data, the conv filter size was fixed as 7×7. Both the number of filters and layers were tested. To model biological RGC data, several sets of parameters in convolution layers, including the number of layers, the number, and size of convolution filters were explored. The prediction performance is robust against these changes of parameters. Therefore, we adopted the filter size of 15×15 in the first conv layer and 3×3 in the second conv layer.

For training CNNs with biophysical RGC model data, we generated a dataset consisting of 600K training samples of white noise images, and an additional set of 10K samples for testing. The training labels are a train of binary spikes with 0 and 1 generated by the model.

For biological RGC data recorded in the salamander, there are about 320K training samples of white images and labels as the number of spikes as in [0 5] for each image. The test data have 300 samples, which were repeatedly presented to the retina for about 200 trials.

The procedure of training a CNN RGC model is shown in Fig. 1(d), where either binary spike train or firing rate of biophysical and biological test data can be compared to the predicated firing as CNN output for calculation of the Pearson correlation coefficient (CC) as a performance measure. A Poisson loss is used to optimize the CNN output to match the spiking labels. The final nonlinearity after the dense layer is a standard softplus function.

For preprocessing of data, a standard technique of spike-triggered average was applied to obtain a 3-D spatiotemporal receptive field filter [28]. The singular value decomposition of this 3-D filter yields a temporal filter and a spatial receptive field [42].

Two versions of CNNs were used.

Version I: The first version of CNN has only spatial filters without temporal filters to be fitted. For this, data were temporally correlated first by convolving every pixel of the entire set of stimulus images with the temporal filter first along

the temporal dimension [30], [43]. In this way, a sequence of spatial images was obtained as inputs with the corresponding spike train as output labels for the CNN model, such that this CNN makes analysis focusing on the spatial structure of receptive fields. In addition, this CNN has much fewer parameters, for example, when the temporal filter of interest is lasting for 600 ms with 30 Hz, then CNN parameters are 20 times less.

Version II: The second version of CNN is the full model with both spatial and temporal filters to be learned from the RGC data. This usually results in a large number of parameter that causes unavoidable problems for traditional statical models [27], [38]. The recent advancements of deep learning make it relatively easier to fit the data with high-dimensional parameters. Both versions of CNNs were used and compared to see the effects on the performance.

When CNNs are used for the object recognition task by learning a set of natural images, it is important to visualize what kind of features of natural images are learned by conv filter [4]. Similarly, here one can also visualize the image features represented by each CNN conv filter. The feature here means the response-weighted average feature, which is the spike-triggered average generated by a batch of random noise stimulus and its average activation from the corresponding feature map in the first layer.

III. RESULTS

By using both clearly defined biophysical model and real retinal data, we show that CNNs are interpretable when single RGC was modeled with the benefit to clarify what has been learned in the network structure components of CNNs. Recently, a variation of non-negative matrix factorization was used to analyze the RGC responses to white noise images and identify a number of subunits as BCs of one RGC [30]. With this picture in mind, here we address the question of what types of network structure components can be revealed by CNN when it is used to model the single RGC response.

A. Subunits of Modeled RGC as CNN Filters

We set up a biophysical RGC model with four subunits as in Fig. 1(b), which is resembling a 2-layer network with one layer of subunits and one layer of single RGC. By using a set of white noise images, the model generated a sequence of spikes to simulate a minimal neural network of the RGCs. With the input of stimulus images and the output of RGC spikes, we can train a CNN as in Fig. 1(c) to predict the simulated spikes generated by the RGC model.

Given that our focus is looking into the structure of network components learned in CNNs, we varied a number of parameters to train the CNNs, in particular, the number of conv filters from 1 to 16. When only one conv filter is used, the learned filter has a similar structure as the receptive field of the modeled RGC as in Fig. 2 that can be obtained by the standard method called spike-triggered average [28] (see Methods). When there are more conv filters, there are more fine structure of the receptive field as filters learned by CNNs. We found that when training the CNN with four conv filters, the outcome filters resemble the subunits used in the biophysical RGC model.
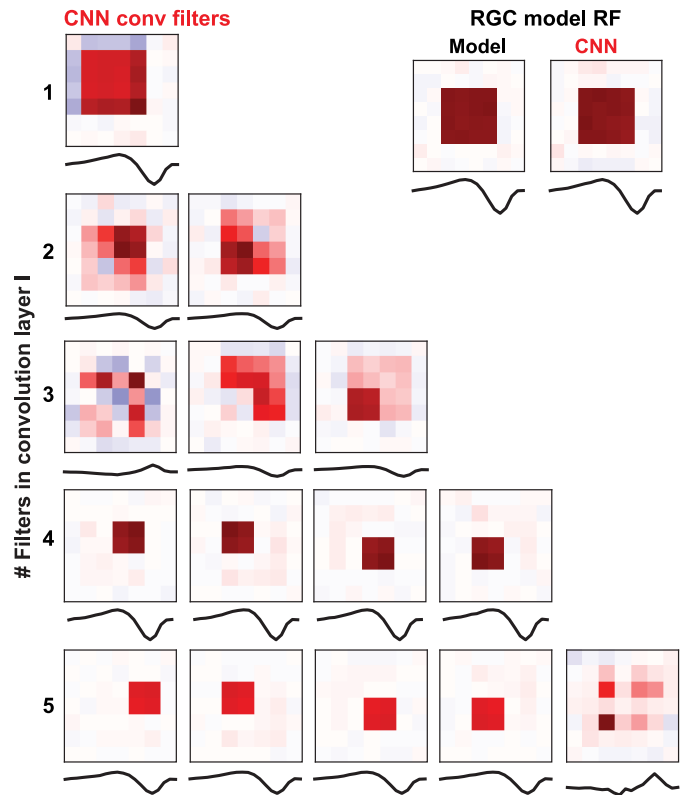

Fig. 2. Subunit structure of RGCs model data revealed by CNNs. Visualizing conv filters learned in a CNN with one layer of conv filters. The number of conv filters is from 1 to 5, where both spatial and temporal filters are learned by CNNs. (Inset) spatial receptive field and temporal filter of the modeled RGC computed by spike-triggered average (left) and CNN (right). Note the filter size is 7 × 7, and the full size of the receptive field is 8 × 8.

The outcome filters learned by the CNNs are convergent in the sense that there are only four "effective" filters similar to the model subunits, the rest of the filters are noise, for example when there are five conv filters in Fig. 2. This observation is similar to a recent study where the non-negative matrix factorization was used for identifying the subunits [30].

The nature of the black box of CNNs often forces researchers to tune the parameters of CNNs according to the performance that is usually the accuracy of the tasks such as image classification. Here, the performance is then the correlation between CNNs output and modeled RGCs spiking response. Not surprisingly, CNNs can give a good performance for predicting the RGCs response, which is consistent with the previous studies of various types of visual neurons [11], [15], [23]. More important, here we also found when there is enough number of conv filters, increasing the number of conv filters does not make the performance better as shown by an evolution of filter change with an increasing number of filters as in Fig. 3. The performance is convergent when the number of conv filter reaches 5.

In addition, the number of the conv layer is tuning the CNN performance to reach the saturation level. There is no difference when there are two layers or three layers of CNN conv filters. When there are two or more layers of conv filters, the performance is stabilized independently of the number of conv filters used. In either case, when there is only one conv filter
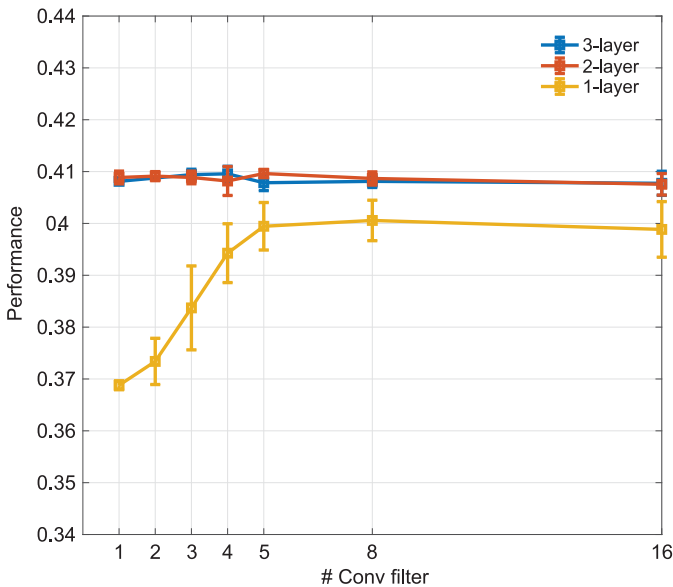
Fig. 3. CNN performance saturated when there are more conv filters and layers. Data points represented by 10 runs with mean ± standard derivation (STD).

used, it yields a result where the receptive filed of RGC is learned as one conv filter of CNN.

Therefore, these results show that increasing the number of conv filters and layers does not increase the performance when enough components are used to capture the underlying biophysical properties of the RGC model. In other words, CNN parameters could be highly redundant when setting up to a larger number. Such redundancy of parameters is widely observed for deep-learning models [44], [45]. This point will be shown by the biological data below in detail.

Altogether, these results suggest that CNNs can identify the underlying hidden network structure components within the RGCs model by only looking at the input stimulus images and the output spiking response.

B. Subunits of Biological RGC as CNN Filters

To further characterizing the structure components of CNNs in detail, we use CNNs to learn the biological RGC data with similar images of white noise and spiking responses. We first use CNNs to study temporally correlated RGCs data where no temporal filter is needed to be learned by CNNs (see Version I CNN model in Methods) with the benefit of fewer parameters of CNNs to be learned. Similar to the results of the RGC model above, the outputs of the CNN model can recover fine structures of the receptive field of RGC data very well as in Fig. 4(a). We also found that the learned conv filters converge to a set of localized subunits whereas the rest of filters are noisy and close to zero as in Fig. 4(b). The size of these localized filters is comparable to that in BCs around 100 μm [30].

In addition, the features (see Methods) represented by these localized conv filers are also localized. Given the example RGC is an OFF type cell that responds to the dark part of images strongly, most features have similar OFF peaks resulted

from the OFF BC-like filters. These OFF features tile the space of the receptive field of RGC. Interestingly, there are some features with ON peaks, which plays a role as inhibition in the retinal circuit. A few features have some complex structures mixed with OFF and ON peaks, which are mostly resulted from the less localized filters. However, if the filters are pure noise, the resulting features are pure noise without any structure embedded. Besides filters and features, the CNN model generates a good prediction of RGC response as in Fig. 4(c). These observations are similar across different RGCs recorded.

When there are 32 conv filters used in the CNN, there are enough number of conv filters to fit the data. Given there are many redundant filters [minimal 16 filters close to zero in Fig. 4(b)] unused in CNN learning, they unlikely reflect any interesting biophysical properties of RGC data, so could be no contribution for the performance of CNN. Indeed, CNN performance maintains at a similar level when the conv filters are pruned such that the only effective filters are used in CNN. The selection of effective filters was quantified with the spatial autocorrelation of conv filters [30]. The pruning results of a population of RGCs show that the performance is similar to a small subset of effective conv filters in Fig. 5. When pruning is done with randomly selected conv filters out of 32, the performance drops to zero as in Fig. 5 (right).

These results confirm the observation shown in the biophysical model RGC, where the CNN performance cannot be increased with more parameters than enough. In terms of biological RGC, the subunits are the upstream BCs [30]. Thus, the conv filters play a functional role as BCs when using CNNs to model biological RGCs.

Next, we examine the effect of sample size and noise on the CNN model. For the same cell shown in Fig. 4, the size of samples/images was changed in a wide range from 1% to 100%, in this way, the corresponding number of spikes recorded in RGC is reduced to about 150 spikes with 1% of training data. Both CNN performance and loss after training are dependent on how much data used for training as in Fig. 6(a). Surprisingly, with only about 150 spikes from 1% of data, we can still obtain some level of performance with CC about 0.35 as in Fig. 6(ai). Although the receptive field with such a small amount of spikes is not good [Fig. 6(aii)], the performance of CNNs does not drop that much (CC is 0.35 versus 0.75, but with 150 versus 15K spikes). Thus, CNNs seems to need only a relatively small set of spikes for training to obtain reasonable performance. Note with 30% of data (4.5K spikes), the performance is almost similar to the full data. In this sense, the CNNs seem to be much less data demanding than traditional biophysical RGC models [27], [38]. The resulting filters and features of CNNs are also worse, however, the spareness of filters still holds although 32 filters are used.

In contrast to the sample size, the noise has a much larger effect on the CNN model as in Fig. 6(b). Keeping the sample size unchanged, we replaced part of data images with irrelevant noise images, for instance, 70% of noise means there are 30% of data and 70% of noise images. Although the data percentage is the same as 30%, the CNN trained with noise has a much worse performance (CC ≈ 0.4) compared to the same amount of data without noise (CC ≈ 0.7). Fig. 6(bi)–(biii)
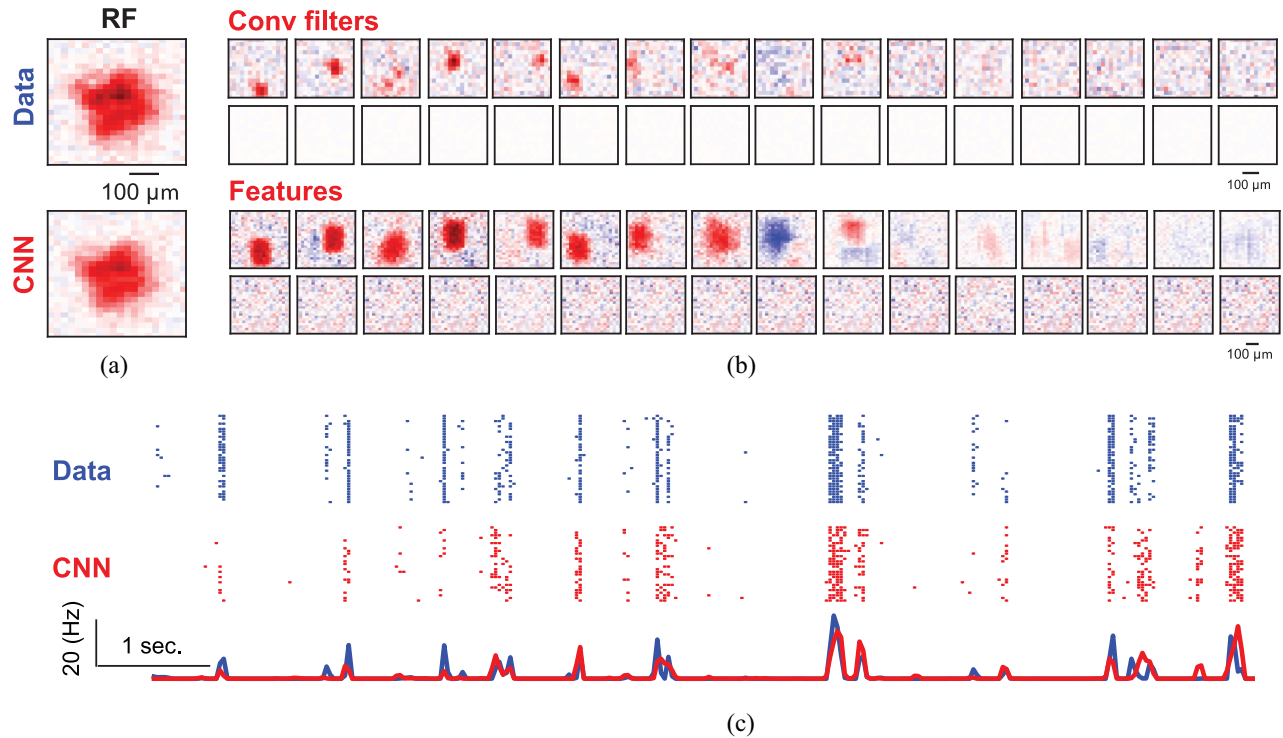
Fig. 4.   Subunit structures of biological RGC revealed by CNN. (a) Receptive fields of one RGC computed by STA and CNN prediction. (b) Visualizing CNN model components of conv filters and average features represented by each filter. (c) Neuronal response predicted by CNN visualized by RGC data spike rasters (upper), CNN spike rasters (middle), and their average firing rates (bottom).
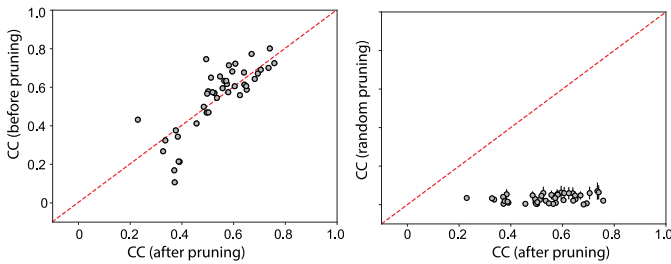

Fig. 5.   Pruning conv filters in CNNs. (Left) The CNN model performance (CC) maintained at a similar level after pruning CNN by using only a subset of effective (nonzero) filters as in Fig. 4. (Right) CNN performance dropped to zero when pruning CNN with the same number of parameters but a randomly selected subset of filters. Error bars indicate ten random prunings (mean ± STD).

shows an example result of training CNN with complete noise images. Even in this case, the redundancy of filters makes most filters close to zeros.

C. Transfer Learning Across Different RGCs

Given there is a population of RGCs recorded experimentally and then modeled by CNNs, one can study the behavior of transfer learning, or the generalization ability of CNNs model, that is, using the CNN model learned from one RGC to predicate the response of another RGC. Several scenarios of transfer learning are commonly used in deep learning, including using a pretrained model directly, fine-tuning a pretrained model, or fixing features of a pretrained model but adjusting

the dense layer [46]. Here, we took the approach to use a pretrained CNN directly from one RGC to other different RGCs. This is suitable for our experimental setup, as a large full-size of white noise images were presented to all RGCs of a population at one time. Different RGCs are sitting at different spatial locations of images, therefore they are seeing parts of the entire image. However, due to the nature of white noise images, the statistics of the ensemble input images are the same, or at least closely similar to Gaussian, across different RGCs. Thus, one expects that the transfer learning of the CNN model, in this case, has a good performance.

However, we found there is a large diversity of transfer learning performance across different RGCs as shown in Fig. 7, where there are four example cells showing their CNNs model predictions (diagonal traces labeled as "Self") and the corresponding transfer learning behaviors (off-diagonal traces). The CNN performance characterized by CC shows the tendency that the best performance is always from the CNN model trained with their own cell. For instance, cell 1 has a good test performance with its own CNN model, denoted as CNN 1. When CNN 1 is transferred to the other three cells, the prediction power is lower than cell 1. However, CNN 3 is transferred to cell 1 with a better performance of CC = 0.64 compared to CC = 0.56 for cell 3 itself. Interestingly, all CNN models have a reasonably good performance for cell 1, even though CNN 4 has a fairly good performance for cell 4 itself.

To further investigate the ability of transfer learning of the CNN model, we collect a population of ten RGCs (including those four cells in Fig. 7). The population plots in Fig. 8 confirm the observation above. The scatter plot of self-learning
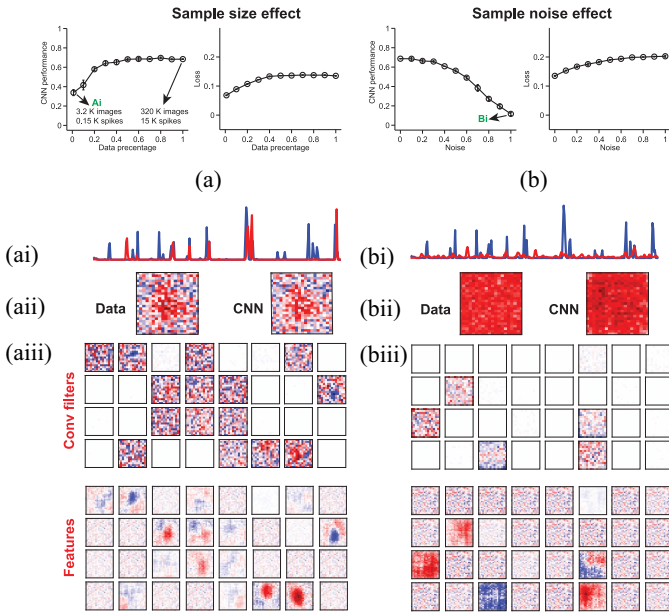
Fig. 6. Effects of training sample size and noise on the performance of CNNs. (a) Effect of sample size changing from 1% (0.01) to 100% (1) of the entire training set images on the performance of CNN (CC, upper) and the loss of training (Loss, bottom), where 1% of training data has about 3.2K images and 0.15K spikes (3.05K of nonspiking labels as zero) due to the spare firing property of RGC. Data point of ai is shown in (ai)–(aiii). (ai) CNN output versus test data firing rate. (aii) receptive field of data (left) versus CNN prediction (right). (aiii) conv. filters and features leaned by CNN. (b) Similar to (a) but for the effect of noise in sample images, where the ratio of noise is changing from 0% (without noise) to 1 (100% of noise without data). (bi)–(biii) Similar to (ai)–(aiii).

Fig. 8. CNNs transfer prediction for a population of ten RGCs. (a) Performance matrix of ten CNNs across different RGCs, where the diagonal ones are self-training (Target to Target, purple), each row is transferring the CNN trained by target cell to all other cells (Target to Others, fix-CNN, green), and each column is transferring all other CNNs to the target cell (Others to Target, fix-data, light blue). The first four cells are the same ones as in Fig. 7. The rows and columns of the performance matrix have the same meaning as in Fig. 7. (b) Performance matrix shown as a scatter plot of self-training versus fix-CNN. (c) Performance matrix is shown as a scatter plot of self-training versus fix-data.

with lowest CC in self-learning, when its CNN is transferred to other cells, its CNN has a better performance.

Similarly, as above, the performances are always lower when other CNNs are transferred to the target cells themselves [Fig. 8 (right)]. This indicates that each CNN model is really optimized for the target cell after training. For the best cell who has the highest CC by CNN, other CNNs trained with other cells also have good performance in general. In contrast, for the worst cell who has the lowest CC by CNN, other CNNs trained with other cells also have bad performance.

These results suggest that the CNN learned from each cell is very specific to that particular cell. The resulting CNNs, therefore, learns to obtain a minimal neural network that carries out the essential computations done by that cell.

D. Full CNN Model for Biological RGC Data

The results above on biological RGC data were studied by a CNN without temporal filter learned. Now we consider the full CNN model (Version II CNN model, see Methods), where both temporal and spatial filters are needed. Similar to the RGC model data, CNN can recover both spatial filters as receptive field and temporal filters as shown in Fig. 9(a). With the full CNN model, there are 20 times more parameters than that of temporally correlated data. Both conv filters and features visualized in CNN have still a located spatial structure and good temporal filer shape for a subset of filters. Not surprisingly, the prediction of CNNs for neuronal response has a good performance as well.
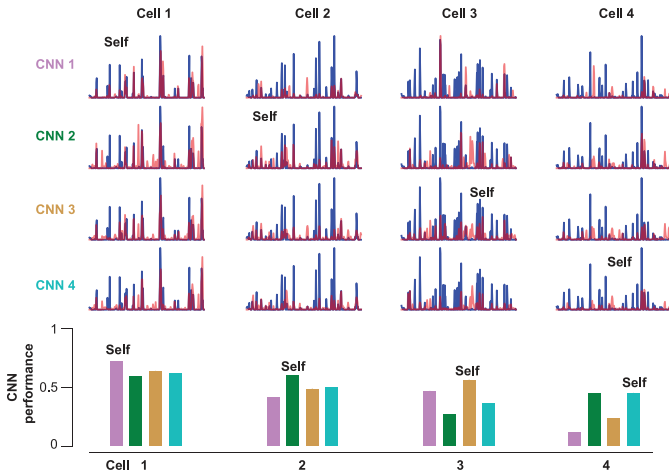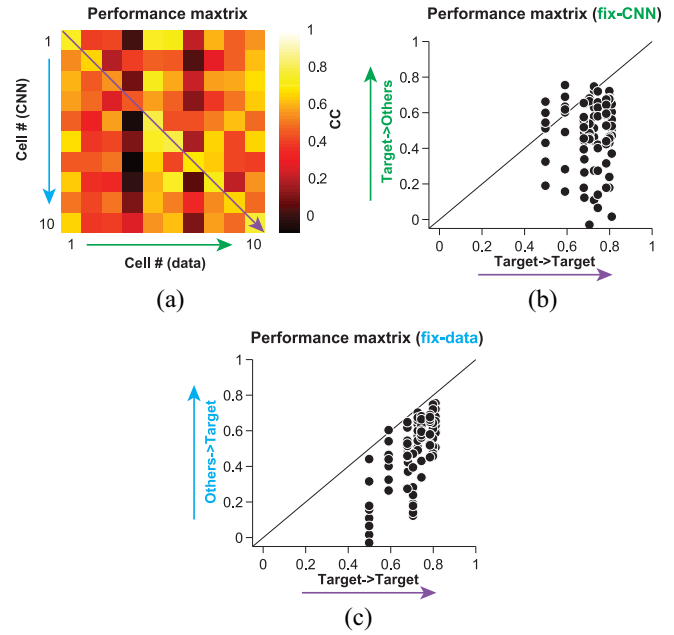

Fig. 7. CNNs transfer learning across different cells. (Upper) Four example cells used to train the CNN models. Learned CNNs are then transferred across different RGCs for prediction. Over this 4 × 4 matrix, the diagonal ones are the cases of self-training (Target to Target) with each cell's data. cell 1 is the same cell as in Fig. 4. CNN 1 is the model obtained by cell 1. The first row represents that CNN 1 model is transferred to predict the test data of cells 2–4. The first column represents that three CNN models (CNN 2–4) from the other three cells (cells 2–4) are transferred to predict the test data of cell 1. (Bottom) Performance of this 4 × 4 matrix. The first four points are calculated for cell 1 from CNN 1–4. Different CNNs are colored in different colors.

(Target → Target) versus transfer learning (Target → Others) in Fig. 8 (left) shows that the CNN model has reasonable good performance for both self-learning and transfer learning, yet the results are quite diverse. Note that even for the worst cell
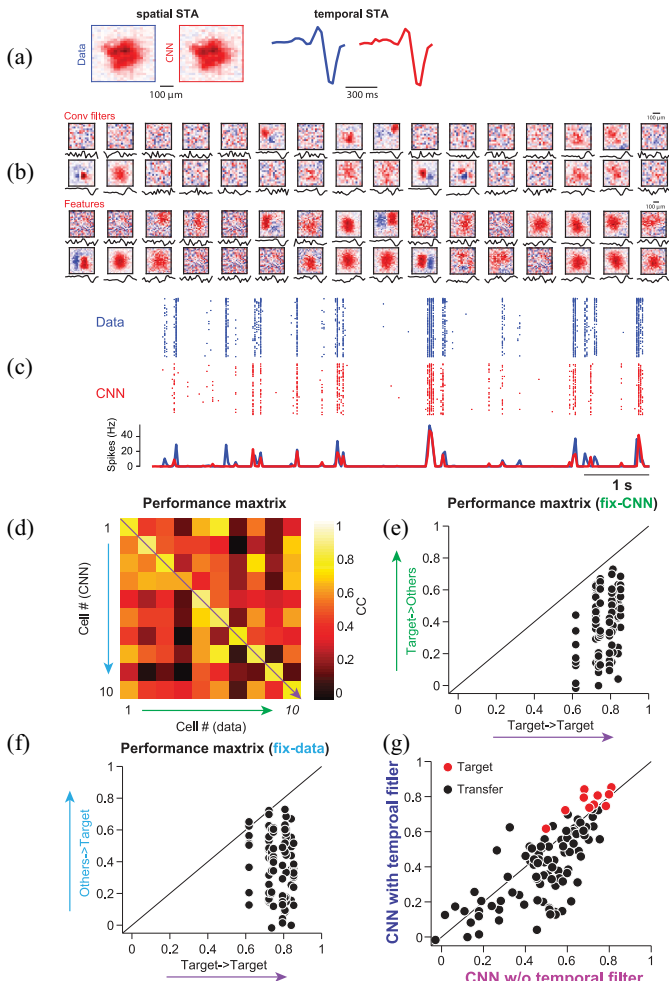
Fig. 9. Spatial and temporal filters of biological RGC data revealed by full CNN model. (a) Receptive fields as spatial STA of the example cell and CNN prediction. Temporal filters are also recovered by CNN. (b) Visualizing CNN model components of both conv filters and average features represented by each filter in both spatial and temporal dimensions. (c) Neuronal response predicted CNN visualized by RGC data spike rasters (upper), CNN spike rasters (middle), and their firing rates. (d)–(f) Transfer prediction by full CNN models for a population of ten RGCs. All plots have the same meanings as Fig. 8. (g) CNN prediction improved with temporal filter included, but transfer prediction is worse in general. Performance (CC) matrix shown as a scatter plot of CNN without temporal filter (Fig. 8) versus full CNN with temporal filter. Black indicates transfer prediction and red indicates target prediction.

Similar to Fig. 8, we also tested the transferring learning ability of the full CNNs model for the same population of ten RGCs. The results in Fig. 9 show a similar tendency of transferring learning for the CNN model. Yet, there are some differences between the two versions of CNN models, which can be seen by comparison in Fig. 9. For the target case, where each CNN was trained by using that particular RGC data, the full CNN yields a better performance than the CNN without temporal filter learned. However, when full CNNs are transferred between different cells, their performance is in general worse than reduced CNNs. That indicates that the full CNN with both spatial and temporal filters is more specific to the particular cell used for training. In turn, such a cell-specific CNN cannot be used to explain other cells. Therefore, this confirms the result that CNN indeed learns the underlying

neuronal computation carried out by the biological cell, which cannot be transferred across different cells.

Given there are 20 times more parameters in the full CNN, the conv filters have less clearly localized structures than those in the CNN of reduced temporal correlations when comparing the filters in Figs. 4 and 9. This seems to be caused by the limited sample size of biological RGC data. When the biophysical RGC model is used for both versions of CNN, there is no difference in terms of the structure of conv filters, see the results shown in [39]. Therefore, depending on the questions to be addressed, one may want to choose the simple or full version of CNN for modeling static images or dynamical videos, as videos contain strong temporal correlations that have a strong impact on the adaptation of neuronal dynamics [29].

E. Transfer Learning Between Different Stimulus Images

Above we tested transfer learning of CNNs across different cells. Here, we further test the transfer learning of CNNs from each cell but for different input images. So far, the CNN model of each GC trained by white noise images, one can test the ability of transfer prediction by directly applying the learned CNNs to natural scenes.

For this, a sequence of 300 natural images was presented to the retina, where each image was briefly presented for 200 ms and followed by 800-ms empty scene. Such a protocol of stimulation is to leaving out temporal adaption for each image (see details in [41]). Therefore, we used a temporal decorrelated CNN model (Version I) to fit the neuronal response of this sequence of flashing images.

Surprisingly, CNNs learned by white noise images can predict the spiking response of natural images quite well. Fig. 10(a) shows the result of one example cell. For a clear illustration, we only show a partial sequence of 50 images. The detailed result is demonstrated with five images together with three cells in Fig. 10(b). Each RGC reads a different part of images as the receptive field is located differently. Interestingly, CNNs predict the responses well when all three cells fire for image 1, in addition, CNNs can also predict when one of three cells is not firing with spikes for images 2–4. Even when all three cells are not firing completely for image 5, the outputs of CNNs are also silent.

These resulting in transfer learning across image domains, compared to those across different RGCs, suggest that CNNs intend to learn a cell-specific neural network in which conv filters play a role as upstream subunit cells that connect to a particular RGC. Thus, CNNs serve a model of neural system identification to reveal the underlying computations of the RGCs.

IV. SUMMARY AND DISCUSSIONS

In recent years, system identification of neural coding based on neural networks has been greatly improved by the development of deep CNN [1], [10], [11], where multiple layers and huge banks of local subunit filters within the network are significant characteristics. Besides its powerful ability to many practical tasks [1], the underlying structure is mimicking a hierarchical organization of the brain [10], [11]. However,
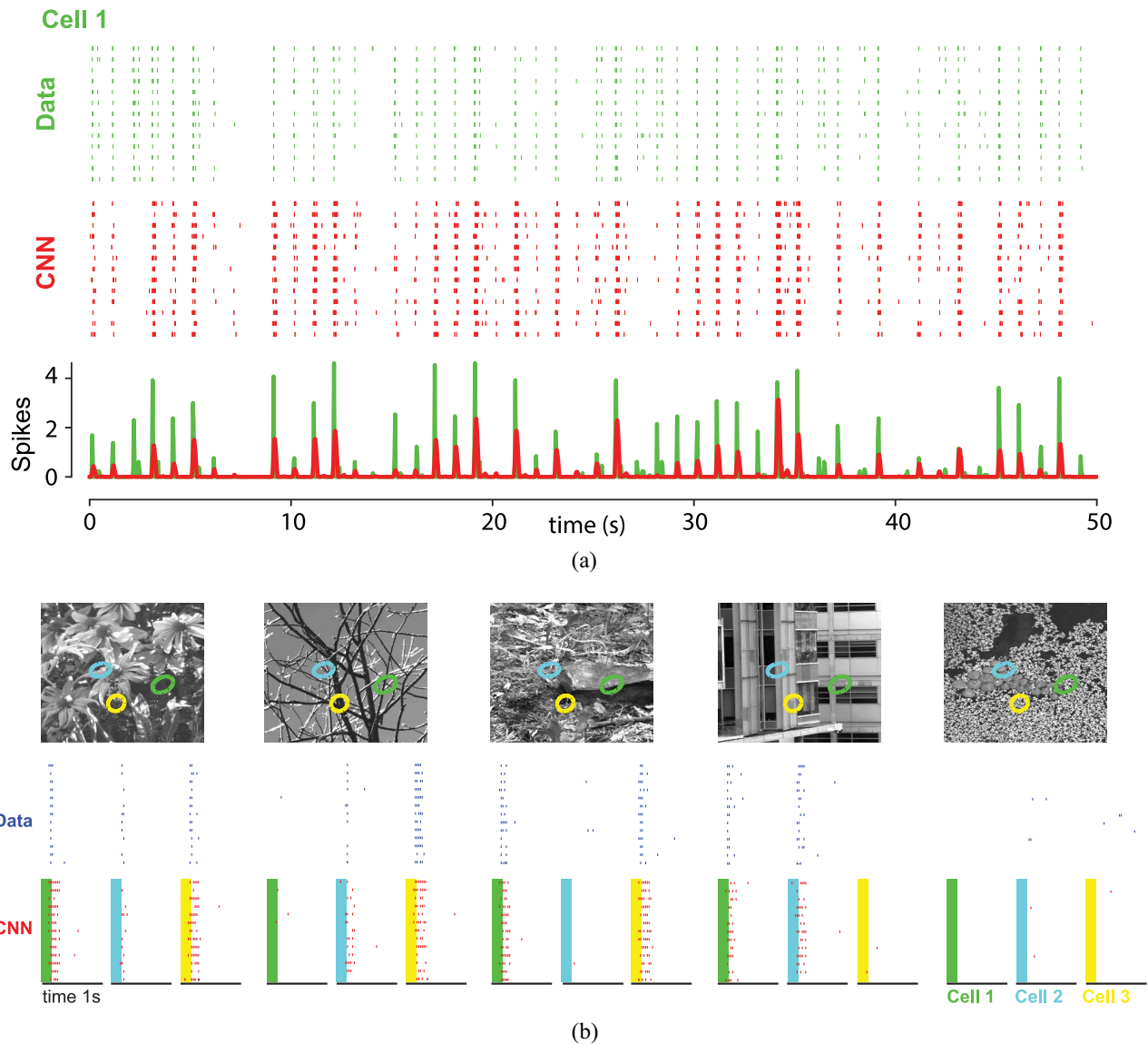
Fig. 10. Transfer prediction of natural images by CNNs trained with white noise images. (a) Performance of CNN for the example RGC [green cell in (b)] with 50 images (one image per second). (b) Performance of CNN with four example images and three RGCs. (Top) Five images overlaid with the outlines of the receptive field of three RGCs colored in green, light blue, and yellow. (Bottom) Spike response of each GC for each image (blue) together with spikes sampled from CNN output (red). Each image was presented for 200-ms long (colored shadow window) then followed by 800-ms gray period. Note RGC response is delayed after the onset of an image. Each image triggers one RGC in a different manner with spiking or nonspiking depending on the texture of the image and specific RGC.

there is no first principle about designing the structure of a hierarchical deep-learning network [2]. Recent works begin to look into the details of network structure, in particular, with the potential connections to the biological brain [3], [11].

Hereby focusing on single RGC, we found that CNNs can learn their parameters in an interpretable fashion, and CNNs network components are close to the biological underpinnings of the retinal circuit. With the benefit of the relative well-understood retinal circuit, our results suggest that the building blocks of CNNs are meaningful when they are applied to neuroscience for revealing network structure components.

Deep CNNs are useful for modeling the abstract level of vision information [10], [11] and neural coding in general [47] in neuroscience. This article simplified the approach used by the previous studies [10], [11], [13] for the higher part of

the visual cortex, where interpretable structure components of deep CNNs are difficult due to the complex hierarchical layers from the retina to the IT cortex in the brain. By training CNNs with white noise images, our current work also simplified the interpretation, compared to the studies where natural images were used for model visual neurons with CNNs [15], [21], [25], since white noise images have the benefit to mapping out the receptive field of the visual neurons [28], [30].

Unlike most of the recent studies, which are driven by the goal that the better performance of neural response can be archived by using either feedforward and recurrent neural networks (or both), here we focus on the fine structure of the receptive field in the retinal circuit. Along with several recent papers [23], [25], [48], characterizing the receptive field of visual neurons is important for understanding the filters

leaned by the CNNs. Given the retina has a relatively clear and simple circuit, and the eye has (almost) no feedback connections from the cortical cortex, it is a suitable model system as a feedforward neural network, similar to the principle of CNNs. Certainly, the contributions from the inhibitory neurons, such as horizontal cells and amacrine cells, play a role in the function of the retina. In this sense, the potential neural networks with lateral inhibition and/or recurrent units are desirable [15], [16].

Our approach is suitable to address other difficult issues of deep learning, such as transfer learning, since the domain of images seen by single RGCs is local and less complicated than the global structure of entire natural images. In the first case, it is surprising that the transfer learning across different RGCs is not perfect given that the stimulus distribution is the same since the ensembles of white noise image stimuli for all RGCs converge to a naive Gaussian distribution. However, a further thought indicates that the CNNs learned from each cell is rather specific with a particular set of filters in conv layers. In this article, we only explore the filters of CNNs. Future work is needed to investigate other components, such as nonlinearities, of CNNs.

Here, we also test the ability of transfer learning across different types of stimulus, that is, transfer between white noise images to natural images. Traditionally, dynamical videos have been used in the studies of using CNNs with the neuronal signals of interest as fMRI data [49], for neuronal spiking response, a sequence of flash images is often used [11], [15], [20]. One needs to consider the case of how to use CNNs for neuronal spikes to study dynamical visual scenes, that is, continuous videos [50]. Future work is needed in this direction of studying the retinal computation under dynamical natural scenes.

CNNs here modeled for the retinal RGCs can be further used for the reconstruction of natural images based on the responses of retinal RGCs [50], [51]. Embedding the encoder/decoder into the retinal prosthesis has been suggested as a promising direction for visual restoration [52]. Such an approach of studying spike encoding and decoding of visual scenes with neural spikes will be crucial for the next generation of neuromorphic computing, including artificial visual system [53], where the data format processed on chips are digital spikes [54]. One expects that the close interaction of the algorithms based on spike data and CNNs [55]–[58] with neuromorphic chips [59], [60] will greatly expand our computing capacity.

ACKNOWLEDGMENT

REFERENCES

[1] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] L. N. Smith and N. Topin, "Deep convolutional neural network design patterns," 2016. [Online]. Available: arXiv:1611.00847.

[3] A. H. Marblestone, G. Wayne, and K. P. Kording, "Toward an integration of deep learning and neuroscience," *Front. Comput. Neurosci.*, vol. 10, p. 94, Sep. 2016.

[4] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.

[5] H. Demis, K. Dharshan, S. Christopher, and B. Matthew, "Neuroscience-inspired artificial intelligence," *Neuron*, vol. 95, no. 2, pp. 245–258, Jul. 2017.

[6] Y. Lecun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2010, pp. 253–256.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[8] E. P. Simoncelli and B. A. Olshausen, "Natural image statistics and neural representation," *Annu. Rev. Neurosci.*, vol. 24, no. 24, p. 1193, 2001.

[9] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2018–2025.

[10] N. Kriegeskorte, "Deep neural networks: A new framework for modeling biological vision and brain information processing," *Annu. Rev. Vis. Sci.*, vol. 1, no. 1, pp. 417–446, Nov. 2015.

[11] D. L. K. Yamins and J. J. Dicarlo, "Using goal-driven deep learning models to understand sensory cortex," *Nat. Neurosci.*, vol. 19, no. 3, pp. 356–365, 2016.

[12] D. Yamins, H. Hong, C. Cadieu, and J. J. Dicarlo, "Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3093–3101.

[13] D. L. K. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo, "Performance-optimized hierarchical models predict neural responses in higher visual cortex," *Proc. Nat. Acad. Sci. USA*, vol. 111, no. 23, pp. 8619–8624, May 2014.

[14] S.-M. Khaligh-Razavi and N. Kriegeskorte, "Deep supervised, but not unsupervised, models May explain IT cortical representation," *PLoS Comput. Biol.*, vol. 10, no. 11, Nov. 2014, Art. no. e1003915.

[15] L. McIntosh, N. Maheswaranathan, A. Nayebi, S. Ganguli, and S. Baccus, "Deep learning models of the retinal response to natural scenes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1369–1377.

[16] E. Batty *et al.*, "Multilayer recurrent network models of primate retinal ganglion cell responses," in *Proc. 5th Int. Conf. Learn. Represent.*, 2017, pp. 24–26.

[17] P. J. Vance *et al.*, "Bioinspired approach to modeling retinal ganglion cells using system identification techniques," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1796–1808, May 2018.

[18] N. Maheswaranathan *et al.*, "Deep learning models reveal internal structure and diverse computations in the retina under natural scenes," 2018. [Online]. Available: https://www.biorxiv.org/content/10.1101/340943v1.

[19] B. Vintch, J. A. Movshon, and E. P. Simoncelli, "A convolutional subunit model for neuronal responses in macaque v1," *J. Neurosci.*, vol. 35, no. 44, pp. 14829–14841, 2015.

[20] J. Antolík, S. B. Hofer, J. A. Bednar, and T. D. Mrsic-Flogel, "Model constrained by visual hierarchy improves prediction of neural responses to natural scenes," *PLoS Comput. Biol.*, vol. 12, no. 6, Jun. 2016, Art. no. e1004927.

[21] W. F. Kindel, E. D. Christensen, and J. Zylberberg, "Using deep learning to reveal the neural code for images in primary visual cortex," 2017. [Online]. Available: arXiv:1706.06208.

[22] S. A. Cadena *et al.*, "Deep convolutional models improve predictions of macaque v1 responses to natural images," 2017. [Online]. Available: https://www.biorxiv.org/content/10.1101/201764v1.

[23] D. Klindt, A. S. Ecker, T. Euler, and M. Bethge, "Neural system identification for large populations separating 'what' and 'where'," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3509–3519.

[24] M. R. Whiteway, K. Socha, V. Bonin, and D. A. Butts, "Characterizing the nonlinear structure of shared variability in cortical neuron populations using neural networks," 2018. [Online]. Available: https://www.biorxiv.org/content/10.1101/407858v1.

[25] J. Ukita, T. Yoshida, and K. Ohki, "Characterization of nonlinear receptive fields of visual neurons by convolutional neural network," 2018. [Online]. Available: https://www.biorxiv.org/content/10.1101/348060v1.

[26] R. J. Rowekamp and T. O. Sharpee, "Cross-orientation suppression in visual area v2," *Nat. Commun.*, vol. 8, Jun. 2017, Art. no. 15739.

[27] J. M. McFarland, Y. Cui, and D. A. Butts, "Inferring nonlinear neuronal computation based on physiologically plausible inputs," *PLoS Comput. Biol.*, vol. 9, no. 7, Jul. 2013, Art. no. e1003143.

[28] E. J. Chichilnisky, "A simple white noise analysis of neuronal light responses," *Network*, vol. 12, no. 2, pp. 199–213, 2001.

[29] J. K. Liu and T. Gollisch, "Spike-triggered covariance analysis reveals phenomenological diversity of contrast adaptation in the retina," *PLoS Comput. Biol.*, vol. 11, no. 7, Jul. 2015, Art. no. e1004425.

[30] J. K. Liu *et al.*, "Inference of neuronal functional circuitry with spike-triggered non-negative matrix factorization," *Nat. Commun.*, vol. 8, no. 1, p. 149, Jul. 2017.

[31] S. Jia, Z. Yu, A. Onken, Y. Tian, T. Huang, and J. K. Liu, "Neural system identification with spike-triggered non-negative matrix factorization," 2018. [Online]. Available: arXiv:1808.03958.

[32] M. Helmstaedter, K. L. Briggman, S. C. Turaga, V. Jain, H. S. Seung, and W. Denk, "Connectomic reconstruction of the inner plexiform layer in the mouse retina," *Nature*, vol. 500, no. 7461, pp. 168–174, 2013.

[33] H. Zeng and J. R. Sanes, "Neuronal cell-type classification: Challenges, opportunities and the path forward," *Nat. Rev. Neurosci.*, vol. 18, no. 9, pp. 530–546, 2017.

[34] R. E. Marc, B. W. Jones, C. B. Watt, J. R. Anderson, C. Sigulinsky, and S. Lauritzen, "Retinal connectomics: Towards complete, accurate networks," *Progress Retinal Eye Res.*, vol. 37, pp. 141–162, Nov. 2013.

[35] H. S. Seung and U. Sümbül, "Neuronal cell types and connectivity: Lessons from the retina," *Neuron*, vol. 83, no. 6, pp. 1262–1272, 2014.

[36] J. R. Sanes and R. H. Masland, "The types of retinal ganglion cells: Current status and implications for neuronal classification," *Annu. Rev. Vis. Sci.*, vol. 38, pp. 221–246, Jul. 2015.

[37] J. B. Demb and J. H. Singer, "Functional circuitry of the retina," *Annu. Rev. Vis. Sci.*, vol. 1, pp. 263–289, Nov. 2015.

[38] J. W. Pillow *et al.*, "Spatio-temporal correlations and visual signalling in a complete neuronal population," *Nature*, vol. 454, no. 7207, pp. 995–999, 2008.

[39] Q. Yan, Z. Yu, F. Chen, and J. K. Liu, "Revealing structure components of the retina by deep learning networks," in *Proc. NIPS Symp. Interpretable Mach. Learn.*, 2017. [Online]. Available: https://arxiv.org/abs/1711.02837

[40] T. Gollisch and M. Meister, "Rapid neural coding in the retina with relative spike latencies," *Science*, vol. 319, no. 5866, pp. 1108–1111, 2008.

[41] A. Onken, J. K. Liu, P. C. R. Karunasekara, I. Delis, T. Gollisch, and S. Panzeri, "Using matrix and tensor factorizations for the single-trial analysis of population spike trains," *PLoS Comput. Biol.*, vol. 12, no. 11, Nov. 2016, Art. no. e1005189.

[42] J. L. Gauthier *et al.*, "Receptive fields in primate retina are coordinated to sample visual space more uniformly," *PLoS Biol.*, vol. 7, no. 4, Apr. 2009, Art. no. e1000063.

[43] J. Kaardal, J. D. Fitzgerald, M. J. Berry, and T. O. Sharpee, "Identifying functional bases for multidimensional neural computations," *Neural Comput.*, vol. 25, no. 7, pp. 1870–1890, 2013.

[44] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and N. de Freitas, "Predicting parameters in deep learning," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2013, pp. 2148–2156.

[45] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *Fiber*, vol. 56, no. 4, pp. 3–7, 2015.

[46] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2014, pp. 3320–3328.

[47] J. I. Glaser, R. H. Chowdhury, M. G. Perich, L. E. Miller, and K. P. Kording, "Machine learning for neural decoding," 2017. [Online]. Available: arXiv:1708.00909.

[48] A. S. Ecker *et al.*, "A rotation-equivariant convolutional neural network model of primary visual cortex," 2018. [Online]. Available: arXiv:1809.10504.

[49] H. Wen, J. Shi, Y. Zhang, K.-H. Lu, J. Cao, and Z. Liu, "Neural encoding and decoding with deep learning for dynamic natural vision," *Cerebr. Cortex*, vol. 28, no. 12, pp. 1–25, 2017.

[50] Y. Zhang *et al.*, "Reconstruction of natural visual scenes from neural spikes with deep neural networks," 2019. [Online]. Available: arXiv:1904.13007.

[51] N. Parthasarathy *et al.*, "Neural networks for efficient bayesian decoding of natural images from retinal neurons," in *Advances in Neural Information Processing Systems 30*, I. Guyon *et al.*, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 6437–6448.

[52] S. Nirenberg and C. Pandarinath, "Retinal prosthetic strategy with the capacity to restore normal vision," *Proc. Nat. Acad. Sci. USA*, vol. 109, no. 37, pp. 15012–15017, Aug. 2012.

[53] Z. Yu *et al.*, "Towards the next generation of retinal neuroprosthesis: Visual computation with spikes," 2020. [Online]. Available: arXiv:2001.04064.

[54] S. Dong, T. Huang, and Y. Tian, "Spike camera and its coding methods," in *Proc. Data Compress. Conf. (DCC)*, 2017, p. 437.

[55] Y. Hu, H. Tang, Y. Wang, and G. Pan, "Spiking deep residual network," 2018. [Online]. Available: arXiv:1805.01352.

[56] Q. Xu, Y. Qi, H. Yu, J. Shen, H. Tang, and G. Pan, "CSNN: An augmented spiking based framework with perceptron-inception," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2018, pp. 1646–1652.

[57] R. Xiao, H. Tang, P. Gu, and X. Xu, "Spike-based encoding and learning of spectrum features for robust sound recognition," *Neurocomputing*, vol. 313, pp. 65–73, Nov. 2018.

[58] Q. Xu, J. Peng, J. Shen, H. Tang, and G. Pan, "Deep CovDenseSNN: A hierarchical event-driven dynamic framework with spiking neurons in noisy environment," *Neural Netw.*, vol. 121, pp. 512–519, Jan. 2020.

[59] S. Esser *et al.*, "Convolutional networks for fast, energy-efficient neuromorphic computing." *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 41, pp. 11441–11446, 2016.

[60] M. Davies *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan./Feb. 2018.

[61] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: arXiv:1409.1556.

Qi Yan received the B.S. degree from the Department of Automation, Tsinghua University, Beijing, China, in 2015, where he is currently pursuing the Ph.D. degree.

His current research interests include computer vision, artificial intelligence, and reinforcement learning.

Yajing Zheng received the B.S. degree from Sichuan University, Sichuan, China, in 2017. She is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science, Peking University, Beijing, China.

Her research interests include neuroscience, brain-inspired computing, machine learning, and spiking neural network.

Shanshan Jia received the master's degree from the Smart City College, Beijing Union University, Beijing, China, in 2018.

She is currently a Research Assistant with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing. Her current research interests include artificial intelligence, brain-inspired computing, and computational neuroscience.

Yichen Zhang received the B.S. degree in computer science from Northwestern Polytechnical University, Xi'an, China, in 2016. He is currently pursuing the Ph.D. degree with Peking University, Beijing, China.

His research interests include neuron simulation, computational neuroscience, and artificial intelligence.

Zhaofei Yu (Member, IEEE) received the B.S. degree from the Hong Shen Honors School, College of Optoelectronic Engineering, Chongqing University, Chongqing, China, in 2012, and the Ph.D. degree from the Automation Department, Tsinghua University, Beijing, China, in 2017.

He is currently a Postdoctoral Fellow with the National Engineering Laboratory for Video Technology, School of Electronics Engineering and Computer Science, Peking University, Beijing. His current research interests include artificial intelligence, brain-inspired computing, and computational neuroscience.

Feng Chen (Member, IEEE) received the B.S. and M.S. degrees in automation from Saint-Petersburg Polytechnic University, Saint Petersburg, Russia, in 1994 and 1996, respectively, and the Ph.D. degree from the Automation Department, Tsinghua University, Beijing, China, in 2000.

He is currently a Professor with Tsinghua University. His current research interests include computer vision, brain-inspired computing, and inference in graphical models.

Yonghong Tian (Senior Member, IEEE) received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2005.

He is currently a Full Professor with the School of Electronics Engineering and Computer Science, Peking University, Beijing, and the Deputy Director of the AI Research Center, Peng Cheng Laboratory, Shenzhen, China. He has authored or coauthored over 170 technical articles in refereed journals and conferences. His research interests include computer vision, multimedia big data, and brain-inspired computation.

Prof. Tian was a recipient of the Chinese National Science Foundation for Distinguished Young Scholars in 2018, two National Science and Technology Awards and three Ministerial-Level Awards in China, the 2015 EURASIP Best Paper Award for the *EURASIP Journal on Image and Video Processing*, and the Best Paper Award of IEEE BigMM 2018. He co-initiated IEEE International Conference on Multimedia Big Data (BigMM) and served as a TPC Co-Chair for BigMM 2015, and also served as the Technical Program Co-Chair for IEEE ICME in 2015, IEEE ISM in 2015, and IEEE MIPR in 2018 and 2019, and the Organizing Committee Member of ACM Multimedia in 2009, IEEE MMSP in 2011, IEEE ISCAS in 2013, and IEEE ISM in 2015 and 2016. He has been the Steering Committee Member of IEEE ICME since 2018 and IEEE BigMM since 2015, and the TPC Member of more than ten conferences, such as CVPR, ICCV, ACM KDD, AAAI, ACM MM, and ECCV. He was an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA from August 2014 to August 2018. He has been an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since January 2018, the *IEEE Multimedia Magazine* since January 2018, and IEEE ACCESS since January 2017. He is a Senior Member of CIE and CCF, and a member of ACM.

Tiejun Huang (Senior Member, IEEE) received the bachelor's and master's degrees in computer science from the Wuhan University of Technology, Wuhan, China, in 1992 and 1995, respectively, and the Ph.D. degree in pattern recognition and intelligent system from the Huazhong University of Science and Technology, Wuhan, in 1998.

He is currently a Professor with the School of Electronic Engineering and Computer Science, Peking University, Beijing, China, where he is also the Director of the Institute for Digital Media Technology. He has authored or coauthored over 100 peer-reviewed papers and three books. His research area includes video coding, image understanding, digital right management, and digital library.

Prof. Huang is a member of the Board of Director for Digital Media Project, the Advisory Board of the IEEE Computing Society, and the Board of the Chinese Institute of Electronics.

Jian K. Liu received the Ph.D. degree in mathematics from the University of California at Los Angeles, Los Angeles, CA, USA, in 2009.

He is currently a Lecturer with the Centre for Systems Neuroscience, University of Leicester, Leicester, U.K. His area of research includes computational neuroscience and brain-like computation.