

XÂY DỰNG PHẦN CỨNG CHO MẠNG NƠON PCNN VÀ ỨNG DỤNG TRONG RÚT TRÍCH ĐẶC TRƯNG ẢNH

A HARDWARE IMPLEMENTATION OF PULSE COUPLED NEURAL NETWORK (PCNN) FOR IMAGE FEATURE EXTRACTION

Nguyễn Ngọc Hùng, Hoàng Trọng Thức, Bùi Trọng Tú
Trường Đại học Khoa học Tự Nhiên, ĐHQG – HCM

TÓM TẮT

Pulse Couple Neural Network (PCNN) là mô hình mạng nơon nhân tạo được phát minh dựa trên mô hình vỏ não thị giác của động vật. So với các phương pháp xử lý ảnh truyền thống, PCNN có những ưu điểm như: không cần tập dữ liệu huấn luyện, bất biến với các phép biến đổi hình học của ảnh, chống nhiễu mạnh, độ ổn định cao. Trong bài báo này, nhóm tác giả sẽ trình bày kết quả nghiên cứu mô hình PCNN và ứng dụng mô hình trong rút trích đặc trưng ảnh và nhận dạng ảnh. Mô hình PCNN đề xuất sẽ được kiểm chứng bằng kết quả mô phỏng trên phần mềm Matlab và sau đó tiến hành thiết kế phần cứng cho mạng nơon PCNN dựa trên công nghệ FPGA. Kết quả minh họa là khối rút trích đặc trưng của ảnh với ngõ ra là các vector đặc trưng. Lỗi bình phương trung bình (MSE) giữa vector đặc trưng của ảnh ở ngõ vào và vector đặc trưng chuẩn được dùng như là một tiêu chuẩn để nhận dạng, phân loại ảnh. Ảnh được sử dụng ở đây là các ảnh mức xám. Kết quả thực nghiệm thu được trên phần mềm Matlab và trên cơ sở phần cứng hoàn toàn tương đồng nhau.

Từ khóa: PCNN, vector đặc trưng, MSE

ABSTRACT

Couple Pulse Neural Network (PCNN) is the artificial neural network model was invented based on the visual cortex model of animals. Compared with other traditional processing methods, PCNN has some advantages such as not require training data, invariant to geometrical transformations of an image, robust against noise, highly stable structure. In this paper, the authors present results of a research on PCNN model and its applications in image feature extraction and image recognition. The proposed PCNN model is verified by the simulation results obtained from both Matlab software and FPGA implemetation. The demonstration is an image feature extraction block whose output is a feature vector of the input image. The mean square error (MSE) between the feature vector of the input image and standard feature vector is used as the criteria to recognize, classify the image. Images are used for this study are gray level ones. Experimental results show good agreements between Matlab software implementation and hardware implementation.

Key words: PCNN, feature vector, MSE

I. GIỚI THIỆU

Pulse-Coupled Neural Network (PCNN) là một loại mạng nơon nhân tạo được sử dụng chủ yếu trong lĩnh vực xử lý ảnh [1], [2]. Ứng dụng của PCNN bắt nguồn từ mô hình nơon của Echoron [3]. Mô hình nơon của Echoron được phát triển dựa trên quan sát tế bào thần kinh vỏ não thị giác của mèo và mô phỏng hoạt động của tế bào thần kinh thị giác. Vì

thuật toán của mô hình PCNN bắt nguồn trực tiếp từ việc nghiên cứu đặc tính thị giác của động vật có vú, nó rất phù hợp cho xử lý ảnh, như: rút trích đặc trưng của ảnh [4], phân đoạn ảnh [5], [6], dò cạnh [7], giảm nhiễu và làm mịn ảnh [8]. PCNN khác với các mạng nơon đa lớp truyền thống, nó là mô hình một lớp đơn nhưng chúng tỏ phù hợp cho xử lý ảnh thời gian thực [9]. PCNN sẽ tạo ra một chuỗi các ảnh xung nhị phân ở ngõ ra với ngõ vào là

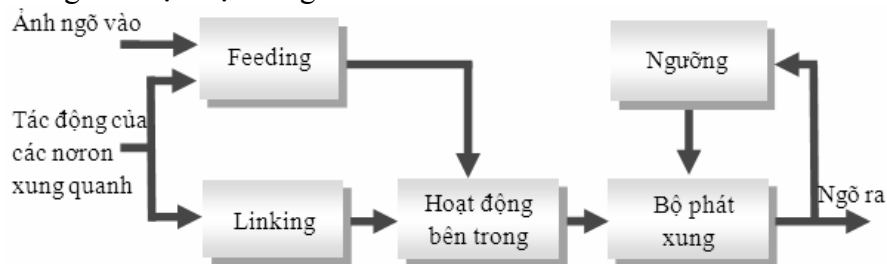
các giá trị mức xám của ảnh. Dựa trên các ảnh xung nhị phân, hệ thống sẽ trích xuất ra một vector đặc trưng của ảnh. Trong lĩnh vực thị giác máy tính có nhiều phương pháp và thuật toán đã được sử dụng để rút trích đặc trưng của ảnh [10]. Các phương pháp truyền thống tỏ ra khá hiệu quả để rút trích các đặc trưng của các ảnh đơn giản như: các dạng hình học đơn giản hoặc các ký tự. Tuy nhiên, chúng rất nhạy với nhiễu; thậm chí cho kết quả không chính xác trong trường hợp các ảnh phức tạp. Bên cạnh đó, các phương pháp này chủ yếu được thực hiện trên phần mềm, do vậy cần nhiều thời gian để xử lý. Trong khi đó, do có cấu trúc đơn giản, PCNN khá linh hoạt khi thực hiện xuống phần cứng và là một phương pháp tiềm năng cho các ứng dụng xử lý thời gian thực [9]. Trong bài báo này, tác giả sẽ trình bày một thiết kế phần cứng cho mạng nơron PCNN và ứng dụng để rút trích đặc trưng của ảnh mức xám.

Nội dung chính của bài báo gồm năm phần. Phần I và phần II giới thiệu nội dung và trình

bày cấu trúc mô hình mạng nơron PCNN. Phương pháp rút trích đặc trưng của ảnh và quá trình thực hiện phần cứng cho mạng nơron PCNN được trình bày trong phần III. Phần IV trình bày các kết quả thực nghiệm thu được trên phần mềm Matlab và phần cứng FPGA. Các kết luận được đưa ra ở phần V.

II. MÔ HÌNH MẠNG NƠN PCNN

PCNN chỉ đơn giản là một mạng lưới đơn lớp hai chiều với các kết nối bộ phận, cục bộ giữa các nơron. Kích thước của PCNN tương ứng với kích thước của ảnh ngõ vào. PCNN dao động dựa trên các kích thích dao động từ bên ngoài và các kích thích cục bộ ở bên trong mạng. Các kích thích từ bên ngoài chính là giá trị cường độ của các điểm ảnh ở ngõ vào, các kích thích cục bộ ở bên trong chính là các tín hiệu dao động từ các nơron lân cận xung quanh. Sơ đồ khối mô tả các thành phần cấu tạo nên mạng PCNN được thể hiện trong hình 1.



Hình 1: Sơ đồ khối các thành phần cơ bản của PCNN

Sự mô tả toán học đầy đủ của PCNN được đưa ra bởi Ranganath H S, Kuntimad G và Johnson JL [2]. Thuật toán được thực

$$F_{ij}[n] = S_{ij} + F_{ij}[n-1] \cdot e^{-\alpha_F} + V_F \cdot \sum_{k,l} M_{ijkl} \cdot Y_{ij}[n-1] \quad (1)$$

$$L_{ij}[n] = L_{ij}[n-1] \cdot e^{-\alpha_L} + V_L \cdot \sum_{k,l} W_{ijkl} \cdot Y_{ij}[n-1] \quad (2)$$

$$U_{ij}[n] = F_{ij}[n] \cdot (1 + \beta \cdot L_{ij}[n]) \quad (3)$$

$$Y_{ij}[n] = \begin{cases} 1 & \text{if } U_{ij}[n] > T_{ij}[n] \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$T_{ij}[n] = T_{ij}[n-1] \cdot e^{-\alpha_T} + V_T \cdot Y_{ij}[n-1] \quad (5)$$

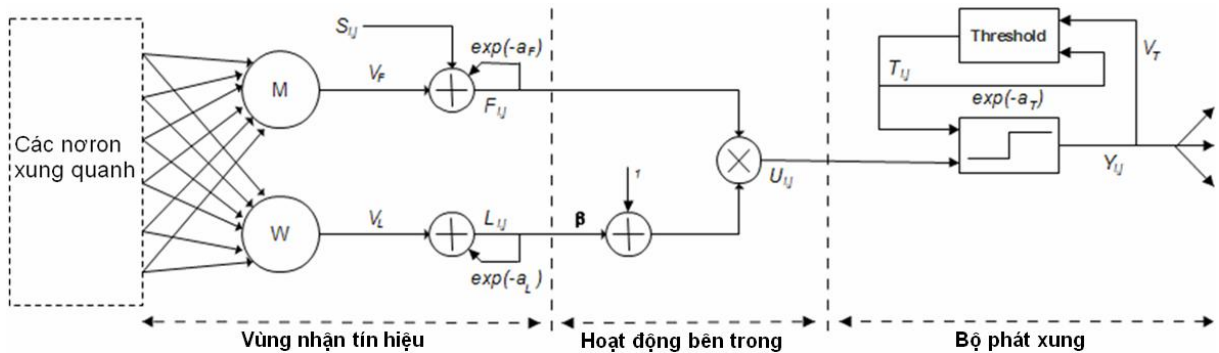
Trong đó, S_{ij} là kích thích ở ngõ vào tương ứng là các mức xám chuẩn hóa của các điểm ảnh ở vị trí (i, j) , $F_{ij}[n]$ là ngõ vào hồi tiếp của nơron tại vị trí (i, j) , và $L_{ij}[n]$ là toán hạng liên kết. β là hệ số liên kết. $U_{ij}[n]$ là giá trị hoạt động bên trong của nơron, và $T_{ij}[n]$ là ngưỡng

hiện bởi sự lặp lại liên tục ở ngõ vào và ngõ ra, với các phương trình toán học như sau:

động. $Y_{ij}[n]$ là giá trị dao động ở ngõ ra của nơron, có giá trị nhị phân 0 hoặc 1. M và W là các ma trận hằng số trọng số sinh học (kết nối) cho các ngõ vào feeding và các ngõ vào linking một các tương ứng và phụ thuộc vào khoảng cách giữa các nơron, thông thường M

và W được chọn giống nhau. α_F , α_L , và α_T là các hằng số suy giảm theo thời gian của $F_{ij}[n]$, $L_{ij}[n]$ và $T_{ij}[n]$ một cách tương ứng. V_F , V_L và

V_T là các điện thế vốn có của $F_{ij}[n]$, $L_{ij}[n]$ và $T_{ij}[n]$ một cách tương ứng.



Hình 2: Cấu trúc chi tiết của mạng nơon PCNN

Hình 2 là cấu trúc chi tiết của mạng nơon PCNN. Mỗi nơon nhận tín hiệu từ hai ngõ vào, là ngõ vào hồi tiếp $F_{ij}[n]$ và ngõ vào liên kết $L_{ij}[n]$, sau đó phát ra tín hiệu hoạt động bên trong $U_{ij}[n]$. Khi $U_{ij}[n]$ lớn hơn giá trị ngưỡng động $T_{ij}[n]$, PCNN sẽ tạo ra chuỗi xung $Y_{ij}[n]$.

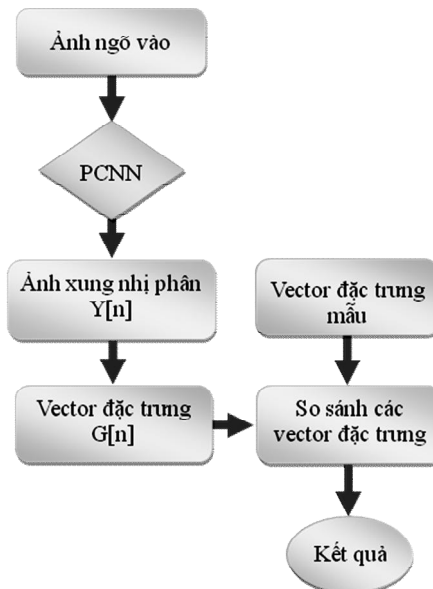
được biến đổi thành tín hiệu trong miền thời gian một chiều. Đó chính là vector đặc trưng của ảnh, có tính chất bất biến và đơn nhất. Vector tín hiệu thời gian này có thể được dùng trong các ứng dụng phân loại hoặc nhận dạng.

III. RÚT TRÍCH ĐẶC TRƯNG CỦA ẢNH DÙNG MẠNG NƠON PCNN

Quá trình thực hiện từng bước được mô tả trong hình 3. Việc tính tổng số dao động của các nơon ở ngõ ra của các ảnh xung sẽ tạo ra một tín hiệu thay đổi theo thời gian. Đây chính là cơ sở của phép biến đổi không gian ảnh từ hai chiều sang một chiều.

3.1 Phương pháp thực hiện

Một ảnh ở ngõ vào được biến đổi thành một tập các ảnh xung khi qua khối PCNN. Các ảnh xung nhị phân trong không gian hai chiều



Hình 3: Hệ thống nhận dạng cơ bản dựa trên PCNN

3.2 Vector tín hiệu thời gian (Time signal vector)

Mỗi ảnh xung nhị phân sẽ được biến đổi thành một vector thông tin đơn nhất, gọi là

vector đặc trưng của ảnh (feature vector), hay còn gọi là vector tín hiệu trong miền thời gian (Time signal vector) [2] và được tính theo phương trình (6). Thực nghiệm chứng tỏ rằng hầu hết các ảnh đều có vector tín hiệu thời

gian riêng biệt và bất biến với các sự thay đổi như xoay, thay đổi tỷ lệ, dịch chuyển hay sự nghiêng lệch của ảnh.

$$G[n] = \sum_{i,j} Y_{i,j}[n] \quad (6)$$

Có thể dùng vector đặc trưng để thực hiện công đoạn nhận dạng ảnh, tuy nhiên để tăng độ chính xác ta thường dùng giá trị entropy của ảnh và sai số bình phương trung bình (Mean Square Error- MSE) được tính lần lượt theo phương trình (7) và (8). Entropy là một yếu tố thống kê phản ánh chất lượng thông tin của ảnh [1]. Trong đó, p_1 và p_0 biểu diễn xác suất khi $Y_{ij}[n] = 1$ và $Y_{ij}[n] = 0$ ở ngõ ra một cách riêng lẻ.

$$E_n(p) = -p_1 \log_2 p_1 - p_0 \log_2 p_0 \quad (7)$$

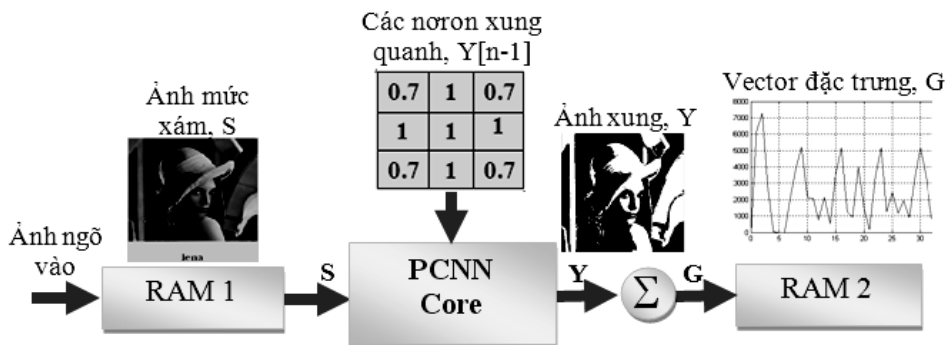
MSE giữa các vector đặc trưng được tính theo phương trình (8).

$$MSE = \frac{1}{N} \sum_{n=0}^N [E'_n(p) - E_n(p)]^2 \quad (8)$$

Ở đây, $E'_n(p)$ và $E_n(p)$ là các giá trị entropy của vector đặc trưng chuẩn và vector đặc trưng đang xét đến.

3.3 Thực hiện phần cứng cho PCNN

Tác giả sử dụng ngôn ngữ mô tả phần cứng VHDL để thiết kế phần cứng cho mô hình mạng nơron PCNN. Sau đó, kết quả thiết kế được ánh xạ lên các FPGA để kiểm tra hoạt động của mạng PCNN. Sơ đồ khối cho quá trình thực hiện phần cứng cho mạng nơron PCNN được thể hiện trong hình 4. Về cơ bản, chúng ta cần hai khối bộ nhớ cho quá trình xử lý của PCNN. Một bộ nhớ dùng để lưu trữ ảnh gốc ban đầu, và một bộ nhớ để lưu trữ kết quả thu được. Kết quả ở ngõ ra chính là các vector đặc trưng của ảnh, G. Các vector đặc trưng này được sử dụng cho quá trình nhận dạng hoặc phân loại đối tượng.



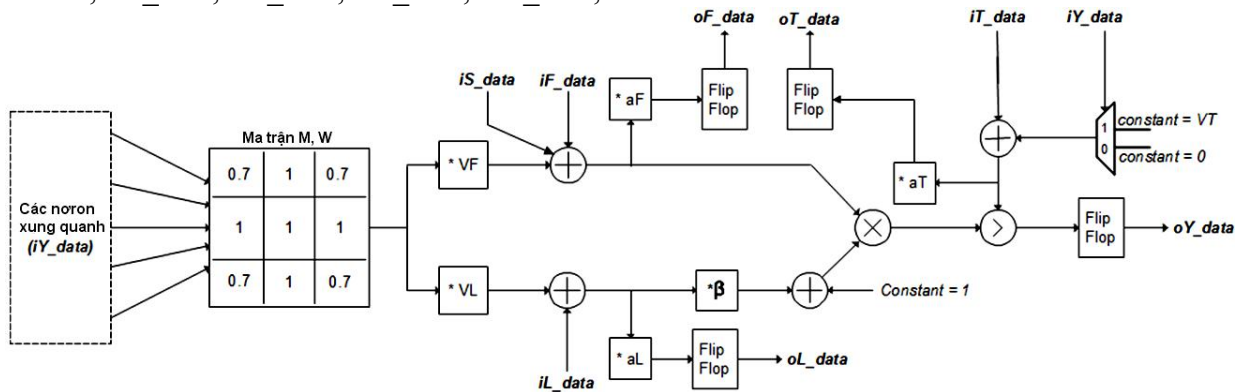
Hình 4: Sơ đồ hệ thống phần cứng mạng nơron PCNN

Dữ liệu từ bộ nhớ được đưa vào bên trong PCNN để kích thích các nơron dao động. Bên trong PCNN, các nơron sẽ tác động và ảnh hưởng qua lại lẫn nhau. Mỗi nơron sẽ bị ảnh hưởng bởi sự dao động của các nơron lân cận gần nhất, $Y(n-1)$. $Y(n-1)$ là giá trị dao động trước đó của các nơron xung quanh. Sự ảnh hưởng này được đặc trưng bởi ma trận trọng số kết nối giữa các nơron. Các giá trị của ma trận trọng số M, W bằng 1 đối với các nơron ở gần nhất và có giá trị giảm dần đối với các nơron ở vị trí xa hơn. Cấu trúc chi tiết thực hiện phần cứng cho khối PCNN được thể hiện trong hình 5. Trong đó các giá trị aF, aL và aT chính là các hằng số suy giảm tương ứng với các giá trị $e^{-\alpha F}, e^{-\alpha L}$ và $e^{-\alpha T}$ trong phương trình hoạt động của PCNN. Cấu trúc bên trong PCNN chủ yếu bao gồm các bộ cộng, bộ nhân,

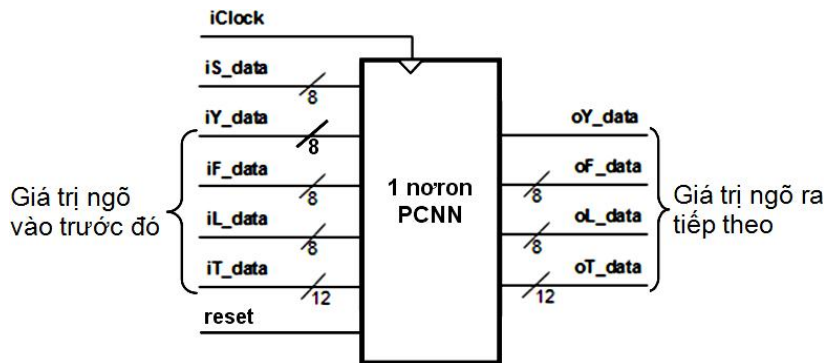
một bộ so sánh và một bộ đa hợp dùng để thay đổi giá trị ngưỡng khi nơron hoạt động. Các flipflop đóng vai trò là các thanh ghi dùng để lưu trữ các giá trị hoạt động bên trong: F, L, T của mạng và các giá trị này được cập nhật liên tục sau mỗi lần lặp của PCNN. Hình 6 là sơ đồ khối cấu tạo một nơron PCNN. Trong quá trình hoạt động, tín hiệu iS_data truyền tải các giá trị mức xám của ảnh đến khối PCNN để xử lý sau mỗi chu kỳ xung clock. Như vậy, với cấu trúc thiết kế này, tốc độ xử lý của nơron là một điểm ảnh trên một chu kỳ clock. iY_data là giá trị xung dao động của các nơron xung quanh tác động lên nơron đang xét. oY_data là giá trị xung dao động ở ngõ ra của nơron. Khi một nơron dao động sẽ tạo ra một xung ở ngõ ra và giá trị này được hồi tiếp về để gia tăng giá trị ngưỡng oT_data cho nơron. iF_data, iL_data và iT_data là các giá trị đặc trưng cho

sự hoạt động bên trong của neuron, ban đầu các giá trị này được khởi tạo bằng 0 và liên tục thay đổi sau mỗi bước lặp của PCNN. Trong hình 6, iF_data , iL_data , iT_data , oF_data ,

oL_data , oT_data là các giá trị dao động tương ứng ở ngõ vào trước đó và ở ngõ ra tiếp theo của mạng.



Hình 5: Cấu trúc thực hiện phần cứng core PCNN



Hình 6: Sơ đồ khối một neuron PCNN

Các giá trị mức xám của ảnh, iS_data , chính là các kích thích ban đầu để cho các neuron trong mạng hoạt động và ảnh hưởng suốt trong quá trình hoạt động của mạng. Dựa trên các ảnh xung được tạo ra sau mỗi lần lặp của PCNN, vector đặc trưng của ảnh được tạo thành và lưu trữ lại làm cơ sở dữ liệu cho quá trình xử lý tiếp theo. Kích thước của vector

đặc trưng G tương ứng với số lần lặp của PCNN.

IV. KẾT QUẢ THỰC HIỆN

4.1 Các thông số của PCNN

Trong bài báo này chúng ta sử dụng các thông số của PCNN trong bảng 1. Các thông số này được gán dựa trên thực nghiệm và vào ứng dụng cụ thể.

Thông số	α_F	α_L	α_T	V_F	V_L	V_T	β
Giá trị	0.5	1.0	1.0	0.02	0.02	10	0.2

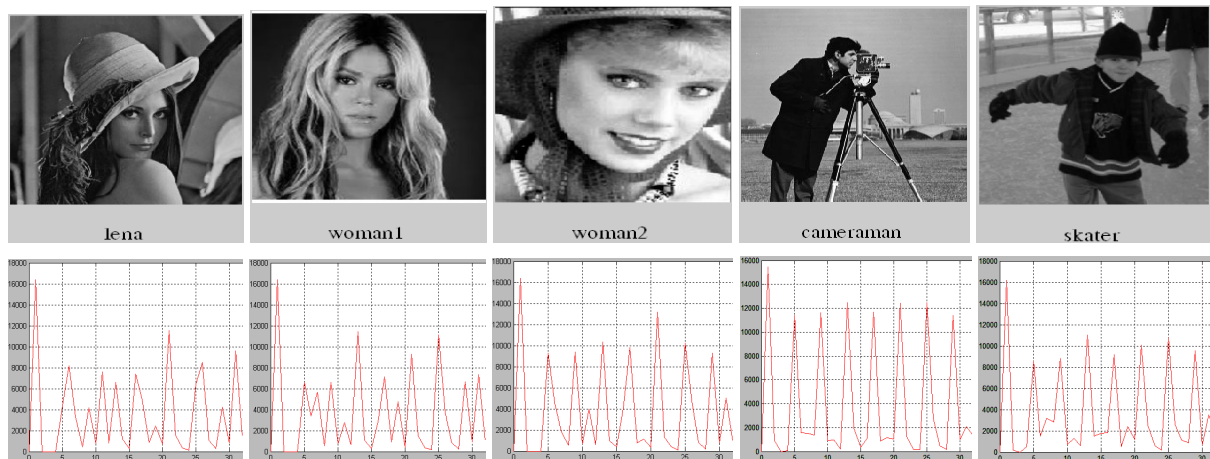
Bảng 1: Các thông số cơ bản của PCNN

Các ma trận trọng số kết nối M và W có kích thước 3×3 , được thể hiện trong hình 5. Giá trị tại các vị trí trong ma trận được tính bằng $1/r$, với r là khoảng cách tại vị trí ta muốn tính cho đến tâm của ma trận được tính theo công thức tính khoảng cách Euclide.

4.2 Kết quả mô phỏng trên Matlab

Tập dữ liệu được sử dụng là các ảnh mức xám có kích thước 128 x128 pixel. Các ảnh này lần lượt được xử lý và biến đổi thành các ảnh xung nhị phân $Y[n]$ ở ngõ ra của PCNN. Từ đó tạo ra các vector đặc trưng của ảnh G . Các kết quả mô phỏng được thực hiện dựa trên

phần mềm Matlab. Số lần lặp của PCNN là 32 lần. Vector đặc trưng tương ứng với mỗi ảnh sẽ có 32 phần tử, mỗi phần tử đặc trưng cho giá trị thông tin của ảnh ở ngõ vào sau mỗi bước lặp. Sau đó MSE giữa các vector đặc trưng được tính toán để phục vụ công đoạn nhận dạng ảnh.



Hình 7: Các ảnh khác nhau và các vector đặc trưng tương ứng

Hình 7 thể hiện các vector đặc trưng thu được tương ứng với các ảnh khác nhau. Có thể thấy là với các ảnh khác nhau sẽ có vector đặc trưng khác nhau. Nếu giả định ảnh cần nhận dạng là ảnh Lena. Giá trị MSE giữa các vector

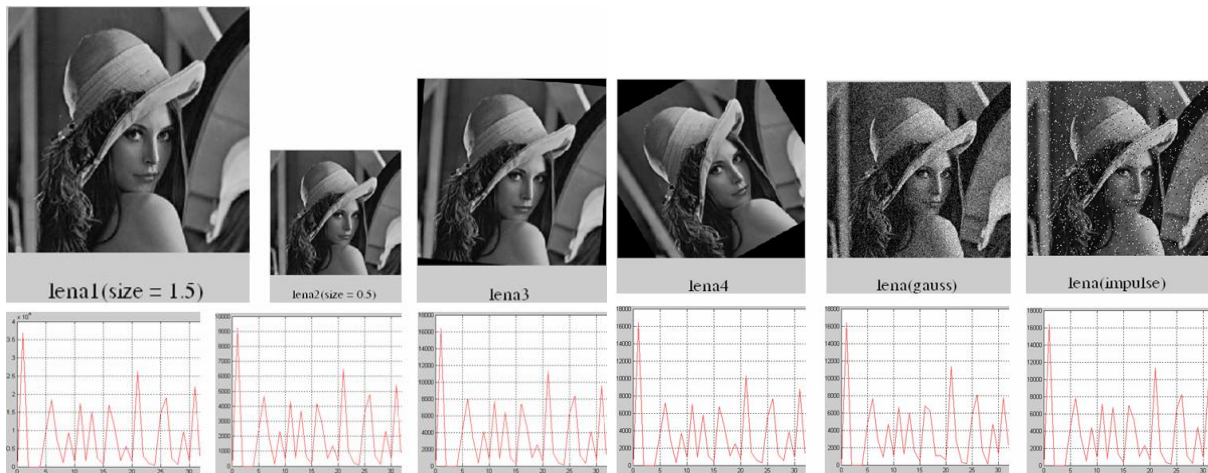
đặc trưng của các ảnh khác so với ảnh Lena được tính toán và thể hiện trong bảng 2. Có thể thấy giá trị MSE thu được sẽ khác nhau, và MSE nhỏ nhất khi hai ảnh tương đồng nhau.

Ảnh mẫu	Lena				
Ảnh test	Lena	woman1	woman2	skater	cameraman
MSE	0	0.0321	0.0499	0.0685	0.1382

Bảng 2: MSE giữa các vector đặc trưng của các ảnh khác nhau

Trong trường hợp ảnh cần nhận dạng bị nhiễu hoặc có những sự thay đổi về hình học. Giả sử, cho ảnh lena thay đổi kích thước (lớn, nhỏ), bị nghiêng hoặc bị quay một góc và bị ảnh hưởng bởi nhiễu Gaussian hoặc nhiễu xung. Chúng ta cũng thu được kết quả thể hiện trong hình 8 và bảng 3. Các vector đặc trưng trong hình 8 có mức độ tương quan rất cao, vị trí các đỉnh và tỉ lệ độ lớn có mức độ trùng khớp cao.

Các kết quả thực nghiệm thu được ở trên chứng tỏ rằng PCNN là một phương pháp hiệu quả trong lĩnh vực xử lý ảnh. Vector đặc trưng được tạo ra có tính chất đơn nhất đặc trưng cho từng ảnh và bất biến với các thay đổi hình học của ảnh, không nhạy với nhiễu. Các kết quả mô phỏng thu được trên phần mềm Matlab là nền tảng để thực hiện thiết kế phần cứng cho mạng neuron PCNN.



Hình 8: Sự thay đổi của ảnh lena và các vector đặc trưng tương ứng

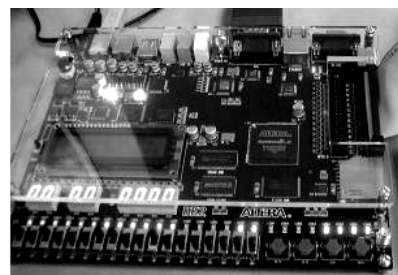
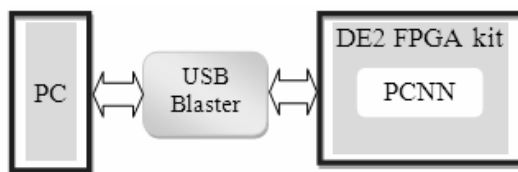
Ảnh mẫu	Lena					
Ảnh test	Lena1	Lena2	Lena3	Lena4	Lena gauss	Lena impulse
MSE	0.00014	0.00027	0.00076	0.0036	0.0049	0.00047

Bảng 3: MSE giữa các vector đặc trưng của các ảnh lena khác nhau

4.3 Kết quả thực hiện trên FPGA

Sơ đồ phần cứng FPGA rút trích đặc trưng ảnh dùng thuật toán PCNN được thể hiện trong hình 9. Trong nghiên cứu này, kit FPGA DE2 của Altera được sử dụng để hiện thực hoá thuật toán PCNN trên phần cứng. Sử dụng cấu trúc phần cứng được thể hiện trong hình 5, toàn bộ hệ thống phần cứng PCNN được cài đặt lên kit DE2 thông qua chuẩn giao tiếp USB Blaster. Tần số hoạt động của chip FPGA Cyclone II là 50 MHz. Dữ liệu ảnh từ bộ nhớ

được đưa trực tiếp vào khối PCNN sau mỗi chu kỳ của xung clock. Như vậy, mỗi điểm ảnh cần khoảng 20 ns để xử lý. Với ảnh mức xám sử dụng có kích thước 128 x 128 pixel, 8 bit/pixel. Giả sử mỗi ảnh cần từ 50 đến 70 lần lặp, khi đó PCNN có thể xử lý 60 ảnh/1 giây. Đây là tốc độ thích hợp cho nhiều ứng dụng thời gian thực. Tương tự kết quả thu được ở trên, đầu ra của thiết kế là vector đặc trưng của ảnh ngõ vào. Mỗi vector gồm 32 phần tử.



Hình 9: Sơ đồ khối giao tiếp phần cứng

Hoạt động của mạng PCNN được mô phỏng trên môi trường Modelsim để kiểm tra tính đúng đắn của hệ thống và sau đó ảnh xạ lên kit FPGA DE2. Tài nguyên phần cứng sử dụng trong thiết kế được tóm lược trong bảng 5, thiết kế tốn 2529 phần tử logic tương ứng 8% tài nguyên của chip Cyclone II. So với kết quả đạt được trước đây, hiệu suất của PCNN đã được cải tiến. Bảng 5 là kết quả so sánh

phần cứng thu được khi thực hiện PCNN trên chip FPGA FLEX 10K [9] và Cyclone II của Altera. Đối với chip FLEX 10K, tốc độ xử lý của PCNN có thể đạt được tối đa 4.7 triệu điểm ảnh/giây. Tốc độ này có thể lên tới 50 triệu điểm ảnh/giây nếu sử dụng chip Cyclone II. Điều này chứng tỏ mạng neuron PCNN rất linh động khi thực hiện xuống phần cứng.

Family	Cyclone II
Device	FP2C35F672C6
Total logic elements	1876 / 33216 (6%)
Total registers	1579
Total memory bits	145280 / 483840 (30%)
Total PLLs	0 / 4(0%)

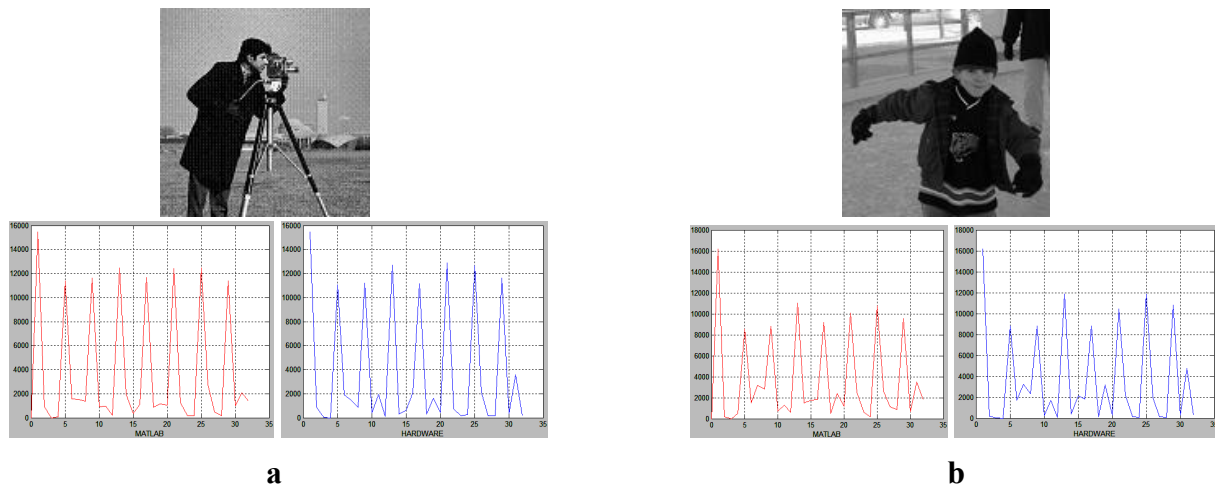
Bảng 4: Tài nguyên phần cứng đã dùng.

Family	FLEX 10K	Cyclone II
Device	EPF 10K70	FP2C35F672C6
Total logic elements	1228 / 3744 (32%)	1876 / 33216 (6%)
Performance (pixels/s)	4.7 M	50 M

Bảng 5: So sánh kết quả phần cứng đạt được trên các chip FPGA

So sánh các vector đặc trưng thu được tương ứng với các ảnh Cameraman và Skater trong môi trường Matlab và trên phần cứng được trình bày ở hình 10. Các đỉnh của vector

chính là các vị trí mà số lượng các neuron dao động là nhiều nhất. Ngược lại, các vị trí thấp nhất tương ứng với số dao động của các neuron là ít nhất.



Hình 10: So sánh các vector đặc trưng thu được tương ứng trên Matlab và FPGA trong 2 trường hợp: a) ảnh Cameraman, b) ảnh Skater

Kết quả thực nghiệm thu được chứng tỏ rằng vị trí dao động của các neuron và sự tương đồng về độ lớn giữa các vector đặc trưng thu được trên phần mềm Matlab và trên FPGA của từng ảnh là hoàn toàn giống nhau. Việc thực hiện phần cứng mạng neuron PCNN trên phần cứng đã cải tiến tốc độ xử lý của neuron nhanh hơn so với các thuật toán xử lý trên phần mềm. Nhưng quan trọng hơn, thuật toán PCNN có thể nhúng trên một phần cứng có tài nguyên hạn chế. Đó chính là tiền đề để thực hiện một hệ thống xử lý ảnh thời gian thực trên phần cứng dùng thuật toán PCNN.

V. KẾT LUẬN

Kết quả nghiên cứu chứng tỏ sự thành công của việc áp dụng mạng neuron PCNN trong lĩnh vực tiền xử lý ảnh, cụ thể trong bài báo này là rút trích đặc trưng của ảnh mức xám. Quá trình hoạt động của mạng PCNN được kiểm chứng trong môi trường Matlab và sau đó tiến hành thiết kế trên phần cứng FPGA. Vector đặc trưng của ảnh thu được trong cả hai phương pháp là giống nhau. Vector đặc trưng chính là sự chuyển đổi biểu diễn thông tin của ảnh từ không gian 2 chiều sang không gian 1 chiều. Điều này giảm bớt đáng kể số lượng thông tin dùng để biểu diễn ảnh, tiết kiệm được tài nguyên bộ nhớ và theo đó các quá trình xử lý sẽ được thực hiện dễ dàng và hiệu quả hơn.

Thiết kế phần cứng được trình bày trong bài báo là tiền đề để xây dựng các hệ thống nhúng xử lý ảnh thời gian thực nhằm giải quyết các bài toán xử lý ảnh phức tạp hơn như: nhận dạng mặt người, dò tìm chuyển động,..., thông

qua kết hợp PCNN với các thuật toán xử lý khác như ICA (Independent Component Analysis) hay PCA (Principal component analysis),...

TÀI LIỆU THAM KHẢO

- [1] Zhaobin Wang, Yide Ma, Feiyan Cheng, Lizhen Yang, *Review of pulse-coupled neural networks*, Image and Vision Computing, 2009.
- [2] T. Lindblad and J.M. Kinser, *Image Processing Using Pulse-Coupled Neural Networks*, Springer, 2005.
- [3] J.L. Johnson and M.L. Padgett, *PCNN models and applications*, IEEE Transactions on Neural Networks, 1999.
- [4] Xiaodong Gu, *Feature Extraction using Unit-linking Pulse Coupled Neural Network and its Applications*, Springer Science+Business Media, 10 November 2007.
- [5] D. Yamaoka, Y. Ogawa, K. Ishimura, M. Wada, *Motion segmentation using pulse-coupled neural network*, SICE kmual Conference in Fukui, 2003.
- [6] Y. Ma, R. Dai and L. Li, *Automated image segmentation using pulse-coupled neural networks and image's entropy*, Journal of Chinese Institute of Communication, 2002.
- [7] Liang Zhou, Yu Sun, Jianguo Zheng, *Automated Color Image Edge Detection Using Improved PCNN Model*, WSEAS Transactions on Computers, Issue 4, Volume 7, April 2008.
- [8] Y. Ma and H. Zhang, *A new image denoising algorithm combined PCNN with gray-scale morphology*, Journal of Beijing University of Posts & Telecommunications, 2008.
- [9] Vega-Pineda J, Chacon-Murguia MI, Camarillo-Cisneros R, *Synthesis of Pulsed-Coupled Neural Networks in FPGAs for Real-Time Image Segmentation*, IEEE World Congress on Computational Intelligence Proceedings, July 16-21 2006.
- [10] Thomas Lindblad, *Inherent Features of Wavelets and Pulse Coupled Neural Networks*, IEEE Trans. on neuralnetworks, pp.607-614, 1999.
- [11] Waldemark Joakim, Millberg Mikael and Lindblad Thomas, *Implementation of a Pulsed-Coupled Neural Networks in FPGA*, International Journal of Neural Systems, Vol.10, No.3.pp. 171-177, June 2000.
- [12] R. Forgác, I. Mokriš, *Algorithm for Pulse Coupled Neural Network Parameters Estimation*, IEEE 7th International Conference on Computational Cybernetics, November 26-29, 2009.
- [13] Yide Ma and Lian Li et al., *Principle of Pulse-Coupled Neural Network and Its Applications*, Science Press, Beijing, China, 2006.
- [14] Hitoshi Yamada, Yuuki Ogawa, Kosei Ishimura, Mitsuo Wada, *Face Detection using Pulse-Coupled Neural Network*, SICE Annual Conference in Fukui, August 4-6, 2003.
- [15] Kinser.J.M, *A Simplified Pulse-Coupled Neural Network*, Proceedings, SPIE, Vol. 2760, No. 3, 1996.