# Applications of Deep Learning and Reinforcement Learning to Biological Data

Mufti Mahmud⬦, *Senior Member, IEEE*, Mohammed Shamim Kaiser, *Senior Member, IEEE*, Amir Hussain, *Senior Member, IEEE*, and Stefano Vassanelli

*Abstract*—**Rapid advances in hardware-based technologies during the past decades have opened up new possibilities for life scientists to gather multimodal data in various application domains, such as *omics*, *bioimaging*, *medical imaging*, and *(brain/body)–machine interfaces*. These have generated novel opportunities for development of dedicated data-intensive machine learning techniques. In particular, recent research in deep learning (DL), reinforcement learning (RL), and their combination (deep RL) promise to revolutionize the future of artificial intelligence. The growth in computational power accompanied by faster and increased data storage, and declining computing costs have already allowed scientists in various fields to apply these techniques on data sets that were previously intractable owing to their size and complexity. This paper provides a comprehensive survey on the application of DL, RL, and deep RL techniques in mining biological data. In addition, we compare the performances of DL techniques when applied to different data sets across various application domains. Finally, we outline open issues in this challenging research area and discuss future development perspectives.**

*Index Terms*—**Bioimaging, brain–machine interfaces, convolutional neural network (CNN), deep autoencoder (DA), deep belief network (DBN), deep learning performance, medical imaging, omics, recurrent neural network (RNN).**

## I. INTRODUCTION

**T**HE need for novel healthcare solutions and continuous efforts in understating the biological bases of pathologies has pushed extensive research in biological sciences over the last two centuries [1]. Recent technological advancements in life sciences have opened up possibilities not only to study biological systems from a holistic perspective, but provided unprecedented access to molecular details of living organisms [2], [3]. Novel tools for DNA sequencing [4], gene expression (GE) [5], bioimaging [6], neuroimaging [7], and brain–machine interfaces [8] are now available to the scientific community. However, considering the inherent complexity of biological systems together with the high dimensionality,

diversity, and noise contaminations, inferring meaningful conclusion from such data is a huge challenge [9]. Therefore, novel instruments are required to process and analyze biological big data that must be robust, reliable, reusable, and accurate [10]. This has encouraged numerous scientists from life and computing sciences disciplines to embark in a multidisciplinary approach to demystify functions and dynamics of living organisms, with remarkable progress reported in biological and biomedical research [11]. Thus, many techniques of artificial intelligence (AI), in particular machine learning (ML), have been proposed over time to facilitate recognition, classification, and prediction of patterns in biological data [12].

Conventional ML techniques can be broadly categorized in two large sets—*supervised* and *unsupervised*. The methods pertaining to the *supervised* learning paradigm classify objects in a pool using a set of known annotations/attributes/features. On the other hand, the *unsupervised* learning techniques form groups/clusters among the objects in a pool by identifying their similarity, and then use them for classifying the unknowns. Further, the other category, *reinforcement learning* (RL), allows a system to learn from the experiences it gains through interacting with its environment (see Section II-B for details).

Popular *supervised* methods include artificial neural networks (ANN) [13] and their variants [e.g., multilayer perceptron (MLP)], support vector machines (SVMs) [14], linear classifiers [15], Bayesian statistics [16], k-nearest neighbors (kNNs) [17], hidden Markov model (HMM) [18], and decision trees [19]. Popular *unsupervised* methods include autoencoders [20], expectation maximization [21], self-organizing maps [22], k-means [23], fuzzy [24], and density-based [25] clustering.

A large body of evidence shows that the abovementioned methods and their respective variants can be successfully applied to biological data coming from various sources, e.g., *Omics* (covers data from genetics and (gen/transcript/epigen/prote/metabol)omics [26]), *Bioimaging* (covers data from (sub)cellular images acquired by diverse imaging techniques [27]), *Medical Imaging* (covers data from (medical/clinical/health) imaging mainly through diagnostic imaging techniques [28]), and *(brain/body)–machine interfaces (BMIs)* (covers electrical signals generated by the brain and the muscles and acquired using appropriate sensors [29], [30]).

Broadly, AI can be thought to have evolved parallelly in two main directions—expert systems (ES) and ML [see Fig. 1(h)]. Focusing on the latter, ML extracts features from training

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

2                                                                                                                    IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS
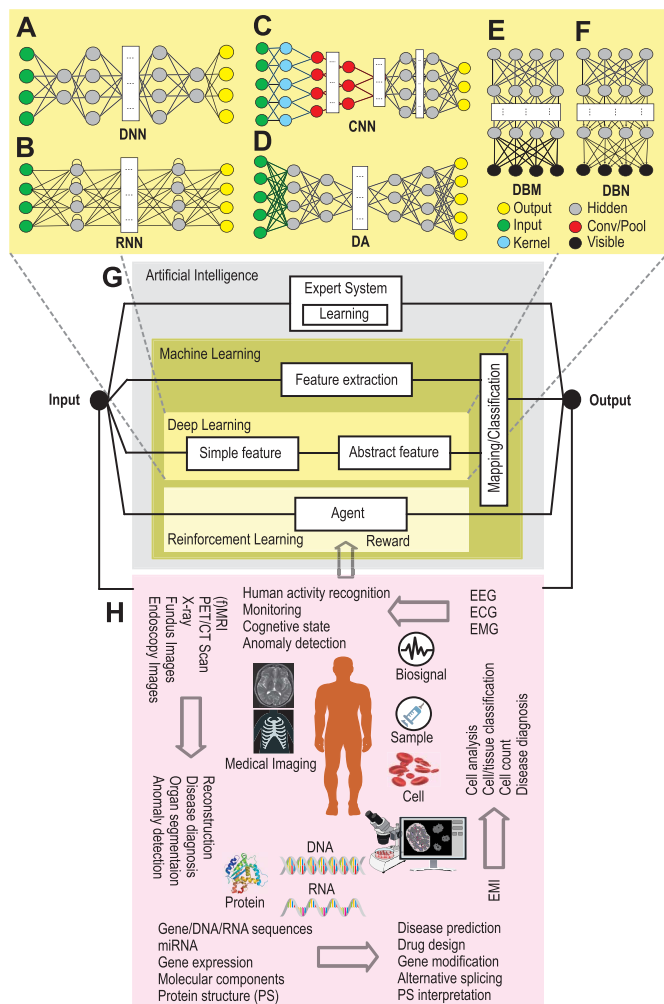


Fig. 1. Possible representation of the DL, RL, and deep RL frameworks for biological applications. (a)–(f) Popular DL architectures. (g) Schematic of the learning framework as a part of AI. Broadly, AI can be thought to have evolved parallelly in two main directions—ES and ML. ES takes expert decisions from given factual data using rule-based inferences. ML extracts features from data mainly through statistical modeling and provides predictive output when applied to unknown data. DL, being a subdivision of ML, extracts more abstract features from a larger set of training data mostly in a hierarchical fashion resembling the working principle of our brain. The other subdivision, RL, provides a software agent that gathers experience based on interactions with the environment through some actions and aims to maximize the cumulative performance. (h) Possible applications of AI to biological data.

data set(s) and make models with minimal or no human intervention. These models provide predicted outputs based on test data. Deep learning (DL), being a subdivision of ML, extracts more abstract features from a larger set of training data mostly without human supervision. RL, being the other subdivision of ML, is inspired by psychology. It provides a software agent that gathers experience based on interactions with the environment through some actions and aims to maximize the cumulative performance.

In recent years, DL, RL, and deep RL methods are poised to reshape the future of ML [31]. Over the last decade, works pertaining to DL, RL, and deep RL were extensively reviewed from different perspectives. In a topical review, Schmidhuber [32] provided a detailed time line of significant DL developments (for both supervised and unsupervised),

RL and evolutionary computation, and DL in feedforward and recurrent neural networks (RNNs) for RL. Other reviews are focusing on the applications of DL in health informatics [33], biomedicine [34], and bioinformatics [35]. On the other hand, Kaelbling *et al.* [36] discussed RL from the perspective of the tradeoff between exploitation and exploration, RL's foundation via Markov decision theory, the learning mechanism using delayed reinforcement, construction of empirical learning models, use of generalization and hierarchy, and reported some exemplifying RL systems implementations. Glorennec [37] provides a brief overview of the basis of RL with explicit descriptions of Q- and fuzzy Q-learning. With respect to applications in solving dynamic optimization problems, Gosavi [38] surveys Q-learning, temporal differences (TDs), semi-Markov decision problems, stochastic games, policy gradients, and hierarchical RL with the detailed underlying mathematics. In addition, Li [39] analyzed the recent advances of deep RL on—deep Q-network (DQN) with its extensions, asynchronous methods, policy optimization, reward, and planning as well as different applications, including games (e.g., AlphaGo, robotics, and chatbot), neural architecture design, natural language processing, personalized Web services, healthcare, and finance.

Despite the popularity of the topic and application potential to diverse disciplines, a comprehensive review is missing that focuses on data from different Biological application domains while providing a performance comparison across techniques. This review is intended to fill this gap: it provides a brief overview on DL, RL, and deep RL concepts, followed by the state-of-the-art applications of these techniques and performance comparison between various DL approaches. Finally, it identifies and outlines some open issues and speculates about future perspectives.

As for the organization of the rest of this paper, Section II provides a conceptual overview of DL, RL, and deep RL techniques, introducing the reader to underlying theory. Section III presents state-of-the-art applications of these techniques to various biological domains. Section IV presents test results and performance comparison of DL techniques applied on data sets pertaining to different biological domains. Section V highlights open issues and future perspectives, and, finally, Section VI presents some concluding remarks.

## II. CONCEPTUAL OVERVIEW

### A. Deep Learning

The core concept of DL is to learn data representations through increasing abstraction levels. Almost in all levels, more abstract representations at a higher level are learned by defining them in terms of less abstract representations at lower levels. This type of hierarchical learning process is very powerful as it allows a system to comprehend and learn complex representations directly from the raw data [40], making it useful in many disciplines [41].

Several DL architectures have been reported in the literature, including deep neural network (DNN), RNN, convolutional neural network (CNN), deep autoencoder (DA), deep Boltzmann machine (DBM), deep belief network (DBN), deep residual network, deep convolutional inverse graphics network, and so on. For the sake of brevity, only the ones widely used

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MAHMUD *et al.*: APPLICATIONS OF DL AND RL TO BIOLOGICAL DATA 3

with biological data are briefly summarized in the following. However, interested readers are directed to the references mentioned in each section for concrete mathematical details behind each architecture.

*1) Deep Neural Network:* DNN [Fig. 1(a)] [42] is inspired by the brain's visual input processing mechanism, which takes place at multiple levels (i.e., starting with cortical area "V1" and then passing to area "V2" and so on) [32]. The standard neural network (NN) is extended to have multiple hidden layers with nonlinear modules embodied in each hidden layer allowing it to learn part-hole of the representations. Though this formulation has been successfully used in many applications, the training process is slow and cumbersome.

*2) Recurrent Neural Network:* RNN [Fig. 1(b)] [43] is an NN model designed to detect structures in streams of data [44]. Unlike feedforward NN that performs computations unidirectionally from input to output, an RNN computes the current state's output depending on the outputs of the previous states. Due to this "memory-like" property, despite learning problems related to vanishing and exploding gradients, RNN applications have gained popularity in many fields involving streaming data (e.g., text mining, time series, and genomes). In recent years, two main variants, such as bidirectional RNN [45] and long short-term memory (LSTM) [46], have also been developed and increasingly applied in biological applications [47], [48].

*3) Convolutional Neural Network:* CNN [Fig. 1(c)] [49] is a multilayer NN model [50], inspired by the neurobiology of visual cortex, that consists of convolutional layer(s) followed by fully connected layer(s). In between these two types of layers, there may exist subsampling steps. They get the better of DNNs, which have difficulty in scaling well with multidimensional locally correlated input data. Therefore, the main application of the CNN has been in data sets, where the number of nodes and parameters required to be trained is relatively large (e.g., image analysis). Exploiting the "stationary" property of an image, convolution filters (CFs) can learn data-driven kernels. Applying such a CF along with a suitable pooling function reduces the features that are supplied to the fully connected network to classify. However, in case of large data sets, even this can be daunting and can be solved using sparsely connected networks. Some of the popular CNN configurations include AlexNet [51], VGGNet [52], and GoogLeNet [53].

*4) Deep Autoencoder:* DA architecture [Fig. 1(d)] [54] is obtained by stacking a number of autoencoders that are data driven NN models (i.e., unsupervised) designed to reduce data dimension by automatically projecting incoming representations to a lesser dimensional space than that of the input. In an autoencoder, an equal amount of units are used in the input/output layers and less units in the hidden layers. (Non)linear transformations are embodied in the hidden layer units to encode the given input into smaller dimensions [55]. Despite that it requires a pretraining stage and suffers from vanishing error, this architecture is popular for its data compression capability and have many variants, e.g., denoising autoencoder [54], sparse autoencoder [56], variational autoencoder [57], and contractive autoencoder [58].

*5) (Restricted) Boltzmann Machine:* A (restricted) Boltzmann machine [(R)BM] is an undirected probabilistic generative model representing specific probability distributions [59]. It is also considered as a nonlinear feature detector. The learning process of (R)BM is based on optimizing its parameters for a set of given observations to obtain the best possible fit of the probability distribution through Gibbs sampling (a Markov chain Monte Carlo (MC) method [60]) [61]. BM has symmetrical connections among its units and has one visible layer with (multiple) hidden layers. Usually, the learning process of a BM is slow and computationally expensive and, thus, requires long to reach equilibrium statistics [40]. By restricting the intralayer units of a BM to connect among themselves, a bipartite graph is formed (i.e., an RBM has a visible and a hidden layer), where the learning inefficiency is solved [59]. Stacking multiple RBMs as learning elements yields the following two DL architectures.

*a) Deep Boltzmann Machine:* DBM [Fig. 1(e)] [62] is a stack of undirected RBMs. Being undirected, there is a feedback process among the layers, where feature inference from higher level units affect the inference of lower level units. Despite this powerful inference mechanism that allows an input's alternative interpretations through concurrent competition at all levels of the model, estimating model parameters from data remains difficult. Gradient-based methods (e.g., persistent contrastive divergence [63]) fail to explore the model parameters sufficiently [62]. Though this learning problem is overcome by pretraining each RBM in a layerwise greedy fashion, with outputs of the hidden variables from lower layers as input to upper layers [59], the time complexity remains high and may not be suitable for large training data sets [64].

*b) Deep Belief Network:* DBN [Fig. 1(f)] [65] is formed by ordering several RBMs in a way that one RBM's latent layer is linked to the subsequent RBM's visible layer. The connections of DBN are downward directed to its immediate lower layer, except that the upper two layers are undirected [65]. Thus, a DBN is a hybrid model with the first two layers as undirected graphical model and the rest being directed generative model. The different layers are learned in a layerwise greedy fashion and fine-tuned based on required output [33]; however, the training procedure is computationally demanding.

*B. Reinforcement Learning*

Rooted in behavioral psychology, RL is a distinctive member of the ML family. An RL problem is solved by learning new experiences through trial-and-error. An RL *agent* is trained; as such, its *actions* to interact with the *environment* maximize the cumulative *reward* resulting from the interactions. Generally, RL problems are modeled and solved using Markov decision processes (MDPs) theory through MC and dynamic programming (DP) [66].

The learning of an agent is a continuous process, where interactions with the environment occur at discrete time steps. In a typical RL cycle (at time $t$), the agent receives the environment's state (i.e., *state*, $s_t$) and selects an action ($a_t$) to interact. The environment responds to the action and

progresses to a new state $(s_{t+1})$. The reward $(r_{t+1})$ that the agent either receives or not for the selected action associated with the *transition* $(s_t, a_t, s_{t+1})$ is also determined [66]. Accordingly, after each cycle, the agent updates the *value function* $V(s)$ or *action-value function* $Q(s, a)$ based on a certain *policy*, where *policy* $(\pi)$ is a function that maps states $s \in S$ to actions $a \in A$, i.e., $\pi : S \rightarrow A \Rightarrow a = \pi(s)$ [36].

A possible way to solve the RL problem is to describe the environment as MDP with a set of state-value function pairs, a set of actions, a policy, and a reward function. The value function can be separated to solve state-value function $(V)$ or action-value function $(Q)$. In the state-value function, the expected outcome, of being in state $s$ following policy $\pi$, is determined by sum of the rewards at future time steps with a given discount factor $(\gamma \in [0, 1])$, i.e., $V^{\pi}(s) = \mathbb{E}_{\pi}(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s)$. And in the action-value function, the expected outcome, of being in state $s$ taking action $a$ following policy $\pi$, is determined by sum of the rewards for each state action pairs, i.e., $Q^{\pi}(s, a) = \mathbb{E}_{\pi}(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s, a_t = a)$.

The MDP can be solved, and the optimum policy can be achieved through DP by: either starting with an initial policy or improving it iteratively (*policy iteration*), or starting with arbitrary value function and recursively refining an estimate of an improved state-value or action-value function to compute an optimal policy and its value (*value iteration*) [67]. In the simplest case, the state-value function for a given policy can be estimated using the Bellman expectation equation as: $V^{\pi}(s) = \mathbb{E}_{\pi}(r_{t+1} + \gamma V^{\pi}(s_{t+1})|s_t = s)$. Considering this as a *policy evaluation* process, an improved and eventually optimal policy $(\pi^*)$ can be achieved by taking actions greedily that maximizes the state-action value. But in scenarios with unknown environments, modelfree methods are to be used without the MDP. In such cases, instead of the state-value function, the action-value function can be maximized to find the optimal policy $(\pi^*)$ using a similar policy evaluation and improvement process, i.e., $Q^{\pi}(s, a) = \mathbb{E}_{\pi}(r_{t+1} + \gamma Q^{\pi}(s_{t+1}, a_{t+1})|s_t = s, a_t = a)$. There are several learning techniques, e.g., MC, TD, and state-action-reward-state-action (SARSA), which describe various aspects of the modelfree policy evaluation and improvement process [68].

However, in real-world RL problems, the state-action space is very large, and storing a separate value function for every possible state is cumbersome. In such situations, generalization of the value function through *function approximation* is required. For example, the $Q$ value function approximation is able to generalize to unknown states by calculating a function $(\hat{Q})$ for a given state action pair $(s, a)$, i.e., $\hat{Q}(s, a, \mathbf{w}) \approx Q^{\pi}(s, a) = \mathbf{x}(s, a)^{\top}\mathbf{w}$. In other words, a rough approximation of the $Q$ function is obtained from the feature vector representing $(s, a)$ pair $(\mathbf{x})$ and the provided parameter $(\mathbf{w}$ which is updated using MC or TD learning) [69]. This approximation allows to improve the $Q$ function by minimizing the loss between the true and approximated values (e.g., using gradient descent), i.e., $J(\mathbf{w}) = \mathbb{E}_{\pi}((Q_{\pi}(s, a) - \hat{Q}(s, a, \mathbf{w}))^2)$. Examples of differentiable function approximators include NN, linear combinations of features, decision tree, nearest neighbor, and Fourier bases [70].

### C. Deep Reinforcement Learning

The autonomic capability to learn without any feature crafting makes RL a powerful tool applicable to many disciplines, but it falls short in cases when the data dimensionality is large and the environment is nonstationary [71]. Also, DL's capability to learn complex patterns is sometimes prone to misclassification [72]. To mitigate, in recent years, RL algorithms have been successfully combined with a deep NN [39] giving rise to novel learning strategies. This integration has been used either in approximating RL functions using deep NN architectures or in training deep NN using RL.

The first notable example of such an integration is the DQN [31], which combines Q-learning with a deep NN. The DQN agent, when presented with high-dimensional inputs, can successfully learn policies using RL. The action-value function is approximated for optimality using the deep CNN. The deep CNN, using experience replay and target network, overcomes the instability and divergence sometimes experienced while approximating Q-function with shallow NN.

Another deep RL algorithm is the double DQN, which is an extension of the DQN algorithm [73]. In certain situations, the DQN suffers from substantial overestimations inherited from the implemented Q-learning, which are overcome by replacing the Q-learning of the DQN with a double Q-learning algorithm [74]. The DQN learns two value functions, by assigning an experience randomly to update one of them, resulting in two sets of weights. During every update, one set determines the greedy policy, while the other determines its value. Other deep RL algorithms include deep deterministic policy gradient, continuous DQN, asynchronous $N$-step Q-learning, Dueling network DQN, prioritized experience replay, deep SARSA, asynchronous advantage actor-critic, and actor-critic with experience replay [39].

## III. APPLICATIONS TO BIOLOGICAL DATA

The above-outlined techniques, also available as open-source tools (see [75] for a mini review of tools based on DL), have been used in mining biological data. The applications, as reported in literature, are provided in the following for data from each of the application domains.

Table I summarizes state-of-the-art applications of DL and RL to biological data [see Fig. 1(h)]. It also reports on individual applications in each of these domains and the data type on which the methods have been applied.

### A. Omics

Some DL and RL methods have been extensively used in *Omics* (such as genomics, proteomics, or metabolomics) research to extract features, functions, structure, and molecular dynamics from the raw biological sequence data (e.g., DNA, RNA, and amino acids). Specifically, mining sequence data is a challenging task. Different analyses (e.g., GE profiling, splicing junction prediction, sequence specificity prediction, transcription factor determination, and protein–protein interaction evaluation) dealing with different types of sequence data have been reported in the literature.

TABLE I

SUMMARY OF (DEEP) (REINFORCEMENT) LEARNING APPLICATIONS TO BIOLOGICAL DATA

| App. Dom. | Purpose | Data Type | Method & Reference | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | DL | | | | | RL |
| | | | DA | DBN | DNN | CNN | RNN$^\psi$ | |
| Omics | GE, DD, DmsPS, PrSD, DrD, PsDs | SJP/Se, miRNA, Prtn, Gene | [76–78] | [79–82] | [83–91] | [92–98] | [99], [100] | [101–104] |
| Bioimaging | Seg, BCaD, CACI, CCnt | EMI | [105] | | [106], [107] | [108–115] | | |
| Medical Imaging | Seg, ID, DD | [f/s]MRI | [116–118] | [119–122] | [123][124]$^\upsilon$ | [125–130] | [131] | |
| | | CT | [132] | | [133] | [134–141] | | |
| | | PET | [117] | [120], [121], [142] | [124]$^\upsilon$ | [143] | | |
| | | OthI | [144] | [145] | [146] | [138], [147–157] | [131] | [158] |
| BMI | MA, MD, ER, CogS | EEG | [159], [160] | [161–168][169]$^\gamma$ | [170], [171] | [172–177] | [178–181] | [182], [183] |
| | | EMG | | | | [184], [185] | | |
| | | ECG | | [186–188] | [189] | | | |
| | | NS | | | | | | [190–196] |

Legends: App. Dom.–Application Domain; $^\psi$–including LSTM; $^\upsilon$–using DPN; $^\gamma$–using Sparse-DBN; **Abbreviations in 'Purpose' column**: GE–Gene Expression; DD–Disease Diagnosis (including Alzheimer's/ Huntington/ Schizophrenia/ Mental/ Emotional State); DmsPS–DNA methylation state prediction [85], Pathogenicity [86], Sequence assembly [101–103]; PrSD–Protein Structure (binding) Determination; DrD–Drug Discovery; Seg–Segmentation; BCaD–Breast Cancer Detection; CACI–Classification & Analysis of Cell Image; CCnt–Cell Count; ID–Image Denoising; MA–Motor Action decoding; MD–Movement Decoding; ER–Emotion Recognition; CogS–Cognitive State Determination;
**Abbreviations in 'Data Type' column**: SJP–Splice Junction Prediction; Se–[DNA/RNA/ChIP/DNase] sequence & microarray GE; Prtn–Protein properties; OthI–Images not used elsewhere (e.g., UlS/EMI/EnI/XRay/CFI/MMM/cardiac MRI); NS–Neural Spikes.

To identify splicing junction at the DNA level, a tedious job to do manually, Lee and Yoon [79] proposed a DBN-based unsupervised method to perform the autoprediction. Profiling GE is a demanding job. Chen *et al.* [83] exploited a DNN-based method for GE profiling on RNA-seq and microarray-based GE Omnibus data set. The ChIP-seq data were preprocessed, using CNN, into a 2-D matrix where each row denoted a gene's transcription factor activity profile [92]. Also, somatic point mutation-based cancer classification was performed using the DNN [90]. In addition, DA-based methods have been used for feature extraction in cancer diagnosis and classification (Fakoor *et al.* [76] used the sparse DA method) in combination with related gene identification (Danaee *et al.* [77] used stacked denoising DA) from GE data.

Alipanahi *et al.* [93] used a deep CNN structure to predict DNA- and RNA-binding proteins' [(D/R)BPs] role in alternative splicing and examined the effect of disease associated genetic variants (GVs) on transcription factor binding and GE. Zhang *et al.* [84] developed a DNN framework to model structural features of RBPs. Pan and Shen [82] proposed a hybrid CNN-DBN model to predict RBP interaction sites and motifs on RNAs. Quang *et al.* [86] proposed a DNN model to annotate and identify pathogenicity in GV.

Identifying the best discriminative genes/microRNAs (miRNAs) is a challenging task. Ibrahim *et al.* [80] proposed a group feature selection method from genes/miRNAs based on expression profile using DBN and active learning. CNN was used to interpret noncoding genome by annotating them [94]. Also, Zeng *et al.* [95] employed CNN to predict the binding between DNA and protein. Zhou and Troyanskaya [96] proposed a CNN-based approach to identify noncoding GV, which was also used by Huang *et al.* [97] for a similar purpose. Park *et al.* [99] proposed an LSTM-based tool to automatically predict miRNA precursor. Also, Lee *et al.* [100] presented a deep RNN framework for automatic miRNA target prediction.

DNA methylation (DM) causes DNA segment activity alteration without affecting the sequence, and thus, detecting its state in a sequence is important. Angermueller *et al.* [85] used a DNN-based method to estimate a DM state by predicting the changes in single nucleotides and uncovering sequence motifs.

Proteomics pose many complex computational problems to solve. Estimating complete protein structures from biological sequences, in 3-D space, is a complex and NP-hard problem. Alternatively, the protein structures can be divided into independent subproblems (e.g., torsion angle, access surface area, and dihedral angles) and solved in parallel, and estimate the secondary protein structures (2-PS). Predicting compound–protein interaction (CPI) is very interesting from drug discovery point of view and tough to solve.

Heffernan *et al.* [87] proposed an iterative DNN scheme to solve these subproblems for 2-PS. Wang *et al.* [98] utilized a deep CNN to predict 2-PS. Li [78] proposed the DA learning-based model to reconstruct a protein structure based on a template. Also, DNN-based methods to predict CPI [88], [89], [91] have also been reported.

In medicine, model organisms are often used for translational research. Chen *et al.* [81] used bimodal DBNs to predict responses of human cells under certain stimuli

based on responses of rat cells obtained with the same stimuli.

RL has also been used in omics, for example, Chuang *et al.* [101] used binary particle swarm optimization (PSO) and RL to predict bacterial genomes, Ralha *et al.* [102] used RL through a system called BioAgent to increase the accuracy of biological sequence annotation, and Bocicor *et al.* [103] solved the problem of DNA fragment assembly using the RL-based framework. Zhu *et al.* [104] proposed a hybrid RL method, with text mining, for constructing protein–protein interaction networks.

### B. Bioimaging

In biology, DL architectures targeted on pixel levels of a biological image to train the NN. Ning *et al.* [108] used CNN for pixelwise image segmentation of nucleus, cytoplasm, cell, and nuclear membranes using electron microscope image (EMI). Reduced pixel noise and better abstract features of biological images can be obtained by adding multiple layers. Ciresan *et al.* [109], [110] employed deep CNNs to identify mitosis in histology images of the breast, and a similar architecture was also used to find neuronal membranes and automatically segment neuronal structures in EMI. Xu *et al.* [105] used a stacked sparse DA architecture to identify nuclei in the histopathology images of the breast cancer. Xu *et al.* [106] classified colon cancer images using multiple instance learning (MIL) from DNN learned features.

Besides pixel-level analysis, DL has also been applied to cell- and tissue-level analysis. Chen *et al.* [107] employed DNN in labelfree cell classification. Parnamaa and Parts [111] used the CNN to automatically detect fluorescent protein in various subcellular localization patterns using microscopy images of yeast. Ferrari *et al.* [112] used CNNs to count bacterial colonies in agar plates. Kraus *et al.* [113] integrated both the segmentation and the classification in a model, which can be utilized to classify the microscopy images of the yeast. Flow cytometry is used in cellular biology through a cycle analysis to monitor different stages of a cell cycle. Eulenberg *et al.* [114] proposed a deep flow model, combining nonlinear dimension reduction with CNN, to analyze single-cell flow cytometry images. Furthermore, a CNN architecture was employed to segment and recognize neural stem cells in images taken by bright field microscope [115], and DBN for analyzing gold immunochromatographic strip [197].

### C. Medical Imaging

DL and RL architectures have been widely used in analyzing medical images obtained from—magnetic resonance [(f/s)MRI], CT scan, positron emission tomography (PET), radiography/fundus (e.g., X-ray and CFI), microscope, ultrasound (UlS)—to denoise, segment, classify, detect anomalies, and diseases from these images.

Segmentation is a process of partitioning an image based on some specific patterns. Sirinukunwattana *et al.* [156] reported the results of the gland segmentation competition from colon histology images. Kamnitsas *et al.* [130] proposed a 3-D dual pathway CNN to simultaneously process multichannel

MRI and segment lesions related to tumors, traumatic injuries, and ischemic stroke. Stollenga *et al.* [131] segmented neuronal structures from 3-D EMI and brain MRI using multidimensional RNN. Fritscher *et al.* [134] used a deep CNN for volume segmentation from head-neck region's CT scans. Havaei *et al.* [123], [125] segmented brain tumor from MRI using CNN and DNN. Brosch and Tam [119] proposed a DBN-based manifold learning method of 3-D brain MRI. Cardiac MRIs were segmented for heart's left ventricle using the DBN [145], and blood pool (BP) and myocardium (MC) using the CNN [157]. Mansoor *et al.* [116] automatically segmented anterior visual pathway from MRI sequences using a stacked DA model. Lerouge *et al.* [133] proposed the DNN-based method to label CT scans.

Success of many medical image analysis methods depends on image denoising. Gondara [144] proposed a denoising technique utilizing convolutional denoising DA, and validated it with mammograms and dental radiography, while Agostinelli *et al.* presented an adaptive multicolumn stacked sparse denoising DA method for image denoising, which was validated using CT scan images of the head [132].

Detecting anomaly in medical images is widely used for disease diagnosis. Several models were applied to detect Alzheimer's disease (AD) and mild cognitive impairment (MCI) from MRI and PET scans, including DA [117], [118], DBM [120], RBM [121], and multimodal stacked deep polynomial network (MStDPN) [124].

Due to its facilitating structure, a CNN has been the most popular DL architecture for an image analysis. The CNN was applied to classify breast masses from mammograms (MMM) [151]–[155], diagnose AD using different neuroimages (e.g., brain MRI [126], brain CT scans [135], and (f)MRIs [128]), and rheumatoid arthritis from hand radiographs [150]. The CNN was also used extensively: on CT scans to detect anatomical structure [136], sclerotic metastases of spine along with colonic polyps and lymph nodes (LNs) [137], thoracoabdominal LN and interstitial lung disease (ILD) [139], pulmonary nodules [138], [140], [141]; on (f)MRI and diffusion tensor images to extract deep features for brain tumor patients' survival time prediction [129]; on MRI to detect neuroendocrine carcinoma [127]; on UlS images to diagnose breast lesions [138] and ILD [147]; on CFI to detect hemorrhages [148]; on endoscopy images to diagnose digestive organ-related diseases [149]; and on PET images to identify oesophagal carcinoma and predict responses of neoadjuvant chemotherapy [143].

In addition, a DBN was successfully applied to identify attention deficit hyperactivity disorder [142], and schizophrenia and Huntington disease from (f/s)MRI [122]. And, a DNN-based method was proposed to successfully identify the fetal abdominal standard plane in UlS images [146].

RL was used in segmenting transrectal UlS images to estimate location and volume of the prostate [158].

### D. (Brain/Body)–Machine Interfaces

DL and RL methods have been applied to BMI signals [e.g., electroencephalogram (EEG), electrocardiogram (ECG),

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MAHMUD *et al.*: APPLICATIONS OF DL AND RL TO BIOLOGICAL DATA

7

and electromyogram (EMG)] mainly from (brain) function decoding and anomaly detection perspectives.

Various DL architectures have been used in classifying EEG signals to decode Motor Imagery (MoI). The CNN was applied in the classification pipeline using—augmented common spatial pattern (CSP) features which covered various frequency ranges [172], features based on combined selective location, time, and frequency attributes, which were then classified using DA [173], and signal's dynamic energy representation [174]. The DBN was also employed—in combination with softmax regression to classify signal frequency information as features [161] and in conjunction with Ada-boost algorithm to classify single channels [162]. DNN was used with variance-based CSP features to classify MoI EEG [171] and to find neural patterns occurring at each time points in single trials, where the input heatmaps were created with a layerwise relevance propagation technique [170]. In addition, MoI EEG signals were classified by denoising DA using multifractal attribute features [159].

The DBN was used by Li *et al.* [163] to extract low-dimensional latent features as well as critical channel selection that led to an early framework for affective state classification using EEG signals. In a similar work, Jia *et al.* [164] used a semisupervised approach with active learning to train DBN and generative RBMs for the classification. Later, using differential entropy as features to train the DBN, Zheng and Lu examined dominant frequency bands and channels of EEG in an emotion recognition system [165]. Jirayucharoensak *et al.* used principal component analysis (PCA) extracted power spectral densities from each EEG channel, which were corrected by covariate shift adaptation to reduce nonstationarity, as features to stacked DA to detect emotion [160]. Tripathi *et al.* [175] explored the DNN (with Softmax activator and Dropout) and CNN [198] (with Tan Hyperbolic, Max Pooling, Dropout, and Softplus) for emotion classification from the DEAP data set using EEG signals and response face video. Using the similar data from the MAHNOB-HCI data set, Soleymani *et al.* [178] detected continuous emotion using the LSTM-RNN. Channelwise CNN and its variant with RBM [176], and autoregressive-model based features with sparse-DBN [169], was used to estimate driver's cognitive states using EEG data.

In another approach to model cognitive events, EEG signals were transformed to time-lagged multispectral images and fed to the CNN for learning the spectral and spatial representations of each image, followed by an adapted LSTM to find the temporal patterns in the image sequence [179].

The DBN has been employed in classifying EEG signals for anomaly detection in diverse scenarios, including online waveform classification [166], AD diagnosis [167], integrated with HMM to understand sleep phases [168]. To detect and predict seizures, the CNN was used for classification of synchronization patterns [177]; RNN predicted specific signal features related to seizure after being trained with data preprocessed by wavelet decomposition [180]. Also, a lapse of responsiveness warning system was proposed using the LSTM [181].

Using the CNN, Atzori *et al.* [184] and Park and Lee [185] decoded hand movements from EMG signals.

ECG arrhythmias were successfully detected using the DBN [188] and DNN [189]. The DBN was also used to classify ECG signals acquired with two leads [187], and in combination with nonlinear SVM and Gaussian kernel [186].

RL has also been applied in the BMI research. Concentrating mainly on controlling (prosthetic/robotic) devices, several studies have been reported, including mapping neural activity to intended behavior through coadaptive BMI [using TD($\lambda$)] [190], symbiotic BMI (using actor-critic) [191], a testbed targeting center-out reaching task in primates for creating more realistic BMI control models [192], Hebbian RL for adaptive control by mapping neural states to prosthetic actions [193], BMI for unsupervised decoding of cortical spikes in multistep goal-directed tracking task [using Q($\lambda$)] [194], adaptive BMI capable of adjusting to dramatic reorganizing neural activities with minimal training and stable performance over long duration (using actor-critic) [195], and BMI for efficient nonlinear mapping of neural states to actions through sparsification of state-action mapping space using quantized attention-gated kernel RL as an approximator [196]. Also, Lampe *et al.* [182] proposed BMI capable of transmitting imaginary movement-evoked EEG signals over the Internet to remotely control a robotic device, and Bauer and Gharabaghi [183] combined RL with the Bayesian model to select dynamic thresholds for an improved performance of restorative BMI.

## IV. PERFORMANCE ANALYSIS AND COMPARISON

Comparative test results, in the form of performances/accuracies of each DL technique when applied to the data coming from *Omics* (Fig. 2), *Bioimaging* (Fig. 3), *Medical Imaging* (Fig. 4), and *BMIs* (Fig. 5), are summarized in the following to facilitate the reader in selecting the appropriate method for h(is/er) research. The reported performances can be regarded as a metric to evaluate the strengths/weaknesses of a particular technique with a given set of parameters on a specific data set. It should be noted that several factors (e.g., data preprocessing, network architecture, feature selection and learning, and parameters' optimization) collectively determine the accuracy of a method.

In Figs. 2–5, each group of bars indicates the accuracies/performances of comparable DL or non-DL techniques when applied to the same data and reported in an individual study. And, each bar in a group shows the (mean) performance of different runs of a technique on either multiple subjects/data sets (for means, error bar is $\pm$ standard deviation). Many of the acronyms used in the performance comparison are defined in the legends of the figures.

### A. Omics

Fig. 2(a) reports that DBN outperforms other methods in predicting splice junction when applied to two data sets from Whole Human Genome database (GWH-donor and GWH-acceptor) and two from UCSC genome database (UCSC-hg19 and UCSC-hg38) [79]. In the GWH data sets, the DBN-based method achieved superior F1-score (0.81 and 0.75) against the SVM-based methods with radial basis (SVM-RBF; 0.77 and 0.67), sigmoid (SVM-SF; 0.71 and 0.56) functions, and other splicing techniques, such as gene splicer (GS; 0.74 and 0.75) and splice machine (SM; 0.77 and 0.73).

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

8                                                                                          IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS
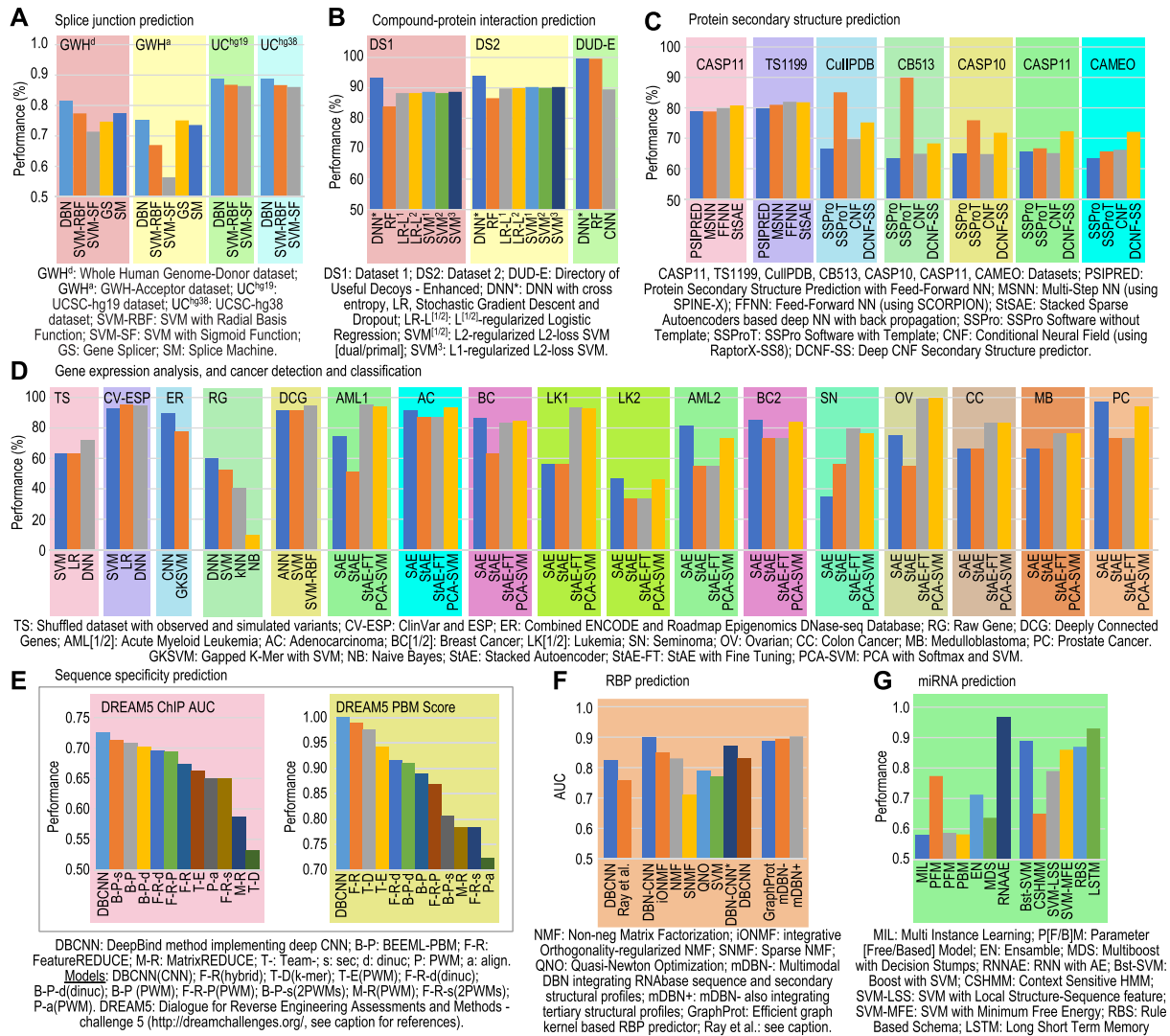
Fig. 2. Performance comparison of representative DL techniques when applied to Omics data in (a) predicting splice junction, (b) CPIs, (c) secondary/tertiary structures of proteins, (d) analyzing GE data and classifying and detecting cancers from them, (e) predicting DNA- and RNA-sequence specificity (details about DREAM5 can be found in [201] and [202]), (f) RNA binding proteins, and (g) micro-RNA precursors. Ray *et al.*'s method [199].

Also, in the UCSC data sets, the DBN achieved highest classification accuracy (0.88 and 0.88) in comparison with SVM-RBF (0.868 and 0.867) and SVM-SF (0.864 and 0.861).

Performance comparison of CPI is shown in Fig. 2(b). Tested over two CPI data sets, a DNN-based method (DNN*) achieved superior prediction accuracy (93.2% in data set 1 and 93.8% in data set 2) compared with other methods based on RF (83.9% and 86.6%), logistic regression (88.3% using LR-L$^1$ and 89.9% using LR-L$^2$), and SVM (88.7% and 90.3% using SVM$^3$) [88]. In another study, a similar DNN* was applied on the DUD-E data set, where it achieved higher accuracy (99.6%) over RF-based (99.58%) and CNN-based (89.5%) methods [89]. As per the accuracies reported in [88], the RF-based method had lower values in comparison with the LR- and SVM-based methods, which had similar values. When applied on DUD-E data set (reported in [89]), the RF-based method outperforms the CNN-based method. This may be attributed to the fact that, classification problems are data dependent and despite being one of the best classifiers [200], RF performs poorly on the DUD-E data set.

In predicting 2-PS, DL-based methods outperform other methods [see Fig. 2(c)]. When applied on two data sets (CASP11 and TS1199), the stacked sparse autoencoder (StSAE)-based method achieved superior prediction accuracy (80.8% and 81.8%) in comparison with other NN-based methods (FFNN: 79.9% and 82%, MSNN: 78.8% and 81%, and PSIPRED: 78.8% and 79.7%) [87]. Another DL method with conditional neural fields, when tested on five different data sets (CullPDB53, CB513, CASP1054, CASP1155, and CAMEO), better predicted the 2-PS (Q8 accuracy: 75.2%, 68.3%, 71.8%, 72.3%, and 72.1%) in comparison with other nontemplate-based methods (SSPro: 66.6%, 63.5%, 64.9%, 65.6%, and 63.5% and CNF: 69.7%, 64.9%, 64.8%, 65.1%, and 66.2%) [98]. However, when a template of solved protein structure from protein data bank (PDB) was used, the SSPro with template obtained the best accuracy (SSProT: 85.1%, 89.9%, 75.9%, 66.7%, and 65.7%).

To annotate GV in identifying pathogenic variants from two data sets [TS and CVESP in Fig. 2(d)], a DNN-based method performed better (72.2% and 94.6%) than LR-based (63.5%

and 95.2%) and SVM-based (63.1% and 93%) methods. Another CNN-based approach to predict DNA sequence accessibility was tested on data from ENCODE and REC databases and was reported to outperform gapped k-mer SVM method (a mean area under the receiver operating characteristic curve or AUC of 0.89 versus 0.78) [94]. In classifying cancer based on somatic point mutation using raw TCGA data containing 12 cancer types, a DNN-based method outperformed non-DL methods [60.1% versus (SVM: 52.7%, kNN: 40.4%, and Naive Bayes (NB): 9.8%)] [90]. To detect breast cancer using GE data from the TCGA database, a stacked denoising autoencoder (StDAE) was employed to extract features. According to the reported accuracies of non-DL classifiers (ANN, SVM, and SVM-RBF), StDAE outperformed other feature extraction methods, such as PCA and kernel-PCA (SVM-RBF classification accuracies for StDAE, PCA, and kernel-PCA were 98.26%, 89.13%, and 97.32%, respectively) [77]. Also, deeply connected genes were better classified with StDAE extracted features (accuracies—ANN: 91.74%, SVM: 91.74%, and SVM-RBF: 94.78%) [77]. Another study on classifying cancer, using 13 different GE data sets taken from the literature, reported that the use of PCA in data dimensionality reduction, before applying SAE, StAE, and StAE-FT for feature extraction, facilitates more accurate extraction of features [except AC and OV in Fig. 2(d)] for classification using SVM with Gaussian kernel [76].

Sequence specificities of (D/R)BPs prediction was performed more accurately using a deep CNN-based method in comparison with other non-DL methods that participated at the DREAM5[1] challenge [93]. As seen in Fig. 2(e), the CNN based method (DeepBind-CNN or DBCNN) outperformed other methods in ChIP AUC values (top two values- DBCNN: 0.726 versus BEEML-PBM_sec: 0.714) and PBM scores (two top scores- DBCNN: 0.998 versus FeatureREDUCE: 0.985) [201], [202].

Moreover, in predicting RBP, DL-based methods outperformed non-DL methods, as seen in Fig. 2(f). As reported using CLIP AUC values, the DBCNN outperforms Ray *et al.* (0.825 versus 0.759) [93], multimodal DBN outperforms GraphProt (0.902 versus 0.887) [84], and DBN-CNN hybrid outperforms NMF-based methods (0.9 versus 0.85) [82].

Also, in predicting miRNA precursor Fig. 2(g), an RNN with DA outperformed other non-DL methods [RNNAE: 0.97 versus (MIL: 0.58, PFM: 0.77, PFM: 0.59, PBM: 0.58, EN: 0.71, and MDS: 0.64)] [100]. And LSTM outperformed SVM methods [LSTM: 0.93 versus (Boost-SVM: 0.89, CSHMM: 0.65, SVM-LSS: 0.79, SVM-MFE: 0.86, and RBS: 0.8)] [99].

*B. Bioimaging*

A DNN was used for detecting 12 different cellular compartments from microscopy images and reported to have achieved a classification accuracy of 87% compared to 75% for RF [111]. The mean performance of the detection was 83.24% ± 5.18% using the DNN and 69.85% ± 6.33% using the RF [Fig. 3(a)]. In classifying flow cytometry images for

[1]DREAM5 challenge: http://dreamchallenges.org/ and [201], [202].
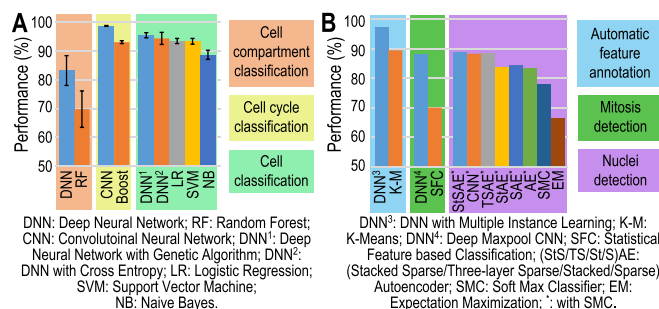


Fig. 3. Performance comparison of some DL and conventional ML techniques when applied to bioimaging application domain. (a) Performances in classifying EMIs for cell compartments, cell cycles, and cells. (b) Performances in analyzing images to automatically annotate features and detect mitosis and cell nuclei.

cell-cycle phases, a deep CNN with nonlinear dimension reduction outperformed boosting (98.73% ± 0.16% versus 93.1% ± 0.5%) [Fig. 3(a)] [114]. The DNN, trained using a genetic algorithm with AUC as a cost function, performed labelfree cell classification at higher accuracy (95.5% ± 0.9%) than SVM with Gaussian kernel (94.4% ± 2.1%), LR (93.5% ± 0.9%), NB (93.4% ± 1%), and DNN trained with cross entropy (88.7% ± 1.6%) [Fig. 3(a)] [107].

Colon histopathology images were classified with higher accuracy using the DNN and MIL (97.44%) compared with K-means clustering (89.43%) [106] [Fig. 3(b)]. Deep maxpooling CNN detected mitosis in breast histology images with higher accuracy (88%) in comparison with statistical-feature-based classification (70%) [Fig. 3(b)] [109]. Using StSAE with softmax classifier (SMC), nuclei were more accurately detected from breast cancer histopathology images (88.8% ± 2.7%) when compared with other techniques with SMC–CNN (88.3% ± 2.7%), three-layer SAE (88.3% ± 1.9%), StAE (83.7% ± 1.9%), SAE (84.5% ± 3.8%), AE (83.5% ± 3.3%), SMC alone (78.1% ± 4%), and EM (66.4% ± 4%) [Fig. 3(b)] [105].

*C. Medical Imaging*

Comparative test results on the performance of various DL/non-DL techniques in segmenting medical images to detect pathology or organ parts are reported in Fig. 4(a). A multiscale, dual pathway, 11-layered, 3-D CNN-based method with conditional random fields outperformed an RF method (DSC metric values: 63±16.3 versus 54.8±18.5) when segmenting brain lesion in MRIs obtained from the TBI database. The classifier's accuracy improved when three similar networks were ensembled (i.e., used as an ensembled method) and their outputs were averaged (64.5±16.3 versus 63±16.3) [130]. In a similar task, a (two pathway/cascaded) CNN trained using a two-phase training procedure, with local and global features, outperformed other methods participating at the MCCAI-BRATS2013[2] as reported using Dice coefficients (InputCascadeCNN: 0.88 versus Tustison: 0.87) [125]. The StAE-based method performed similarly or superior to other non-DL methods [see "ON" in Fig. 4(a), DSC values–StAE:

[2]See [203] for MICCAI-BRATS2013.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

10            IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS



**A**   Image Segmentation

MS3D(CNN-CRF): Multi Scale 3D Deep CNN coupled with a 3D fully connected Conditional Random Field; ESM: Ensemble Segmentation Method; ESM-CRF: ESM with CRF; ICCNN: Input Cascade CNN; MAP-MRF: Maximum A Priori estimation with Markov Random Fields; LCCNN: Local Cascade CNN; LPCNN: Local Path CNN; DF-CFCRF: Density Forest with Classification Forest and CRF; MRF-HM: Markov Random Field with Histogram Matching; MCP-LOO: Multi Channel Patch with Leave-One-Out; StAE: Stacked Autoencoder; MAF: Multi Atlas Framework; DCAN: Deep Contour-Aware Network; MPCNN[1/2]: Multi Path CNN [with/without] Border Netowrks; UCNN-MHF: U-shape CNN with Morphological Hole-Filling; MFR-FCN: Fully Conv. Nets with Multi-level Feature Representations; UCNN-CCL: U-shaped CNN with Connected Component Labeling; CNN-StAE: CNN with StAE; TLVSF: 3D Time Left Ventricle Segmentation Framework; DRLS-DBN: Distance Regularized with DBN; GMM-DP: Gaussian-Mixture Model with Dynamic Programming; MF-GVF: Morphological Filtering with Gradient Vector Flow; TSST-RRDP: Topological Stable-State Thresholding with Region Restricted Dynamic Programming; DCNN: Dilated CNN; TVRF: Total Variation RF; 3DUNet: 3D UNet CNN architecture; 3DDSN: 3D Deep Supervision Network.

**B**   Image denoising

Noisy: Noisy data; NL-Mean: Non Local Mean; MedF: Median Filter; CNNDAE[a]: Stacked Denoising Autoencoders with Convolutional Layers using small dataset; CNNDAE[b]: CNNDAE using combanied dataset; MCStSDA: Multi-Column Stacked Sparse DAE; AMCStSDA: Adaptive MCStSDA. GN: Gaussian Noise; PN: Poisson Noise; SPN: Salt & Pepper Noise; SN: Speckle Noise.

**C**   Anomaly and disease detection - I (Breast mass detection)

L/U: Labelled/Unlabelled features; CNN3: 2 convolutional layers with 1 fully connected layer; CNN2: 1 convolutional layer with 1 fully connected layer; HGD: Histogram of Gradient Divergence; HOG: Histogram of Oriented Gradients; DeCAF: Deep Convolutional Activation Feature; CNNPT: CNN with Pre-Training; RF-CNNPT: RF with CNNPT; RF-HF: RF with Hand-crafted Features; SFS-LDA: Sequential Forward Selection with LDA; AANN: Auto-Associator NN; SCNN: Soft Clustered NN; SVM-ELM: SVM with Extreme Learning Machine; SSVM: Structured SVM; CNN-LSVM: CNN with Linear SVM; SVM-MLP: SVM with MLP; DWT-GMB: Discrete Wavelet Transform with Gray Level Co-occurrence Matrix with Back Propagating NN.

**D**   Anomaly and disease detection - II (AD/MCI/NC detection)

SAE: Sparse Autoencoder; DSAE: Deep Sparse Autoencoder; DBM: Deep Boltzmann Machine; MK-SVM: Multi-Kernel SVM; DSAE-SLR: Deep Sparse Autoencoder with Softmax Logistic Regressor; SAE-3DCNN: SAE with 3 Dimentional CNN; ISML: Inherent Structure-based Multiview Learning; DDL: Deep Learning with Dropout; DSA-3DCNN: Deeply Supervised Adaptable 3 Dimensional CNN; SK-SVM: Single Kernel SVM; SRBMD: Stacked Restricted Boltzmann Machine With Dropout; SRBMWOD: Stacked Restricted Boltzmann Machine Without Dropout; MK-SVM: Multi-Kernel SVM; MTL: MultiTask Learning; SDSAE: Stacked Denoising Sparse Autoencoder; MSDPN-SVM: Multimodal Stacked Deep Polynomial Netowrk with SVM; MSDPN-LC: MSDPN with Linear Classifier.

**E**   Anomaly and disease detection - III (Other disease detection)

SVM-RBF: SVM with RBF; RT-SVM: Ranklet Transform with SVM; CT-SVM: Curvelet Transform with SVM; StDAE: Stacked Denoising Autoencoder; DAE-BDT: DAE with Binary Decision Tree; QRA: Quantitative Radiomics Approach; MC-CNN: Multi-crop CNN; MTANN: Massive-Training ANN; SCNN: Shallow CNN; LeNet: CNN's LeNet arch.; RDCNN: Relatively Deep CNN; AlexNet: CNN's AlexNet arch.; FT-AlexNet: Fine Tuned AlexNet; SIFT-SVM: Scale Invariant Feature Transform with Locality Constraint based Vector Sparse Coding and SVM; CNN-SVM: CNN with SVM; HCNN-ELM: Hybrid CNN with Extreme Learning Machine; HF: Handcrafted Features; HF-SIFT: HF with SIFT; HF2D-CNN: HF with 2D CNN; HFCNN-PCA: HF with 3D CNN and PCA; HoG-SVM: Histogram of oriented Gradients with SVM; 3/1S-CNN: 3/1 Slices CNN; GB: Gradient Boosting; *: with PCA.
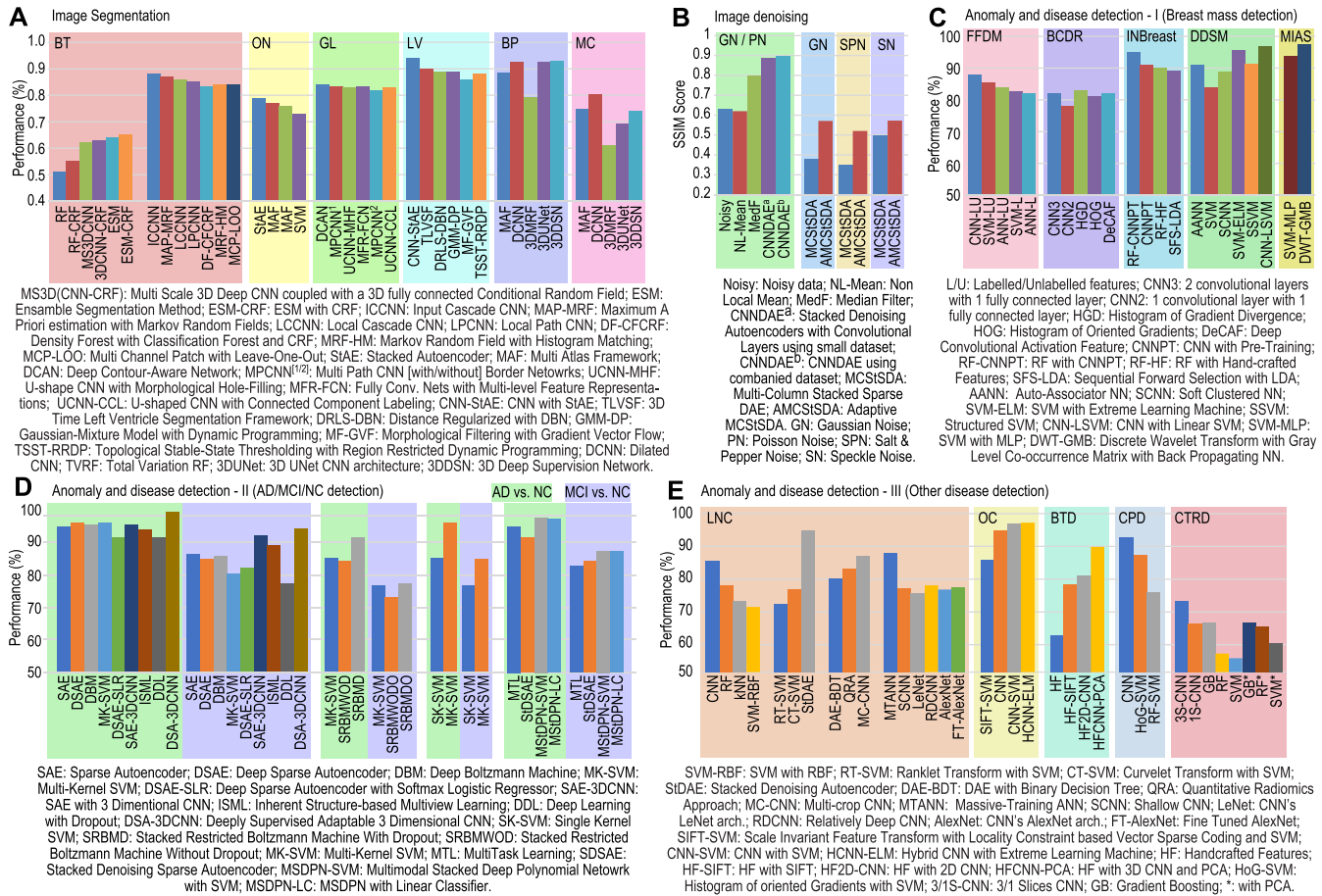
Fig. 4.   Performance comparison of representative DL techniques when applied to medical imaging. (a) Performance of image segmentation techniques in segmenting tumors (BT: brain tumor), and different organ parts (ON: optic nerve, GL: gland, LV: left ventricle of heart, BP: blood pool, and MC: myocardium). (b) Image denoising techniques to improve image quality during the presence of GN, PN, SPN, and SN. (c) Detecting anomalies and diseases in mammograms. (d) Classification and detection of AD and MCI, along with healthy controls (NC). (e) Performance of prominent techniques for LNC, organ classification, brain tumor detection, colon polyp detection, and chemotherapy response detection.

0.79 versus (MAF: 0.77, MAF: 0.76, and SVM: 0.73)] in segmenting optic nerve from MRI data [116]. Several DL methods were evaluated for identifying glands in colon histology images, and a DCAN-based method outperformed other CNN-based methods at the GlaS contest[3] [DCAN: 0.839 versus (MPCNN[1]: 0.834, UCNN-MHF: 0.831, MFR-FCN: 0.833, MPCNN[2]: 0.819, and UCNN-CCL: 0.829)] [156]. Also, left ventricles were segmented from cardiac MRI, where a CNN-StAE-based method outperformed other methods [CNN-StAE: 0.94 versus (TLVSF: 0.9, DRLS-DBN: 0.89, GMM-DP: 0.89, MF-GVF: 0.86, and TSST-RRDP: 0.88)] [204]. In segmenting volumetric medical images for BP and MC, CNN-based methods outperformed other methods as reported using Dice coefficients—BP (MAF: 0.88, DCNN: 0.93, 3DMRF: 0.87, TVRF: 0.79, 3DUNet: 0.926, and 3DDSN: 0.928) and MC (MAF: 0.75, DCNN: 0.8, 3DMRF: 0.61, TVRF: 0.5, 3DUNet: 0.69, and 3DDSN: 0.74) [157].

DL-based methods outperformed other methods in denoising MMM and dental radiographs [144], and brain CT scans [132] [Fig. 4(b)]. StDAE-CNN performed more accurate denoising in the presence of Gaussian noise (GN)/Poisson noise (PN) as reported using structural SIMilarity (SSIM)

index scores (Noisy: 0.63, NL Means: 0.62, MedFilt: 0.8, CNNDAE[a]: 0.89, and CNNDAE[b]: 0.9) [144]. Adaptive MCStSDA outperformed MCStSDA in denoising CT images as reported using peak signal-to-noise ratio (PSNR) values for GN, salt and pepper (SPN), and speckle noise (SN) (SSIM[4] scores—GN: 26.5 versus 22.7, SPN: 25.5 versus 22.1, and SN: 26.6 versus 25.1) [132].

CNN-based methods performed very well in detecting breast mass and lesion in MMM obtained from different data sets [see Fig. 4(c)]. For MMM obtained from the FFDM database, trained with labeled and unlabeled features, CNN outperformed other methods (CNN-LU: 87.9%, SVM-LU: 85.4%, ANN-LU: 84%, SVM-L: 82.5%, and ANN-L: 81.9%) [153]. In detecting masses in MMM from the BCDR database, the CNN with two convolution layers and one fully connected layer (CNN3) performed similar to other methods (CNN3:82%, HGD: 83%, HOG: 81%, and DeCAF: 82%), and the CNN with one convolution layer and one fully connected layer performed poorly (CNN2: 78%) [151]. Pretrained CNN with RF outperformed other methods [e.g., RF with handcrafted features and sequential forward selection with linear discriminant analysis (LDA)] while analyzing MMM

[3]See [156] for MICCAI15-GlaS.

[4]SSIM = (PSNR − 15.0865)/20.069, with $\sigma_{fg} = 10^2$ [205].

from the INBreast database (RF-CNNPT: 95%, CNNPT: 91%, RF-HF: 90%, and SFS-LDA: 89%) [155]. In yet another study, the CNN with a linear SVM outperformed other methods on MMM from the DDSM database (CNN-LSVM: 96.7%, SVM-ELM: 95.7%, SSVM: 91.4%, AANN: 91%, SCNN: 88.8%, and SVM: 83.9%) [152]. However, for the MMM from the MIAS database, a DWT with backpropagating NN outperformed its SVM/CNN counterparts (DWT-GMB: 97.4% versus SVM-MLP: 93.8%) [152].

Despite having been all applied on images from the ADNI database, reported methods displayed different performances in detecting and classifying "AD versus NC" and "MCI versus NC" (AD and MCI, in short) varied greatly [see Fig. 4(d)]. An approach employing deep-supervised-adaptable 3-D-CNN (DSA-3-D-CNN) outperformed other DL and non-DL methods, as reported using their accuracies, in detecting AD and MCI [AD, MCI—DSA-3-D-CNN: 99.3%, 94.2% versus (DSAE: 95.9% and 85%, DBM: 95.4% and 85.7%, SAE-3-D-CNN: 95.3% and 92.1%, SAE: 94.7% and 86.3%, DSAE-SLR: 91.4% and 82.1%, MK-SVM: 96% and 80.3%, ISML: 93.8% and 89.1%, and DDL: 91.4% and 77.4%)] [126]. A stacked RBM with dropout-based method outperformed the same method without dropout and multikernel-based SVM method in detecting AD and MCI [AD, MCI—SRBMDO: 91.4% and 77.4% versus (SRBMWODO: 84.2% and 73.1% and MK-SVM: 85.3% and 76.9%)] [121]. In another method with StAE extracted features, MK-SVM was more accurate than SK-SVM method [(AD, MCI)—MK-SVM: 85.3% and 76.9% versus SK-SVM: 95.9% and 85%)] [117]. Another method, where features from MRI and PET were fused and learned using MStDPN algorithm, outperformed other multimodal learning methods in detecting AD and MCI (AD, MCI—MTL: 95.38% and 82.99%, StSDAE: 91.95% and 83.72%, MStDPN-SVM: 97.13% and 87.24%, and MStDPN-LC: 96.93% and 86.99%) [124].

Different techniques reported varying accuracies in detecting a range of anomalies from different medical images [Fig. 4(e)]. The CNN had a better accuracy in classifying ILD (85.61%) when compared with RF (78.09%), kNN (73.33%), and SVM-RBF (71.52%) [147]. Lung nodule classification (LNC) was accurately done using StDAE (95%) compared with RT-SVM (72.5%) and CT-SVM (77%) [138]. Multicrop CNN achieved a better accuracy (87.14%) than the DAE with binary DT (80.29%) and quantitative radiomics-based approach (83.21%) in LNC [140]. MTANNs outperformed CNNs variants in LNC [MTANN: 88.06% versus (SCNN: 77.09%, LeNet: 75.86%, RDCNN: 78.13%, AlexNet: 76.85%, and FT-AlexNet: 77.55%)] [141]. A hierarchical CNN with extreme learning machine (ELM) outperformed other CNN and SVM methods [HCNN-NELM: 97.23% versus (SIFT-SVM: 89.79%, CNN: 95%, and CNN-SVM: 97.05%)] in classifying digestive organs [149]. A multichannel CNN with PCA and handcrafted features better detected BT (89.85%) in comparison with 2-D-CNN (81.25%), scale-invariant transform (78.35%), and manual classification with handcrafted features (62.8%) [129]. A 2-D-CNN-based method trained with stochastic gradient descent learning outperformed other non-DL methods [AUC

Fig. 5. Accuracy comparison of DL and conventional ML techniques when applied to BMI signals. (a) Performance comparison in detecting motor imagery, recognizing emotion and cognitive states (ER), and detecting anomaly (AnD) from EEG signals. (b) Accuracies of MD from EMG signals. (c) Accuracies of ECG signal classification.

values—CNN: 0.93 versus (HoG-SVM: 0.87 and RF-SVM: 0.76)] in detecting colon polyp from CT colonographs [137]. In addition, 3Slice-CNN was successfully employed to detect chemotherapy response in PET images, which outperformed other shallow methods (3S-CNN: 73.4% ± 5.3%, 1S-CNN: 66.4% ± 5.9%, GB: 66.7% ± 5.2%, RF: 57.3% ± 7.8%, SVM: 55.9% ± 8.1%, GB-PCA: 66.7% ± 6%, RF-PCA: 65.7% ± 5.6%, and SVM-PCA: 60.5% ± 8%) [143].

## D. (Brain/Body)–Machine Interfaces

Test results in the form of performance comparison of DL/non-DL methods applied to EEG data to detect MoI, emotion and affective state, and anomaly are shown in Fig. 5(a).

A linear parallel CNN with MLP classified EEG energy dynamics more accurately (70.6%) than SVM (67%), MLP (65.8%), and CNN (69.6%) to detect MoI from the BCI Competition (BCIC) IV-2a data set [174]. A CNN better classified frequency complementary feature maps of augmented CSP and SFM as features (68.5% and 69.3%) than filter-bank CSP (67%) to detect MoI from the BCIC IV-2a data set [172]. CNN, StAE, and their combination (CNN-StAE) were tested in classifying MoI from BCIC IV-2b EEG data. Using time, frequency, and location information as features, CNN-StAE achieved best accuracy (77.6% ± 2.1%) in comparison with SVM (72.4% ± 5.7%), CNN (74.8% ± 2.3%), and StAE (57.7% ± 5.5%) [173]. A DBN with Ada-boost-based classifier had higher accuracy (~81%) than SVM (~76%) in classifying hand movements from EEG [162]. Another DBN-based method reported better accuracy (0.84) using frequency representations of EEG (using FFT and wavelet package decomposition) rather than FCSP (0.8), and CSP (0.76) in classifying MoI [161]. A DNN-based method, with layerwise relevance propagation heatmaps,

performed comparably (75%) with CSP-LDA (82%) in MoI classification [170].

A deep learning network (DLN) was built using StAE with PCA and covariate shift adaptation to classify valence and arousal states from DEAP EEG data with multichannel power spectral densities as features. The mean accuracy of the DLN was $52.7\% \pm 9.7\%$ compared with SVM ($40.1\% \pm 7.9\%$) [160]. A supervised DBN-based method classified affective states more accurately ($75.6\% \pm 4.5\%$) when compared with SVM ($64.7\% \pm 4.6\%$) by extracting deep features from thousands of low-level features using the DEAP EEG data [163]. A DBN-based method, with differential entropy as features, explored critical frequency bands and channels in EEG, and classified three emotions (positive, neutral, and negative) with higher accuracy (86.1%) than SVM (83.9%) [165]. As reported through Az-score, in predicting driver's drowsy and alert state from EEG data, CCNN and CNNR methods outperformed (79.6% and 82.8%, respectively) other DL (CNN: $71.4\% \pm 7.5\%$ and DNN: $76.5\% \pm 4.4\%$) and non-DL (LDA: $52.8\% \pm 4\%$ and SVM: $50.4\% \pm 2.5\%$) methods [176].

A DBN was used to model, classify, and detect anomalies from EEG waveforms. It has been reported that using raw data, a comparable classification and superior anomaly detection accuracy ($50\% \pm 3\%$) can be achieved compared with SVM ($48\% \pm 2\%$) and kNN ($40\% \pm 2\%$) classifiers [166]. Another DBN- and HMM-based method performed comparable sleep stage classification from raw EEG data ($67.4\% \pm 12.9\%$) with respect to DBN with HMM and features ($72.2\% \pm 9.7\%$) and Gaussian observation HMM with features ($63.9\% \pm 10.8\%$) [168].

Fig. 5(b) shows the performances of various methods in movement decoding (MD) from (s)EMG. A CNN-based method's hand movement classification accuracy using three sEMG data sets (from the Ninapro database) were comparable with other methods [CNN versus (kNN, SVM, RF)]—data set 1: $66.6\% \pm 6.4\%$ versus ($60\% \pm 8\%$, $62.1\% \pm 6.1\%$, and $75.3\% \pm 5.7\%$), data set 2: $60.3\% \pm 7.7\%$ versus ($68\% \pm 7\%$, $75\% \pm 5.8\%$, and $75.3\% \pm 7.8\%$), and data set 3: $38.1\% \pm 14.3\%$ versus ($38.8\% \pm 11.9\%$, $46.3\% \pm 7.9\%$, and $46.3\% \pm 7.9\%$) [184]. Another method, a user-adaptive one, using CNN with deep feature learning, decoded movements more accurately compared with SVM ($95\% \pm 2\%$ versus $80\% \pm 10\%$) [185].

Fig. 5(c) compares different techniques' performances in classifying ECG signals from MIT-BIH arrhythmia database and detecting anomalies in them. A nonlinear SVM with Gaussian kernel ($DBN^1$) outperformed (98.5%) NN (97.5%), LDA (96.2%), SVM with genetic algorithm ($SVM^1$: 96%), SVM with kNN ($SVM^2$: 98%), Wavelet with PSO (88.8%), and CA (94.3%) in classifying ECG features extracted using the DBN [186]. Comparable accuracy in classifying ECG beats was obtained using the DBN with softmax ($DBN^2$: 98.8%) compared to SVM with Wavelet and independent component analysis (ICA) ($SVM^3$: 99.7%), SVM with higher order statistics and Hermite coefficients ($SVM^4$: 98.1%), SVM with ICA ($SVM^5$: 98.8%), DT (96.1%), and Dynamic Bayesian network (DyBN: 98%) [187]. Using the DBN (with contrastive divergence and persistent contrastive divergence

learning), arrhythmias were classified more accurately (98.6%) in comparison with block $NN^1$ (98.1%), feedforward-based NN with PSO ($NN^2$: 97.0%), mixture of experts (97.6%), and LDA (95.5%) [188].

## V. Open Issues and Future Perspectives

Overall, it is believed that the brain solves problems through reinforcement learning and neuronal networks organized as hierarchical processing systems. Though since the 1950s the field of AI has been trying to adopt and implement this strategy in computers, notable progress has been seen only recently due to our better understanding about learning systems, increase of computational power, decline of computing costs, and last but not least, the seamless integration of different technological and technical breakthroughs. However, there are still situations where these methods fail, underperform against traditional methods, and, therefore, must be improved. In the following, we outline, what in our opinion are, the shortcomings of current techniques and existing open research challenges, and speculate about some future perspectives that will facilitate further development and advancement of the field.

The combined computational capability and flexibility provided by the two prominent ML methods (i.e., DL and RL) also has limitations [33]. Both of these methods require heavy computing power and memory, and therefore, are not worthy of being applied to moderate size data sets. Additionally, the theory of DL is not completely understood, making the high-level outcomes obscure and difficult to interpret. This turns into a situation when the models are considered as "Black box" [206]. In addition, like other ML techniques, DL is also susceptible to misclassification [72] and over-classification [207]. Furthermore, in representing action-value pairs in RL, it is not possible to use all nonlinear approximators, which may cause instability or even divergence in some cases [31]. Also, bootstrapping makes many of the RL algorithms NP-hard and inapplicable to real-time applications, as they are too slow to converge and in some cases too dangerous (e.g., autonomous driving). Moreover, very few of the existing techniques support harnessing the potential power of distributed and parallel computation through cloud computing. Arguably, in case of cloud, distributed, and parallel computing, data privacy and security concerns are still prevailing [208], and real-time processing capability of the gigantic amount of experimentally acquired data is still underdeveloped [209], [210].

To proceed toward mitigating shortcomings and addressing open issues, firstly, improving the existing theoretical foundations of DL on the basis of experimental data becomes crucial to be able to quantify the performances of individual NN models [211]. These improvements should be able to address issues, such as specific assessment of an individual model's computational complexity and learning efficiency in relation to well-defined parameter tuning strategies, and the ability to generalize and topologically self-organize based on data-driven properties. Also, novel data visualization techniques should be incorporated so that the interpretation of data becomes intuitive and less cumbersome. In terms of learning strategies, updated hybrid on- and off-policy with new advances in

optimization techniques is required. The problems pertaining to observability of RL are yet to be completely solved, and optimal action selection is still a huge challenge.

As seen in Table I, there are timely opportunities to employ deep RL in biological data mining, for example, deriving dynamic information from biological data coming from multiple levels to reduce data redundancy and discover novel biomarkers for disease detection and prevention. Also, new unsupervised learning for deep RL methods is required to shrink the necessity of large sets of labeled data at the training phase. Multitasking and multiagent learning paradigm should advance in order to cope with dynamically changing problems.

In addition, to keep up with the rapid pace of data growth in biological application domains, computational infrastructures in terms of distributed and parallel computing tailored to those applications are needed.

## VI. Conclusion

The recent bliss of technological advancement in life sciences came with the huge challenge of mining the multimodal, multidimensional, and complex biological data. Triggered by that call, interdisciplinary approaches have resulted in the development of cutting edge machine learning-based analytical tools. The success stories of ANN, deep architectures, and reinforcement learning in making machines more intelligent are well known. Furthermore, computational costs have dropped, computing power has surged, and *quasi*-unlimited solid-state storage is available at a reasonable price. These factors have allowed to combine these learning techniques to reshape machines' capabilities to understand and decipher complex patterns from biological data. To facilitate wider deployment of such techniques and to serve as a reference point for the community, this paper provides a comprehensive survey of the literature of techniques' usability with different biological data. Further, a comparative study is presented on performances of various DL techniques, when applied to data from different biological application domains, as reported in the literature. Finally, some open issues and future perspectives are highlighted.

## Acknowledgment

## References

[1] W. Coleman, *Biology in the Nineteenth Century: Problems of Form, Function and Transformation*. New York, NY, USA: Cambridge Univ. Press, 1977.

[2] L. Magner, *A History of the Life Sciences*. New York, NY, USA: Marcel Dekker, 2002.

[3] S. Brenner, "The revolution in the life sciences," *Science*, vol. 338, no. 6113, pp. 1427–1428, 2012.

[4] J. Shendure and H. Ji, "Next-generation DNA sequencing," *Nature Biotechnol.*, vol. 26, no. 10, pp. 1135–1145, 2008.

[5] M. L. Metzker, "Sequencing technologies—The next generation," *Nature Rev. Genet.*, vol. 11, no. 1, pp. 31–46, 2010.

[6] R. Vadivambal and D. S. Jayas, *Bio-Imaging: Principles, Techniques, and Applications*. Boca Raton, FL, USA: CRC Press, 2016.

[7] R. A. Poldrack and M. J. Farah, "Progress and challenges in probing the human brain," *Nature*, vol. 526, no. 7573, pp. 371–379, 2015.

[8] M. A. Lebedev and M. A. L. Nicolelis, "Brain-machine interfaces: From basic science to neuroprostheses and neurorehabilitation," *Phys. Rev.*, vol. 97, no. 2, pp. 767–837, 2017.

[9] V. Marx, "Biology: The big challenges of big data," *Nature*, vol. 498, no. 7453, pp. 255–260, 2013.

[10] Y. Li and L. Chen, "Big biological data: Challenges and opportunities," *Genomics Proteomics Bioinform.*, vol. 12, no. 5, pp. 187–189, 2014.

[11] P. Wickware, "Next-generation biologists must straddle computation and biology," *Nature*, vol. 404, no. 6778, pp. 683–684, 2000.

[12] A. L. Tarca, V. J. Carey, X.-W. Chen, R. Romero, and S. Draghici, "Machine learning and its applications to biology," *PLoS Comput. Biol.*, vol. 3, no. 6, p. e116, 2007.

[13] J. J. Hopfield, "Artificial neural networks," *IEEE Circuits Devices Mag.*, vol. 4, no. 5, pp. 3–10, Sep. 1988.

[14] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[15] G.-X. Yuan, C.-J. Ho, and C.-J. Lin, "Recent advances of large-scale linear classification," *Proc. IEEE*, vol. 100, no. 9, pp. 2584–2603, Sep. 2012.

[16] D. Heckerman, "A tutorial on learning with Bayesian networks," in *Learning in Graphical Models*, M. I. Jordan, Ed. Amsterdam, The Netherlands: Springer, 1998, pp. 301–354.

[17] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 1, pp. 21–27, Jan. 1967.

[18] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Mag.*, vol. 3, no. 1, pp. 4–16, Jan. 1986.

[19] R. Kohavi and J. Quinlan, "Data mining tasks and methods: Classification: Decision-tree discovery," in *Handbook of Data Mining and Knowledge Discovery*, W. Klosgen and J. Zytkow, Eds. New York, NY, USA: Oxford Univ. Press, 2002, pp. 267–276.

[20] G. E. Hinton, "Connectionist learning procedures," *Artif. Intell.*, vol. 40, nos. 1–3, pp. 185–234, 1989.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. B, Methodol.*, vol. 39, no. 1, pp. 1–38, 1977.

[22] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.

[23] G. H. Ball and D. J. Hall, "ISODATA, a novel method of data analysis and pattern classification," Stanford Res. Inst., Stanford, CA, USA, Tech. Rep. NTIS AD 699616, 1965.

[24] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybern.*, vol. 3, no. 3, pp. 32–57, 1973.

[25] J. A. Hartigan, *Clustering Algorithms*. New York, NY, USA: Wiley, 1975.

[26] M. W. Libbrecht and W. S. Noble, "Machine learning applications in genetics and genomics," *Nature Rev. Genet.*, vol. 16, no. 6, pp. 321–332, 2015.

[27] A. Kan, "Machine learning applications in cell image analysis," *Immunol. Cell Biol.*, vol. 95, no. 6, pp. 525–530, 2017.

[28] B. Erickson, P. Korfiatis, Z. Akkus, and T. Kline, "Machine learning for medical imaging," *RadioGraphics*, vol. 37, no. 2, pp. 505–515, 2017.

[29] C. Vidaurre, C. Sannelli, K. R. Müller, and B. Blankertz, "Machine-learning-based coadaptive calibration for brain-computer interfaces," *Neural Comput.*, vol. 23, no. 3, pp. 791–816, 2010.

[30] M. Mahmud and S. Vassanelli, "Processing and analysis of multichannel extracellular neuronal signals: State-of-the-art and challenges," *Front. Neurosci.*, vol. 10, p. 248, Jun. 2016, doi: 10.3389/fnins.2016.00248.

[31] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.

[32] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[33] D. Ravi *et al.*, "Deep learning for health informatics," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 4–21, Jan. 2017.

[34] P. Mamoshina, A. Vieira, E. Putin, and A. Zhavoronkov, "Applications of deep learning in biomedicine," *Mol. Pharmaceutics*, vol. 13, no. 5, pp. 1445–1454, 2016.

[35] S. Min, B. Lee, and S. Yoon, "Deep learning in bioinformatics," *Brief Bioinform.*, vol. 18, no. 5, pp. 851–869, 2016.

[36] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 237–285, Jan. 1996.

[37] P. Y. Glorennec, "Reinforcement learning: An overview," in *Proc. ESIT*, 2000, pp. 17–35.

[38] A. Gosavi, "Reinforcement learning: A tutorial survey and recent advances," *INFORMS J. Comput.*, vol. 21, no. 2, pp. 178–192, 2009.

[39] Y. Li, "Deep reinforcement learning: An overview," *CoRR*, vol. abs/1701.07274, Sep. 2017.

[40] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.

[41] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[42] A. M. Saxe, J. L. McClelland, and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," *CoRR*, vol. abs/1312.6120, Feb. 2014.

[43] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," in *Proc. ICLR*, 2014, pp. 1–13.

[44] J. L. Elman, "Finding structure in time," *Cognit. Sci.*, vol. 14, no. 2, pp. 179–211, Mar. 1990.

[45] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[46] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[47] Z. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *CoRR*, vol. abs/1506.00019, Oct. 2015.

[48] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[49] T. Wiatowski and H. Bölcskei, "A mathematical theory of deep convolutional neural networks for feature extraction," *CoRR*, vol. abs/1512.06293, Dec. 2015.

[50] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*, M. Arbib, Ed. Cambridge, MA, USA: MIT Press, 1998, pp. 255–258.

[51] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, Apr. 2015.

[53] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, 2015, pp. 1–9.

[54] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

[55] P. Baldi, "Autoencoders, unsupervised learning and deep architectures," in *Proc. ICUTLW*, 2012, pp. 37–50.

[56] M. Ranzato, C. Poultney, S. Chopra, and Y. LeCun, "Efficient learning of sparse representations with an energy-based model," in *Proc. NIPS*, 2006, pp. 1137–1144.

[57] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *CoRR*, vol. abs/1312.6114, May 2014.

[58] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. ICML*, 2011, pp. 833–840.

[59] R. Salakhutdinov and G. E. Hinton, "Deep Boltzmann machines," in *Proc. AISTATS*, 2009, pp. 448–455.

[60] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[61] A. Fischer and C. Igel, "An introduction to restricted Boltzmann machines," in *Proc. CIARP*, 2012, pp. 14–36.

[62] G. Desjardins, A. C. Courville, and Y. Bengio, "On training deep Boltzmann machines," *CoRR*, vol. abs/1203.4416, Mar. 2012.

[63] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *Proc. ICML*, 2008, pp. 1064–1071.

[64] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.

[65] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[66] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[67] D. L. Poole and A. K. Mackworth, *Artificial Intelligence: Foundations of Computational Agents*. New York, NY, USA: Cambridge Univ. Press, 2017.

[68] L. Busoniu, R. Babuska, B. De Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*. Boca Raton, FL, USA: CRC Press, 2010.

[69] R. S. Sutton *et al.*, "Fast gradient-descent methods for temporal-difference learning with linear function approximation," in *Proc. ICML*, 2009, pp. 993–1000.

[70] T. Schaul, D. Horgan, K. Gregor, and D. Silver, "Universal value function approximators," in *Proc. ICML*, 2015, pp. 1312–1320.

[71] F. Woergoetter and B. Porr, "Reinforcement learning," *Scholarpedia*, vol. 3, no. 3, p. 1448, 2008.

[72] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. CVPR*, 2015, pp. 427–436.

[73] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," *CoRR*, vol. abs/1509.06461, 2015.

[74] H. V. Hasselt, "Double Q-learning," in *Proc. NIPS*, 2010, pp. 2613–2621.

[75] B. J. Erickson, P. Korfiatis, Z. Akkus, T. Kline, and K. Philbrick, "Toolkits and libraries for deep learning," *J. Digit. Imag.*, vol. 30, no. 4, pp. 400–405, 2017.

[76] R. Fakoor, F. Ladhak, A. Nazi, and M. Huber, "Using deep learning to enhance cancer diagnosis and classification," in *Proc. ICML*, 2013.

[77] P. Danaee, R. Ghaeini, and D. A. Hendrix, "A deep learning approach for cancer detection and relevant gene identification," in *Proc. Pacific Symp. Biocomput.*, vol. 22. 2016, pp. 219–229.

[78] H. Li, Q. Lyu, and J. Cheng, "A template-based protein structure reconstruction method using deep autoencoder learning," *J. Proteomics Bioinform.*, vol. 9, no. 12, pp. 306–313, 2016.

[79] T. Lee and S. Yoon, "Boosted categorical restricted Boltzmann machine for computational prediction of splice junctions," in *Proc. ICML*, 2015, pp. 2483–2492.

[80] R. Ibrahim, N. A. Yousri, M. A. Ismail, and N. M. El-Makky, "Multi-level gene/MiRNA feature selection using deep belief nets and active learning," in *Proc. IEEE EMBC*, Aug. 2014, pp. 3957–3960.

[81] L. Chen, C. Cai, V. Chen, and X. Lu, "Trans-species learning of cellular signaling systems with bimodal deep belief networks," *Bioinformatics*, vol. 31, no. 18, pp. 3008–3015, 2015.

[82] X. Pan and H.-B. Shen, "RNA-protein binding motifs mining with a new hybrid deep learning based cross-domain knowledge integration approach," *BMC Bioinform.*, vol. 18, no. 1, p. 136, 2017.

[83] Y. Chen, Y. Li, R. Narayan, A. Subramanian, and X. Xie, "Gene expression inference with deep learning," *Bioinformatics*, vol. 32, no. 12, pp. 1832–1839, 2016.

[84] S. Zhang *et al.*, "A deep learning framework for modeling structural features of rna-binding protein targets," *Nucl. Acids Res.*, vol. 44, no. 4, p. e32, 2016.

[85] C. Angermueller, H. J. Lee, W. Reik, and O. Stegle, "DeepCpG: Accurate prediction of single-cell DNA methylation states using deep learning," *Genome Biol.*, vol. 18, no. 1, p. 67, 2017.

[86] D. Quang, Y. Chen, and X. Xie, "DANN: A deep learning approach for annotating the pathogenicity of genetic variants," *Bioinformatics*, vol. 31, no. 5, pp. 761–763, 2015.

[87] R. Heffernan *et al.*, "Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning," *Sci. Rep.*, vol. 5, p. 11476, Jun. 2015.

[88] K. Tian, M. Shao, S. Zhou, and J. Guan, "Boosting compound-protein interaction prediction by deep learning," *Methods*, vol. 110, pp. 64–72, Nov. 2016.

[89] F. Wan and J. Zeng, "Deep learning with feature embedding for compound-protein interaction prediction," *bioRxiv*, p. 086033, Jan. 2016.

[90] Y. Yuan *et al.*, "DeepGene: An advanced cancer type classifier based on deep learning and somatic point mutations," *BMC Bioinform.*, vol. 17, no. 17, p. 476, 2016.

[91] M. Hamanaka *et al.*, "CGBVS-DNN: Prediction of compound-protein interactions based on deep learning," *Mol. Inform.*, vol. 36, nos. 1–2, p. 1600045, 2017, doi: 10.1002/minf.201600045.

[92] O. Denas and J. Taylor, "Deep modeling of gene expression regulation in erythropoiesis model," in *Proc. ICMLRL*, 2013, pp. 1–5.

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

[94] D. R. Kelley, J. Snoek, and J. L. Rinn, "Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks," *Genome Res.*, vol. 26, no. 7, pp. 990–999, 2016.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MAHMUD *et al.*: APPLICATIONS OF DL AND RL TO BIOLOGICAL DATA

15

[95] H. Zeng, M. D. Edwards, G. Liu, and D. K. Gifford, "Convolutional neural network architectures for predicting DNA–protein binding," *Bioinformatics*, vol. 32, no. 12, pp. 121–127, 2016.

[96] J. Zhou and O. G. Troyanskaya, "Predicting effects of noncoding variants with deep learning-based sequence model," *Nature Methods*, vol. 12, no. 10, pp. 931–934, 2015.

[97] Y. F. Huang, B. Gulko, and A. Siepel, "Fast, scalable prediction of deleterious noncoding variants from functional and population genomic data," *Nature Genet.*, vol. 49, no. 4, pp. 618–624, 2017.

[98] S. Wang, J. Peng, J. Ma, and J. Xu, "Protein secondary structure prediction using deep convolutional neural fields," *Sci. Rep.*, vol. 6, p. 18962, Jan. 2016.

[99] S. Park, S. Min, H. Choi, and S. Yoon, "deepMiRGene: Deep neural network based precursor microrna prediction," *CoRR*, vol. abs/1605.00017, Apr. 2016.

[100] B. Lee, J. Baek, S. Park, and S. Yoon, "deepTarget: End-to-end learning framework for microRNA target prediction using deep recurrent neural networks," *CoRR*, vol. abs/1603.09123, Aug. 2016.

[101] L.-Y. Chuang, J.-H. Tsai, and C.-H. Yang, "Operon prediction using particle swarm optimization and reinforcement learning," in *Proc. ICTAAI*, 2010, pp. 366–372.

[102] C. G. Ralha, H. W. Schneider, M. E. M. T. Walter, and A. L. C. Bazzan, "Reinforcement learning method for BioAgents," in *Proc. SBRN*, 2010, pp. 109–114.

[103] M.-I. Bocicor, G. Czibula, and I.-G. Czibula, "A reinforcement learning approach for solving the fragment assembly problem," in *Proc. SYNASC*, 2011, pp. 191–198.

[104] F. Zhu, Q. Liu, X. Zhang, and B. Shen, "Protein-protein interaction network constructing based on text mining and reinforcement learning with application to prostate cancer," in *Proc. BIBM*, 2014, pp. 46–51.

[105] J. Xu *et al.*, "Stacked sparse autoencoder (SSAE) for nuclei detection on breast cancer histopathology images," *IEEE Trans. Med. Imag.*, vol. 35, no. 1, pp. 119–130, Jan. 2016.

[106] Y. Xu, T. Mo, Q. Feng, P. Zhong, M. Lai, and E. I.-C. Chang, "Deep learning of feature representation with multiple instance learning for medical image analysis," in *Proc. ICASSP*, 2014, pp. 1626–1630.

[107] C. L. Chen *et al.*, "Deep learning in label-free cell classification," *Sci. Rep.*, vol. 6, no. 1, 2016, Art. no. 21471.

[108] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano, "Toward automatic phenotyping of developing embryos from videos," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1360–1371, Sep. 2005.

[109] D. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *Proc. MICCAI*, 2013, pp. 411–418.

[110] D. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," in *Proc. NIPS*, 2012, pp. 2843–2851.

[111] T. Pärnamaa and L. Parts, "Accurate classification of protein subcellular localization from high-throughput microscopy images using deep learning," *G3*, vol. 7, no. 5, pp. 1385–1392, 2017.

[112] A. Ferrari, S. Lombardi, and A. Signoroni, "Bacterial colony counting with convolutional neural networks in digital microbiology imaging," *Pattern Recognit.*, vol. 61, pp. 629–640, Jan. 2017.

[113] O. Z. Kraus, J. L. Ba, and B. J. Frey, "Classifying and segmenting microscopy images with deep multiple instance learning," *Bioinformatics*, vol. 32, no. 12, pp. i52–i59, 2016.

[114] P. Eulenberg *et al.*, "Deep learning for imaging flow cytometry: Cell cycle analysis of Jurkat cells," *bioRxiv*, p. 081364, Oct. 2016.

[115] B. Jiang, X. Wang, J. Luo, X. Zhang, Y. Xiong, and H. Pang, "Convolutional neural networks in automatic recognition of trans-differentiated neural progenitor cells under bright-field microscopy," in *Proc. IMCCC*, 2015, pp. 122–126.

[116] A. Mansoor *et al.*, "Deep learning guided partitioned shape model for anterior visual pathway segmentation," *IEEE Trans. Med. Imag.*, vol. 35, no. 8, pp. 1856–1865, Aug. 2016.

[117] H.-I. Suk and D. Shen, "Deep learning-based feature representation for ad/mci classification," in *Proc. MICCAI*, 2013, pp. 583–590.

[118] B. Shi, Y. Chen, P. Zhang, C. D. Smith, and J. Liu, "Nonlinear feature transformation and deep fusion for Alzheimer's disease staging analysis," *Pattern Recognit.*, vol. 63, pp. 487–498, Mar. 2017.

[119] T. Brosch and R. Tam, "Manifold learning of brain MRIs by deep learning," in *Proc. MICCAI*, 2013, pp. 633–640.

[120] H.-I. Suk, S.-W. Lee, and D. Shen, "Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis," *NeuroImage*, vol. 101, pp. 569–582, Nov. 2014.

[121] F. Li, L. Tran, K. H. Thung, S. Ji, D. Shen, and J. Li, "A robust deep model for improved classification of AD/MCI patients," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 5, pp. 1610–1616, Sep. 2015.

[122] S. M. Plis *et al.*, "Deep learning for neuroimaging: A validation study," *Frontiers Neurosci.*, vol. 8, p. 229, Aug. 2014.

[123] M. Havaei *et al.*, "Brain tumor segmentation with deep neural networks," *Med. Image Anal.*, vol. 35, pp. 18–31, Jan. 2017.

[124] J. Shi, X. Zheng, Y. Li, Q. Zhang, and S. Ying, "Multimodal neuroimaging feature learning with multimodal stacked deep polynomial networks for diagnosis of Alzheimer's disease," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 1, pp. 173–183, Jan. 2018.

[125] M. Havaei, N. Guizard, H. Larochelle, and P.-M. Jodoin, "Deep learning trends for focal brain pathology segmentation in MRI," in *Machine Learning for Health Informatics*, A. Holzinger, Ed. Cham, Switzerland: Springer, 2016, pp. 125–148.

[126] E. Hosseini-Asl, G. L. Gimel'farb, and A. El-Baz, "Alzheimer's disease diagnostics by a deeply supervised adaptable 3D convolutional network," *CoRR*, vol. abs/1607.00556, Jul. 2016.

[127] J. Kleesiek *et al.*, "Deep MRI brain extraction: A 3D convolutional neural network for skull stripping," *NeuroImage*, vol. 129, pp. 460–469, Apr. 2016.

[128] S. Sarraf and G. Tofighi, "DeepAD: Alzheimer's disease classification via deep convolutional neural networks using MRI and fMRI," *bioRxiv*, p. 070441, Aug. 2016.

[129] D. Nie, H. Zhang, E. Adeli, L. Liu, and D. Shen, "3D deep learning for multi-modal imaging-guided survival time prediction of brain tumor patients," in *Proc. MICCAI*, 2016, pp. 212–220.

[130] K. Kamnitsas *et al.*, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2017.

[131] M. F. Stollenga, W. Byeon, M. Liwicki, and J. Schmidhuber, "Parallel multi-dimensional LSTM, with application to fast biomedical volumetric image segmentation," in *Proc. NIPS*, 2015, pp. 2998–3006.

[132] F. Agostinelli, M. R. Anderson, and H. Lee, "Adaptive multi-column deep neural networks with application to robust image denoising," in *Proc. NIPS*, 2013, pp. 1493–1501.

[133] J. Lerouge, R. Herault, C. Chatelain, F. Jardin, and R. Modzelewski, "IODA: An input/output deep architecture for image labeling," *Pattern Recognit.*, vol. 48, no. 9, pp. 2847–2858, 2015.

[134] K. Fritscher, P. Raudaschl, P. Zaffino, M. F. Spadea, G. C. Sharp, and R. Schubert, "Deep neural networks for fast segmentation of 3D medical images," in *Proc. MICCAI*, 2016, pp. 158–165.

[135] X. W. Gao, R. Hui, and Z. Tian, "Classification of CT brain images based on deep learning networks," *Comput. Methods Programs Biomed.*, vol. 138, pp. 49–56, Jan. 2017.

[136] J. Cho, K. Lee, E. Shin, G. Choy, and S. Do, "Medical image deep learning with hospital PACS dataset," *CoRR*, vol. abs/1511.06348, Nov. 2015.

[137] H. R. Roth *et al.*, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1170–1181, May 2016.

[138] J. Z. Cheng *et al.*, "Computer-aided diagnosis with deep learning architecture: Applications to breast lesions in US images and pulmonary nodules in CT scans," *Sci. Rep.*, vol. 6, p. 24454, Apr. 2016.

[139] H.-C. Shin *et al.*, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.

[140] W. Shen *et al.*, "Multi-crop convolutional neural networks for lung nodule malignancy suspiciousness classification," *Pattern Recognit.*, vol. 61, pp. 663–673, Jan. 2017.

[141] N. Tajbakhsh and K. Suzuki, "Comparing two classes of end-to-end machine-learning models in lung nodule detection and classification: MTANNs vs. CNNs," *Pattern Recognit.*, vol. 63, pp. 476–486, Mar. 2017.

[142] D. Kuang and L. He, "Classification on ADHD with deep learning," in *Proc. CCBD*, 2014, pp. 27–32.

[143] P.-P. Ypsilantis *et al.*, "Predicting response to neoadjuvant chemotherapy with PET imaging using convolutional neural networks," *PLoS ONE*, vol. 10, no. 9, p. e0137036, 2015.

[144] L. Gondara, "Medical image denoising using convolutional denoising autoencoders," in *Proc. ICDMW*, 2016, pp. 241–246.

[145] T. A. Ngo, Z. Lu, and G. Carneiro, "Combining deep learning and level set for the automated segmentation of the left ventricle of the heart from cardiac cine magnetic resonance," *Med. Image Anal.*, vol. 35, pp. 159–171, Jan. 2017.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

16

IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS

[146] H. Chen *et al.*, "Standard plane localization in fetal ultrasound via domain transferred deep neural networks," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 5, pp. 1627–1636, Sep. 2015.

[147] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1207–1216, 2016.

[148] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Fast CNN training using selective data sampling: Application to hemorrhage detection in color fundus images," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1207–1216, May 2016.

[149] J.-S. Yu, J. Chen, Z. Q. Xiang, and Y.-X. Zou, "A hybrid convolutional neural networks with extreme learning machine for WCE image classification," in *Proc. ROBIO*, 2015, pp. 1822–1827.

[150] S. Lee, M. Choi, H.-S. Choi, M. S. Park, and S. Yoon, "FingerNet: Deep learning-based robust finger joint detection from radiographs," in *Proc. IEEE BioCAS*, Oct. 2015, pp. 1–4.

[151] J. Arevalo *et al.*, "Representation learning for mammography mass lesion classification with convolutional neural networks," *Comput. Methods Programs Biomed.*, vol. 127, pp. 248–257, Apr. 2016.

[152] Z. Jiao, X. Gao, Y. Wang, and J. Li, "A deep feature based framework for breast masses classification," *Neurocomputing*, vol. 197, pp. 221–231, Jul. 2016.

[153] W. Sun, T. B. Tseng, J. Zhang, and W. Qian, "Enhancing deep convolutional neural network scheme for breast cancer diagnosis with unlabeled data," *Comput. Med. Imag. Graph.*, vol. 57, pp. 4–9, Apr. 2017.

[154] T. Kooi *et al.*, "Large scale deep learning for computer aided detection of mammographic lesions," *Med. Image Anal.*, vol. 35, pp. 303–312, Jan. 2017.

[155] N. Dhungel, G. Carneiro, and A. P. Bradley, "A deep learning approach for the analysis of masses in mammograms with minimal user intervention," *Med. Image Anal.*, vol. 37, pp. 114–128, Apr. 2017.

[156] K. Sirinukunwattana *et al.*, "Gland segmentation in colon histology images: The glas challenge contest," *Med. Image Anal.*, vol. 35, pp. 489–502, Jan. 2017.

[157] Q. Dou *et al.*, "3D deeply supervised network for automated segmentation of volumetric medical images," *Med. Image Anal.*, vol. 41, pp. 40–54, Oct. 2017.

[158] F. Sahba, H. R. Tizhoosh, and M. M. A. Salama, "Application of reinforcement learning for segmentation of transrectal ultrasound images," *BMC Med. Imag.*, vol. 8, no. 1, p. 8, 2008.

[159] J. Li and A. Cichocki, "Deep learning of multifractal attributes from motor imagery induced EEG," in *Proc. ICONIP*, 2014, pp. 503–510.

[160] S. Jirayucharoensak, S. Pan-Ngum, and P. Israsena, "EEG-based emotion recognition using deep learning network with principal component based covariate shift adaptation," *Sci. World J.*, vol. 2014, pp. 1–10, Sep. 2014.

[161] N. Lu, T. Li, X. Ren, and H. Miao, "A deep learning scheme for motor imagery classification based on restricted Boltzmann machines," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 566–576, Jun. 2017.

[162] X. An, D. Kuang, X. Guo, Y. Zhao, and L. He, "A deep learning method for classification of EEG data based on motor imagery," in *Proc. ICIC*, 2014, pp. 203–210.

[163] K. Li, X. Li, Y. Zhang, and A. Zhang, "Affective state recognition from EEG with deep belief networks," in *Proc. BIBM*, 2013, pp. 305–310.

[164] X. Jia, K. Li, X. Li, and A. Zhang, "A novel semi-supervised deep learning framework for affective state recognition on eeg signals," in *Proc. IEEE BIBE*, Nov. 2014, pp. 30–37.

[165] W.-L. Zheng and B.-L. Lu, "Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks," *IEEE Trans. Auton. Mental Develop.*, vol. 7, no. 3, pp. 162–175, Sep. 2015.

[166] D. F. Wulsin, J. R. Gupta, R. Mani, J. A. Blanco, and B. Litt, "Modeling electroencephalography waveforms with semi-supervised deep belief nets: Fast classification and anomaly measurement," *J. Neural Eng.*, vol. 8, no. 3, p. 036015, 2011.

[167] Y. Zhao and L. He, "Deep learning in the EEG diagnosis of Alzheimer's disease," in *Proc. ACCV*, 2015, pp. 340–353.

[168] M. Längkvist, L. Karlsson, and A. Loutfi, "Sleep stage classification using unsupervised feature learning," *Adv. Artif. Neural Syst.*, vol. 2012, p. 107046, May 2012.

[169] R. Chai *et al.*, "Improving eeg-based driver fatigue classification using sparse-deep belief networks," *Front. Neurosci.*, vol. 11, p. 103, Mar. 2017.

[170] I. Sturm, S. Lapuschkin, W. Samek, and K.-R. Müller, "Interpretable deep neural networks for single-trial EEG classification," *J. Neurosci. Methods*, vol. 274, pp. 141–145, Dec. 2016.

[171] S. Kumar, A. Sharma, K. Mamun, and T. Tsunoda, "A deep learning approach for motor imagery eeg signal classification," in *Proc. APWC CSE*, 2016, pp. 34–39.

[172] H. Yang, S. Sakhavi, K. K. Ang, and C. Guan, "On the use of convolutional neural networks and augmented CSP features for multiclass motor imagery of EEG signals classification," in *Proc. EMBC*, 2015, pp. 2620–2623.

[173] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of EEG motor imagery signals," *J. Neural Eng.*, vol. 14, no. 1, p. 016003, 2017.

[174] S. Sakhavi, C. Guan, and S. Yan, "Parallel convolutional-linear neural network for motor imagery classification," in *Proc. EUSIPCO*, 2015, pp. 2786–2790.

[175] S. Tripathi, S. Acharya, R. D. Sharma, S. Mittal, and S. Bhattacharya, "Using deep and convolutional neural networks for accurate emotion classification on DEAP dataset," in *Proc. IAAI*, 2017, pp. 4746–4752.

[176] M. Hajinoroozi, Z. Mao, and Y. Huang, "Prediction of driver's drowsy and alert states from eeg signals with deep learning," in *Proc. IEEE CAMSAP*, Dec. 2015, pp. 493–496.

[177] P. Mirowski, D. Madhavan, Y. LeCun, and R. Kuzniecky, "Classification of patterns of EEG synchronization for seizure prediction," *Clin. Neurophysiol.*, vol. 120, no. 11, pp. 1927–1940, 2009.

[178] M. Soleymani, S. Asghari-Esfeden, M. Pantic, and Y. Fu, "Continuous emotion detection using EEG signals and facial expressions," in *Proc. ICME*, 2014, pp. 1–6.

[179] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from EEG with deep recurrent-convolutional neural networks," *CoRR*, vol. abs/1511.06448, Feb. 2016.

[180] A. Petrosian, D. Prokhorov, R. Homan, R. Dasheiff, and D. Wunsch, II, "Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG," *Neurocomputing*, vol. 30, nos. 1–4, pp. 201–218, 2000.

[181] P. R. Davidson, R. D. Jones, and M. T. R. Peiris, "EEG-based lapse detection with high temporal resolution," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 5, pp. 832–839, May 2007.

[182] T. Lampe, L. D. J. Fiederer, M. Voelker, A. Knorr, M. Riedmiller, and T. Ball, "A brain-computer interface for high-level remote control of an autonomous, reinforcement-learning-based robotic system for reaching and grasping," in *Proc. IUI*, 2014, pp. 83–88.

[183] R. Bauer and A. Gharabaghi, "Reinforcement learning for adaptive threshold control of restorative brain-computer interfaces: A Bayesian simulation," *Frontiers Neurosci.*, vol. 9, p. 36, Feb. 2015.

[184] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers Neurorobot.*, vol. 10, p. 9, 2016.

[185] K.-H. Park and S.-W. Lee, "Movement intention decoding based on deep learning for multiuser myoelectric interfaces," in *Proc. IWW-BCI*, 2016, p. 2.

[186] M. Huanhuan and Z. Yue, "Classification of electrocardiogram signals with deep belief networks," in *Proc. IEEE CSE*, Dec. 2014, pp. 7–12.

[187] Y. Yan, X. Qin, Y. Wu, N. Zhang, J. Fan, and L. Wang, "A restricted Boltzmann machine based two-lead electrocardiography classification," in *Proc. BSN*, 2015, pp. 1–9.

[188] Z. Wu, X. Ding, and G. Zhang, "A novel method for classification of ECG arrhythmias using deep belief networks," *J. Comput. Intell. Appl.*, vol. 15, no. 4, p. 1650021, 2016.

[189] M. M. Al Rahhal, Y. Bazi, H. AlHichri, N. Alajlan, F. Melgani, and R. R. Yager, "Deep learning approach for active classification of electrocardiogram signals," *Inf. Sci.*, vol. 345, pp. 340–354, Jun. 2016.

[190] J. DiGiovanna, B. Mahmoudi, J. Fortes, J. C. Principe, and J. C. Sanchez, "Coadaptive brain–machine interface via reinforcement learning," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 1, pp. 54–64, Jan. 2009.

[191] B. Mahmoudi and J. C. Sanchez, "A symbiotic brain-machine interface through value-based decision making," *PLoS ONE*, vol. 6, no. 3, p. e14760, 2011.

[192] J. C. Sanchez *et al.*, "Control of a center-out reaching task using a reinforcement learning brain-machine interface," in *Proc. IEEE NER*, Apr./May 2011, pp. 525–528.

[193] B. Mahmoudi, E. A. Pohlmeyer, N. W. Prins, S. Geng, and J. C. Sanchez, "Towards autonomous neuroprosthetic control using Hebbian reinforcement learning," *J. Neural Eng.*, vol. 10, no. 6, p. 066005, Oct. 2013.

[194] F. Wang, K. Xu, Q. Zhang, Y. Wang, and X. Zheng, "A multi-step neural control for motor brain-machine interface by reinforcement learning," *Appl. Mech. Mater.*, vol. 461, pp. 565–569, 2013.

[195] E. A. Pohlmeyer, B. Mahmoudi, S. Geng, N. W. Prins, and J. C. Sanchez, "Using reinforcement learning to provide stable brain-machine interface control despite neural input reorganization," *PLoS ONE*, vol. 9, no. 1, p. e87253, 2014.

[196] F. Wang *et al.*, "Quantized attention-gated kernel reinforcement learning for brain–machine interface decoding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 4, pp. 873–886, Apr. 2017.

[197] N. Zeng, Z. Wang, H. Zhang, W. Liu, and F. E. Alsaadi, "Deep belief networks for quantitative analysis of a gold immunochromatographic strip," *Cognit. Comput.*, vol. 8, no. 4, pp. 684–692, 2016.

[198] J. Li, Z. Zhang, and H. He, "Hierarchical convolutional neural networks for EEG-based emotion recognition," *Cognit. Comput.*, pp. 1–13, Dec. 2017. doi: 10.1007/s12559-017-9533-x.

[199] D. Ray *et al.*, "A compendium of RNA-binding motifs for decoding gene regulation," *Nature*, vol. 499, no. 7457, pp. 172–177, 2013.

[200] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3133–3181, 2014.

[201] D. Marbach *et al.*, "Wisdom of crowds for robust gene network inference," *Nature Methods*, vol. 9, no. 8, pp. 796–804, 2012.

[202] M. T. Weirauch *et al.*, "Evaluation of methods for modeling transcription factor sequence specificity," *Nature Biotechnol.*, vol. 31, no. 2, pp. 126–134, 2013.

[203] B. H. Menze *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[204] M. Avendi, A. Kheradvar, and H. Jafarkhani, "Fully automatic segmentation of heart chambers in cardiac MRI using deep learning," *J. Cardiovascular Magn. Reson.*, vol. 18, no. 1, p. P351, 2016.

[205] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. ICPR*, 2010, pp. 2366–2369.

[206] D. Erhan, A. Courville, and Y. Bengio, "Understanding representations learned in deep architectures," Univ. Montreal, Montreal, QC, Canada, Tech. Rep. 1355, 2010.

[207] C. Szegedy *et al.*, "Intriguing properties of neural networks," *CoRR*, vol. abs/1312.6199, Feb. 2014.

[208] M. Mahmud, M. M. Rahman, D. Travalin, P. Raif, and A. Hussain, "Service oriented architecture based Web application model for collaborative biomedical signal analysis," *Biomed. Eng.*, vol. 57, pp. 780–783, Sep. 2012.

[209] M. Mahmud, R. Pulizzi, E. Vasilaki, and M. Giugliano, "QSpike tools: A generic framework for parallel batch preprocessing of extracellular neuronal signals recorded by substrate microelectrode arrays," *Frontiers Neuroinform.*, vol. 8, p. 26, Mar. 2014.

[210] M. Mahmud, R. Pulizzi, E. Vasilaki, and M. Giugliano, "A Web-based framework for semi-online parallel processing of extracellular neuronal signals recorded by microelectrode arrays," in *Proc. MEA Meeting*, 2014, pp. 202–203.

[211] P. Angelov and A. Sperduti, "Challenges in deep learning," in *Proc. ESANN*, 2016, pp. 489–495.

**Mufti Mahmud** (GS'08–M'11–SM'16) born in Thakurgaon, Bangladesh. He received the B.Sc. degree in computer science from the University of Madras, Chennai, India, in 2001, the M.Sc. degree in computer science from the University of Mysore, Mysore, India, in 2003, the M.S. degree in bionanotechnology from the University of Trento, Trento, Italy, in 2008, and the Ph.D. degree in information engineering from the University of Padova, Padua, Italy, in 2011.

He served at various positions in industry and academia in India, Bangladesh, Italy, and Belgium. With over 60 publications in leading journals and conferences, he is a neuroinformatics expert. He has released two open source toolboxes (SigMate and QSpike Tools) for processing and analysis of brain signals. His current research interests include healthcare and neuroscience big data analytics, advanced machine learning applied to biological data, assistive brain–machine interfacing, computational neuroscience, personalized and preventive (e/m)-healthcare, Internet of Healthcare Things, cloud computing, security and trust management in cyber-physical systems, and crowd analysis.

Dr. Mahmud was a recipient of the Marie-Curie Fellowship. He serves as an Associate Editor of *Cognitive Computation* journal (Springer-Nature), successfully organized numerous special sessions in leading conferences, served many internationally reputed conferences in different capacities, such as programme, organization, and advisory committee member, and also served as a referee for many high-impact journals.

**Mohammed Shamim Kaiser** (SM'16) received the B.Sc. (Hons.) and M.S. degrees in applied physics electronics and communication engineering from the University of Dhaka, Dhaka, Bangladesh, in 2002 and 2004, respectively, and the Ph.D. degree in telecommunications from the Asian Institute of Technology, Khlong Nung, Thailand, in 2010.

He is currently an Associate Professor with the Institute of Information Technology, Jahangirnagar University, Dhaka. His current research interests include multihop wireless networks, big data analytics, and cyber security.

Dr. Kaiser is a Life Member of the Bangladesh Electronic Society, the Bangladesh Physical Society, and the Bangladesh Computer Society, a member of IEICE, Japan, and an ExCom Member of IEEE, Bangladesh Section.

**Amir Hussain** (SM'97) received the B.Eng. and Ph.D. degrees from the University of Strathclyde, Glasgow, U.K.

He joined the University of Stirling, Stirling, U.K., in 2000, where he is currently a Professor of computing science and the Founding Director of the Cognitive Big Data Informatics Laboratory. He has authored over 300 publications, including over a dozen books and over 100 journal papers, conducted and led collaborative research with industry, partnered in major European and international research programs, and supervised over 30 Ph.D. students. His current research interests include multidisciplinary with a focus on brain-inspired, multimodal cognitive big data technology for solving complex real-world problems.

Dr. Hussain is a fellow of the U.K.'s Higher Education Academy and a Senior Fellow of the Brain Sciences Foundation. He received the Research Fellowship at the University of Paisley and the Research Lectureship at the University of Dundee. He is a Founding Chief Editor of the *Cognitive Computation* journal (Springer) and an Associate Editor for a number of other leading journals. He has served as an Invited Speaker and the Organizing Committee Co-Chair for over 50 top international conferences and workshops. He has served as the founding Publications Co-Chair of the INNS Big Data Section and the Annual INNS Conference on Big Data, and the Chapter Chair of the IEEE, U.K., and RI's Industry Applications Society.

**Stefano Vassanelli** received the M.D. and the Ph.D. degrees in molecular and cellular biology and pathology from the University of Padova, Padua, Italy, in 1992 and 1999, respectively.

From 1993 to 2000, he was with the Oregon Graduate Institute of Science and Technology, Portland, OR, USA, for one year, where he was investigating the mitochondrial proton transporter uncoupling protein, and the Max-Planck Institute for Biochemistry, Martinsried, Germany, for five years, where he was involved in the first neuron-transistor capacitive coupling. In 2001, he founded the NeurChip Laboratory, University of Padova. Since 2001, he has been involved in the development of neurotechnologies for recording, stimulation, and processing of signals generated by neuronal networks. He is currently a Professor of physiology and teaching both for the engineering and medical faculties.