Deep-Reinforcement-Learning-Based Images Segmentation for Quantitative Analysis of Gold Immunochromatographic Strip

Nianyin Zeng, Han Li, Zidong Wang*, Weibo Liu, Songming Liu, Fuad E. Alsaadi and Xiaohui Liu

***Abstract*—Gold immunochromatographic strip (GICS) is a widely used lateral flow immunoassay technique. A novel image segmentation method is developed in this paper for quantitative analysis of GICS based on the deep reinforcement learning (DRL), which can accurately distinguish the test line and the control line in the GICS images. The deep belief network (DBN) is employed in the deep Q network in our DRL algorithm. Meanwhile, the multi-factor learning curve is introduced in the DRL algorithm to dynamically adjust the capacity of the replay buffer and the sampling size, which leads to enhanced learning efficiency. It is worth mentioning that the states, actions, and rewards in the developed DRL algorithm are determined based on the characteristics of GICS images. Experiment results demonstrate the feasibility and reliability of the proposed DRL-based image segmentation method and show that the proposed new image segmentation method outperforms some existing image segmentation methods for quantitative analysis of GICS images.**

***Index Terms*—Deep reinforcement learning, image segmentation, deep belief network, image segmentation, multi-factor learning curve, gold immunochromatographic strip.**

I. Introduction

Served as an important lateral flow immunoassay technique, the gold immunochromatographic strip (GICS) has been successfully applied to biomedical and related areas for determining the target analyte in the specimens, especially under the non-laboratory environment due to its short analysis time and high stability [12], [14], [37], [43]. With the purpose of improving the performance of the GICS, researchers have devoted tremendous efforts to designing various biochemical reaction systems (so as to investigate the quantitative properties of the strips) and developing GICS quantitative instruments [9], [28], [39], [42], [43].

Notice that the image-based quantitative analysis method (which aims to recognize the test line and the control line in GICS images) has become an attractive research topic for developing GICS quantitative instruments [40], [42], [43]. In fact, it is of critical importance to distinguish the test line and control line in the GICS images as two lines significantly affect the subsequent quantification. Up to now, a large number of image processing methods have been utilized to segment the two lines in the GICS images, such as the cellular neural network, the fuzzy c-means algorithm, and the deep belief network (DBN) [7], [15], [40], [41], [43]. It is worth mentioning that there are two main challenging problems in dealing with the acquired GICS images: 1) the quality of the GICS image is poor due to the existence of unavoidable noise in the GICS image caused by environment factors like temperature and humidity; and 2) the boundary between the two lines and the background is irregular and blurry, especially the images with low concentration where the test line is too shallow to be recognized. To address the above mentioned challenges, our goal is to develop an effective image segmentation method to recognize the two lines for quantitative analysis of GICS.

Recently, reinforcement learning (RL) has become an attractive research topic in artificial intelligence and has achieved a great success in various areas [2], [13], [16], [17], [24], [31], [36], [38]. The aim of a RL algorithm is to maximize the cumulative rewards by learning strategies through the interaction with the environment. Nevertheless, traditional RL algorithms have the problem of lack of scalability [2]. To overcome this drawback of the RL, the deep RL (DRL) algorithm has been put forward by employing the popular deep neural networks (DNNs) in RL algorithms. Up to now, DRL algorithms have achieved a great success in many research fields, such as intelligent control [3], [17], [24], strategy analysis [13], [31] and image processing [38]. Served as a popular DRL algorithm, the deep Q-learning network (DQN) proposed in [23] has shown competitive performance for high-dimensional problems [1], [8]. According to the characteristics of GICS images, the image segmentation problem can be regarded as the process of finding the optimal boundary of test and control lines. Hence, it is natural to develop an effective segmentation method based on the DRL algorithm to segment the GICS images for quantitative analysis of GICS.

This work was supported in part by International Science and Technology Cooperation Project of Fujian Province of China under Grant 2019I0003, in part by the Korea Foundation for Advanced Studies, in part by the Fundamental Research Funds for the Central Universities of China under Grant 20720190009, in part by The Open Fund of Engineering Research Center of Big Data Application in Private Health Medicine of China under Grant KF2020002, and in part by The Open Fund of Provincial Key Laboratory of Eco-Industrial Green Technology-Wuyi University of China.

N. Zeng, H. Li and S. Liu are with the Department of Instrumental and Electrical Engineering, Xiamen University, Fujian 361005, China. Email: zny@xmu.edu.cn

Z. Wang is with the College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China. He is also with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom. Email: Zidong.Wang@brunel.ac.uk

W. Liu and X. Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom.

F. E. Alsaadi is with the Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia.

*Corresponding author.

To apply the DRL algorithm to GICS images segmentation, the state, action, and reward of DRL algorithm are defined according to the characteristics of GICS images. In general, the architecture of the network has significant impacts on the performance of the DQN algorithm. Notice that the DBN has been successfully applied to segment the two lines in the GICS images in [41]. Hence, a seemingly natural idea is to employ a DBN to establish the feature space for the DRL algorithm due to its competitive representation learning ability. In addition, the learning efficiency of the DRL algorithm is decreased if the capacity of replay buffer and the size of random sampling are fixed. In the DQN, the replay buffer capacity and the sampling size are related to the discount factor and learning rate. As such, the multi-factor learning curve proposed in [35] has been introduced in this work to dynamically adjust the replay buffer capacity and the sampling size with hope to enhance the leaning efficiency of DRL algorithm. The developed DBN-based DRL algorithm possesses two advantages: 1) the developed DBN-based DRL algorithm is capable of extracting the state features with higher accuracy than that of the standard RL algorithm; and 2) the learning efficiency of the developed DBN-based DRL algorithm is improved, which benefits the image segmentation process.

The main contributions of our work can be outlined as follows: 1) a DRL-based image segmentation method is developed for accurately recognizing the control and test lines in GICS images; 2) a modified deep Q-learning algorithm is proposed by utilizing the DBN and the multi-factor learning curve to enhance the feature extraction ability and the learning efficiency of the DRL algorithm; and 3) the DRL algorithm is applied to the quantitative analysis of GICS for the first time.

The remainder of this paper is organized as follows. In Section II, the preliminaries about RL, Q-learning, and Deep Q-learning are presented. The developed DRL algorithm combined with the DBN and the multi-factor learning curve are presented in Section III. GICS images and the introduced image segmentation method based on the proposed DRL algorithm are described in Section IV. Experiment results and performance evaluation are discussed in section V. Finally, conclusions are drawn in VI.

II. Preliminaries

In this section, the background of the basic RL system, Q-learning and deep Q-learning are presented.

A. Reinforcement Learning

RL is a popular machine learning method which aims to learn satisfactory policies to solve sequential decision problems by optimizing a cumulative reward signal [27], [32]. Generally, a typical RL model is defined by a 4-tuple (*S, A, P, R*), where S denotes the state space, A represents the action space, P is the state transition probability, and R stands for the reward function [27], [33]. Through the process of interaction with the environment, the agent learns to act in a specific state to obtain the maximum future rewards. Notice that it is of practical significance to balance the short-term and long-term benefits of the agent while making decisions [25]. According to the cumulative learning experience, the RL agent is capable of selecting the best action for each state with the maximized cumulative rewards.

B. Q-Learning

It should be noticed that the Markov decision process is used in the RL algorithm. The reward r as well as the next state s' are obtained according to following formula:

$$p(s', r|s, a) = P\{S_t = s', R_t = r|S_{t-1} = s, A_{t-1} = a\} \quad (1)$$

where s denotes the current state, s' is the next state, t represents the time step, a is the action.

The long-term cumulative discounted rewards G_t is shown as follows:

$$G_t = \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k \quad (2)$$

where R_k is the received reward at time step k, and γ is a discount factor which represents the trade-off between short-term and long-term gains.

Meanwhile, the state-value function based on Eq. (1) is defined by:

$$Q_\pi(s, a) = E_\pi[G_t|S_t = s, A_t = a] \quad (3)$$

Then, the optimal action for each state is determined when the state-value function Eq. (3) reaches the best solution. In the RL algorithm, the *Q*-table is employed to store $Q(s, a)$ for all states. The Q-learning algorithm aims to optimize the state-value function by iteratively updating the *Q*-table [10], [34]. The updating equation of the *Q*-table is given as follows [33]:

$$\begin{aligned} Q(S_t, A_t) \leftarrow & Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_{a \in A(S_{t+1})} Q(S_{t+1}, a) \\ & - Q(S_t, A_t)] \end{aligned} \quad (4)$$

where α stands for the learning rate.

C. Deep Q-Learning

Essentially, Q-learning is a binary discrete function:

$$Q(S, A) = f(s, a) \quad (5)$$

Although the traditional Q-learning algorithm performs well in the low-dimensional state space, the performance of the Q-learning algorithm is not satisfactory in the high-dimensional state space. In the high-dimensional state space, the *Q*-table is not able to cover all the states, and the large amount of data leads to high computational burden. Therefore, the function fitting method is employed to solve the limitation of Q-learning algorithm for high-dimensional problem.

Owing to its power in function approximation, neural networks have been introduced in the Q-learning algorithm. The deep Q-learning network (DQN) proposed in [23] which is a combination of a neural network and Q-learning has shown competitive performance for RL problems [1], [8]. In [24], a double network structure has been put forward to describe the correlation between the state-action value function and the

update target, where the two networks involved are named as the Q-network and the target network, respectively.

The output of the Q-network represents the solution of the state-value function $Q(S, A)$, and the output of target network serves as the label of the Q-network. It is remarkable that the two networks have same structures, and the parameter updating of two networks is asynchronous. The parameters of the Q-network update at every iteration. The parameters of the target network update at every C iterations. Note that the parameters of the target network remain unchanged when the Q-network updates the parameters [24].

It should be pointed out that there exists inevitable correlation between the samples. Therefore, a replay buffer is used in the DQN to reduce the correlation between samples and increase the sample efficiency [2]. That is, a replay buffer is employed to store the samples generated by the agent interacting with the environment. During the training process of the DQN, a small batch of samples are selected from the replay buffer. Then, the parameters of the Q-network and the target network are updated by using the stochastic gradient descent method, which greatly reduces the correlation among samples to solve the local optimal problem to some extent. It should be noted that the architecture of the network has significant impacts on the performance of the DQN algorithm.

III. Improved Deep Reinforcement Learning

In this section, the proposed modified DQN and the multi-factor learning curve are discussed.

A. Modified DQN

In this paper, the deep belief network (DBN) [11], [26], [41] is selected as the Q-network and the target network, of which the structure is shown in Fig. 1.

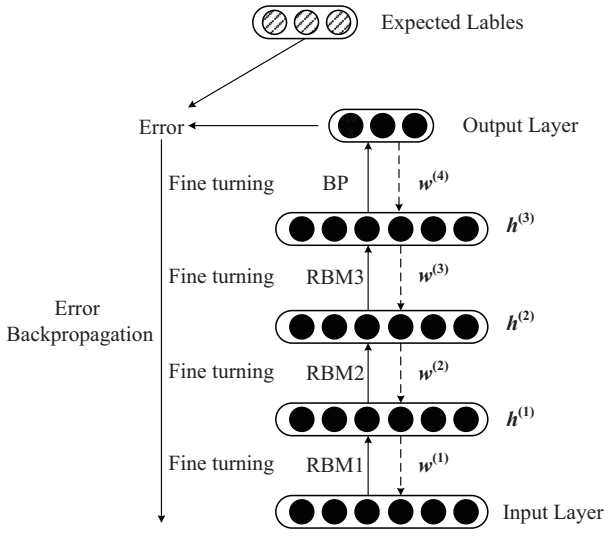


Fig. 1. Schematic diagram of a DBN

The DBN is composed of stacked restricted Boltzmann machines (RBMs) and a back propagation (BP) layer. In a DBN, the greedy learning algorithm is utilized to optimize the weights of the network layer-by-layer, which is divided into two phases. The first phase is pre-training, which performs in a bottom-up manner. The second phase is the fine-tuning phase, where the error is propagated from up to bottom to adjust the parameters of the entire network.

For the jth node of the output layer in a DBN, we assume that the actual output is o_j and the expected output is e_j. The sensitivity δ_j is computed by using the following formula:

$$\delta_j = o_j(1 - o_j)(e_j - o_j) \tag{6}$$

For the lth hidden layer, y_i is the output of the ith node, and the sensitivity δ_i is calculated via the following equation:

$$\delta_i^l = y_i^l(1 - y_i^l) \sum_j w_{ij}^l \delta_j^{l+1} \tag{7}$$

The weight and bias of a DBN are updated as follows:

$$w_{ij}^l = w_{ij}^l + \varepsilon_{\text{fine-tuning}} y_i^l \delta_j^{l+1} \tag{8}$$

$$b_j^l = b_j^l + \varepsilon_{\text{fine-tuning}} \delta_j^{l+1} \tag{9}$$

The schematic diagram of the DQN embedded with DBN algorithm is depicted in Fig. 2:

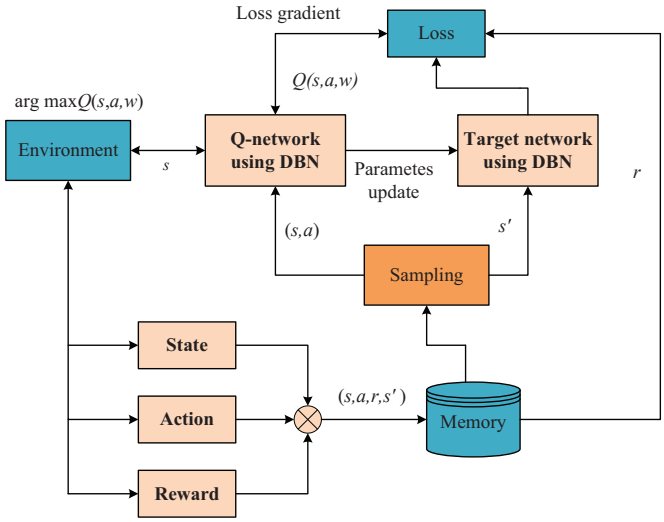


Fig. 2. The schematic diagram of DQN embedded with DBN algorithm

The loss function of DQN is shown as follows:

$$\text{Loss}(\theta, Q, y) = \frac{1}{2}[y(s,a) - Q(s,a,\theta)]^2 \tag{10}$$

where $y(s,a)$ represents the label of Q-network, which is determined through maximizing the value of state-value function:

$$y(s,a) = r + \max_{a'} Q(s', a', \theta^-) \tag{11}$$

where θ^- denotes the parameters of the target network, and θ^- is fixed during the computation of $y(s,a)$.

B. Multi-factor Learning Curve

Through the training process of a DQN algorithm, the learning efficiency is decreased if the capacity of replay buffer and the size of random sampling are fixed. In [29], the prioritized replay sampling method is utilized to ensure that the DQN is able to obtain satisfactory training samples with a large probability, which eventually leads to enhanced training efficiency. Notice that the optimized replay buffer can add high quality samples, but ignores the influence of the replay buffer capacity.

It should be noted that the relationship between the training steps and the learning ability of the DQN is similar to the core ideal of learning curve [35]. The theory of learning curve aims to describe the process that an individual enhances the learning ability through the accumulation of experience. The learning curve model is mainly divided into two categories, which are the single factor model and the multi-factor model. In general, the leaning ability of an individual is related to several factors. In the DQN, the replay buffer capacity and the sampling size are related to the discount factor γ and learning rate α. Hence, the multi-factor learning curve is utilized in this paper to dynamically adjust the replay buffer capacity and the sampling size with hope to enhance the leaning efficiency of the DQN.

In this paper, the updating rules of the replay buffer capacity and the sampling size based on the learning curve model are given by:

$$R \leftarrow R\frac{1}{k}(steps)^{-\frac{\log\gamma}{\log 2}} * \left\lceil \frac{steps}{C} \right\rceil^{-\frac{\log(1-\alpha)}{\log 2}} \tag{12}$$

$$N \leftarrow N\frac{1}{k}(steps)^{-\frac{\log\gamma}{\log 2}} * \left\lceil \frac{steps}{C} \right\rceil^{-\frac{\log(1-\alpha)}{\log 2}} \tag{13}$$

where *R* denotes the capacity of replay buffer; *N* represents the sampling size; $steps$ denotes the current training step; *k* stands for the performance of the first training whose default value is 1; γ represents the discount factor; α is the learning rate, and C is the iteration number. It should be mentioned that the discount factor γ has an effect on the capacity of the replay buffer and the sampling size at each iteration, while the learning rate α has an effect on the replay buffer and the sampling size at every C iterations.

The pseudocode of the proposed DRL algorithm for the GICS images segmentation in this paper is described as follows:

Algorithm 1: Improved DRL Algorithm

Initialize replay memory *D* to capacity *R*, minibatch size *N*, training steps $steps$, learning rate α and reward discount γ
Initialize Q-network with random weights θ
Initialize target network with weights $\theta^- = \theta$
For episode=1, *M* **do**
  Initialize sequence and calculate initial state s_1
  For t=1, *T* **do**
    Calculate the output $\arg\max_a Q(s_t, a, \theta)$ of Q-network
    Select a random action a_t with a certain probability otherwise select $a_t = \arg\max_a Q(s_t, a, \theta)$
    Execute action a_t, observe reward r_t and next state s_{t+1} according to the Eqs. (14) and (15)
    Update the capacity of replay buffer *R*:

$$R \leftarrow R\frac{1}{k}(steps)^{-\frac{\log\gamma}{\log 2}} * \left\lceil \frac{steps}{C} \right\rceil^{-\frac{\log(1-\alpha)}{\log 2}}$$

    Store transition (s_t, a_t, r_t, r_{t+1}) in replay memory *D*
    Update minibatch size *N*:

$$N \leftarrow N\frac{1}{k}(steps)^{-\frac{\log\gamma}{\log 2}} * \left\lceil \frac{steps}{C} \right\rceil^{-\frac{\log(1-\alpha)}{\log 2}}$$

    Sample random minibatch of transitions (s_i, a_i, r_i, r_{i+1})
    Set $y_i = r_i + \gamma \max_{a'} Q(s_i, a', \theta^-)$
    Perform a gradient descent step to update the weights θ of Q-network:

$$L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i, \theta)^2$$

    Every *C* steps reset $\theta^- = \theta$
    Set $steps \leftarrow steps + 1$
  End For
End For

IV. Deep Reinforcement Learning-Based GICS Images Segmentation

In this section, the background of the GICS images and the detailed DRL-based image segmentation are presented.

A. GICS Images

It is worth mentioning that the control line and test line of the GICS become red when the specimen containing the target analyte passes through the strip. Seven GICS images with different concentrations of human chorionic gonadotropin (hCG) are presented in Fig. 3.

Fig. 3. Images of GICS with different concentrations of hCG.

It should be pointed out that the concentration of the target analyte influences the intensity of the lines, especially the test line. In this case, the quantitative analysis of the GICS can be accomplished by measuring the signal intensity of test and control lines [40]. It can be seen in Fig. 3 that the boundary of the line is too blurry to be recognized, which indicates that it is difficult to precisely segment the control and test lines from the background. As such, we aim to propose an effective image segmentation method with hope to accurately segment the control and test lines.

In this paper, the main objective is to develop an image segmentation method based on the DRL algorithm to accurately segment the GICS images for the quantitative analysis.

In our simulation, each image is divided into two sub-images (including the control line and the test line, respectively) to reduce the computational complexity. In addition, a DRL algorithm is proposed based on the characteristics of the GICS.

B. Processing GICS Images via Deep Reinforcement Learning

The RL agent starts with the top or bottom of each column, and moves either down/up or remains unchanged. The corresponding reward is determined according to the position of the agent. In order to segment the sub-image, it is necessary to find the upper and lower edges of the target line (test line or control line). The edge of the target line is determined when the upper and lower points of the boundary in each column are found.

1) Definition of state: The grayscale intensity of neighbor pixels near the target is used to represent the state of target pixel. The state vector $winsize*winsize$ is obtained by setting the size of the moving window to $winsize$. The mirroring method is applied to fill the pixels outside the window.

2) Definition of action: In this paper, the agent performs two actions, that is, moves or remains unchanged. Action A is expressed as:

$$A = \begin{cases} 0 & \text{stop} \\ 1 & \text{down/up} \end{cases} \tag{14}$$

3) Definition of reward: To evaluate the segmentation performance of the image, the manually segmented image is utilized as the benchmark. In this work, the coincidence rate C_r between the manually segmented image and the segmented image by using the algorithm can be used as the criteria for determining the reward. It should be mentioned that the lower the coincidence, the worse the segmentation performance. In this paper, 0.9 is set as threshold of the coincidence rate. A positive reward is offered if the coincidence rate is bigger than the threshold. On the contrary, a negative value is given if the coincidence rate is smaller than the threshold. The coincidence rate and reward are defined as follows:

$$C_r = \frac{\sum p_f \cap \sum p_{f|opt} + \sum p_b \cap \sum p_{b|opt}}{\sum p_f + \sum p_b} \tag{15}$$

$$R = \begin{cases} 10 * C_r & C_r \geq 0.9 \\ -1 & C_r < 0.9 \end{cases} \tag{16}$$

where p_f and p_b represent the pixels in the foreground (test line and control line) and background (other area), respectively; $p_{f|opt}$ and $p_{b|opt}$ are the foreground and background pixels of the manually segmented image (which is the optimal segmented image), respectively.

The flowchart of the GICS image segmentation method based on the developed DRL algorithm is displayed in Fig. 4.

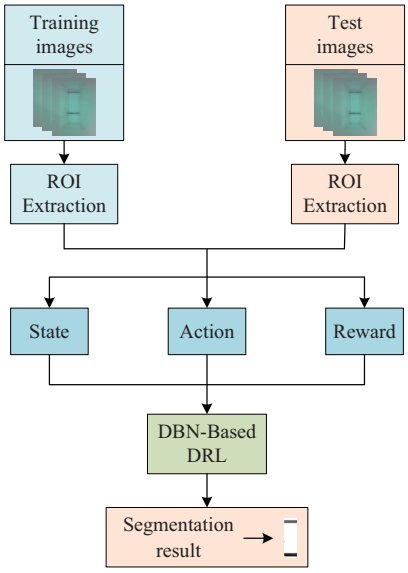


Fig. 4. The flowchart of GICS image segmentation based on the DRL.

V. Experiment Results and Discussions

In this section, the performance of the developed DRL algorithm is evaluated in terms of the segmentation results and the quantitative analysis of GICS images. Here, 10 GICS images with different concentrations of hCG are selected as the training set. Especially, each GICS image is divided into two sub-images, which contain the control line and the test line, respectively. The size of each sub-image is set as 115*270.

In our simulation, a DBN consists of two RBMs, where the number of input node is respectively set to be 100 and 9 for each RBM when the $winsize$ equals to 3. The learning rate α is set to be 0.1, the initial capacity of the replay buffer is set to be 100, and the frequency of changing target network parameters is 20.

A. Image Segmentation

To comprehensively evaluate the performance of the image segmentation, a standard RL algorithm and the developed DRL algorithm are applied to segment the GICS images. For simplicity, three typical segmentation results are shown in Fig. 5, where the left column shows pre-processed images (denoted by original images), the middle and right columns are segmentation results obtained by the RL algorithm and the proposed DRL algorithm, respectively.

It can be seen in Fig. 5 that both the RL algorithm and the proposed DRL algorithm achieve satisfactory results. In order to quantitatively evaluate the segmentation accuracy, a similarity indicator η is employed, which is defined by:

$$\eta = \frac{B_s \cap B_r + F_s \cap F_r}{B_s + F_s} \times 100\% \tag{17}$$

where the subscript index r represents the manually segmented image (labeled image); s stands for the segmentation result; B and F stand for the background and foreground area, respectively. According to Eq. (17), the segmentation accuracy of test and control lines in each test image is listed in Table. I. The total segmentation accuracy is equal to the average of corresponding segmentation accuracy of the control and test lines.

In Table. I, we can see that the developed DRL algorithm outperforms the RL algorithm in the segmentation of GICS

TABLE I
THE SEGMENTATION ACCURACY OF THE GICS IMAGES

Concentration	Control line		Test line		Total accuracy	
	RL (%)	DRL (%)	RL (%)	DRL (%)	RL (%)	DRL (%)
75ml	93.12	94.76	94.03	96.03	93.58	**95.40**
100ml	94.95	96.95	94.01	97.00	94.48	**96.98**
150ml	95.08	97.08	93.56	95.56	94.32	**96.32**
200ml	95.62	97.62	94.36	98.36	94.99	**97.99**
300ml	95.79	98.79	94.40	96.40	95.10	**97.56**
400ml	94.21	97.21	96.12	99.22	95.17	**98.22**
500ml	95.57	96.57	94.60	98.60	95.09	**97.58**
Average	94.90	97.00	94.45	97.31	94.68	**97.16**

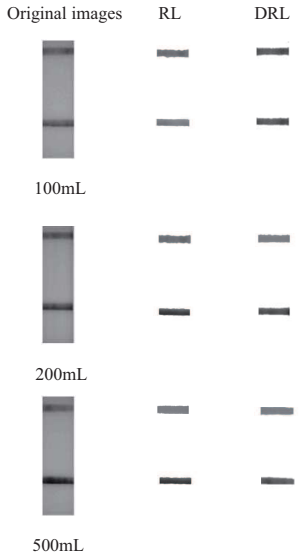


Fig. 5. Segmentation results of the GICS images.

images with different concentrations of hCG, and the total accuracy of DRL algorithm is 2.48% higher than the result obtained by the RL algorithm. Meanwhile, a total segmentation accuracy of 97.16% demonstrates the reliability and efficiency of the developed DRL algorithm. Furthermore, the peak signal-to-noise ratio (PSNR) is utilized to evaluate the segmentation performance of the proposed DRL algorithm. The larger the PSNR value, the better the performance of image segmentation. In order to calculate the PSNR of GICS images, a binary mask is set up to classify the image pixels, where the pixel value of foreground (the test and control line) is set as 1 and the pixel value of background (other areas) is set as 0. The PSNR is calculated by:

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{18}$$

$$\mathrm{PSNR} = 20 * \log_{10}\left[\frac{MAX_I}{\sqrt{\mathrm{MSE}}}\right] \tag{19}$$

where MSE denotes the mean square error between the normalized original image I and the masked image K; MAX_I is the maximum range of the input image data type.

The results of PSNR are illustrated in Fig. 6. It can be seen in Fig. 6 that the developed DRL algorithm has demonstrated outstanding performance with satisfactory PSNR values in the segmentation of GICS images. Especially, it provides higher accuracy than the methods developed in [40].

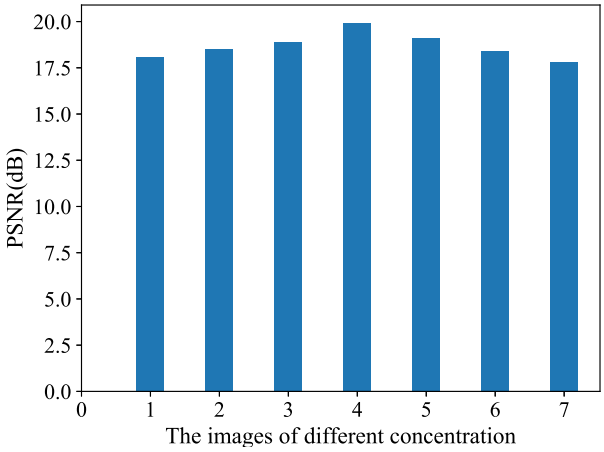


Fig. 6. PSNR values of GICS images with different concentrations.

B. Quantitative Analysis

The objective of segmenting GICS images is to quantitatively determine the concentration of the target analyte in a specimen. Therefore, the relative integral optical density (RIOD) is selected as an indicator for the quantitative analysis of the concentration of hCG [41]. RIOD is defined as:

$$\mathrm{RIOD} = \frac{IOD_T}{IOD_C} = \frac{\sum_{i=1}^{N}\log\frac{G_{avg}}{G_i}}{\sum_{j=1}^{M}\log\frac{G_{avg}}{G_j}} \tag{20}$$

where IDO_T and IDO_C denote the integral optical density of the test line and the control line, respectively; *N* stands for the number of pixels in the test line; *M* denotes the number of pixels in the control line; G_{avg} is the average grayscale pixel

TABLE II
THE ROID VALUES OF DIFFERENT TEST SAMPLES

Concentration(mL)	75	100	150	200	300	400	500
ROID	0.7012	0.9376	1.0432	1.6634	2.0643	2.7968	3.7941

value of the whole image; G_i and G_j represent the grayscale of pixel value in the test line and the control line, respectively. The RIOD values for 7 different concentrations of hCG are listed in Table. II.

In Table. II, it can be seen that the RIOD value distinguishes the concentration of the target analyte very well. Meanwhile, a functional relationship between the RIOD and the concentration of the target analyte is obtained by using the least square method. The scatterplot between the RIOD and the concentration of hCG and the fitted line based on Table. II are shown in Fig. 7.

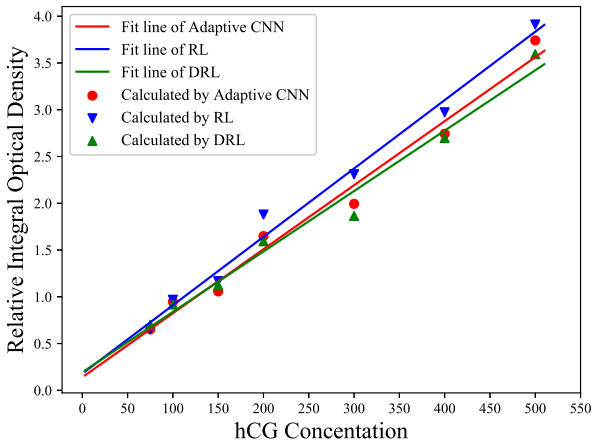


Fig. 7. The scatterplot and fitted line between the RIOD and the concentration of hCG.

In particular, the results obtained by using the RL algorithm and the cellular neural network (CNN) [40] are presented in Fig. 7. To make it convenient for comparison, the fitted linear equation of the DRL algorithm, RL algorithm and the CNN are given in Table. III. In addition, the correlation coefficients of the three methods are also shown in Table. III, where the correlation coefficient of the proposed DRL algorithm is 0.973, which is the best among the three methods. In this case, the concentration of hCG and RIOD value are linear correlated. To summarize, the RIOD value can be used to determine the concentration of target analyte in a specimen for the quantitative analysis of GICS.

VI. CONCLUSIONS

In this paper, a novel image segmentation method based on the DRL algorithm is proposed for the quantitative analysis of GICS images. In our work, the state, action, and the reward are defined based on the characteristics of GICS images. In addition, the DBN is utilized in the DQN algorithm, where

TABLE III
THE COMPARISON OF FITTING RESULTS AMONG THREE METHODS

Method	Fitted linear equation	Correlation coefficient
DRL	y=0.006x+0.241	0.973
RL	y=0.007x+0.179	0.963
CNN	y=0.006x+0.236	0.969

the capacity of the replay buffer and the sampling size are dynamically changed according to the multi-factor learning curve, which can effectively enhance the learning efficiency. Experiments on GICS images with different concentrations have been carried out. Experiment results have demonstrated that the developed DRL algorithm is capable of providing a satisfactory performance in terms of several indices.

In the future, we aim to develop advanced image processing approaches for the quantitative analysis of GICS [4], [5], and apply our developed DRL algorithm to other research areas, for example, discrete-time switched complex networks [6], [22], [46], networked systems [30], [45], [47], [48], and multiagent systems [44]. We can also integrate the developed DRL algorithm with the latest optimization techniques proposed in [18]–[21]. For example, where a variety of effective optimization methods have been developed for petroleum engineering, which saved a lot of investment for the petroleum industry.

REFERENCES

[1] D. Akimov and I. Makarov, Deep reinforcement learning in vizdoom first-person shooter for health gathering scenario, In: *Proceedings of the International Conference on Advances in Multimedia*, Valencia, Spain, 2019, pp. 1-6.

[2] K. Arulkumaran, M. Deisenroth, M. Brundage, and A. Bharath, Deep reinforcement learning: a brief survey, *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26-38, 2017.

[3] D. Bertsekas, Feature-based aggregation and deep reinforcement learning: A survey and some new implementations, *IEEE/CAA Journal of Automatica Sinica* vol. 6, no. 1, pp. 1-31, 2018.

[4] J. Cao, Z. Wu, J. Wu, and W. Liu, Towards information-theoretic K-means clustering for image indexing, *Signal Processing*, vol. 93, no. 7, pp. 2026–2037, 2013.

[5] J. Cao, B. Wang, and D. Brown, Similarity based leaf image retrieval using multiscale R-angle description, *Information Sciences*, vol. 374, pp. 51–64, 2016.

[6] Y. Chen, Z. Wang, L. Wang, and W. Sheng, Mixed H_2/H_∞ state estimation for discrete-time switched complex networks with random coupling strengths through redundant channels, *IEEE Transactions on Neural Networks and Learning Systems*, in press, DOI: 10.1109/TNNLS.2019.2952249.

[7] L. Chuang, J. Hwang, H. Chang, F. Chang, and S. Jong, Rapid and simple quantitative measurement of α-fetoprotein by combining immunochromatographic strip test and artificial neural network image analysis system, *Clinica Chimica Acta*, vol. 348, no. 1-2, pp. 87-93, 2004.

[8] H. Cuayahuitl, Deep reinforcement learning for conversational robots playing games, In: *Proceedings of the IEEE-RAS International Conference on Humanoid Robotics*, Birmingham, UK, 2017, pp. 771-776.

[9] K. Faulstich, R. Gruler, M. Eberhard, and K. Haberstroh, Developing rapid mobile POC systems, part 1: devices and applications for lateral-flow immunodiagnostics, *IVD Technology*, vol. 13, no.6, pp. 47-53, 2007.

[10] A. Gosavi, Simulation-based optimization: parametric optimization techniques and reinforcement learning second edition, *Operations Research Computer Science Interfaces*, vol. 55, pp. 473-476, 2015.

[11] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups, *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82-97, 2012.

[12] J. Kaur, K. Singh, R. Boro, K. Thampi, M. Raje, G. Varshney, and C. Suri, Immunochromatographic dipstick assay format using gold nanoparticles labeled protein-hapten conjugate for the detection of atrazine, *Environmental Science & Technology*, vol. 41, no. 14, pp. 5028-5036, 2007.

[13] M. Lanctot, V. Zambaldi, A. Gruslys, A. Lazaridou, K. Tuyls, J. Perolat, D. Silver, and T. Graepel, A unified game-theoretic approach to multiagent reinforcement learning, *Advances in Neural Information Processing Systems*, pp. 4190-4203, 2017.

[14] J. Li, A. Ouellette, L. Giovangrandi, D. Cooper, A. Ricco, and G. Kovacs, Optical scanner for immunoassays with up-converting phosphorescent labels, *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 5, pp. 1560-1571, 2008.

[15] Y. Li, N. Zeng, and M. Du, A novel image methodology for interpretation of gold immunochromatographic strip, *Journal of Computers*, vol. 6, no. 3, pp. 540-547, 2011.

[16] Y. Li, Deep reinforcement learning: an overview, *arXiv:1810.06339*, 2018.

[17] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, Continuous control with deep reinforcement learning, *arXiv:1509.02971v6*, 2019.

[18] Y. Liu, Q. Cheng, Y. Gan, Y. Wang, Z. Li and J. Zhao, Multi-objective optimization of energy consumption in crude oil pipeline transportation system operation based on exergy loss analysis, *Neurocomputing*, vol. 332, pp. 100–110, Mar. 2019.

[19] Y. Liu, S. Chen, B. Guan and P. Xu, Layout optimization of large-scale oil-gas gathering system based on combined optimization strategy, *Neurocomputing*, vol. 332, pp. 159–183, Mar. 2019.

[20] Y. Liu and G. Chen, Optimal parameters design of oilfield surface pipeline systems using fuzzy models, *Information Sciences*, vol. 120, no. 1, pp. 13-21, 1999. 1999, 120(1): 13-21

[21] Y. Liu, Theories and methodologies of petroleum engineering optimization design. Beijing: Petroleum Industry Press, 1994.

[22] L. Ma, Z. Wang, C. Cai, and F. E. Alsaadi, A dynamic event-triggered approach to H_∞ control for discrete-time singularly perturbed systems with time-delays and sensor saturations, *IEEE Transactions on Systems, Man, and Cybernetics-Systems*, in press, DOI: 10.1109/TSMC.2019.2958529.

[23] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, Playing atari with deep reinforcement learning, *arXiv:1312.5602v1*, 2013.

[24] V. Mnih, K. Kavukcuoglu, D. Silver, A. Rusu, J. Veness, M. Bellemare, A. Graves, M. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, Human-level control through deep reinforcement learning, *Nature*, vol. 518, no. 7540, pp. 529-533, 2015.

[25] E. Mocanu, D. Mocanu, P. Nguyen, A. Liotta, M. Webber, M. Gibescu, and J. Slootweg, On-line building energy optimization using deep reinforcement learning, *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3698-3708, 2019.

[26] A. Mohamed, G. Dahl, and G. Hinton, Acoustic modeling using deep belief networks, *IEEE Transactions on Audio Speech and Language Processing*, vol. 20, no. 1, pp. 14-22, 2012.

[27] P. Montague, Reinforcement learning: An introduction, *Trends in Cognitive Sciences*, vol. 3, no. 9, pp. 360, 1999.

[28] S. Qian and H. Bau, A mathematical model of lateral flow bioreactions applied to sandwich assays, *Analytical Biochemistry*, vol. 322, no. 1, pp. 89-98, 2003.

[29] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, Prioritized Experience Replay, *arXiv:1511.05952v4*, 2016.

[30] Y. Shen, Z. Wang, B. Shen, F. E. Alsaadi and F. E. Alsaadi, Fusion estimation for multi-rate linear repetitive processes under weighted Try-Once-Discard protocol, *Information Fusion*, vol. 55, pp. 281–291, Mar. 2020.

[31] D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Vandendriessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, Mastering the game of Go with deep neural networks and tree search, *Nature*, vol. 529, no. 7587, pp. 484-489, 2016.

[32] Z. Wang, Z. Shi, Y. Li, and J. Tu, The optimization of path planning for multi-robot system using Boltzmann policy based Q-learning algorithm, In: *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Shenzhen, China, 2013, pp. 1199-1204.

[33] C. Watkins and P. Dayan, Q-learning, *Machine Learning*, vol. 8, no. 3-4, pp. 279-292, 1992.

[34] R. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning, *Machine Learning*, vol. 8, no. 3-4, pp. 229-256, 1992.

[35] T. Wright, Factors affecting the cost of airplanes, *Journal of the Aeronautical Sciences*, vol. 3, no. 4, pp. 122-128, 1936.

[36] L. Xue, C. Sun, D. Wunsch, Y. Zhou, F. Yu, An adaptive strategy via reinforcement learning for the prisoner's dilemma game, *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 301-310, 2017.

[37] P. Yager, T. Edwards, E. Fu, K. Helton, K. Nelson, and M. Tam, Microfluidic diagnostic technologies for global public health, *Nature*, vol. 442, no. 7101, pp. 412-418, 2006.

[38] K. Yu, C. Dong, L. Lin, and C. Loy, Crafting a toolchain for image restoration by deep reinforcement learning, In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 2443-2452.

[39] N. Zeng, Z. Wang, Y. Li, M. Du, J. Cao, and X. Liu, Time series modeling of nano-gold immunochromatographic assay via expectation maximization algorithm, *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 3418-3424, 2013.

[40] N. Zeng, Z. Wang, B. Zineddin, Y. Li, M. Du, L. Xiao, X. Liu, and T. Young, Image-based quantitative analysis of gold immunochromatographic strip via cellular neural network approach, *IEEE Transactions on Medical Imaging*, vol. 33, pp. 1129-1136, 2014.

[41] N. Zeng, Z. Wang, H. Zhang, W. Liu, and F. E. Alsaadi, Deep belief networks for quantitative analysis of gold immunochromatographic strip, *Cognitive Computation*, vol. 8, pp. 684-692, 2016.

[42] N. Zeng, H. Li, Y. Li, and X. Luo, Quantitative analysis of immunochromatographic strip based on convolutional neural network, *IEEE Access*, vol. 7, no. 1, pp. 16257-16263, 2019.

[43] N. Zeng, Z. Wang, H. Zhang, K. Kim, Y. Li, and X. Liu, An improved particle filter with a novel hybrid proposal distribution for quantitative analysis of gold immunochromatographic strips, *IEEE Transactions on Nanotechnology*, vol. 18, no. 1, pp. 819-829, 2019.

[44] W. Zhang, Z. Wang, Y. Liu, D. Ding, and F. E. Alsaadi, Sampled-data consensus of nonlinear multiagent systems subject to cyber attacks, *International Journal of Robust and Nonlinear Control*, vol. 28, no. 1, pp. 53–67, 2018.

[45] D. Zhao, Z. Wang, Y. Chen and G. Wei, Proportional-integral observer design for multi-delayed sensor-saturated recurrent neural networks: A dynamic event-triggered protocol, *IEEE Transactions on Cybernetics*, in press, 10.1109/TCYB.2020.2969377

[46] Z. Zhao, Z. Wang, L. Zou and H. Liu, Finite-horizon H_∞ state estimation for artificial neural networks with component-based distributed delays and stochastic protocol, *Neurocomputing*, vol. 321, pp. 169–177, Dec. 2018.

[47] L. Zou, Z. Wang, Q.-L. Han, and D. Zhou, Moving horizon estimation of networked nonlinear systems with random access protocol, *IEEE Transactions on Systems, Man, and Cybernetics-Systems*, in press, DOI:10.1109/TSMC.2019.2918002.

[48] L. Zou, Z. Wang, Q.-L. Han, and D. Zhou, Moving horizon estimation for networked time-delay systems under Round-Robin protocol, *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5191-5198, 2019.

Nianyin Zeng was born in Fujian Province, China, in 1986. He received the B.Eng. degree in electrical engineering and automation in 2008 and the Ph.D. degree in electrical engineering in 2013, both from Fuzhou University. From October 2012 to March 2013, he was a Research Assistant in the Department of Electrical and Electronic Engineering at the University of Hong Kong.

Currently, he is an Assistant Professor with the Department of Instrumental & Electrical Engineering of Xiamen University. Especially, Dr. Zeng is an ISEF Fellow founded by the Korea Foundation for Advance Studies and also a Visiting Professor at the Korea Advanced Institute of Science and Technology from September 2017. His current research interests include intelligent data analysis, computational intelligent, time-series modeling and applications. He is the author or co-author of several technical papers and also a very active reviewer for many international journals and conferences.

Dr. Zeng is currently serving as an Associate Editor for Neurocomputing, and also Editorial Board members for Computers in Biology and Medicine and Biomedical Engineering Online.

Han Li received the bachelor's degree in measurement and control technology and instrumentation from Xiamen University, Xiamen, China, in 2018. He is currently working toward the master's degree in electrical testing technology and instruments at Xiamen University, Xiamen, China. His research interests include medical image processing, intelligent optimization algorithm and deep learning techniques.

Zidong Wang (SM'03-F'14) was born in Jiangsu, China, in 1966. He received the B.Sc. degree in mathematics in 1986 from Suzhou University, Suzhou, China, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology, Nanjing, China.

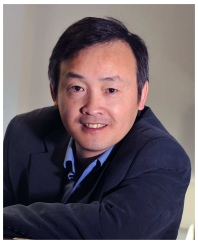
He is currently a Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the UK. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published around 600 papers in refereed international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for Neurocomputing, Deputy Editor-in-Chief for International Journal of Systems Science, and an Associate Editor for 12 international journals including IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology, IEEE Transactions on Neural Networks, IEEE Transactions on Signal Processing, and IEEE Transactions on Systems, Man, and Cybernetics-Part C. He is a Fellow of the IEEE, a Fellow of the Royal Statistical Society and a member of program committee for many international conferences.

Weibo Liu received the B.S. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, UK, in 2015, and the Ph.D. degree in computer science from Brunel University London, Uxbridge, UK, in 2019. He is currently a research fellow with the Department of Computer Science at Brunel University London, Uxbridge, UK. His research interests include big data analysis and deep learning techniques.

Songming Liu received the bachelor's degree in mechanical design and manufacturing and automation from Hefei University of Technology, Hefei, China, in 2017. He is currently working toward the master's degree in electrical testing technology and instruments at Xiamen University, Xiamen, China. His research interests include image processing and reinforcement learning techniques.

Fuad E. Alsaadi received the B.S. and M.Sc. degrees in electronic and communication from King AbdulAziz University, Jeddah, Saudi Arabia, in 1996 and 2002. He then received the Ph.D. degree in Optical Wireless Communication Systems from the University of Leeds, Leeds, UK, in 2011. Between 1996 and 2005, he worked in Jeddah as a communication instructor in the College of Electronics & Communication. He is currently an associate professor of the Electrical and Computer Engineering Department within the Faculty of Engineering, King Abdulaziz University, Jeddah, Saudi Arabia. He published widely in the top IEEE communications conferences and journals and has received the Carter award, University of Leeds for the best PhD. He has research interests in optical systems and networks, signal processing, synchronization and systems design.



Xiaohui Liu received the B.Eng. degree in computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in computer science from Heriot-Watt University, Edinburgh, U.K., in 1988.

He is currently a Professor of Computing at Brunel University. He leads the Intelligent Data Analysis (IDA) Group, performing interdisciplinary research involving artificial intelligence, dynamic systems, image and signal processing, and statistics, particularly for applications in biology, engineering and medicine. Professor Liu serves on editorial boards of four computing journals, founded the biennial international conference series on IDA in 1995, and has given numerous invited talks in bioinformatics, data mining and statistics conferences.