Extended Abstracts of the 2006 International Conference on Solid State Devices and Materials, Yokohama, 2006, pp. 594-595

P-5-6

A Hardware-Implementation-Friendly PCNN for Analog Image-Feature-Generation Circuits

Jun Chen[1] and Tadashi Shibata[2]

[1]Dept. of Electronic Engineering, Univ. of Tokyo, [2]Dept. of Frontier Informatics, Univ. of Tokyo
5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-8561, Japan
Phone: +81-4-7136-3851 Fax: +81-4-7136-3852 E-mail: chenjun@else.k.u-tokyo.ac.jp, shibata@ee.t.u-tokyo.ac.jp

1. Introduction

Pulse-Coupled Neural Networks (PCNNs) are neural models initially introduced for modeling cat's visual cortex and developed for high-performance biomimetic image processing [1]. It has been shown that PCNNs are highly effective tools for image recognition, featuring significant merits of robustness against noise, independence of geometric variations in input patterns, etc [1][2].

However, conventional PCNNs are software-orientated models that are too complicated to implement as VLSI hardware. To employ PCNNs in image recognition VLSIs, a hardware–implementation-friendly PCNN model is proposed here. Two new concepts have been introduced in the model: pulse output with an exponentially-decaying tail, and one-branch dendritic tree. The functions of these two concepts can greatly simplify the PCNN without compromising its performance for image feature generation.

A novel analog image-feature-generation circuit based on the model is proposed. Autonomous running and voltage-mode operation are two features of the circuit. Moreover, the image features generated by the circuit are precision-controllable and are independent of the rotation and translation of input patterns. The superior performance of the circuit has been verified by SPICE simulations based on standard 0.35μm CMOS technology.

2. Hardware-Implementation-Friendly PCNN

The PCNN is composed of a 2D array of output-coupled neurons. Each neuron receives the outputs of its nearest-four neighbors as local stimuli, as well as the intensity of its corresponding pixel in an input image as an external stimulus. Fig. 1 illustrates the differences in neuron structure between conventional PCNNs [1] and the newly-proposed hardware-implementation-friendly PCNN. In the conventional model, the neuron's output is a sequence of pulses. When the pulse enters other neurons as a local stimulus, it is converted to an exponentially-decaying input by convoluting it with an exponential function. The circuit implementation of such conversion would be highly complicated. In the new model, however, the neuron's output is a sequence of pulses with exponentially-decaying tails. Such output can enter other neurons without any conversion and can easily be implemented with a switched-RC circuit described later. In the conventional model, the dendritic tree is divided into two braches: a primary branch (F) and a secondary branch (L), which are then combined in the form of $F(1+\beta L)$. On the other hand, the dendritic tree in

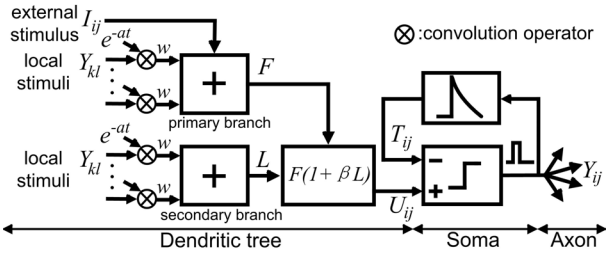

Fig. 1(a) The neuron structure of conventional PCNNs.

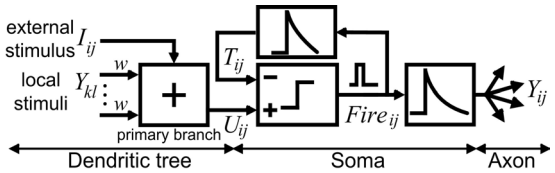

Fig. 1(b) The neuron structure of the new PCNN.

the new model consists of only the primary branch and consequently no multiplier is needed. This simplification is based on the following fact: although the secondary branch plays a role in region-related image processing, it only has a minor effect on image feature generation. The mathematical model of the new PCNN is given below.

$$U_{ij}(t) = I_{ij} + w[Y_{ij}(t) + Y_{i-1,j}(t) + Y_{i+1,j}(t) + Y_{i,j-1}(t) + Y_{i,j+1}(t)] \quad (1)$$

If $U_{ij}(t) > T_{ij}(t)$

$$Y_{ij}(t) = Y_0, T_{ij}(t) = T_0, \sigma_{ij} = t \quad (2)$$

Otherwise

$$Y_{ij}(t) = Y_0 e^{-\alpha_Y (t-\sigma_{ij})}, T_{ij}(t) = T_0 e^{-\alpha_T (t-\sigma_{ij})} \quad (3)$$

$$Fire_{ij}(t) = H(t-\sigma_{ij}) - H(t-\sigma_{ij}-t_d) \quad (4)$$

$$Feature(t) = \sum_{ij} Fire_{ij}(t) \quad (5)$$

Here, *ij* indicates the neuron's position in the 2D network, U is the internal activity, I is the pixel intensity, Y's are neurons' output values, T is the threshold, σ is the latest firing time, *Fire* is the fire signal, *H(t)* is the Heaviside step function, t_d is the width of the fire signal. *Feature(t)*, the sum of the fire signals of all the neurons, is the output of the network, i.e. a time-domain feature of the input image. The rest of the terms are scalars ($0<Y_0$,$0<T_0$, $0<\alpha_Y<\alpha_T$).

3. Analog Image-Feature-Generation Circuit

The analog image-feature-generation circuit is designed based on the above PCNN model. It is composed of

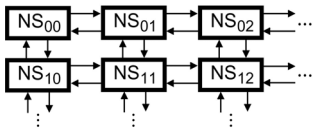

Fig. 2 The image-feature-generation circuit composed of a 2D array of neuron subcircuits (NSs). Each NS corresponds to one pixel in the input image.

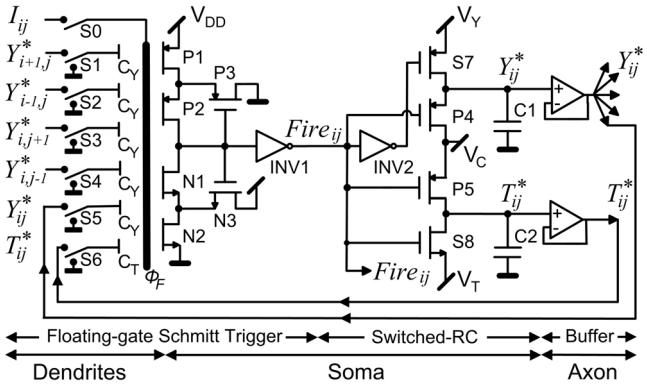

Fig. 3 The structure of the neuron subcircuit (C_T>>C_Y, V_Y>V_C>V_T, $(W/L)_{P5}$>$(W/L)_{P4}$, C1=C2).

a 2D array of neuron subcircuits (NSs, Fig. 2). Each NS corresponds to one pixel in the input image. The NS emulates the dynamics of the PCNN neuron, operating in the following way (Fig. 3). First, S1~S6 are switched to ground and S0 is turned on. Therefore, the potential of the floating gate (Φ_F) becomes I_{ij}, representing the intensity of the NS's corresponding pixel. Then, S0 is turned off and thus I_{ij} is stored in the floating gate. Next, S1~S6 are switched to the input terminals, all of which are then capacitively coupled to the floating gate. Consequently, Φ_F becomes

$$\phi_F = I_{ij} + \frac{C_T T_{ij}^* + C_Y Y_{ij}^* + C_Y Y_{i-1,j}^* + C_Y Y_{i+1,j}^* + C_Y Y_{i,j-1}^* + C_Y Y_{i,j+1}^*}{C_{Total}} \quad (6)$$

which exceeds the low-to-high threshold (V_{M+}) of the Schmitt Trigger (ST). Accordingly, *Fire* goes high and the switches S7, S8 turn on. Therefore, Y^* and T^* are rapidly reset to V_Y and V_T, respectively. As a result, Φ_F falls below the high-to-low threshold (V_{M-}) of the ST. Accordingly, *Fire* goes low and S7, S8 turn off. C1 is then discharged through active resistor P4 and C2 is charged through active resistor P5, resulting in an exponential decay of Y^* and an exponential increase of T^*. Thus Φ_F increases. Once Φ_F reaches V_{M+}, *Fire* goes high again and the reset/decay starts again. *Fire* goes low and high during the reset/decays, generating fire signals like Eq. (4). It can be further proved that the operation of the NS accords with the above PCNN neuron model as long as $V_{M+}=V_C$, $w=C_Y/C_{Total}$, $Y_0=V_Y-V_C$ and $T_0=(V_C-V_T)C_T/C_{Total}$. It is worth noting that the NS is operated in a voltage mode, and the NS runs autonomously after S1~S6 are switched to the input terminals.

The operation of the whole feature generation circuit is as follows. First, pixel intensities are stored in the floating gates of the NSs. Next, the NSs operate simultaneously in the way described above. The *Fire* signals in the NSs are summed up by an analog adder, and the sum becomes the output of the circuit, i.e., a time-domain feature of the input image. The feature's precision can easily be controlled by modifying the length of the running time of the circuit.

4. Simulation Results

Fig. 4 shows the SPICE simulation results of the feature generation circuit. The circuit produces different features for different input patterns (Fig. 4(a) (b) and (c)), showing that the circuit serves as a good tool for pattern discrimination. Meanwhile, the circuit produces highly similar features when the pattern is rotated and translated (Fig. 4(c) and (d)), showing the circuit's capability to generate rotation-independent and translation-independent features.

Furthermore, since analog circuits are more susceptible to fabrication imperfections and transistor characteristic fluctuations, the impact of this phenomenon on our circuit has also been investigated by simulating a more "realistic" circuit, where the threshold voltages of some transistors in the circuit (10%) have been artificially modified by 2%V_{DD}. The simulation result (Fig. 5) shows that the circuit is robust against such fluctuations in transistor characteristics.

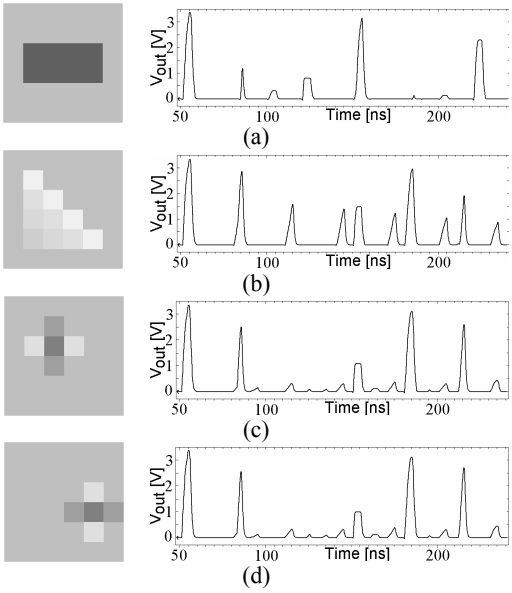

Fig. 4 SPICE simulation results of the circuit. Figures in the left are input images, figures in the right are the corresponding outputs of the circuit, i.e., the time-domain features of the input images.

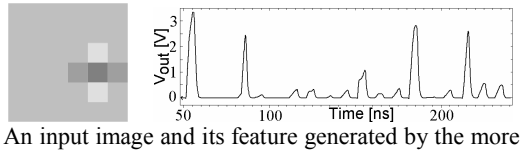

Fig. 5 An input image and its feature generated by the more "realistic" circuit that includes transistor characteristic fluctuations. The feature is similar with that in Fig. 4 (d).

References

[1] L. Johnson and L. Padgett, IEEE Trans. Neural Networks. **10** (3) (1999) 480.
[2] R.C. Muresan, Neurocomputing. **51** (2003) 487.