1-18

Cellular Neural Networks with Update Template by Using Reinforcement Learning

Kazushige NATSUNO, Yoko UWATE and Yoshifumi NISHIO
(Tokushima University)

1. Introduction

In a general way, a conventional CNN can be easy to implement as a hardware by simple structure. However, the conventional CNN is restricted to the performance by spatially-uniform template. Therefore, we propose CNN with update template by using reinforcement learning. In the proposed CNN, the templates are updated by neighboring output values.

From some simulation results, we confirm that the proposed CNN is effective for image processing.

2. Proposed CNN

In this section, we explain the algorithm of the proposed CNN. In the proposed CNN, the template of all cells are changed at every iterations by reinforcement learning. The learning steps in proposed CNN are described as follows.

Step 1 : State and output values are updated according to Eqs. (1) and (2) as follow.

State equation :

$$\begin{aligned}\frac{dv_{xij}}{dt} &= -v_{xij}+\sum_{k=i-r}^{i+r}\sum_{l=j-r}^{j+r}A_{(i,j;k,l)}v_{xkl}(t) \\ &\quad +\sum_{k=i-r}^{i+r}\sum_{l=j-r}^{j+r}B_{(i,j;k,l)}v_{ukl}(t)+T, \\ &\quad (|i-k|\leq 1 \quad , \quad |j-l|\leq 1). \end{aligned} \tag{1}$$

Output equation :

$$v_{yij}(t)=\frac{1}{2}(|v_{xij}(t)+1|-|v_{xij}(t)-1|). \tag{2}$$

Step 2 : The output values of the center cell and neighbor cells are compared. The comparison equation for the $cell(i,j)$ is described Eq. (3).

Comparison Equation :

$$D(i,j;k,l)=|v_y^{center}(i,j)-v_y^{neighbo}(k,l)|. \tag{3}$$

Step 3 : Among the eight calculated values of $D(i,j;k,l)$, the difference values of cell are ranked based on the small value, respectively. In addition, templates of similar and different cells are updated according to Eqs. (3) and (4) as follow.

TD-error :

$$TD-error=\{Delta(i,j;k,l)+\gamma * a_{(i,j;k,l)}^{presently}\}-a_{(i,j;k,l)}^{initial}. \tag{4}$$

Updated equation :

$$a_{(i,j;k,l)}^{updated}=a_{(i,j;k,l)}^{presently}-\alpha *(TD-error). \tag{5}$$

where $a(i,j;k,l)$, α and γ represent the template, a learning rate ($0 \leq \alpha \leq 1$) and a reduction rate ($0 \leq \gamma \leq 1$), respectively.

Step 4 : The step 1 to 3 are repeated every 0.005 [τ].

3. Simulation result

In this section, we show simulation results using the proposed CNN. In this simulation, the parameters of the learning rate and the reduction rate are set to $\alpha = 0.5$ and $\gamma = 0.5$, respectively. Also, templates and threshold are assigned as follows.

Binary template:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \ B = T = 0. \tag{6}$$

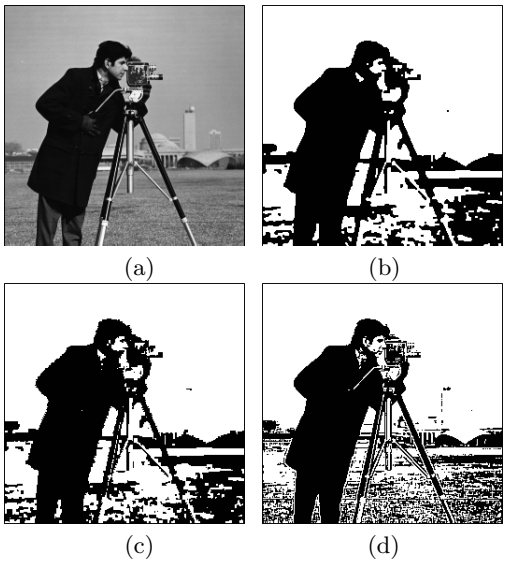

Figure 1: Simulation results. (a) Input image, (b) Conventional CNN, (c) Proposed CNN(Similar cell), (d) Proposed CNN(Different cell).

In Fig. 1(a), the buildings of the background are the indistinct and the detail portions. Figure 1(b) shows the simulation result by the conventional CNN. One the other hand, Fig 1(c) and (d) show the simulation results by using the proposed CNN. In the case of using the conventional CNN, the indistinct portions are not detected. In addition, the detail portions are blacked out in Fig. 1(b). In contrast, in Figs. 1(d), the indistinct portions can be detected. Also, in the case of updated template at similar value, the detail portions are blacked out in Fig. 1(c). On the other hand, in the case of updated template of the different values, the detail portions are shown in Fig. 1(d).

From the simulation results, in the proposed CNN, the output values are changed by update template of selected cell. Therefore, we can say that the proposed CNN is more effective than the conventional CNN.

4. Conclusions

In this study, we proposed CNN with update template by using reinforcement learning. In the proposed CNN, templates are updated by neighboring output values.

From the simulation results, in the proposed CNN, the output values is changed by update template of selected cell. Therefore, we can say that the proposed CNN is more effective than the conventional CNN.